



POLITECNICO
DI TORINO

POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Elettrica

Tesi di Laurea Magistrale

Progettazione di motori elettrici multifase con software open source

Relatore

prof. Gianmario Pellegrino

Candidato

Simone Adamo

matricola: 221713

ANNO ACCADEMICO 2017-2018

Indice

Elenco delle figure	5
Elenco delle tabelle	8
1 Presentazione generale	10
1.1 Introduzione	10
1.2 Identificazione del problema	11
1.2.1 Macchine sincrone brushless: generalità	11
1.2.2 SyR-e: funzionalità e caratteristiche	12
1.3 Obiettivi e risultati attesi	21
2 Integrazione in Octave	23
2.1 Caratteristiche del problema	23
2.2 Caricamento, modifica e salvataggio	26
2.3 Processo di ottimizzazione	30
2.4 Post processing	33
2.5 Risultati	35
2.5.1 Ottimizzazione	35
2.5.2 Post processing	39
2.6 Conclusioni	43
3 Macchine multifase	44
3.1 Introduzione	44
3.2 Macchine multifase: generalità	45
3.3 Implementazione	47
3.3.1 Parametri aggiuntivi e modifiche alla GUI	48
3.3.2 Calcolo e definizione delle proprietà circuitali	50

3.3.3	Trasformazioni di riferimento	53
3.4	Risultati	58
3.4.1	Prototipo trifase	60
3.4.2	Macchina esafase	66
3.4.3	Confronti	70
3.5	Conclusioni	75

Elenco delle figure

1.1	Macchine brushless[1]	11
1.2	Flusso di dati tra SyR-e e FEMM	12
1.3	GUI: finestra Main Data	13
1.4	Tipologie di rotore disponibili	14
1.5	syrmDesign: diagramma x/b e curve di livello	14
1.6	Definizione dei parametri geometrici delle barriere di flusso[2]	15
1.7	GUI: finestra Other Options	16
1.8	GUI: finestra Materials	17
1.9	GUI: finestra Optimization	18
1.10	Esempio dei risultati del processo di ottimizzazione	19
1.11	Esempio dei risultati del post processing del singolo punto di lavoro	20
1.12	Esempio dei risultati del post processing nel piano $i_d - i_q$	20
2.1	GUI: esempio di modifica del disegno in base ai parametri impostati	24
2.2	OpenSave: Selezione del progetto	27
2.3	OpenSave: Modifica dei parametri principali	27
2.4	OpenSave: Modifica delle variabili di ottimizzazione	28
2.5	OpenSave: Modifiche dei parametri di ottimizzazione	29
2.6	OpenSave: Salvataggio	29
2.7	OpenSave: Visualizzazione della macchina	30
2.8	Optimize: Richiesta di svuotamento della cartella <i>partial_optimization</i>	31
2.9	Optimize: Esempio di informazioni visualizzate durante il processo	32
2.10	Optimize: Esempio di output in Octave	33
2.11	post_proc_single_motor: Modifica dei parametri principali	34
2.12	post_proc_single_motor: Esempio dei risultati del singolo punto di lavoro	35
2.13	Sezione del prototipo RawPower	36

2.14	Matlab GUI: fronte di Pareto dell'ottimizzazione	37
2.15	Matlab script: fronte di Pareto dell'ottimizzazione	37
2.16	Octave: fronte di Pareto dell'ottimizzazione	38
2.17	Matlab GUI: obiettivi dell'ottimizzazione	38
2.18	Matlab script: obiettivi dell'ottimizzazione	39
2.19	Octave: obiettivi dell'ottimizzazione	39
2.20	Selezione della macchina nel fronte di Pareto	40
2.21	Sezione della macchina selezionata	40
2.22	Post processing: risultati di coppia e fattore di potenza	41
2.23	Post processing: risultati dei flussi λ_d e λ_q	42
2.24	Post processing: risultati delle caratteristiche flusso-corrente	43
3.1	15 fasi simmetrico	46
3.2	9 fasi asimmetrico	46
3.3	Azionamenti multifase: possibili configurazioni[5]	47
3.4	GUI: nuova finestra Windings	49
3.5	Parametri per la simulazione di condizioni di guasto	49
3.6	GUI: finestra Other	51
3.7	Processo di elaborazione dati in SyR-e	54
3.8	6 fasi	56
3.9	9 fasi	56
3.10	12 fasi	56
3.11	Prototipo esafase asimmetrico in FEMM	60
3.12	Sezione del prototipo RawPower	60
3.13	Prototipo: analisi punti di lavoro caratteristici	62
3.14	Prototipo: analisi punto di lavoro $i_0/50^\circ$	62
3.15	Caratteristiche flusso-corrente del prototipo	63
3.16	Modello magnetico del prototipo: superfici di flusso e coppia	64
3.17	Prototipo: luoghi MTPA e MTPV	65
3.18	Prototipo: setpoint di corrente per MTPA e MTPV	65
3.19	Caratteristiche meccaniche del prototipo	66
3.20	Macchina esafase: analisi punti di lavoro caratteristici	67
3.21	Macchina esafase: analisi punto di lavoro $i_0/50^\circ$	67
3.22	Caratteristiche flusso-corrente della macchina esafase	68
3.23	Modello magnetico della macchina esafase: superfici di flusso e coppia	69
3.24	Macchina esafase: luoghi MTPA e MTPV	69
3.25	Caratteristiche meccaniche della macchina esafase	70

3.26	Densità di corrente in cava	71
3.27	Confronto analisi punto di lavoro $i_o/50^\circ$: λ_d e λ_q	71
3.28	Confronto analisi punto di lavoro $i_o/50^\circ$: coppia e fattore di potenza	72
3.29	Confronto caratteristiche flusso-corrente	72
3.30	Confronto luoghi MTPA e MTPV	73
3.31	Confronto setpoint di corrente per MTPA e MTPV	74
3.32	Confronto coppia all'albero	74
3.33	Confronto potenza	75

Elenco delle tabelle

1.1	Esempio matrice di avvolgimento	16
2.1	Riepilogo dei risultati delle ottimizzazioni	36
2.2	Tempi di elaborazione post processing	42
3.1	Parametri principali del prototipo	61

Sommario

L'obiettivo principale della presente tesi è quello di ampliare le funzionalità di *SyR-e*, acronimo di *Synchronous Reluctance – evolution*, applicativo open source per la progettazione e l'analisi delle macchine sincrone attraverso l'analisi agli elementi finiti e specifici algoritmi di ottimizzazione: sviluppato a partire dal 2009 in collaborazione dal Politecnico di Torino ed il Politecnico di Bari, il software è nato inizialmente come strumento di progettazione ed analisi per motori a riluttanza, ma con il passare del tempo si è evoluto verso l'implementazione di ulteriori tipologie di macchine.

Il lavoro svolto ha fornito al progetto due importanti contributi:

1. rendere compatibili gli script con *GNU Octave*, un'applicazione software open source alternativa a Matlab;
2. includere in *SyR-e* la possibilità di progettare e simulare macchine *multifase*.

Capitolo 1

Presentazione generale

1.1 Introduzione

Le fasi di modellazione, ottimizzazione e simulazione ricoprono un ruolo fondamentale nei processi di progettazione delle macchine elettriche: la possibilità di effettuare valutazioni preliminari permette infatti di individuare le soluzioni migliori in base agli obiettivi progettuali per poter proseguire con la successiva fase di prototipazione.

In tal senso esistono sul mercato software di progettazione assistita che assolvono a questo compito e permettono all'utilizzatore di ottenere informazioni riguardo le grandezze elettromagnetiche, termiche e meccaniche che caratterizzano la macchina sulla base di parametri differenti, quali ingombri, materiali, capacità di raffreddamento, geometrie rotoriche e statoriche.

Il lavoro svolto è stato incentrato sull'ottimizzazione del codice di *SyR-e*, acronimo di *Synchronous Reluctance – evolution*, software open source sviluppato in collaborazione dal Politecnico di Torino ed il Politecnico di Bari per la progettazione delle macchine sincrone attraverso l'utilizzo di analisi agli elementi finiti ed algoritmi di ottimizzazione multi-obiettivo.

Nello specifico, ci si è occupati della compatibilità del software, sviluppato in ambiente *Matlab*, con l'applicazione gratuita per il calcolo scientifico *GNU Octave*: durante il processo di integrazione del codice è stata sviluppata una nuova procedura di modifica e salvataggio manuale dei progetti, in modo tale da renderla più semplice ed accessibile su entrambe le piattaforme all'esterno dell'interfaccia grafica di *SyR-e*.

Successivamente è stata implementata all'interno del software la possibilità di

progettare macchine multifase, in particolare *multit trifase*.

Nel seguito di questo capitolo verranno descritti gli obiettivi e le caratteristiche principali del lavoro svolto, ponendo particolare attenzione sulle funzionalità del software SyR-e e sulle differenze fra gli ambienti Matlab ed Octave.

1.2 Identificazione del problema

1.2.1 Macchine sincrone brushless: generalità

Prima di addentrarsi nella descrizione delle funzionalità e delle caratteristiche di SyR-e è necessario presentare la famiglia di macchine elettriche della cui analisi e progettazione si occupa il software: i motori sincroni a magneti permanenti, tipicamente indicati con il termine *brushless*, fanno parte della famiglia delle macchine in corrente alternata, presentano generalmente a statore un avvolgimento trifase e vengono classificate in base alla configurazione magnetica del rotore (figura 1.1 [1]).

Le macchine *isotrope* presentano proprietà magnetiche costanti nella direzione radiale e tra queste la configurazione più nota è certamente quella a magneti superficiali, cui ci si riferisce tipicamente con la sigla SPM (*Surface Permanent Magnet*).

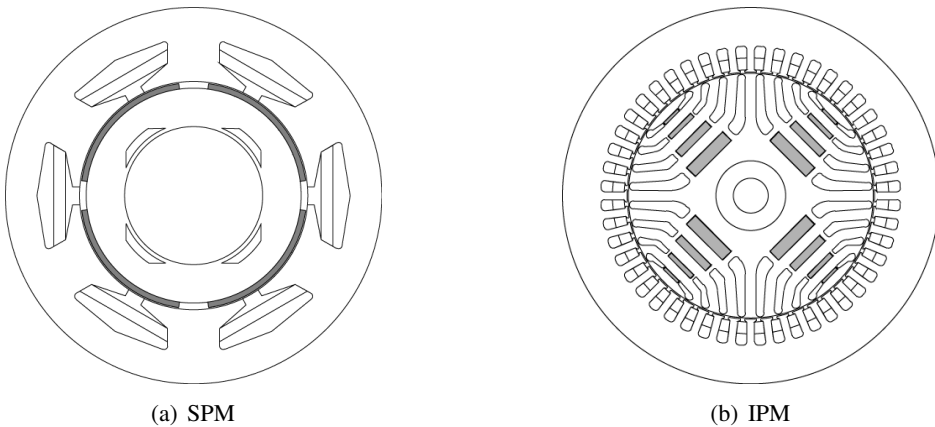


Figura 1.1: Macchine brushless[1]

Della seconda categoria fanno invece parte quelle macchine il cui rotore presenta anisotropia magnetica: quest'ultimo viene infatti opportunamente sagomato per convogliare il flusso e per ospitare eventuali magneti; di questa famiglia fanno infatti parte le macchine brushless a magneti interni, a cui comunemente ci si

riferisce con la sigla IPM, acronimo di *Interior Permanent Magnet*, ed i motori a riluttanza.

La struttura di riferimento di questi ultimi in particolare si caratterizza per il fatto che il rotore è costituito unicamente da lamierini di ferro, tranciati in modo tale da presentare sezioni d'aria (chiamate *barriere di flusso*) che indirizzano il flusso magnetico lungo direzioni ben precise; lo sfruttamento di questi vani d'aria ha portato allo sviluppo dei motori a riluttanza assistiti (*Permanent Magnet Assisted Synchronous Reluctance*), nei quali all'interno delle barriere di flusso vengono collocati magneti opportunamente dimensionati per migliorarne le caratteristiche meccaniche.

La caratteristica certamente più rilevante di queste macchine è l'assenza di contatti striscianti, che risultano essere i componenti che maggiormente contribuiscono alla diminuzione della vita utile e rappresentano in sé un elemento di pericolo per l'ambiente esterno, rendendo di fatto i motori a spazzole non compatibili con applicazioni in ambienti speciali a rischio incendio.

La presenza inoltre di un unico avvolgimento elettrico, in genere trifase e collocato a statore, semplifica lo smaltimento delle perdite, localizzate principalmente nella sezione esterna della macchina: ciò permette l'adozione di sistemi di raffreddamento ridotti ed ovviamente aumenta l'efficienza delle macchine.

D'altro canto il maggior costo dovuto alla presenza dei magneti ed ai particolari processi di lavorazione richiesti rappresenta il vero aspetto negativo di queste macchine, a cui si aggiunge una maggior complessità dei sistemi di controllo real-time.

1.2.2 SyR-e: funzionalità e caratteristiche

Come introdotto nella sezione precedente, tutto il lavoro svolto è stato incentrato sull'ampliamento delle funzionalità dell'applicativo di progettazione open source SyR-e: tale strumento si avvale del software gratuito *FEMM* per l'analisi FEA (*Finite Element Analysis*) per poi elaborarne i risultati all'interno dell'ambiente di

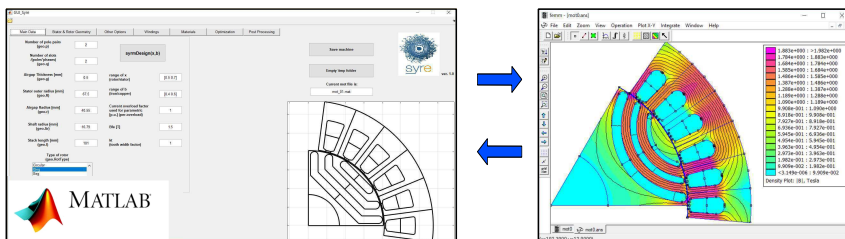


Figura 1.2: Flusso di dati tra SyR-e e FEMM

calcolo Matlab, nel quale è stato scritto e sviluppato; una schematizzazione del flusso di dati scambiati fra i due applicativi è riassunto in figura 1.2.

Tra i vari vantaggi di questa scelta vi è senza dubbio quella di poter creare interfacce grafiche in maniera semplificata tramite l'ausilio del tool di sviluppo dedicato *GUIDE*: nel tempo è infatti stata implementata una GUI (*Graphical User Interface*) che organizza tutte le funzioni e semplifica di fatto l'interazione con il software stesso, sia per quanto riguarda l'impostazione dei parametri di input sia per quello che riguarda l'acquisizione dei dati elaborati.

Lo script *GUI_Syre.m* permette di avviare l'interfaccia grafica, che si presenta con la schermata principale (figura 1.3) in cui viene caricata una macchina di default, di cui vengono riportate i parametri principali ed il disegno della sezione o di parte di essa; è possibile caricare un progetto differente utilizzando il tasto collocato in alto a sinistra, presente in ogni schermata dell'interfaccia.

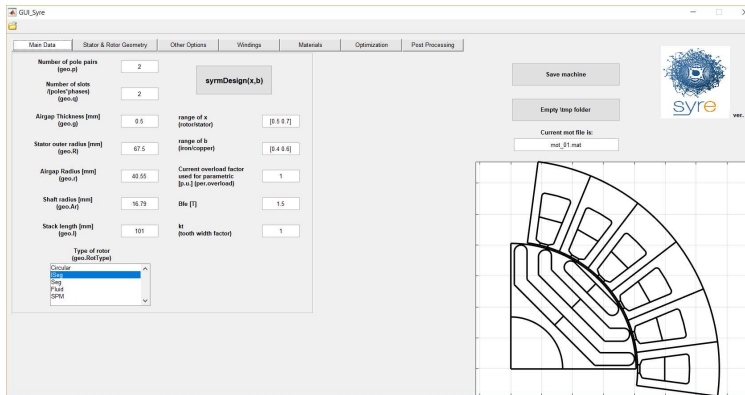


Figura 1.3: GUI: finestra Main Data

Una delle caratteristiche più interessanti è certamente la possibilità di impostare la tipologia di rotore della macchina: sebbene sia nato come strumento di progettazione per macchine sincrone a riluttanza, da cui appunto il nome, SyR-e al momento supporta differenti geometrie di barriera di flusso ed inoltre macchine a magneti superficiali SPM (*Surface Permanent Magnet*).

È possibile accedere alla prima delle funzionalità che offre l'applicativo, ossia un'analisi parametrica di coppia e fattore di potenza sulla base dei dati inseriti tramite la funzione *symrDesign*: tale analisi viene condotta andando a variare due parametri differenti, il rapporto fra il raggio di rotore e quello di statore x ed il rapporto fra i volumi occupati da rame e ferro nello statore b .

Ad elaborazione terminata, i dati vengono riassunti in un grafico che mette in evidenza le curve a coppia costante (rosse in figura 1.5) ed a fattore di potenza

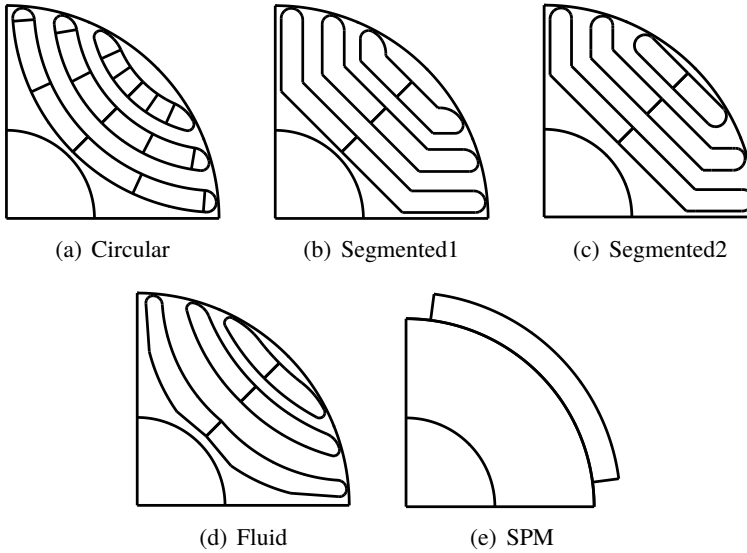


Figura 1.4: Tipologie di rotore disponibili

costante (blu), da cui è possibile inoltre selezionare direttamente le macchine da salvare.

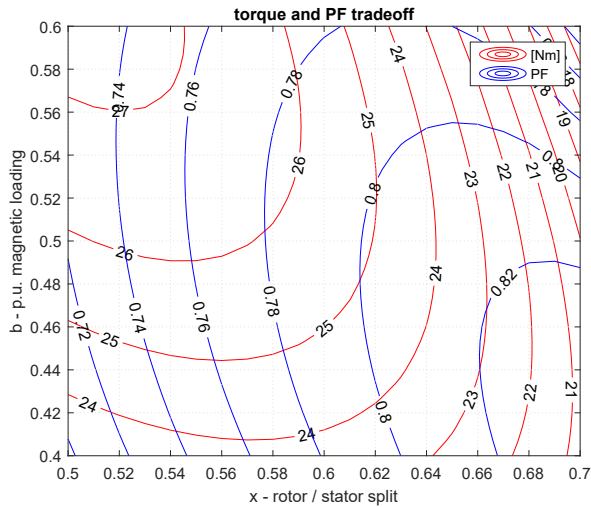


Figura 1.5: syrmDesign: diagramma x/b e curve di livello

Nella schermata *Stator & Rotor Geometry* si ha accesso alla modifica dei parametri caratteristici di rotore e statore: è infatti possibile definire in maniera precisa la geometria delle cave di statore, specificandone la lunghezza *geo.lt*, la larghezza *geo.wt* e l'apertura *geo.acs*, mentre per quanto riguarda il rotore, in base alla tipologia selezionata, possono essere scelte il numero di barriere di flusso e la loro geometria attraverso tre parametri:

alpha rappresenta l'apertura angolare delle estremità di barriera, la prima delle quali viene calcolata a partire dalla semiampiezza del polo (asse *q* in figura 1.6), mentre gli altri vengono determinati come differenza dall'angolo della barriera precedente;

dx definisce lo spostamento delle barriere lungo l'asse posto sulla semiampiezza del polo rispetto alle posizioni standard;

hc determina l'ampiezza della barriera di flusso in senso radiale.

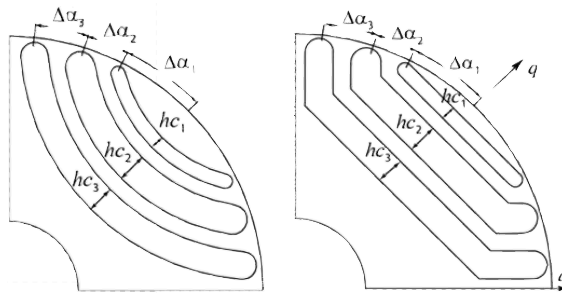


Figura 1.6: Definizione dei parametri geometrici delle barriere di flusso[2]

La finestra *Other Options* (figura 1.7) riassume parametri di natura differente: il primo valore modificabile è quello relativo alle perdite a rotore bloccato *per.loss*, che insieme alla temperatura di progetto dell'avvolgimento permette di calcolare la corrente nominale della macchina *geo.i0* e stimare la resistenza di fase in base alle caratteristiche geometriche dell'avvolgimento stesso; a partire dal modello termico della macchina viene inoltre stimata la temperatura dell'avvolgimento in base a quella impostata per la carcassa.

La velocità massima attesa della macchina risulta necessaria per il dimensionamento dei ponticelli radiali delle barriere, i quali hanno funzione unicamente strutturale: in relazione alla tolleranza meccanica di realizzabilità del lamierino (parametro *geo.pont0*), lo spessore dei ponticelli è in prima approssimazione proporzionale allo sforzo centrifugo al quale è sottoposta la macchina, massimo nella condizione peggiore che coincide appunto con il limite estremo del campo di velocità.

Sono inoltre modificabili i parametri relativi alla mesh utilizzata in FEMM per l'analisi agli elementi finiti: è infatti possibile impostare mesh a densità più o meno elevata in maniera indipendente per le operazioni di ottimizzazione (parametro *geo.K_mesh_MOOA*) e di post processing (*geo.K_mesh*). Nella parte inferiore della finestra compare invece la sezione relativa al dimensionamento dei magneti all'interno delle barriere di flusso (fino ad un massimo di 4): è possibile infatti selezionare la tipologia di magneti da utilizzare e, in base ai dati immessi, il software restituirà le dimensioni della sezione degli stessi.

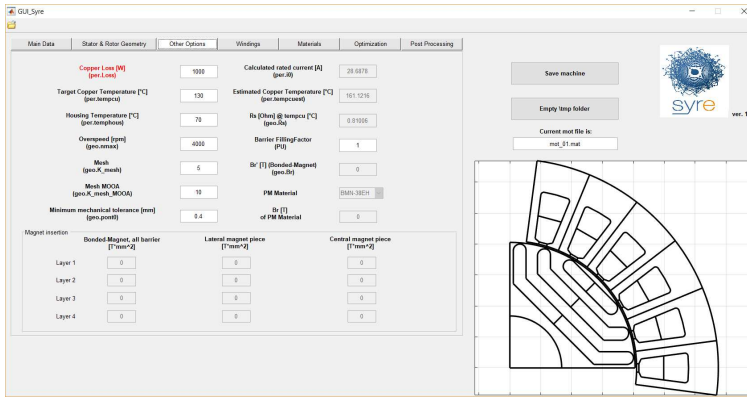


Figura 1.7: GUI: finestra Other Options

Nella schermata *Windings* vengono riportati i dati principali relativi all'avvolgimento, in primis il numero di spire in serie per fase *geo.Ns* ed i fattori di riempimento delle cave e di raccorciamento, rispettivamente *geo.Kcu* e *geo.Kracc*; selezionando inoltre il numero di cave da simulare *geo.Qs*, nella parte inferiore della finestra viene visualizzato lo schema di avvolgimento della macchina all'interno di una tabella di dimensioni $2 \times \text{geo.Kracc}$: la riga superiore corrisponde allo strato esterno del layer, mentre la prima colonna corrisponde alla prima cava, il cui asse di simmetria coincide con l'asse orizzontale.

I valori dello schema d'avvolgimento vengono calcolati da una versione modificata del software open source *Koil*, che si occupa di verificare la fattibilità dell'avvolgimento e genera l'output corretto ogniqualvolta vengano modificati i dati precedenti; un esempio di matrice ottenuta impostando *geo.Qs* = 2 e *geo.Kracc* = 5/6 è riportato nella tabella 1.1.

$$\begin{bmatrix} 1 & 1 & -3 & -3 & 2 & 2 \\ -2 & 1 & 1 & -3 & -3 & 2 \end{bmatrix}$$

Tabella 1.1: Esempio matrice di avvolgimento

Esiste anche la possibilità di progettare avvolgimenti a bobina di dente, che presentano numero di cave/polo/fase non intero: ciò è possibile selezionando l'opzione *Side by side slot layer position*.

All'interno della finestra *Materials* (figura 1.8) è possibile scegliere i materiali da utilizzare per i singoli elementi della macchina: nella parte superiore si può infatti selezionare il ferro di rotore, di statore e dell'albero, il tipo di rame da assegnare all'avvolgimento statorico e le proprietà delle barriere di flusso. Nella parte bassa è invece presente un riquadro che interagisce con i bottoni sottostanti, nello specifico scegliendo la famiglia di materiali di interesse ne verrà visualizzata all'interno del riquadro la lista completa (magneti, ferro o conduttori) ed attraverso il bottone *View prop* è possibile visualizzare le proprietà relative al materiale selezionato.

La modifica dei materiali presenti nella libreria è possibile esternamente alla GUI andando ad editare manualmente i file associati nella cartella *Material library* presente all'interno della directory di SyR-e.

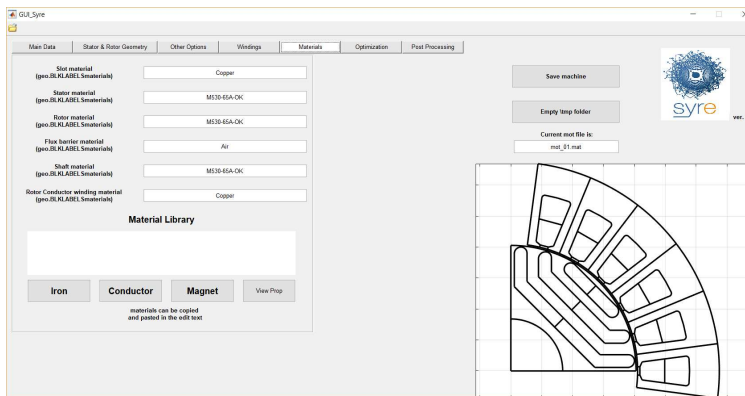


Figura 1.8: GUI: finestra Materials

Le successive due schermate di SyR-e permettono di accedere a due tra le funzionalità più importanti del software: dalla finestra *Optimization* (figura 1.9) è infatti possibile avviare il processo di ottimizzazione a partire dai parametri geometrici impostati nelle finestre precedenti.

Nel software è stato implementato un processo di ottimizzazione in due stadi, dapprima viene eseguito un algoritmo di ottimizzazione multi-obiettivo basato sul concetto di *evoluzione differenziale* [3] che identifica le soluzioni migliori tra quelle generate in maniera stocastica, riassunto nel *fronte di Pareto*, successivamente vengono nuovamente riesaminate le soluzioni migliori con maggior precisione. Nella schermata è possibile modificare tre tipologie di parametri:

- i parametri dell'ottimizzazione possono essere impostati nella parte sinistra della schermata, è possibile infatti scegliere il numero di generazioni del processo evolutivo e la dimensione della popolazione per la singola generazione; inoltre si può impostare l'angolo di rotazione massimo ed il numero di posizioni rotoriche da simulare in maniera indipendente per i due stadi del processo di ottimizzazione.
La corrente da utilizzare invece può essere modificata nella parte destra della finestra in *per unit* rispetto a quella nominale;
- le variabili geometriche da ottimizzare sono invece riportate nella sezione centrale: in base al tipo di macchina scelta precedentemente è possibile accedere ai parametri disponibili selezionando il checkbox e modificare il range di valori nel box corrispondente. Possono essere selezionati numerosi parametri geometrici, quali la dimensione e la posizione delle barriere di flusso, il raggio e la dimensione del traferro ed i parametri geometrici di cava, mentre in ultimo sono presenti i box relativi al γ , angolo del vettore corrente calcolato rispetto all'asse di minima riluttanza (a cui ci si riferisce comunemente come asse d);
- i tre obiettivi del processo di ottimizzazione sono invece selezionabili alla destra dei box dei parametri: è possibile infatti impostare il valore minimo di coppia accettato, il massimo valore del ripple picco-picco ammesso e la massa di rame.

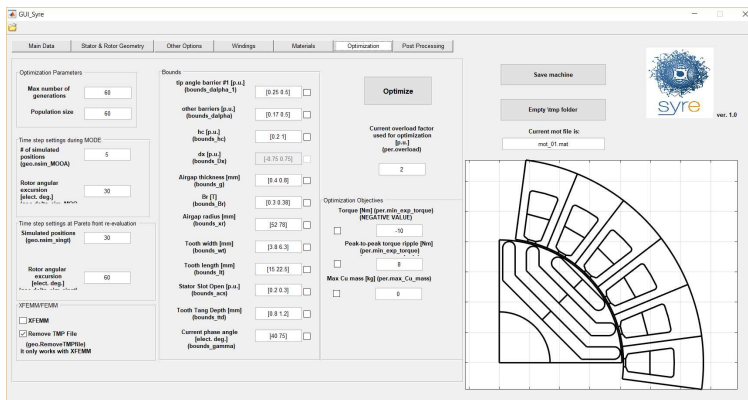


Figura 1.9: GUI: finestra Optimization

I risultati appaiono a video al termine del processo di ottimizzazione ed all'interno della cartella *results* vengono salvate automaticamente le figure riepilogative

dei risultati, che riportano il fronte di Pareto individuato, gli obiettivi ed i valori dei parametri di ottimizzazione associati alle singole macchine ed una coppia di file, con estensione *.fem* e *.mat*, che permettono di caricare nuovamente la rispettiva macchina nella GUI oppure aprire il progetto esternamente in FEMM.

A titolo di esempio in figura 1.10 si riportano i risultati di un'ottimizzazione eseguita impostando il numero di generazioni ed una dimensione delle popolazioni pari a 15 e come parametri di ottimizzazione la posizione delle estremità di barriera e lo spessore delle stesse.

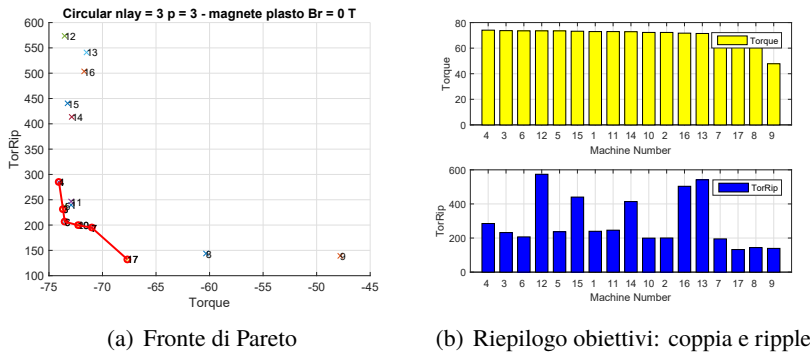


Figura 1.10: Esempio dei risultati del processo di ottimizzazione

La schermata relativa al *Post Processing* mette a disposizione uno strumento di analisi delle grandezze e delle performance della singola macchina in una o più condizioni di funzionamento: nella parte destra della schermata è possibile modificare i parametri utili alla valutazione delle perdite nel ferro sulla base della formula $P_{loss} = k_h f^\alpha B^\beta + k_e (fB)^2$.

Nella parte sinistra, oltre ad alcune impostazioni già incontrate nella sezione relativa all'ottimizzazione, vi è la possibilità di accedere a tre differenti tipologie di analisi tramite il parametro *Current phase angle*: inserendo un'unica coppia di valori corrente/fase viene eseguita l'analisi del singolo punto di lavoro, che permette di ottenere gli andamenti (figura 1.11), in funzione della posizione angolare, di coppia, fattore di potenza (*IPF*) e dei flussi nel riferimento rotante. Inserendo più valori di corrente e γ viene avviata l'analisi contemporanea di più punti di lavoro, con risultati simili al caso precedente.

Attribuendo invece alla fase della corrente il valore convenzionale di 1000 si può procedere alla caratterizzazione elettromagnetica e meccanica della macchina: il software infatti restituisce una serie di curve, parametrizzate in base ai valori assunti dalla corrente in asse diretto i_d e di quadratura i_q , i cui dati possono essere rielaborati per ottenerne di più specifici, come le curve di magnetizzazione od i

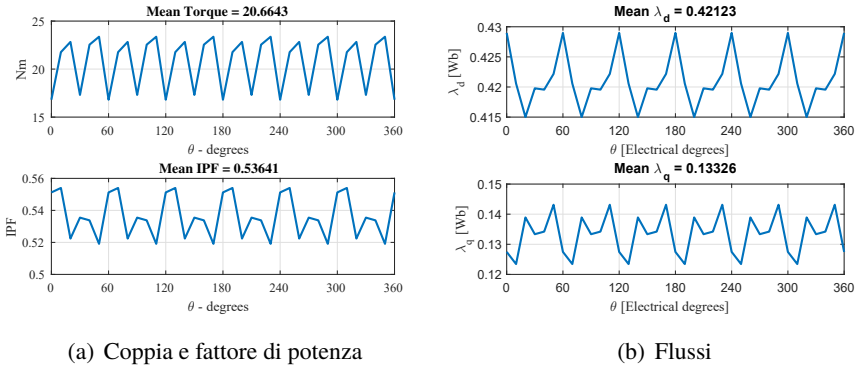


Figura 1.11: Esempio dei risultati del post processing del singolo punto di lavoro

luoghi notevoli in ambito controllistico come *Maximum Torque Per Ampere* oppure *Maximum Torque Per Voltage*.

Un esempio di risultati è riportato in 1.12.

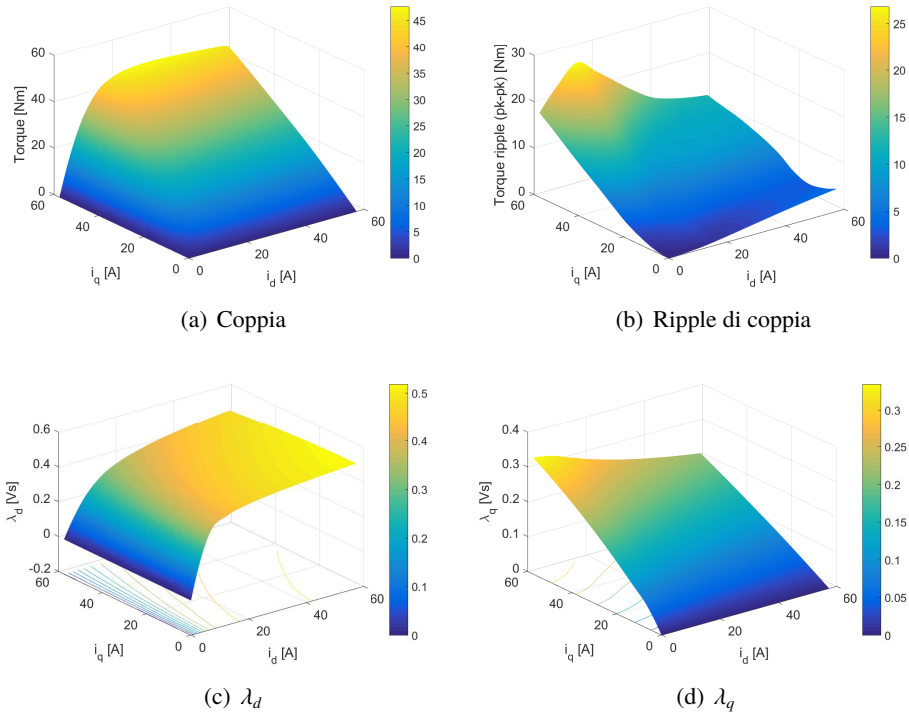


Figura 1.12: Esempio dei risultati del post processing nel piano $i_d - i_q$

1.3 Obiettivi e risultati attesi

Dopo aver presentato brevemente gli aspetti fondamentali del software oggetto di questa tesi è possibile addentrarsi nella descrizione dei problemi affrontati: il lavoro svolto si è infatti sviluppato lungo due strade differenti e ciò ha permesso di verificare in maniera indipendente la correttezza delle soluzioni proposte.

Per prima cosa ci si è occupati del porting di SyR-e all'interno dell'ambiente di calcolo GNU Octave: si è provveduto ad individuare quali funzionalità del software originario fossero compatibili, verificando fin da subito di non poter adottare in maniera semplice l'interfaccia grafica già sviluppata in Matlab; questo ha portato all'implementazione, a partire da alcuni script funzionanti all'esterno della GUI, di nuove funzioni in grado di aprire, modificare, disegnare e salvare progetti in maniera interattiva attraverso apposite finestre di dialogo.

In seguito è stato il turno di due tra le funzionalità più importanti di SyR-e, ossia il processo di ottimizzazione e lo strumento di post processing: sebbene anche queste potessero essere richiamate in maniera indipendente dalla GUI, la non corrispondenza tra funzioni e formati di Matlab e Octave ha richiesto un intenso lavoro di debug e correzione.

Una volta unificato il codice si è passati alla verifica delle modifiche implementate: per far ciò sono stati impostati tre processi di ottimizzazione, uno su Octave e due in ambiente Matlab, avviati rispettivamente attraverso la GUI e lo script dedicato, basate sul modello di una macchina reale e con numero di generazioni e popolazioni elevate in modo da ottenere risultati quanto più significativi possibile.

Per verificare il corretto funzionamento del post processing è stata selezionata una delle soluzioni individuate nelle ottimizzazioni precedenti e, in maniera analoga a quanto già fatto, sono state svolte tre analisi, avviate secondo le modalità descritte per l'ottimizzatore, in specifiche condizioni di lavoro: attraverso il confronto dei risultati sarebbe stato possibile confermare la correttezza delle modifiche introdotte.

Successivamente ci si è occupati di implementare una nuova funzionalità nel software, ossia la progettazione di macchine con avvolgimento statorico multifase, nello specifico *multitrifase*: per fare ciò è stato necessario lavorare principalmente sulla finestra *Windings* della GUI e su tutti le porzioni di codice ad essa connesse, andando ad introdurre opportuni parametri in vista di un futuro aggiornamento di *Koil*, che si occupa di generare le matrici degli schemi d'avvolgimento utilizzati all'interno di SyR-e e che al momento non è in grado di gestire tale casistica.

Gli elementi chiave di questa parte del lavoro sono certamente stati la correzione del calcolo del modulo di corrente nominale i_0 , basato sul massimo valore di perdita

dissipabile, e l'implementazione delle opportune matrici di trasformazione che tenessero conto della diversa posizione nello spazio dei riferimenti delle singole terne: questo è stato possibile dopo un'approfondita ricerca bibliografica che ha permesso di individuare l'approccio più adatto al caso in questione.

Come nella fase precedente, la verifica del codice ha avuto luogo attraverso l'analisi di una macchina definita a partire dal modello di una macchina reale, che ben si adattava ad una realizzazione esafase (o meglio *bi-trifase*): dopo aver caratterizzato elettromagneticamente e meccanicamente la nuova macchina ottenuta è stato possibile confrontare i risultati con quella di partenza e verificare benefici e svantaggi della configurazione proposta.

Capitolo 2

Integrazione in Octave

2.1 Caratteristiche del problema

Come anticipato nella parte introduttiva, SyR-e, seppur sviluppato e disponibile come software open source, era utilizzabile nel pieno delle proprie funzionalità unicamente all'interno del software commerciale Matlab: il primo passo è consistito proprio nel rendere compatibili parte di queste funzionalità nell'ambiente di calcolo open source *GNU Octave*, cercando, quando possibile, di mantenere un unico codice sorgente.

Inizialmente la difficoltà maggiore è consistita nell'impossibilità di adattare l'interfaccia grafica al funzionamento in Octave: veniva quindi meno l'ipotesi di utilizzare lo strumento che più di tutti rendeva intuitiva e relativamente semplice l'interazione col software, soprattutto per quanto riguarda la rappresentazione della sezione di macchina. Variare la tipologia di rotore ed i parametri geometrici attraverso i campi della GUI permette infatti di visualizzare immediatamente le modifiche apportate in quanto la macchina viene ridisegnata ad ogni modifica (si noti in figura 2.1 la differenza tra le due sezioni mostrate, ottenute variando unicamente due parametri di macchina).

La revisione 312, versione di SyR-e su cui è stato svolto il lavoro, includeva già lo script *manual_dataSet.m* che permetteva di modificare la struttura dei parametri base *dataSet* all'esterno dell'interfaccia grafica: questo veniva richiamato a sua volta da *DrawMachineScript.m* e *MODEstart.m*, che rispettivamente eseguivano il salvataggio della macchina definita manualmente sia in file Matlab che in FEMM e l'avvio del processo di ottimizzazione.

Inizialmente ci si è focalizzati sull'analisi ed il debug del secondo script e di tutte

- La gestione dei percorsi delle cartelle e dei singoli file, di cui fa ampio uso il codice di SyR-e, si è dimostrata essere molto differente fra i due ambienti di lavoro: per la modifica ed il salvataggio Octave accetta unicamente come argomenti i nomi estesi degli oggetti, differentemente da Matlab che risulta essere più flessibile; una corretta definizione delle variabili ha permesso di rendere compatibile ed unificare il codice.
- I risultati di ogni ottimizzazione venivano salvati in una cartella dedicata all'interno della directory *results*: oltre ai grafici visualizzati al termine del processo, lo script salvava tutte le soluzioni individuate con estensione *.fem* e *.mat* in modo da poterle caricare nuovamente oppure visualizzare il problema FEMM associato.
Per permettere ai dati salvati di essere leggibili in entrambi gli ambienti di lavoro è stato necessario modificare la procedura di salvataggio, mentre per quanto riguarda i risultati grafici era fondamentale poterli modificare su entrambe le piattaforme.
Una differenza sostanziale ha riguardato però l'estensione dei file, in quanto il formato *FIG* adottato da Matlab è proprietario e risultava complicato accedere ai file creati in uno dei due ambienti sull'altro: pertanto si è deciso di salvare i grafici di output in Octave con estensione *.ofg*, in modo da permetterne l'apertura tramite il comando *hgload* e la successiva modifica.

Il calcolo parallelo è certamente una fra le più interessanti features che offre Matlab: la possibilità di eseguire del codice su più processori in contemporanea permette infatti di sfruttare a pieno l'hardware a disposizione e di diminuire notevolmente il tempo necessario per l'esecuzione dei processi di calcolo.

Il comando *parfor* avvia un gruppo di lavoro locale (*parallel pool*) con tanti elementi quanti sono i core dei processori dell'hardware e distribuisce l'onere computazionale tra i vari workers.

Poichè il *parfor* risultava compatibile con Octave e veniva interpretato in maniera equivalente al comando *for*, si è deciso di dare priorità alle altre funzionalità di SyR-e in modo tale da renderle disponibili, seppur in maniera essenziale, a tutti coloro che non posseggono Matlab.

Una volta sistemato l'ottimizzatore è stato il turno del post processing, accessibile tramite la funzione *post_proc_single_motor.m*, che presentava le stesse problematiche riguardanti il salvataggio dei dati presentati precedentemente, e della funzione di modifica e salvataggio delle macchine: in quest'ultimo caso si è deciso di riscrivere da zero lo script *manual_dataSet* in modo tale da renderne il

funzionamento più intuitivo.

Per far ciò si è ricorso all'utilizzo di apposite finestre di dialogo che permettono di aprire una macchina esistente, modificarne i principali parametri e decidere se salvare o meno il progetto appena definito.

Al termine del processo è possibile visualizzare la sezione della macchina, così come accade in Matlab tramite la GUI.

Nel seguito del capitolo verranno descritte le nuove funzionalità del software, ponendo particolare attenzione sulla procedura e sui singoli elementi che permettono di modificare ed analizzare i progetti in maniera indipendente dall'interfaccia grafica di SyR-e, sia su Matlab che su Octave.

2.2 Caricamento, modifica e salvataggio

Come riportato nella documentazione fornita con SyR-e, prima di iniziare qualunque lavoro è necessario avviare lo script *setupPath.m*, il quale aggiunge i percorsi utilizzati dall'applicativo all'albero delle directory dell'ambiente di lavoro: tale procedura è necessaria ed è sufficiente eseguirla la prima volta che si avvia SyR-e.

Il file che assolve alle funzioni di modifica, salvataggio e visualizzazione delle macchine prende il nome di *OpenSave.m* ed è possibile eseguirlo sia in ambiente Matlab che in Octave: una volta avviato lo script visualizzerà una serie di finestre di dialogo, ognuna delle quali assolve ad una funzione specifica.

La prima finestra che appare (figura 2.2) permette all'utente di selezionare la macchina sui parametri della quale iniziare la definizione del nuovo progetto: viene visualizzata la cartella di lavoro attuale di Matlab/Octave e tutti i file *.fem* presenti in essa; di default viene proposta la stessa macchina caricata all'avvio della GUI, il *mot_01*. Una volta selezionata la macchina voluta cliccando sul pulsante *Open*, lo script acquisisce i dati del progetto scelto attraverso il corrispondente file *.mat*.

A partire dalla prima schermata di selezione vengono visualizzate una serie di finestre che mostrano i parametri principali della macchina selezionata e ne permettono la modifica, la prima delle quali riguarda la tipologia di rotore: le casistiche disponibili sono elencate nella finestra e viene selezionata di default l'opzione corrispondente alla proprietà della macchina scelta.

La successiva finestra di dialogo, chiamata *Edit main parameters*, richiama la schermata principale della GUI e permette di visualizzare i parametri fondamentali.

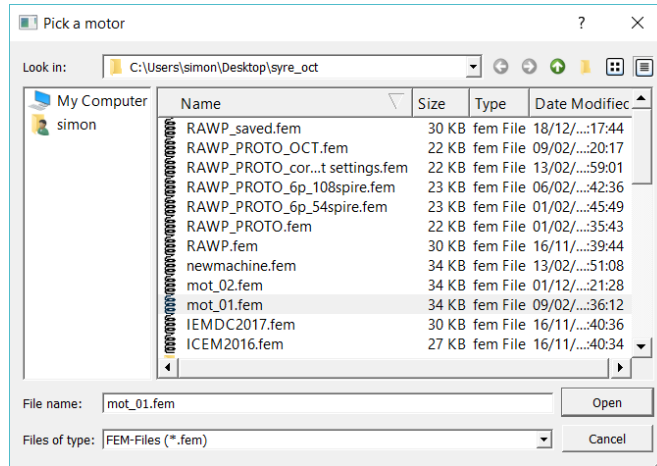


Figura 2.2: OpenSave: Selezione del progetto

Nello specifico è possibile infatti modificare:

1. il numero di paia poli
2. il numero di cave/polo/fase
3. lo spessore del traferro
4. il raggio esterno di statore
5. il raggio esterno di rotore
6. il raggio dell'albero
7. la lunghezza del pacco attivo
8. il numero di barriere di flusso (eccetto nel caso SPM)

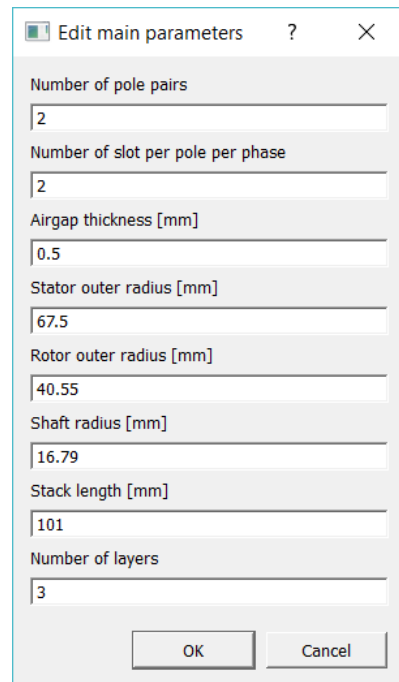


Figura 2.3: OpenSave: Modifica dei parametri principali

Una volta confermato si accede alla sezione *Edit boundary parameters* che, in continuità con la scheda *Optimization* della GUI, permette di scegliere i campi di variazione dei parametri del processo di ottimizzazione: si è deciso in realtà di implementare unicamente le variabili di maggior interesse, nello specifico l'apertura angolare delle singole barriere $\Delta\alpha$, distinte fra quella della barriera più interna e le successive, lo spessore delle barriere di flusso h_c ed infine la fase γ del vettore corrente i_0 nel riferimento rotante.

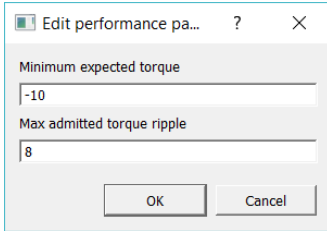
Come si può notare in figura 2.4 l'immissione dei valori avviene singolarmente, viene infatti specificato nella descrizione posta sopra ad ogni box se si tratta del limite superiore oppure inferiore del range di variabilità.

Parameter	Value
Outer barrier angle (lower limit) [PU]	0.25
Outer barrier angle (upper limit) [PU]	0.5
Other barrier angle (lower limit) [PU]	0.17
Other barrier angle (upper limit) [PU]	0.5
Flux barrier width (lower limit) [PU]	0.2
Flux barrier width (upper limit) [PU]	1
Current angle phase (lower limit) [electr. deg.]	40
Current angle phase (upper limit) [electr. deg.]	75

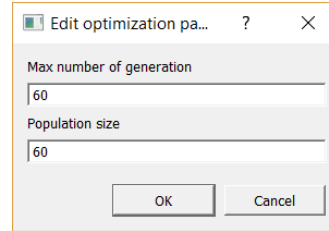
Figura 2.4: OpenSave: Modifica delle variabili di ottimizzazione

La schermata successiva (2.5) è quella relativa agli obiettivi dell'ottimizzazione: nella finestra *Edit performance parameters* possono essere modificati i valori di due degli obiettivi disponibili nell'analisi, in particolare il minimo valore di coppia attesa (da inserire sempre con valore negativo) e il massimo valore del ripple picco-picco di coppia ammesso: durante il processo di ottimizzazione le macchine che presentano parametri di coppia non compatibili con i limiti fissati vengono penalizzate, in modo tale da spingere l'algoritmo a ricercare in regioni differenti

soluzioni auspicabilmente migliori.



(a) Obiettivi



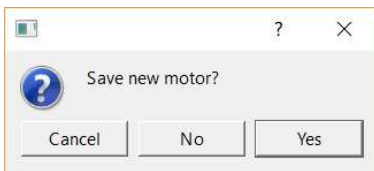
(b) Numero di elementi

Figura 2.5: OpenSave: Modifiche dei parametri di ottimizzazione

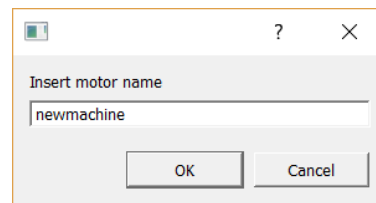
Il processo di modifica vero e proprio termina con l'ultima finestra di dialogo (2.5) che permette invece di impostare il numero di generazioni ed il numero di elementi valutati per ogni generazione: più tali parametri saranno elevati, tanto più il processo di ottimizzazione prenderà in considerazione un numero maggiore di elementi, e di conseguenza maggiori saranno le possibilità di individuare soluzioni migliori.

Ovviamente più ampia sarà la ricerca maggiore sarà l'onere computazionale richiesto, con conseguente aumento considerevole dei tempi di elaborazione.

A questo punto una nuova finestra, riportata in figura 2.6, invita l'utente al salvataggio delle modifiche apportate e, nel caso si decidesse di proseguire, è necessario inserire il nome da attribuire al nuovo progetto: viene suggerito il generico *newmachine*, ma è possibile decidere di sovrascrivere i dati aperti inizialmente.



(a)



(b)

Figura 2.6: OpenSave: Salvataggio

Che si decida o meno di salvare, il processo termina con la visualizzazione della macchina appena definita in una nuova finestra dell'ambiente di lavoro: l'esecuzione è immediata in Matlab, mentre richiede un tempo di elaborazione maggiore in Octave.

È consigliabile lasciar lavorare il programma fino alla visualizzazione completa.

In figura 2.7 è possibile vedere l'output della macchina di default *mot_01* al termine dell'esecuzione.

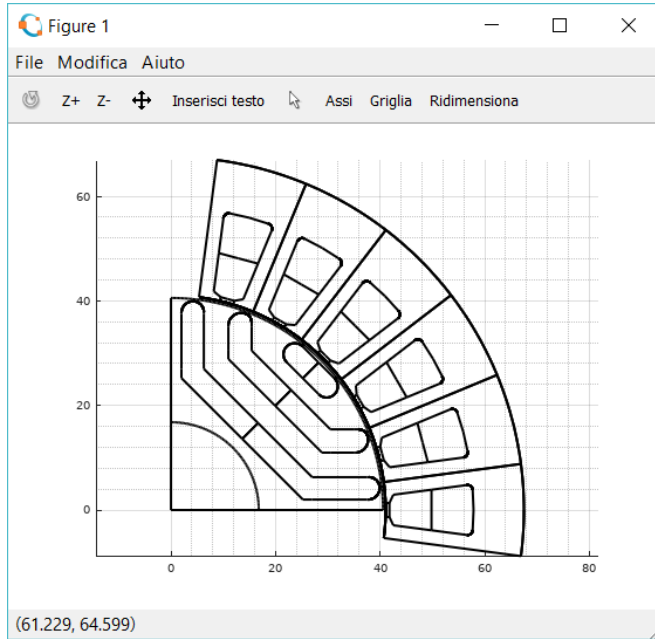


Figura 2.7: *OpenSave: Visualizzazione della macchina*

2.3 Processo di ottimizzazione

Il passo successivo alla creazione di uno strumento multiplatforma che permettesse la modifica dei progetti offline consisteva nel rendere disponibile allo stesso modo lo strumento di ottimizzazione: SyR-e permetteva già di avviare tale funzionalità al di fuori della GUI attraverso il file *MODEstart.m*, previa modifica delle righe iniziali di codice in modo che venisse caricato il file desiderato.

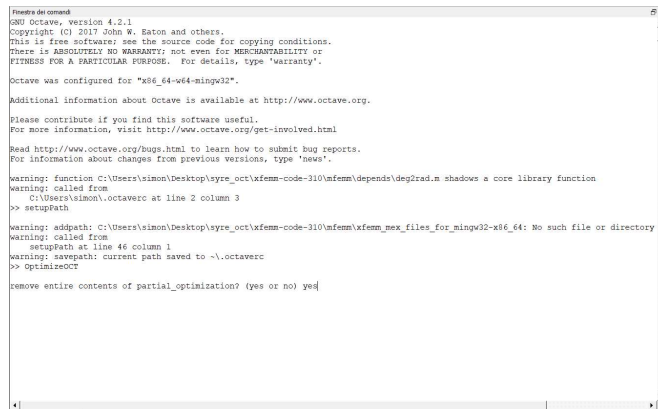
A partire dal codice inizialmente disponibile si è quindi proceduto all'analisi ed al debug di *MODEstart.m* e di tutte le funzioni Matlab ad esso connesse: si è deciso di sostituire la versione iniziale e rinominare il nuovo file *Optimize.m*, presente all'interno della directory di SyR-e.

Una delle azioni consigliate prima di avviare qualunque processo di analisi, ottimizzazione o post processing che sia, è lo svuotamento della cartella *tmp*, nella quale vengono salvate tutte le informazioni relative alle macchine simulate

all'interno di cartelle dedicate il cui nome è generato in maniera casuale: nel caso di ottimizzazioni con popolazioni molto estese il numero di tali elementi può essere veramente rilevante, per evitare quindi conflitti fra i nomi delle cartelle ed in generale di appesantire eccessivamente il disco si consiglia di svuotarla periodicamente.

Avviando lo script appare la finestra di dialogo per la scelta della macchina da ottimizzare, in maniera del tutto simile a quella riportata in figura 2.2.

Una volta avvenuta la selezione in Octave è necessario spostarsi nella finestra dei comandi in quanto viene richiesto all'utente se si intende cancellare i file contenuti nella cartella *partial_optimization*. In tale destinazione vengono infatti salvati parte dei dati relativi al processo di ottimizzazione, che permangono al termine della stessa: è consigliabile eseguire questa procedura digitando la risposta affermativa *yes*.



```

Finestra dei comandi
GNU Octave, version 4.2.1
Copyright (C) 2017 John W. Eaton and others.
This is free software: see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-w64-mingw32".
Additional information about Octave is available at http://www.octave.org.
Please contribute if you find this software useful.
For more information, visit http://www.octave.org/get-involved.html
Read http://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

warning: function C:\Users\simon\Desktop\lyre_oct\xfemm-code-310\mfemm\depends\deg2rad.m shadows a core library function
warning: called from
  C:\Users\simon\.octaverc at line 2 column 3
>> setupPath

warning: addpath: C:\Users\simon\Desktop\lyre_oct\xfemm-code-310\mfemm\xfemm_mex_files_for_mingw32-x86_64: No such file or directory
warning: called from
  setupPath at line 46 column 1
warning: savepath: current path saved to ~\.octaverc
>> OptimizeOCT

remove entire contents of partial_optimization? (yes or no) yes
  
```

Figura 2.8: Optimize: Richiesta di svuotamento della cartella *partial_optimization*

Una volta effettuata la scelta avrà inizio il processo di ottimizzazione: a partire dal progetto selezionato ed in base ai parametri impostati, l'algoritmo estrae in maniera casuale, all'interno dei range di variabilità specificati nel processo di modifica, i valori delle variabili di ottimizzazione da utilizzare. Successivamente definisce la macchina all'interno di FEMM e simula il comportamento elettromagnetico della stessa per un numero di posizioni definite andando a modificare di volta in volta la posizione angolare del rotore.

Nel mentre all'interno della finestra dei comandi sono riepilogate le informazioni relative al processo in corso, nello specifico vengono stampate a schermo il numero della generazione che sta eseguendo l'algoritmo, la dimensione del fronte di Pareto ed il tempo di fine valutazione, mentre per quanto riguarda la singola

macchina vengono riportate le variabili di ottimizzazione generate casualmente raccolte nel vettore RQ e le note riguardanti la realizzabilità geometrica della macchina (eventuali conflitti tra barriere e/o tolleranze meccaniche non rispettate che l'algoritmo provvede a correggere).

Un esempio delle informazioni mostrate è riportato in figura 2.9.

```

#bests decreased
#3 spider is too thin
#10xc inner dx modified for prevent exit of the barrier from pole
#20c 1 layer exit from the rotor
Evaluating 10/10 solution.
RQ =
    0.39659    0.17000    0.49431    0.40664    0.94952    0.65408    0.10172    -0.41472    -0.12520

#1 conflict between two barrier
#3 spider is too thin
#10xc inner dx modified for prevent exit of the barrier from pole
-----
Generation: 2
PES: 30
Pareto Front Size: 4
Evaluation Time: 0 h 3 min 14 sec
End of evolution process: 16-Feb-2018 11:41:19
-----
Evaluating 3-th generation...
-----
Evaluating 1/10 solution.
RQ =
    0.48645    0.31149    0.47512    0.66284    0.20000    0.45656    0.20186    -0.57107    -0.56941

#3 spider is too thin
#10xc inner dx modified for prevent exit of the barrier from pole
Evaluating 2/10 solution.
RQ =
    0.26144    0.22021    0.26501    0.57992    0.73231    0.53578    0.11075    0.24985    -0.14838

```

Figura 2.9: Optimize: Esempio di informazioni visualizzate durante il processo

Il processo di ottimizzazione si compone di due fasi: la prima è quella propriamente di ottimizzazione, nella quale avviene la generazione delle popolazioni attraverso le operazioni di *mutazione* e *cross over* e la simulazione dei singoli elementi delle stesse; nella seconda viene invece operata una rivalutazione più accurata delle macchine appartenenti ai fronti di Pareto delle singole generazioni sulla base della quale avviene il salvataggio e la visualizzazione dei risultati del processo.

Nello specifico viene visualizzato l'insieme delle soluzioni migliori trovate ed il fronte di Pareto globale che determinano, un riepilogo degli obiettivi delle singole macchine, ordinate generalmente in ordine decrescente, ed un numero di figure pari a quello delle variabili di ottimizzazione selezionate, che ne descrivono quindi le caratteristiche geometriche.

I risultati, compresi i file *.fem* e *.mat* associati alle macchine identificate, vengono salvati all'interno della cartella *results* con il nome convenzionalmente ottenuto dalla data e dall'orario del termine del processo: il file omonimo contiene invece il riepilogo del processo, in termini sia di impostazioni preliminari che di risultati, mentre all'interno della cartella corrispondente sono presenti i grafici ed i dettagli delle macchine.

Così come anticipato nella parte introduttiva, per mantenere una continuità con

il codice già presente si è data priorità, nella scelta del formato delle figure da adottare, alla modifica delle stesse: in tal senso il salvataggio dei grafici in ambiente Octave con estensione *.fig* non ne permetteva la successiva riapertura. Per tale motivo è stato identificato il formato più consono *OFIG*, che invece consente di visualizzare e modificare liberamente tramite il comando `hgload('filename')`.

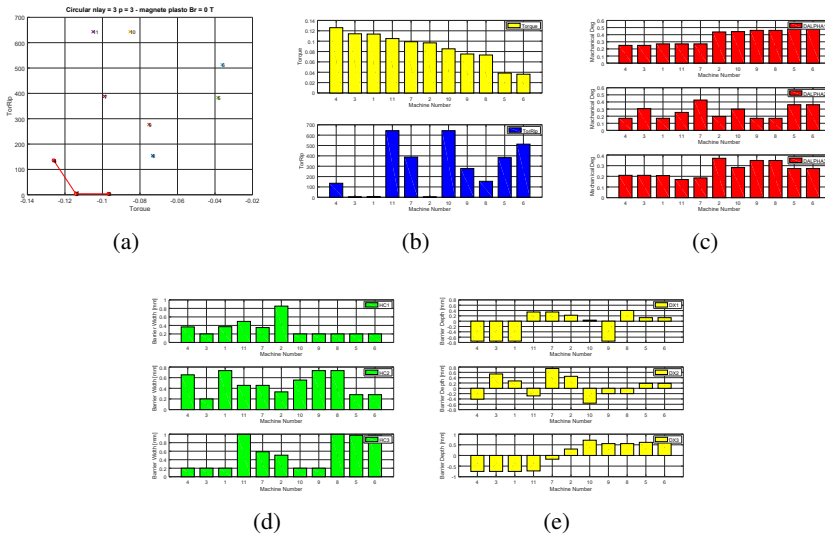


Figura 2.10: Optimize: Esempio di output in Octave

2.4 Post processing

L'ultima funzionalità da implementare in Octave risultava essere quella del post processing, che si occupa di analizzare il comportamento elettromagnetico e meccanico di una singola macchina: questo è infatti il passaggio logicamente successivo al processo di ottimizzazione, che evidenzia le soluzioni migliori ottenute attraverso una ricerca stocastica all'interno dello spazio globale delle soluzioni possibili ma che non le caratterizza se non in quanto agli obiettivi fissati, in genere rappresentati dal valore e dal ripple picco-picco di coppia, ed alle variabili di ottimizzazione.

Il post processing ha invece proprio la funzione di analizzare il comportamento di una macchina, in genere scelta tra quelle appartenenti al fronte di Pareto, in punti di lavoro specifici e permette quindi di valutarne le caratteristiche dei flussi magnetici e delle grandezze meccaniche.

Nella versione di SyR-e oggetto della tesi era già possibile avviare il post processing attraverso la funzione *post_proc_single_motor.m*, presente all'interno della cartella *mfiles*: tale file è stato preso come base per procedere all'implementazione in Octave e si è deciso di non modificarne il nome in quanto già utilizzato per la stessa funzione all'interno della GUI.

Dopo aver avviato lo script viene richiamata una finestra di selezione che permette di scegliere il motore da caratterizzare (2.2): vengono quindi caricati i dati relativi al progetto selezionato ed è possibile procedere, in maniera simile all'interfaccia grafica grazie alla schemata di input 2.11, con la modifica dei seguenti parametri:

1. il modulo della corrente in assi rotanti, espressa in *per unit*
2. la fase γ del vettore corrente in assi rotanti
3. l'induzione magnetica dei magneti, se presenti
4. il numero di posizioni angolari del rotore da simulare
5. la porzione di macchina da simulare, espressa in angoli meccanici
6. il numero dei valori di corrente da simulare nell'intervallo specificato
7. l'opzione di valutazione delle perdite nel ferro

The screenshot shows a window titled "Edit parameters" with a question mark icon and a close button. It contains the following fields:

- current load [p.u.]: [1 1 1 1 1 1 1]
- gamma [deg]: [40 45 50 55 60 65 70 75]
- Br [T]: 0
- number of rotor positions: 6
- angular span (elt. deg.): 60
- points in [0 I_{max}]: 5
- Evaluate loss? [1 - Yes, 0 - No]: 0

At the bottom, there are "OK" and "Cancel" buttons.

Figura 2.11: *post_proc_single_motor*: Modifica dei parametri principali

Come anticipato nella sezione descrittiva delle funzionalità di SyR-e, la funzione di post processing mette a disposizione tre tipi di analisi differenti; nel caso si vogliano valutare i punti di lavoro della macchina è necessario immettere in maniera opportuna i valori di corrente: nello specifico, tali parametri vengono gestiti come elementi vettoriali e risulta quindi necessario che le loro dimensioni coincidano. L'algoritmo infatti associa ogni elemento del vettore modulo di

corrente con il corrispettivo elemento del vettore γ che occupa la medesima posizione.

Un esempio dei risultati ottenuti su Octave a partire dalla macchina numero 3 dell'ottimizzazione i cui risultati sono riportati in figura 2.10 impostando la corrente nominale e $\gamma = 50^\circ$ sono mostrati nella successiva 2.12.

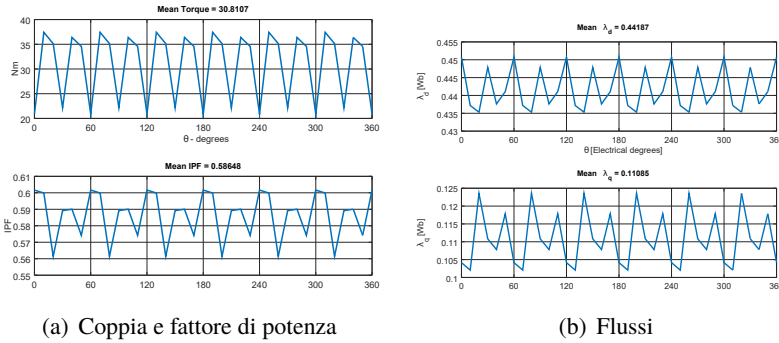


Figura 2.12: *post_proc_single_motor*: Esempio dei risultati del singolo punto di lavoro

Per procedere con l'analisi magnetica della macchina al fine di ottenere il modello completo è necessario specificare il valore massimo di corrente da simulare ed impostare un valore di γ convenzionalmente pari a 1000: in questo caso la macchina viene simulata sulla base di differenti combinazioni di modulo e fase, il risultato di tale operazione sono curve di flussi e coppia parametrizzate su una griglia di valori di corrente.

2.5 Risultati

2.5.1 Ottimizzazione

Dopo aver presentato le modifiche introdotte nell'applicativo e le procedure dettagliate per accedere alle funzionalità all'esterno della GUI risulta necessario verificare le prestazioni del processo di ottimizzazione e del post processing, in particolare nel nuovo ambiente di lavoro: al fine di ottenere risultati significativi il progetto di partenza è stato individuato nel modello del prototipo *RawPower* macchina realizzata per il dipartimento a partire proprio dalle ottimizzazioni e le analisi svolte attraverso SyR-e.

Il prototipo benchmark è una macchina sincrona a riluttanza a 6 poli che presenta a rotore 3 barriere di flusso per polo, ciascuna a geometria circolare (tipologia *Circ*

all'interno di SyR-e), ed uno statore a 36 cave all'interno delle quali è collocato un avvolgimento trifase doppio strato a passo intero: in figura 2.13 è riportata la sezione del lamierino della macchina.

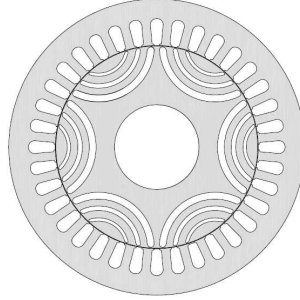


Figura 2.13: Sezione del prototipo RawPower

Si è proceduti con l'esecuzione di tre processi di ottimizzazione, due all'interno dell'ambiente Matlab attraverso l'interfaccia grafica e lo script *Optimize* ed uno in Octave, con gli stessi parametri di ottimizzazione: nello specifico, affinché lo spazio delle soluzioni venisse esplorato in maniera ampia dall'algoritmo, il numero di generazioni e di elementi per la singola generazione sono stati fissati pari a 60, mentre si è deciso di penalizzare tutte le macchine che presentassero coppia minore di 70 Nm.

Sono stati registrati i tempi di elaborazione complessiva, in modo da evidenziare quanto l'assenza del parallel processing su Octave influenzasse la durata dei processi: si tenga presente che tutte e tre le ottimizzazioni sono state eseguite sullo stesso PC equipaggiato con un processore a due core fisici.

I risultati principali sono riassunti nella tabella 2.1.

	<i>Matlab GUI</i>	<i>Matlab Script</i>	<i>Octave</i>
<i>Tempo di elaborazione</i>	13:56:13	13:23:38	23:12:47
<i>Soluzioni identificate</i>	80	74	56
<i>Dimensione fronte di Pareto</i>	5	11	7
<i>Coppia massima [Nm]</i>	75.58	76.93	77.20

Tabella 2.1: Riepilogo dei risultati delle ottimizzazioni

I fronti di Pareto ottenuti sono riportati nelle figure 2.14, 2.15 e 2.16: pur tenendo presente che l'algoritmo genetico di ottimizzazione procede in maniera casuale all'interno dello spazio delle soluzioni, si può notare come nel caso dei processi eseguiti in ambiente Matlab, pur essendo state analizzate lo stesso numero di soluzioni, il numero di macchine non scartate sia maggiore.

Si noti però come il numero di elementi appartenenti ai singoli fronti di Pareto non dipende dal numero di soluzioni analizzate: nonostante nel caso dei processi eseguiti in ambiente Matlab siano state individuate un numero di macchine maggiore, il fronte di quello eseguito tramite GUI conta il numero di elementi minore: ciò ad ulteriore conferma della natura casuale dell'algoritmo di ottimizzazione.

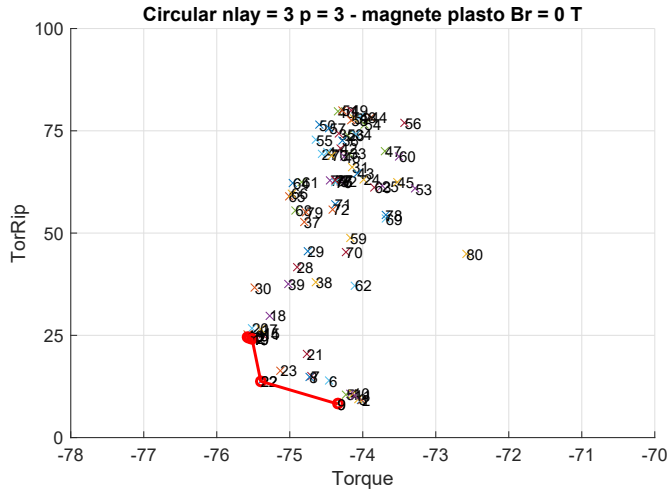


Figura 2.14: Matlab GUI: fronte di Pareto dell'ottimizzazione

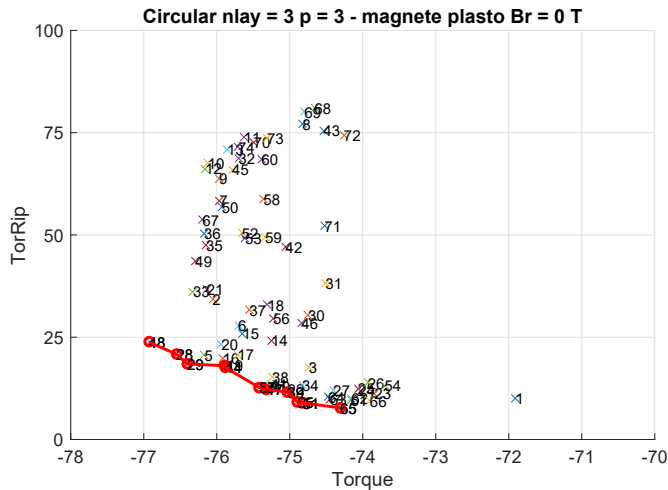


Figura 2.15: Matlab script: fronte di Pareto dell'ottimizzazione

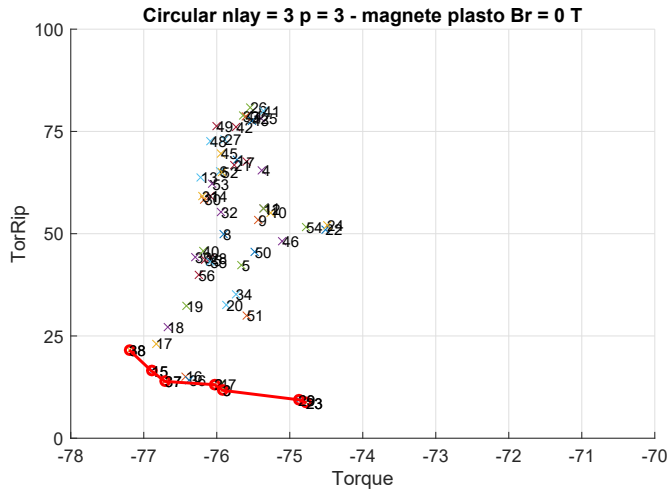


Figura 2.16: Octave: fronte di Pareto dell'ottimizzazione

Nelle figure 2.17, 2.18 e 2.19 sono invece riassunti gli obiettivi delle ottimizzazioni, che nel caso in questione riguardavano il valori di coppia in termini di valor massimo e ripple picco-picco: come si può notare, in tutte e tre le analisi la soluzione migliore per quanto riguarda l'obiettivo di coppia massima è risultato essere maggiore del valore di 75 Nm; si noti come, avendo impostato come obiettivo un valore minimo di coppia atteso pari a 70 Nm siano state scartate tutte le soluzioni che presentavano coppie minori od uguali.

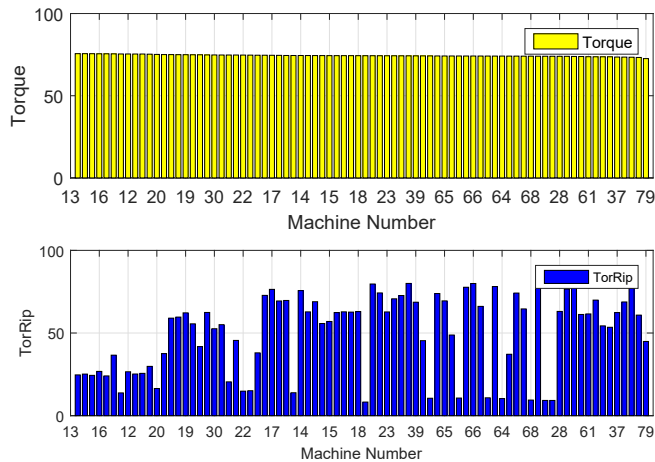


Figura 2.17: Matlab GUI: obiettivi dell'ottimizzazione

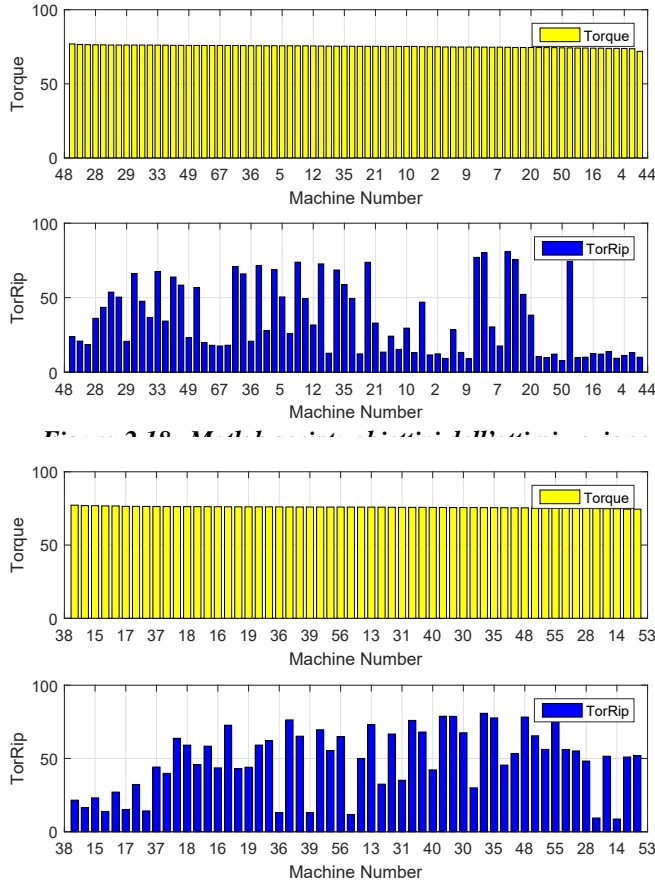


Figura 2.19: Octave: obiettivi dell'ottimizzazione

2.5.2 Post processing

Analizzati i risultati del processo di ottimizzazione ci si è dedicati alla verifica del corretto funzionamento della funzione di *post processing* nei due ambienti di calcolo.

Come primo passo è stata selezionata una fra le soluzioni migliori appartenenti ai tre fronti di Pareto individuati: sulla base della similitudine geometrica con il prototipo è stata individuata la macchina *mot_22* dal processo di ottimizzazione avviato tramite interfaccia grafica in Matlab.

La sezione ed i parametri geometrici principali della macchina scelta sono riportati di seguito in 2.21.

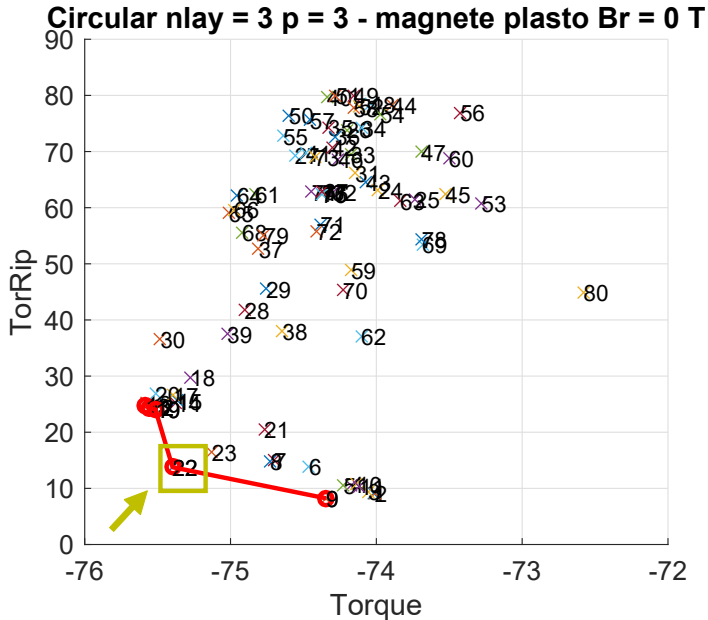


Figura 2.20: Selezione della macchina nel fronte di Pareto

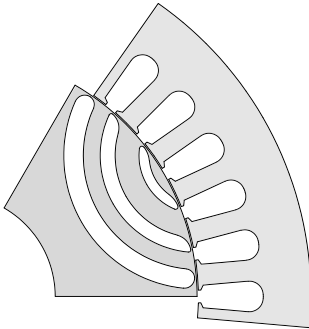


Figura 2.21: Sezione della macchina selezionata

Variabili ottimizzate	
alpha [p.u.]	[0.28 0.35 0.22]
hc [p.u.]	[0.27 0.4 0.6]
dx [p.u.]	[0.15 0.6 -0.19]

Per ottenere risultati confrontabili in maniera significativa i parametri inseriti per le elaborazioni sono stati i medesimi: si è proceduto infatti con la valutazione del comportamento della macchina in un punto specifico, caratterizzato dalla corrente nominale i_0 e dalla fase scelta arbitrariamente $\gamma = 50^\circ$.

L'analisi del singolo punto di lavoro ha permesso di ottenere gli andamenti dei flussi in asse diretto e di quadratura, oltre che quelli relativi alla coppia meccanica ed al power factor.

In figura 2.22 sono riportati i risultati ottenuti dall'analisi descritta in precedenza

nelle tre casistiche differenti, eseguite impostando un numero di posizioni angolari pari a 30: come si può notare immediatamente, le forme d'onda di coppia e fattore di potenza rispetto all'angolo meccanico sono perfettamente sovrapponibili.

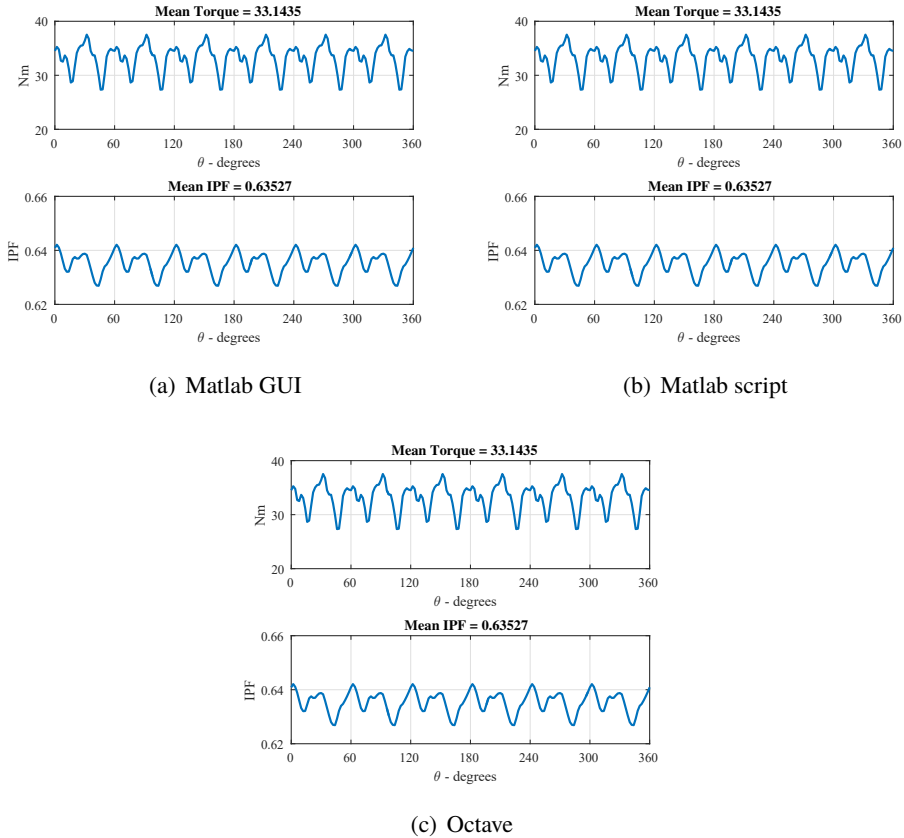
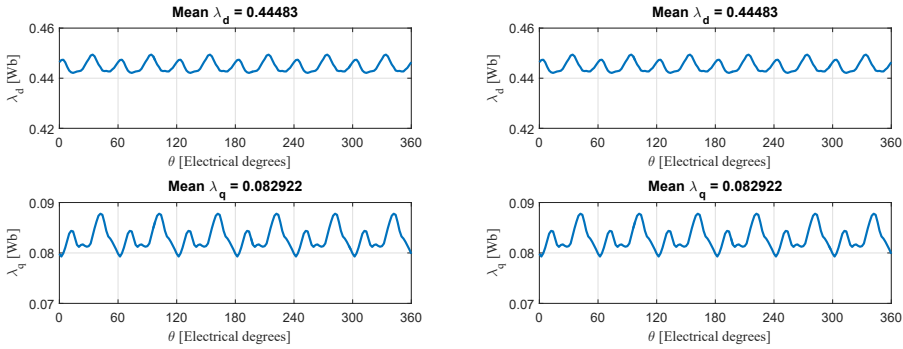


Figura 2.22: Post processing: risultati di coppia e fattore di potenza

Discorso simile vale per gli andamenti dei flussi in asse diretto e di quadratura (figura 2.23), dove si nota l'esatta corrispondenza sia negli andamenti sia per quanto riguarda i valori medi della coppia e del fattore di potenza.

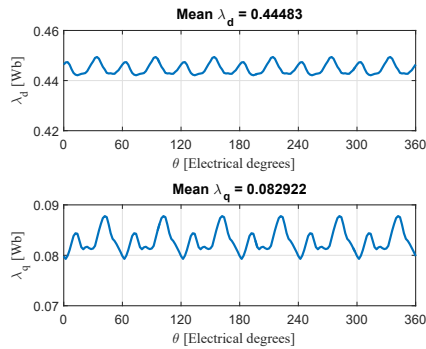
Successivamente si è passati all'elaborazione del modello magnetico della macchina, sono stati quindi impostati una corrente massima pari a due volte quella nominale ed un numero di punti nell'intervallo $[0 \ 2i_0]$ di 10: in questo modo è stato possibile testare la macchina selezionata su una griglia sufficientemente fitta. In analogia con quanto fatto per l'ottimizzatore sono stati registrati i tempi di

completamento dell'elaborazione.



(a) Matlab GUI

(b) Matlab script



(c) Octave

Figura 2.23: Post processing: risultati dei flussi λ_d e λ_q

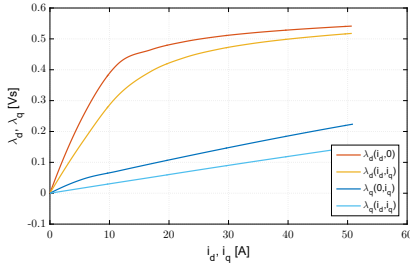
Premesso che il PC su cui sono stati eseguiti i processi risultava lo stesso del caso precedente con un processore a due core fisici, i risultati registrati in termini di velocità di elaborazione sono in linea con quanto già evidenziato per i processi di ottimizzazione: l'assenza del calcolo parallelo fa sì che l'esecuzione in Octave richieda poco meno del doppio dei tempi di elaborazione in Matlab (2.2).

	<i>Matlab GUI</i>	<i>Matlab Script</i>	<i>Octave</i>
<i>Tempo di elaborazione</i>	00:22:01	00:21:30	00:37:19

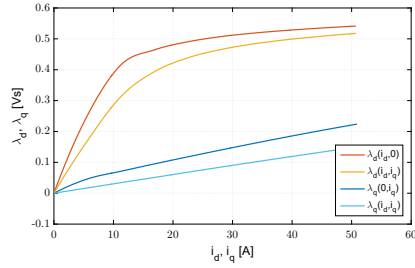
Tabella 2.2: Tempi di elaborazione post processing

In figura 2.24 sono riportati gli andamenti dei flussi di macchina rispetto alle correnti in asse diretto e di quadratura: anche in questo caso si può constatare

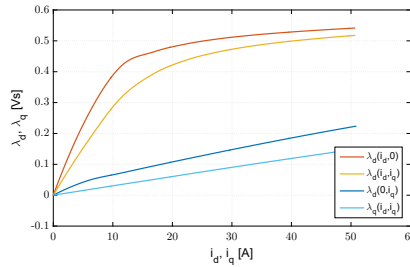
come non vi siano apprezzabili differenze fra i risultati ottenuti nelle tre modalità di analisi eseguite.



(a) Matlab GUI



(b) Matlab script



(c) Octave

Figura 2.24: Post processing: risultati delle caratteristiche flusso-corrente

2.6 Conclusioni

I risultati presentati hanno permesso di dimostrare come il codice sia stato armonizzato.

L'analisi della funzionalità di ottimizzazione ha evidenziato come i risultati dipendano da due fattori principali: la quantità di soluzioni analizzate ed ovviamente la casualità del processo, che determina le caratteristiche degli elementi generati; idealmente se avessero esplorato tutto lo spazio delle soluzioni i tre processi di ottimizzazione sarebbero converiti all'ottimo globale.

Il post processing ha invece dimostrato di produrre gli stessi risultati in tutte e tre le condizioni di verifica: l'unica nota, comune anche al caso precedente, riguarda i tempi di elaborazione, più elevati in Octave a causa della mancanza del calcolo parallelo.

Capitolo 3

Macchine multifase

3.1 Introduzione

La seconda parte dell'elaborato è stata invece incentrata sull'implementazione di una nuova famiglia di macchine, la revisione 312 di SyR-e infatti, sulla quale è stato svolto tutto il lavoro, permette di progettare macchine molto differenti fra loro per geometrie, materiali ma soprattutto per tipologia di rotore: allo stato attuale è possibile scegliere fra cinque diverse geometrie, delle quali è possibile definire nel dettaglio i parametri fondamentali, dal numero di polarità alla quantità ed alla forma delle singole barriere di flusso.

Lo statore può anch'esso essere di fatto caratterizzato in maniera molto approfondita, in particolare per quanto riguarda le variabili dimensionali, quali lo spessore del giogo o la forma delle cave.

La finestra *Windings* è dedicata interamente alla definizione dell'avvolgimento statorico ed i parametri sui quali l'utente può agire sono il fattore di riempimento di cava, definito come rapporto fra la sezione effettiva di rame e la sezione complessiva di cava, che tiene quindi conto della presenza dell'isolamento e del non perfetto sfruttamento dello spazio disponibile, il numero di spire in serie ed il fattore di raccorciamento nel caso si volesse definire un avvolgimento a passo raccorciato. Ulteriore possibilità è quella di inserire all'interno del progetto un avvolgimento concentrato, cui si riferisce tipicamente con l'espressione *a bobina di dente* (dall'inglese *tooth coil*), caratterizzato da un numero di cave/polo/fase frazionario: ciò è possibile andando ad immettere un valore di *geo.q* adeguato nella schermata iniziale della GUI e spuntando l'opzione dedicata nella finestra dell'avvolgimento.

Tutte le opzioni disponibili si riferiscono ad avvolgimenti trifase, che rappresentano lo standard in ambito industriale per quanto riguarda le macchine elettriche; in realtà nel corso degli scorsi decenni un'altra tipologia di macchine ha iniziato ad essere oggetto di ricerche sempre più approfondite: le macchine multifase rappresentano infatti una valida alternativa a quelle trifase a causa dei vantaggi che presentano in termini di tolleranza ai guasti ed efficienza.

Per tale ragione si è quindi deciso di implementare all'interno del codice di SyR-e la possibilità di analizzare una famiglia di macchine multifase particolarmente interessanti, quelle *multit trifase asimmetriche*.

Nel seguito del capitolo verranno presentate le caratteristiche delle macchine citate, descritte le modifiche apportate all'applicativo ed infine verranno mostrati i risultati relativi al confronto fra il prototipo introdotto nel capitolo precedente ed un suo corrispettivo multit trifase.

3.2 Macchine multifase: generalità

Con l'espressione multifase si intendono identificare in maniera del tutto generale tutte quelle macchine elettriche il cui numero di fasi è maggiore di tre, che allo stato dell'arte rappresenta il riferimento per l'industria ed ha permesso di standardizzare la produzione ed il costo della macchine trifase.

Come riportato in [4], l'interesse per le macchine multifase, sebbene note già dalla fine degli anni Sessanta, è esploso a partire dall'ultimo decennio del secolo scorso sulla spinta di tre precisi ambiti: la propulsione navale, la trazione e lo sviluppo del concetto di More Electric Aircraft in ambito aeronautico; ciò è dovuto principalmente a tre ragioni:

1. l'eccitazione dello statore produce un campo con ridotto contenuto di armoniche spaziali nelle macchine multifase rispetto a quelle trifase, che si traduce in maggior efficienza;
2. le macchine multifase hanno una tolleranza al guasto più elevata rispetto a quelle trifase: nel caso di guasto su una fase, le macchine trifasi possono funzionare ma non sono più in grado di avviarsi, cosa invece non vera nel caso delle macchine multifase;
3. in generale le macchine multifase sono meno sensibili alla presenza di armoniche nelle forme d'onda di alimentazione.

Le macchine multifase vengono classificate in base alla posizione angolare delle fasi adiacenti [5]: si parla infatti di *macchine simmetriche* (figura 3.1) quando, essendo n_{ph} il numero di fasi complessivo della macchina, gli assi magnetici di due fasi adiacenti sono sfasate di $\frac{360}{n_{ph}}$ gradi elettrici.

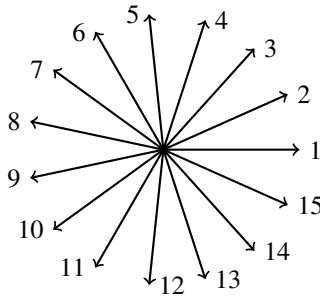


Figura 3.1: 15 fasi simmetrico

Quando invece la macchina presenta un numero di fasi pari oppure un numero di fasi dispari che non sia un numero primo, le fasi possono essere raggruppate e la macchina può essere vista come composta da un certo numero di set: in questo caso si parla di *macchine asimmetriche* (3.2) e due fasi consecutive sono distanziate di un angolo pari a $\frac{180}{n_{ph}}$ gradi elettrici.

Avendo come riferimento il mondo trifase, la cosa più naturale è considerare la macchina composta da un certo numero di terne, ma nulla vieta di comporre le fasi in sets da 5, 7, 11 ecc...

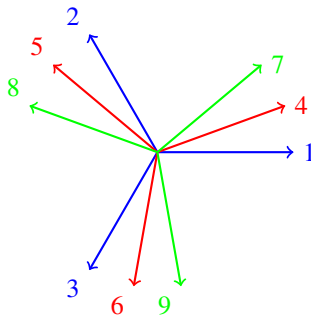


Figura 3.2: 9 fasi asimmetrico

Le differenti configurazioni presentate permettono lo sviluppo di altrettante strutture di azionamenti differenti [5], ognuna caratterizzata da proprietà precise: nel caso di macchine simmetriche è possibile alimentare le singole fasi attraverso un unico inverter a pari numero di fasi (figura 3.3(a)) oppure tramite convertitori dedicati, come mostrato in 3.3(b); tra le due configurazioni proposte, solo la

seconda soddisfa quel criterio di *fault tolerance* che rende vantaggioso l'impiego delle macchine multifase nel caso di guasto al sistema di alimentazione.

Le macchine multifase asimmetriche permettono invece l'adozione di più inverter multifase indipendenti dello stesso tipo (trifase in 3.3(c)) in quanto lo statore è composto da più terne connesse in configurazione a neutro isolato: sebbene rispetto alla configurazione ad inverter monofase indipendenti presenti un livello di affidabilità minore, questa struttura permette l'utilizzo di una tecnologia consolidata come quella degli inverter trifase e ne eredita le proprietà anche per quanto riguarda l'ambito controllistico.

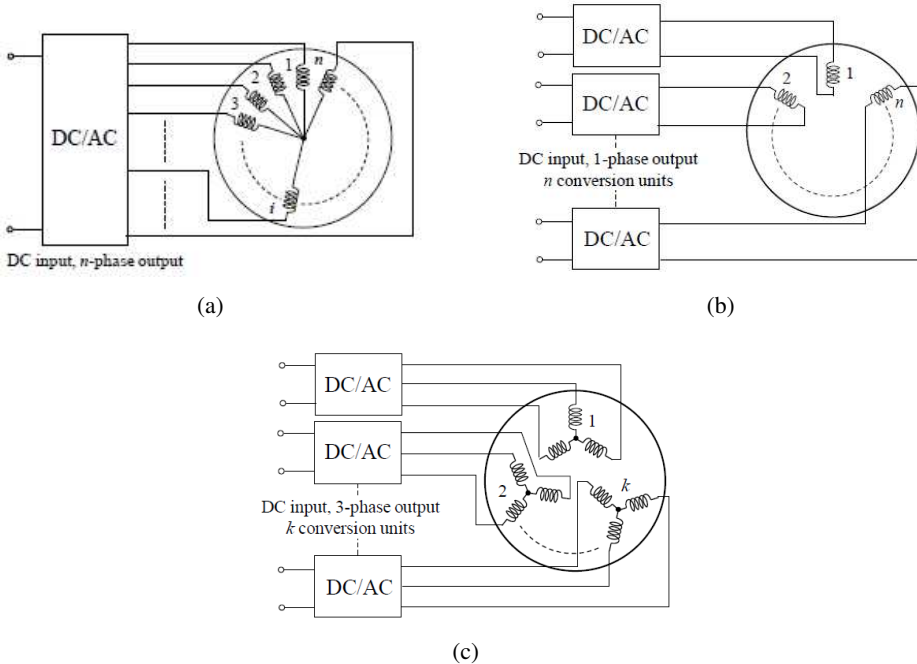


Figura 3.3: Azionamenti multifase: possibili configurazioni[5]

3.3 Implementazione

Come esposto in precedenza, sebbene SyR-e permetta di progettare lo statore ed il corrispettivo avvolgimento in molti dei suoi parametri, in particolare geometrici, ma di fatto l'unica famiglia di avvolgimenti disponibile era quella trifase: tutte le

singole parti che componevano il codice, sia quelle che implementavano funzioni di calcolo che quelle relative alla comunicazione con programmi esterni, sono state dunque scritte partendo da questo assunto.

Volendo lavorare su questo aspetto dell'applicativo ed avendo come riferimento un prototipo a due cave/polo/fase, la tipologia di macchine che risulta naturale derivare è quella *multitrifase*, configurazione multifase asimmetrica nelle quali le fasi, multiple ovviamente di 3, vengono raggruppate in terne e collegate in maniera indipendente ognuno al proprio sistema di alimentazione: nel caso del prototipo in questione, per passare ad una configurazione *bitrifase* sarebbe stato sufficiente riavvolgere la macchina con due terne opportunamente disposte in cava in modo da ottenere una macchina ad una cava/polo/fase.

Nel seguito di questa sezione verranno presentati nel dettaglio gli aspetti teorici del problema, ponendo particolare attenzione ad alcuni elementi di calcolo ed alla questione relativa alle trasformazioni di riferimento, fondamentali per muoversi in maniera bidirezionale dal riferimento del controllo in assi rotanti a quello delle grandezze di fase necessarie per la definizione del problema agli elementi finiti.

3.3.1 Parametri aggiuntivi e modifiche alla GUI

La prima informazione da introdurre all'interno del codice risultava essere quella relativa al numero di terne in cui suddividere l'avvolgimento, per fare ciò è stato quindi introdotto il nuovo campo *Num3PhaseCircuit* all'interno della struttura *dataSet*: quest'ultima risultava essere infatti il punto di partenza per la definizione delle caratteristiche del progetto.

Nello specifico, i valori di *dataSet* possono essere modificati unicamente attraverso l'interfaccia grafica oppure gli strumenti presentati nel capitolo precedente ed elaborati dalla funzione *data0*, che si occupa dell'aggiornamento delle variabili utilizzate durante gli strumenti di analisi e simulazione di SyR-e: questi infatti sono organizzati in strutture specifiche che contengono i parametri geometrici e le opzioni di analisi (struttura *geo*), i valori relativi ai limiti delle variabili di ottimizzazione necessari per limitare lo spazio delle soluzioni (struttura *bounds*), le prestazioni obiettivo dell'ottimizzazione (struttura *per*) ed infine le proprietà relative ai materiali (struttura *mat*).

L'interfaccia grafica in Matlab è stata quindi modificata di conseguenza (figura 3.4): nell'area in basso a destra, al di sopra della tabella contenente lo schema d'avvolgimento è stato inserito un nuovo campo che permette di modificare il numero di terne selezionate; come verrà chiarito in seguito, allo stato attuale è necessario modificare manualmente i valori della tabella d'avvolgimento in

quanto la versione attuale di Koil, su cui l'applicativo si appoggia per la verifica di fattibilità e lo schema di avvolgimento, non è in grado di gestire tale casistica.

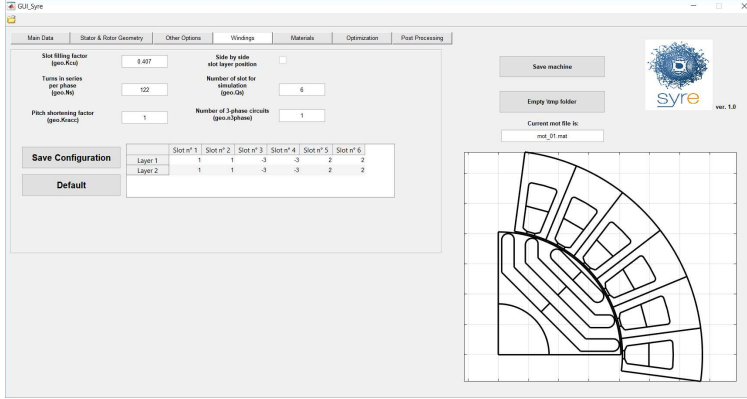
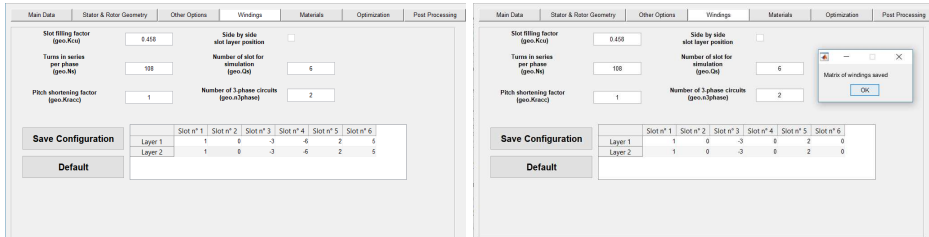


Figura 3.4: GUI: nuova finestra Windings

Con l'introduzione dell'opzione multitrifase si è pensato di aggiungere un'ulteriore funzionalità: oltre alla simulazione della macchina completa di tutte le sue terne è possibile disattivarne una o più in modo tale da analizzare il comportamento della macchina in condizione di guasto.

Per fare ciò si deve sostituire il numero di una delle fasi della terna che si vuole scollegare con il valore predefinito 0: una volta effettuata la sostituzione, cliccando sul pulsante *Save Configuration* le altre fasi appartenenti alla stessa terna verranno automaticamente disabilite, come si può notare dal valore nullo che assumono dopo il salvataggio.



(a) Sostituzione fase 4 con 0

(b) Salvataggio: seconda terna disattivata

Figura 3.5: Parametri per la simulazione di condizioni di guasto

Questo tipo di modifica è stato possibile inserendo una nuova variabile, chiamata *dataSet.WinFlag*, vettore di lunghezza pari al numero di fasi dell'avvolgimento, i cui elementi sono variabili binarie associate allo stato della singola fase: per simulare un guasto al sistema di alimentazione della singola terna, gli elementi

della variabile corrispondenti alle fasi della stessa assumono il medesimo valore. La scelta di una variabile di tale dimensione non è casuale e torna utile nel momento della definizione del problema all'interno di FEMM, in particolare nell'assegnazione delle correnti ai circuiti elettrici.

L'introduzione dell'opzione di disabilitazione delle terne richiedeva uno strumento che permettesse di ritornare indietro alla configurazione iniziale, pertanto si è deciso di introdurre un nuovo pulsante, aggiunto al di sotto di *Save Configuration*: in seguito alla modifica della disposizione delle fasi in cava è sufficiente premere il nuovo pulsante *Default* per reimpostare la configurazione iniziale con tutte le fasi attive.

Per implementare tale funzionalità è stato necessario aggiungere il nuovo campo *DefaultWinMatr* al *dataSet*, in modo tale che il corrispettivo *geo.defaultavv* non fosse interessato dalle modifiche apportate alla matriche dell'avvolgimento tramite la finestra della GUI, che invece manipola la variabile *geo.avv*, e potesse essere richiamato in base alla necessità.

3.3.2 Calcolo e definizione delle proprietà circuitali

Il passo successivo consisteva nell'analizzare le grandezze di macchina per individuare quali ed in quale maniera fossero influenzate dalla nuova configurazione: una volta ottenute le relazioni con i nuovi parametri sarebbe stato possibile dedicarsi all'implementazione di queste ultime all'interno del codice.

L'operazione di riavvolgimento secondo uno schema multitrifase ha una prima evidente conseguenza: a parità di paia poli infatti il numero di cave/polo/fase diminuisce in maniera inversamente proporzionale al numero di terne totali. Una macchina a 6 poli con statore a 36 cave, come il prototipo benchmark, presentava un numero di cave/polo/fase q pari a 2 se l'avvolgimento statorico è di tipo trifase, ma tale valore si dimezzava nel caso in cui venisse avvolto secondo uno schema bitrifase, in quanto per ogni polarità ognuna delle fasi occupa esattamente la metà delle cave della configurazione precedente.

Da questa deriva la seguente considerazione, che ha condotto alla prima scelta progettuale: avendo ogni fase a disposizione un numero minore di cave, se la sezione dei conduttori fosse rimasta invariata il numero di spire in serie per fase N_s avrebbe dovuto anch'esso ridursi nella stessa maniera del numero di q , ossia in maniera inversamente proporzionale al numero di terne.

Ciò però avrebbe voluto dire che, a parità di corrente, il flusso di macchina sarebbe anch'esso stato minore, e di conseguenza la tensione stessa: se si fosse optato per

questa strada l'aggiunta di terne avrebbe comportato di fatto una diminuzione della tensione nominale della macchine.

Questa però non sembrava essere la soluzione migliore da un punto della progettazione, in quanto, ipotizzando che la macchina operi all'interno di un azionamento e che quindi sia abbinata ad un inverter dimensionato sulla sua corrente e tensione massima, l'operazione di riavvolgimento avrebbe richiesto la sostituzione del convertitore oppure un suo sottosfruttamento in tensione: per evitare questa condizione si è deciso quindi di mantenere costante la tensione di macchina, il che comportava non variare la densità di corrente in cava ed il numero di spire.

Sulla base di queste considerazioni è lecito aspettarsi che le resistenze di fase e le correnti siano in qualche modo influenzate dall'operazione di riavvolgimento; prima però di esporre le relazioni fra i vari parametri risulta necessario capire come vengono calcolate ed in che modo l'utente ha la possibilità di modificare e visualizzare tali grandezze.

Nella finestra *Other Options* (figura 3.6) sono riassunti parametri di diversa natura, tra cui quelli relativi all'avvolgimento: in particolare SyR-e integra un modello termico della macchina che permette di stimare, sulla base delle temperature ammesse e delle perdite nel rame, la temperatura dell'avvolgimento.

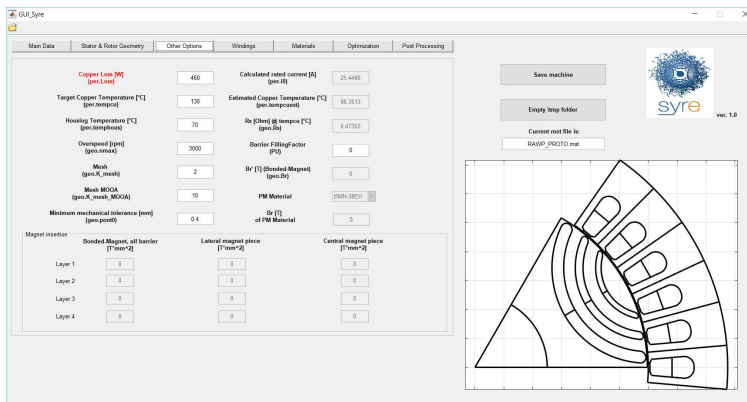


Figura 3.6: GUI: finestra *Other*

È proprio sulle perdite che bisogna soffermarsi in quanto sono fondamentali per la definizione delle correnti e per la stima della resistenza, basate entrambe sul concetto di massime perdite ammissibili; la potenza dissipata sotto forma di calore è infatti strettamente correlata alla tipologia del sistema raffreddamento di cui è dotata la macchina: un raffreddamento a liquido permette infatti di scambiare con

l'ambiente esterno molto più calore rispetto ad uno ad aria.

Più che ai valori di perdita ci si riferisce tipicamente ad una densità di potenza persa, il *carico termico* K_j , ottenuto come rapporto tra le perdite ammesse e la superficie attraverso la quale dovranno essere smaltite, in genere individuata nella superficie esterna del pacco statorico: sulla base di tale informazione vengono quindi calcolati i due parametri citati attraverso la funzione `calc_io`.

In generale, se N_s è il numero di spire in serie per fase, il numero totale di conduttori in cava sarà

$$N_{tot,cond} = 2(3 n_{3ph})N_s = 2mN_s \quad (3.1)$$

dove si tiene conto sia delle andate che dei ritorni dei conduttori, mentre il termine in parentesi può essere inteso come il numero totale di fasi della macchina m .

La sezione del singolo conduttore invece può essere espressa come rapporto fra l'area complessivamente occupata dal rame dei conduttori $S_{tot,cu}$ ed il loro numero totale

$$S_{cu} = \frac{S_{tot,cu}}{N_{tot,cond}} = \frac{S_{tot,cu}}{2mN_s} = \frac{k_{cu} A_{slots}}{2mN_s} \quad (3.2)$$

nella quale l'area complessiva occupata dal rame nelle cave può essere espressa come prodotto tra l'area totale delle cave A_{slots} ed il coefficiente di riempimento k_{cu} , ottenuto come rapporto tra la sezione del rame e quella totale, che tiene conto sia dell'isolante che degli spazi non occupati.

La resistenza di fase può essere correlata ai parametri geometrici della macchina attraverso la seguente relazione

$$R_s = \rho_{cu} N_s \frac{l_{tot}}{S_{cu}} = \rho_{cu} N_s \left[2(l + l_{end}) \right] \frac{2mN_s}{k_{cu} A_{slots}} = \rho_{cu} \frac{4mN_s^2 (l + l_{end})}{k_{cu} A_{slots}} \quad (3.3)$$

dove la lunghezza totale della spira l_{tot} risulta pari al doppio della somma del tratto lineare l e della lunghezza della testata l_{end} . A questo punto è necessario eguagliare le due differenti espressioni delle perdite nel rame P_{cu}

$$P_{cu} = \frac{m}{2} R_s i_0^2 \quad (3.4a)$$

$$P_{cu} = K_j 2\pi R l \quad (3.4b)$$

dove K_j è il carico termico precedentemente introdotto mentre R è il raggio esterno di statore; si ottiene quindi la seguente espressione del modulo del vettore corrente

$$i_0 = \sqrt{K_j k_{cu} \frac{1}{\rho_{cu}} \frac{l}{l + l_{end}} \pi R \frac{A_{slots}}{m^2 N_s^2}} \quad (3.5)$$

e, come è possibile notare, questa risulta inversamente proporzionale al numero di fasi m , e di conseguenza al numero di terne n_{3ph} .

Una volta ottenuto il valore di corrente è immediato calcolare il valore della resistenza di fase attraverso l'equazione

$$R_s = \frac{P_{cu}}{\frac{m}{2} i_0^2} \quad (3.6)$$

Com'era lecito aspettarsi, affinché il flusso di macchina, e quindi la sua tensione, rimanga costante è necessario adottare conduttori di sezione minore, precisamente $S_{cu} \propto n_{3ph}^{-1}$, il che implica che i valori di resistenza di fase aumentino in maniera proporzionale al numero di terne e che, affinché la densità di corrente in cava non vari, il modulo del vettore corrente risulti minore.

3.3.3 Trasformazioni di riferimento

Una volta adattato il processo di calcolo alla nuova casistica rimaneva un ulteriore nodo da sciogliere prima del passaggio alla definizione del problema agli elementi finiti e riguardava l'elaborazione delle grandezze: l'applicativo infatti, una volta calcolato il modulo del vettore corrente in assi rotanti, elaborava le singole componenti in asse diretto i_d e in asse di quadratura i_q attraverso il valore della fase γ , che in base al tipo di analisi avviata poteva risultare o meno costante. Il processo di analisi e simulazione segue infatti lo schema riportato in figura 3.7: a partire dalle coordinate in assi rotanti e dalla posizione del rotore θ rispetto al sistema assoluto, che coincide con la posizione dell'asse magnetico della prima fase, si passa nel sistema in assi fissi $\alpha\beta$ attraverso la *trasformata inversa di Park*

$$\begin{Bmatrix} X_\alpha \\ X_\beta \end{Bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{Bmatrix} X_d \\ X_q \end{Bmatrix} \quad (3.7)$$

e successivamente applicando la *trasformata inversa di Clarke* è possibile passare alle grandezze di fase.

$$\begin{Bmatrix} X_1 \\ X_2 \\ X_3 \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{Bmatrix} X_\alpha \\ X_\beta \end{Bmatrix} \quad (3.8)$$

Solo a questo punto inizia la definizione del problema all'interno di FEMM: vengono inizialmente calcolati tutti i punti necessari per il disegno della macchina, in seguito al quale inizia l'assegnazione delle proprietà dei singoli elementi, dai

materiali dei singoli blocchi alle definizioni dei circuiti elettrici della macchina, ed è proprio qui che vengono utilizzati i dati relativi alle correnti calcolati.

La procedura di simulazione con FEMM è automatizzata, il rotore viene ruotato in base al numero di posizioni angolari fissate in precedenza, vengono rielaborate le correnti da immettere e viene rieseguita l'analisi: una volta terminata la simulazione il programma acquisisce i flussi ed il valore di coppia e prosegue fintanto che non raggiunge il limite imposto.

Ad ogni acquisizione degli output da FEMM l'algoritmo elabora i flussi ritornando nuovamente nel riferimento bifase $\alpha\beta$ tramite la *trasformata diretta di Clarke*

$$\begin{Bmatrix} X_\alpha \\ X_\beta \end{Bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \end{Bmatrix} \quad (3.9)$$

e successivamente passa di nuovo in assi rotanti grazie alla *trasformata diretta di Park*

$$\begin{Bmatrix} X_d \\ X_q \end{Bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{Bmatrix} X_\alpha \\ X_\beta \end{Bmatrix} \quad (3.10)$$

in modo tale da poterli elaborare in maniera significativa per ottenerne le forme d'onda e le mappe di flusso rispetto alle correnti.

Per quanto riguarda il caso in esame, gli elementi critici di tale processo risultavano proprio essere le trasformate poichè il modello matematico dalle quali derivano è basato sull'ipotesi di sistemi elettrici trifase: risultava quindi necessario verificare quali fossero gli approcci ed i modelli sviluppati per affrontare questo tipo di problema.

Una ricerca bibliografica all'interno della letteratura specifica sull'argomento ha fatto emergere due approcci differenti adottati nell'analisi e nel controllo delle macchine multifase [5]: il primo approccio, sviluppato inizialmente per un motore

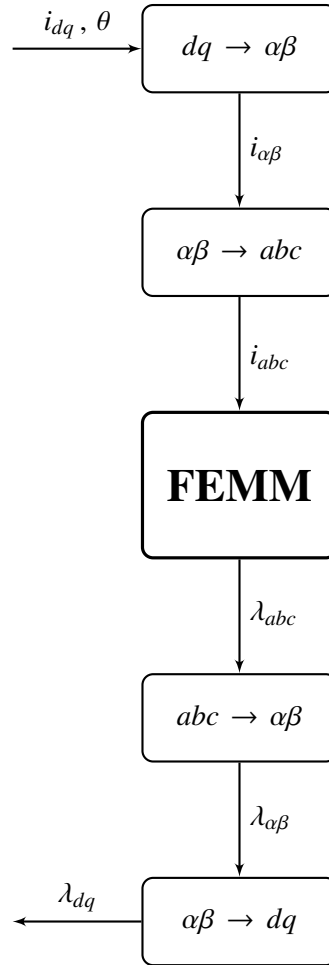


Figura 3.7: Processo di elaborazione dati in SyR-e

ad induzione esafase asimmetrico [6] e successivamente esteso ad un numero maggiore di fasi [4], prevede la scomposizione dello spazio originale attraverso una matrice di trasformazione quadrata di dimensione pari al numero di fasi della macchina; i sottospazi risultanti sono disaccoppiati fra loro e uno solo di questi produce coppia, a differenza degli altri che producono unicamente perdite: per tale motivo prende il nome di *Voltage Space Decomposition*.

Tale modello può essere applicato a qualunque tipologia di macchina multifase, d'altra parte non permette di distinguere i contributi delle singole fasi al comportamento globale della macchina.

Il secondo approccio, chiamato *Multi Statorico* [7], è meno generale del precedente in quanto applicabile a macchine multirifase ma ha il grande vantaggio di mettere in evidenza i contributi alle grandezze di macchina da parte delle singole terne, che a livello controllistico si riflette in gradi di libertà aggiuntivi; mentre la trasformata di Park non subisce variazioni in quanto il riferimento rimane unico per tutte le terne, ciò non è vero per le grandezze di fase: lo scostamento angolare tra gli assi $\alpha\beta$ ed i riferimenti trifase delle singole terne sono diversi l'uno dall'altro. Per tale ragione la matrice di trasformazione, pur rimanendo della stessa dimensione di quella di Clarke, non è più unica ma deve tener conto della posizione relativa fra i sistemi di riferimento.

Poichè la casistica da implementare nel codice di SyR-e risultava essere proprio quella delle macchine multirifase si è quindi deciso di adottare il metodo Multi Statorico.

Sulla base di quanto appena detto è già possibile rendersi conto del primo problema che tale scelta comporta: mentre in precedenza trattando unicamente macchine trifase era possibile dichiarare le matrici delle trasformate di Clarke come costanti, volendo implementare in maniera del tutto generale, senza cioè limitare a priori il numero di terne aggiungibili, era chiaro che le matrici di trasformazione dovessero essere calcolate in maniera automatica di volta in volta in base alla casistica in esame; un'altra possibilità sarebbe stata quella di scriverle manualmente per tutte le casistiche possibili, ma ciò avrebbe reso l'implementazione all'interno del codice più laboriosa e certamente ci si sarebbe fermati ad certo punto, perdendo quindi la possibilità di generalizzare il codice.

Partendo dalla prima trasformazione incontrata durante il processo di elaborazione, per la terna i -esima la matrice inversa avrebbe avuto la seguente forma

$$\begin{bmatrix} \cos \theta_i & \sin \theta_i \\ \cos \left(\theta_i + \frac{2\pi}{3} \right) & \sin \left(\theta_i + \frac{2\pi}{3} \right) \\ \cos \left(\theta_i - \frac{2\pi}{3} \right) & \sin \left(\theta_i - \frac{2\pi}{3} \right) \end{bmatrix} \quad (3.11)$$

dove θ_i risulta essere la differenza angolare fra l'asse α , che coincide con quello della prima fase della prima terna per convenzione, e l'asse della prima fase della terna i -esima; tale angolo dipende ovviamente dal numero di terne in cui sono raggruppate le fasi dell'avvolgimento: se n_{3ph} è il numero di terne dell'avvolgimento, lo scostamento angolare della terna i -esima sarà pari a

$$\theta_{i-1} = \frac{\pi}{3} \frac{i}{n_{3ph}} \quad (3.12)$$

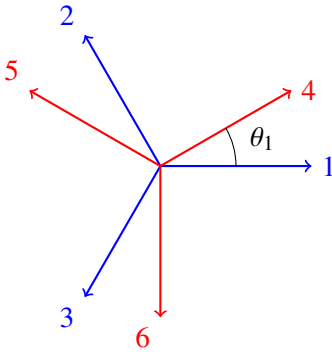


Figura 3.8: 6 fasi

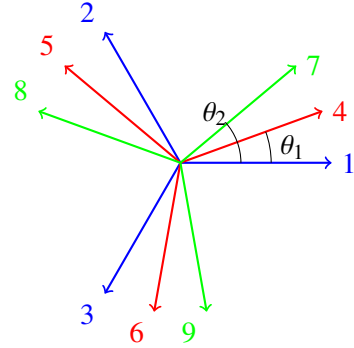


Figura 3.9: 9 fasi

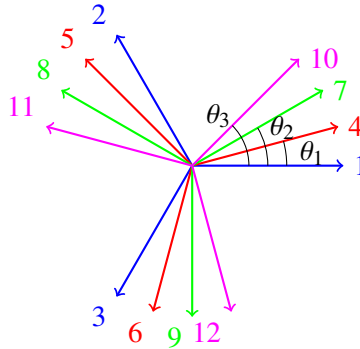


Figura 3.10: 12 fasi

A titolo di esempio si riportano nella seguente tabella i valori di tali angoli in gradi elettrici per alcune casistiche.

n_{3ph}	2	3	4	5
θ_1	30	20	15	12
θ_2		40	30	24
θ_3			45	36
θ_4				48

La struttura del codice permetteva di implementare la trasformazione secondo questo approccio in maniera abbastanza agevole all'interno della funzione dq2abc come segue:

```

1 delta = pi/3; %sextant amplitude
2 theta_i = delta*index/n3phase; %angular difference between
   alpha-axis and first phase of the n3phase-th 3-phase
   circuit
3
4 % Inverse transformation matrix (2 -> 3)
5 T23 = [cos(theta_i)      sin(theta_i)
6        cos(theta_i+2*pi/3) sin(theta_i+2*pi/3)
7        cos(theta_i-2*pi/3) sin(theta_i-2*pi/3)];

```

Tale funzione viene chiamata all'interno di `simulate_xdeg`, per calcolare i valori delle correnti di fase in ognuna delle terne è stato sufficiente aggiungere un ciclo basato sulla variabile incrementale $0 \leq \text{index} \leq n_{3ph}$.

In maniera del tutto complementare a quella precedente, la trasformazione diretta generalizzata dagli assi trifase a quelli bifase veniva definita dalla matrice

$$\frac{2}{3} \begin{bmatrix} \cos \theta_i & \cos \left(\theta_i + \frac{2\pi}{3} \right) & \cos \left(\theta_i - \frac{2\pi}{3} \right) \\ \sin \theta_i & \sin \left(\theta_i + \frac{2\pi}{3} \right) & \sin \left(\theta_i - \frac{2\pi}{3} \right) \end{bmatrix} \quad (3.13)$$

ed implementata in maniera efficace all'interno della funzione `abc2dq`, di cui si riporta la parte di codice interessata dalla modifica.

```

1 delta = pi/3; %sextant amplitude
2 theta_i = delta*index/n3phase; %angular difference between
   alpha-axis and first phase of the n3phase-th 3-phase
   circuit
3
4 % matrice di trasformazione (3 -> 2)
5 T32 = 2/3 * [cos(theta_i)      cos(theta_i+2*pi/3)      cos(
   theta_i-2*pi/3)
6              sin(theta_i)      sin(theta_i+2*pi/3)      sin(
   theta_i-2*pi/3)];

```

In questo caso la trasformata veniva eseguita all'interno del file `post_proc`, nel quale avviene lo scambio dei risultati dell'analisi agli elementi finiti con FEMM e la successiva elaborazione degli stessi; oltre ad implementare, come per la trasformata precedente, un ciclo iterativo che operasse su una terna alla volta, è stato necessario modificare l'output delle componenti del flusso di macchina in assi rotanti, che in accordo con il modello Multi Statorico sono ottenuti come media aritmetica delle componenti di flusso delle singole terne

$$\begin{cases} \lambda_d = \frac{\sum_{i=1}^{n_{3ph}} \lambda_{d,i}}{n_{3ph}} \\ \lambda_q = \frac{\sum_{i=1}^{n_{3ph}} \lambda_{q,i}}{n_{3ph}} \end{cases} \quad (3.14)$$

3.4 Risultati

Terminato il processo di modifica del codice di SyR-e risultava necessario testare le nuove funzionalità implementate: per fare ciò si è usato nuovamente come caso benchmark il prototipo *RawPower* realizzato per il Dipartimento sulla base dei processi di ottimizzazione svolti proprio sull'applicativo in questione, di cui quindi è a disposizione il progetto completo su SyR-e e le caratteristiche delle grandezze elettromagnetiche e meccaniche.

Come anticipato in precedenza, la versione modificata del software Koil integrata in SyR-e, su cui l'applicativo si appoggia per valutare la realizzabilità dell'avvolgimento ed ottenerne la corrispettiva matrice, non è ancora stata aggiornata per

supportare la creazione di avvolgimenti multifase: per tale motivo è stato necessario modificare manualmente i parametri necessari, in modo da poter proseguire con i test.

La procedura di modifica manuale ed il conseguente salvataggio del progetto multifase viene qui descritta:

- il file *RAWP_PROTO.m* associato al modello FEMM del prototipo è stato aperto all'interno dell'ambiente di calcolo Matlab, in modo tale che venissero aggiunte le nuove variabili descritte in precedenza: il caricamento di un progetto, sia all'interno della GUI che con la funzione *OpenSave.m* presentata nel capitolo precedente, è seguito dalla modifica automatica del *dataSet* che, attraverso la funzione *back_compatibility*, aggiunge o rinomina i campi aggiunti con le versioni di SyR-e successive a quella con cui è stata salvata la macchina aperta; ovviamente ciò accade anche per i nuovi parametri necessari per la nuova casistica implementata;
- sono stati quindi modificati i campi *WinMatr* e *DefaultWinMatr* della struttura *dataSet* con la matrice di una macchina avvolta su due strati con due terne e senza raccorciamento;

$$\begin{bmatrix} 1 & 4 & -3 & -6 & 2 & 5 \\ 1 & 4 & -3 & -6 & 2 & 5 \end{bmatrix}$$

- sempre per quanto riguarda la struttura *dataSet*, sono stati modificati i campi relativi al numero di terne dell'avvolgimento (*Num3PhaseCircuit*) e quello relativo alla casistica di guasto *WinFlag*, la cui dimensione deve essere pari a quella del numero di fasi complessive: questa modifica è in realtà già implementata nella GUI in relazione al box relativo al numero di terne nella scheda *Windings*;
- dopo aver salvato le variabili è stato necessario aggiornare il file FEMM associato: per far ciò è stato utilizzato lo script *OpenSave.m*, in alternativa si può utilizzare la GUI caricando e salvando il progetto.

In figura 3.11 è riportata la sezione della macchina salvata in FEMM: è possibile apprezzare come la collocazione delle fasi nelle cave segua esattamente la matrice dell'avvolgimento definita.

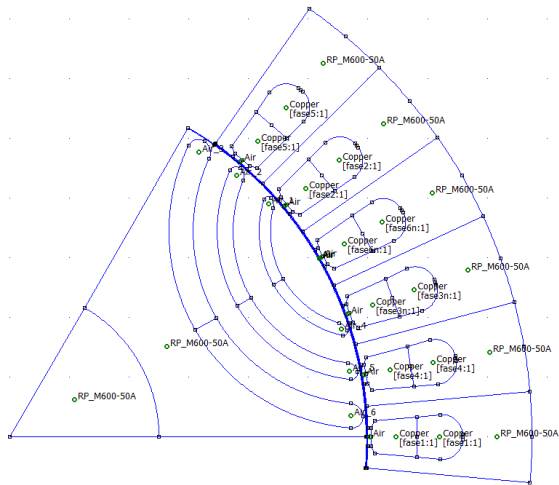
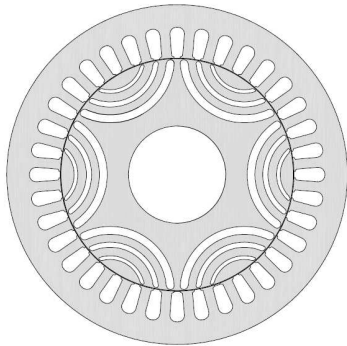


Figura 3.11: Prototipo esafase asimmetrico in FEMM

3.4.1 Prototipo trifase

Il prototipo RawPower è una macchina sincrona a riluttanza a 6 poli che presenta a rotore 3 barriere di flusso per polo, ciascuna a geometria circolare (tipologia *Circ* all'interno di *SyR-e*), ed uno statore a 36 cave all'interno delle quali è collocato un avvolgimento trifase doppio strato a passo intero: nella tabella seguente sono riportati i dati di targa [8] della macchina, mentre in figura 3.12 la sezione del lamierino.



Dati di targa	
Potenza nominale [kW]	2.2
Coppia [Nm]	21.0
Velocità nominale [rpm]	1000
Tensione di alimentazione [V]	157
Frequenza [Hz]	50
Corrente nominale [A]	13.2
Fattore di potenza	0.70
Efficienza	85.0%

Figura 3.12: Sezione del prototipo RawPower

Avendo a disposizione il progetto completo della macchina è stato possibile

ottenere tutte le informazioni riguardanti la geometria della stessa, alcune delle quali riassunte nella tabella 3.1.

Dimensioni	
Diametro esterno [mm]	175
Diametro traferro [mm]	119.35
Spessore traferro [mm]	0.325
Diametro albero [mm]	50
Lunghezza pacco attivo [mm]	110
Dati principali	
Paia poli	3
Numero di cave/polo/fase	2
Numero di barriere per polo	3
Numero di cave statore	36
Numero di spire in serie per fase	108

Tabella 3.1: Parametri principali del prototipo

Per definire invece le prestazioni della macchina sono stati utilizzati i valori presenti di default all'interno del progetto: fissando pertanto le perdite nel rame a 460 W si ottiene nella casistica in oggetto un valore del modulo di corrente pari a 25.45 A ed un valore di resistenza di fase di 0.47Ω , imponendo invece le temperature della carcassa e quella target del rame rispettivamente di 70°C e 130°C si ottiene una stima della temperatura di avvolgimento di 96.4°C .

Per caratterizzare la macchina dal punto di vista magnetico e meccanico è necessario lavorare all'interno della schermata *Post Processing* della GUI: per rendere più agevole il confronto si è deciso in prima battuta di analizzare il comportamento della macchina in punti di lavoro ben precisi, si è poi passati ad un'analisi più approfondita in un range di carico più ampio in modo da valutarne il modello magnetico complessivo ed i limiti di funzionamento.

Le prime condizioni di funzionamento analizzate sono state scelte in modo tale da poter evidenziare eventuali criticità nell'algoritmo, infatti la macchina è stata simulata con un vettore corrente di ampiezza pari a quella nominale posizionato in un caso lungo l'asse diretto, con $\gamma = 0$, nell'altro con solo componente sull'asse di quadratura ($\gamma = 90$): come si poteva immaginare, il flusso di macchina era posizionato sull'unico asse alimentato (figura 3.13).

Il terzo punto di lavoro, caratterizzato sempre dal vettore di corrente nominale ma stavolta con $\gamma = 50$ è stato scelto convenzionalmente per confrontare in maniera immediata le due macchine: in figura 3.14 sono riportati valori medi e forme d'onda relative a coppia, fattore di potenza e flussi.

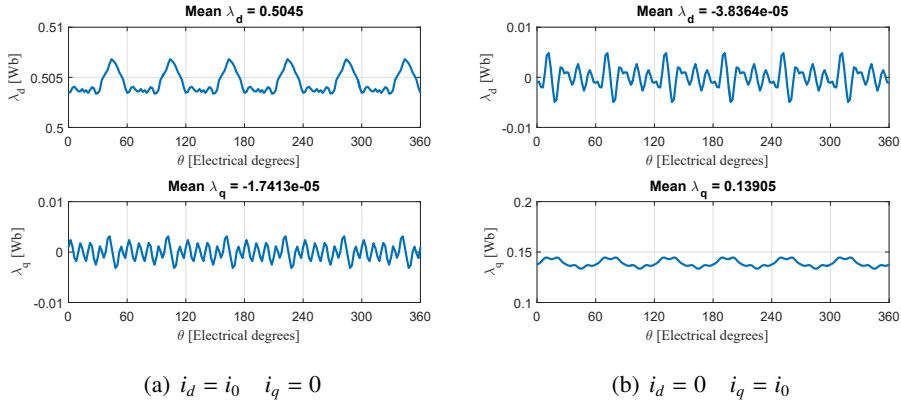


Figura 3.13: Prototipo: analisi punti di lavoro caratteristici

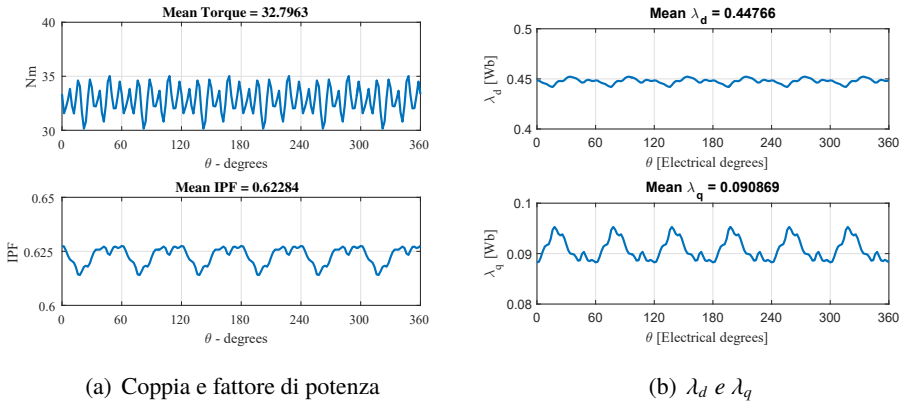


Figura 3.14: Prototipo: analisi punto di lavoro $i_0/50^\circ$

Come anticipato, per caratterizzare la macchina in un campo di funzionamento più ampio è necessario ottenere il modello magnetico completo della stessa e rielaborare i dati attraverso alcuni strumenti esterni all'interfaccia grafica; per selezionare l'analisi in questione è necessario assegnare alla fase del vettore corrente il valore convenzionale di 1000.

Gli altri parametri sono invece stati impostati nel seguente modo: tralasciando l'ampiezza della sezione da simulare invariata, si è inserito un valore massimo di corrente da analizzare pari a 2.5 volte la corrente nominale, che si ricorda essere di 25.45 A, inoltre per ottenere una maggiore precisione sono stati modificati il numero di posizioni di rotore, pari a 15, ed un numero di punti nel range $[0 \ i_{max}]$ di 12, direttamente connessa alla densità della griglia delle coppie di valori di corrente (i_d, i_q) .

In figura 3.15 sono riportate le relazioni tra i flussi e le correnti; come si può notare, oltre al fenomeno della saturazione magnetica che fa tendere ad un asintoto il valore dei flussi al crescere della rispettiva corrente, è molto evidente quella della *cross saturation*: i due assi sono infatti accoppiati, ed una componente di corrente ha un effetto smagnetizzante sulla componente di flusso opposta.

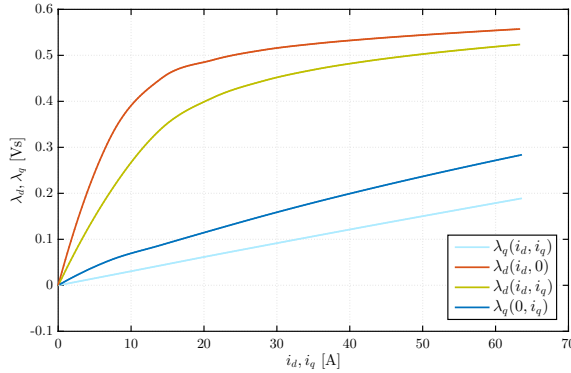


Figura 3.15: Caratteristiche flusso-corrente del prototipo

Nella parte alta del grafico si possono notare le curve relative al flusso diretto, in assenza (in alto) e con corrente di quadratura massima, nella parte bassa in maniera del tutto analoga sono mostrate le relazioni fra il flusso in quadratura e le correnti.

Gli altri grafici ottenuti dall'analisi del modello magnetico mostrano le superfici parametrizzate sulla base della griglia di valori di corrente dei flussi in asse diretto e di quadratura e della coppia, sia in termini di valor medio che di ripple picco-picco: la lettura di tali grafici non risulta agevole, ma i dati che riassumono permettono di ottenere ulteriori informazioni riguardo la macchina in esame.

In figura 3.16 si può notare come una griglia delle coppie di valori (i_d, i_q) molto densa, precisamente composta 256 elementi per ognuna delle due grandezze, ha permesso di ottenere superfici particolarmente dettagliate: analizzare la macchina in un numero così elevato di punti di lavoro risulta necessario per ottenere un'interpolazione significativa delle caratteristiche delle grandezze di interesse e per minimizzare conseguentemente eventuali errori.

Oltre agli output grafici, il vero risultato dell'analisi precedente risultano essere i dati del processo di simulazione i quali vengono salvati all'interno del file *fdfq_idiq_n256.mat*: SyR-e infatti implementa ulteriori strumenti di analisi più specifici, presenti all'interno della cartella *syreManipulateMM*, che forniscono informazioni più dettagliate per quanto riguarda i punti di lavoro caratteristici della macchina, tra cui quelli a coppia massima, e le caratteristiche meccaniche della stessa.

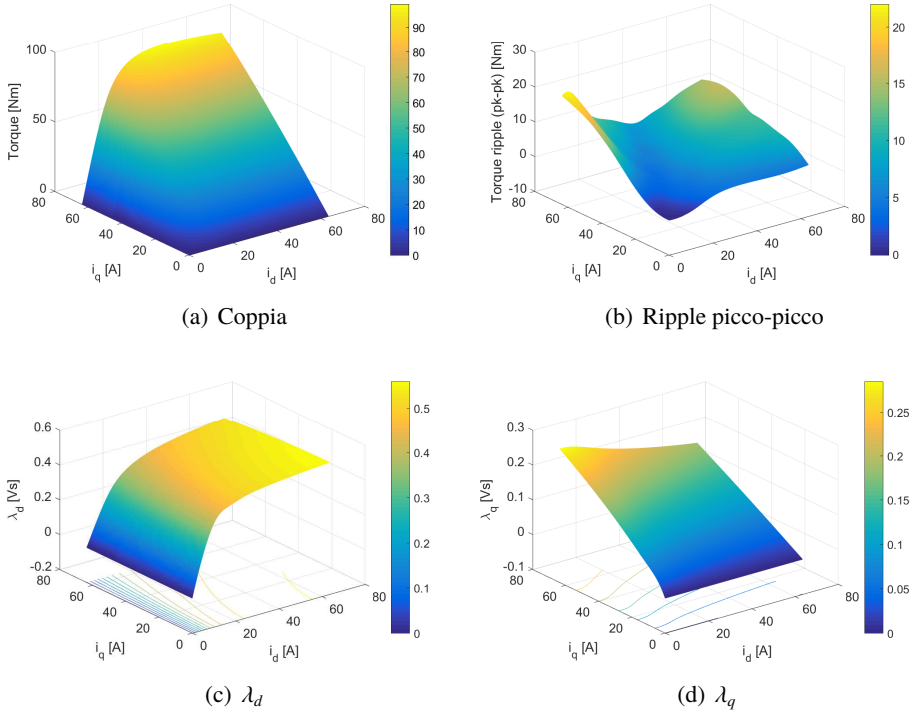


Figura 3.16: Modello magnetico del prototipo: superfici di flusso e coppia

Nello specifico:

- lo script `C_MMLut.m` restituisce alcune mappe ma soprattutto le tabelle dei flussi (chiamate generalmente *lookup table* o *LUT*) che ricoprono un ruolo fondamentale nell'ambito del controllo digitale in quanto permettono di associare attraverso metodi di interpolazione i valori di corrente misurati ai corrispondenti di flusso;
- `C_MtpaMtpvLut.m` elabora i luoghi *Maximum Torque Per Ampere* e *Maximum Torque Per Voltage*, che rappresentano nel riferimento rotante dq le traiettorie del vettore corrente a cui corrisponde la massima coppia erogata dalla macchina: tali andamenti sono influenzati dai limiti massimi di tensione e corrente disponibili, legati alle caratteristiche della macchina e dell'inverter di alimentazione.

Nelle figure 3.17 e 3.18 sono riportati rispettivamente i due luoghi citati

nel riferimento rotante ed le curve di corrente in relazione alla coppia di riferimento;

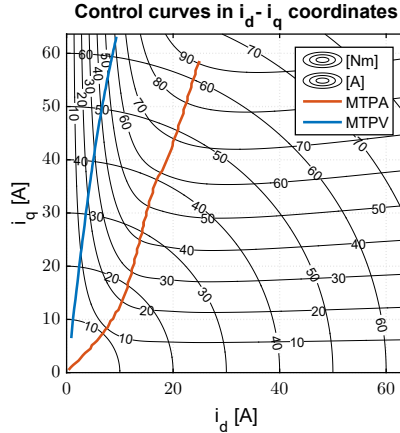


Figura 3.17: Prototipo: luoghi MTPA e MTPV

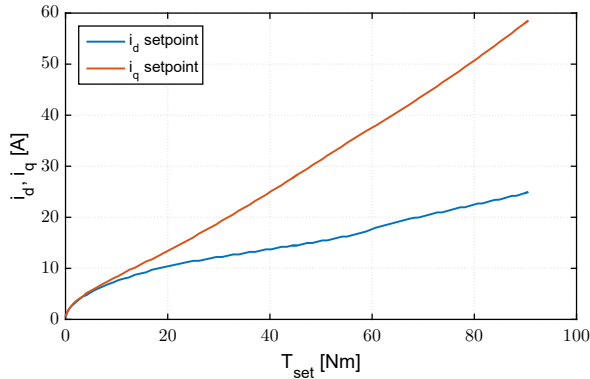


Figura 3.18: Prototipo: setpoint di corrente per MTPA e MTPV

- la funzione `C_OperatingLimits.m` valuta le performance meccaniche della macchina in esame sulla base dei limiti di corrente e tensione massimi inseriti: si ottengono, oltre ai valori di corrente di fase e di tensione rispetto alla velocità, i grafici di coppia e potenza.
- È possibile inoltre fissare quanti valori di corrente valutare: in figura 3.19(a) e 3.19(b) sono riportati i risultati ottenuti selezionando una tensione di 157 V ed una corrente di 26.4 A, pari alla tensione ed a due volte la

corrente nominale del prototipo descritto ad inizio paragrafo, per i due valori caratteristici di corrente i_0 e $2i_0$.

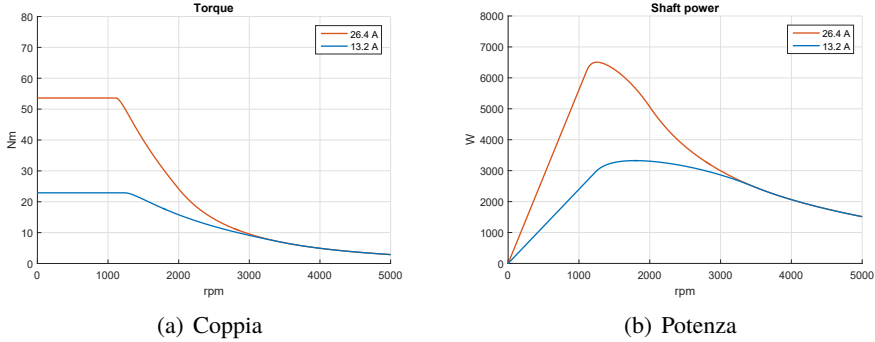


Figura 3.19: Caratteristiche meccaniche del prototipo

3.4.2 Macchina esafase

Come anticipato nella parte introduttiva, la macchina multirifase utilizzata per testare le funzionalità implementate è stata derivata direttamente dal modello del prototipo; questa infatti presentava un numero di cave/polo/fase pari a 2 e la configurazione più immediata da adottare risultava essere quelle esafase asimmetrica (3.8): la seconda terna sarebbe risultata quindi sfasata di 30° elettrici rispetto alla prima, ed ovviamente il numero di cave occupate da ogni fase si sarebbe dimezzato.

Si è deciso inoltre di adottare un avvolgimento a passo intero, così come quello del prototipo.

Primo riscontro del lavoro svolto si è avuto grazie ai parametri relativi all'avvolgimento statorico, nello specifico, mantenuto costante il carico elettrico della macchina non variando le perdite nel rame pari a 460 W, il valore di corrente calcolato risultava pari a 12.72 A, praticamente dimezzato rispetto al caso precedente, mentre la stima della resistenza di fase risultava di 0.95Ω .

Ciò è in accordo con 3.5 e 3.6, in quanto al crescere del numero di terne la corrente diminuisce in maniera proporzionale e il valore di resistenza risulta conseguentemente maggiore.

Le procedure di analisi intraprese seguono esattamente quelle adottate per il motore benchmark: si è partiti innanzitutto eseguendo delle simulazioni su due singoli punti di lavoro, con un valore di corrente pari a quella nominale ma orientati in direzione dei due assi di riferimento. In figura 3.20 sono riportate le

forme d'onda dei flussi, si può notare come nel caso in cui sia nulla una delle due componenti di corrente il flusso corrispondente sia anch'esso praticamente nullo: ciò a conferma del fatto che il modello è stato applicato in maniera efficace.

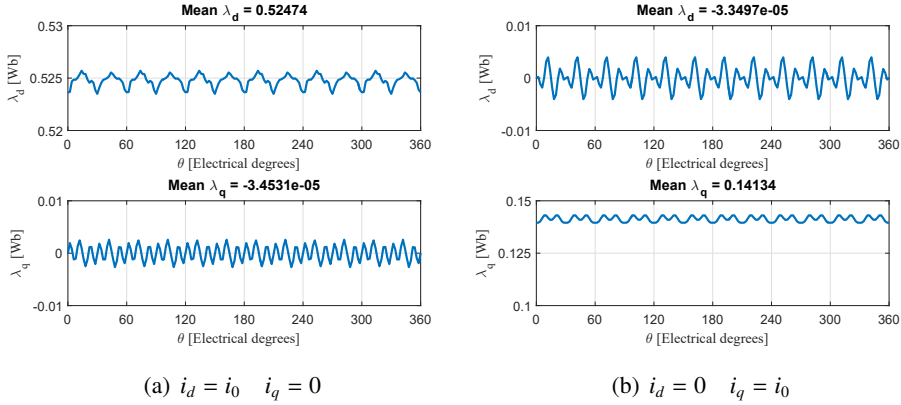


Figura 3.20: Macchina esafase: analisi punti di lavoro caratteristici

Il terzo punto di lavoro, caratterizzato sempre dal vettore di corrente nominale ma stavolta con $\gamma = 50^\circ$ è stato scelto convenzionalmente per confrontare in maniera immediata le due macchine: in figura 3.21 sono riportati i valori medi e le forme d'onda relative a coppia, fattore di potenza e flussi.

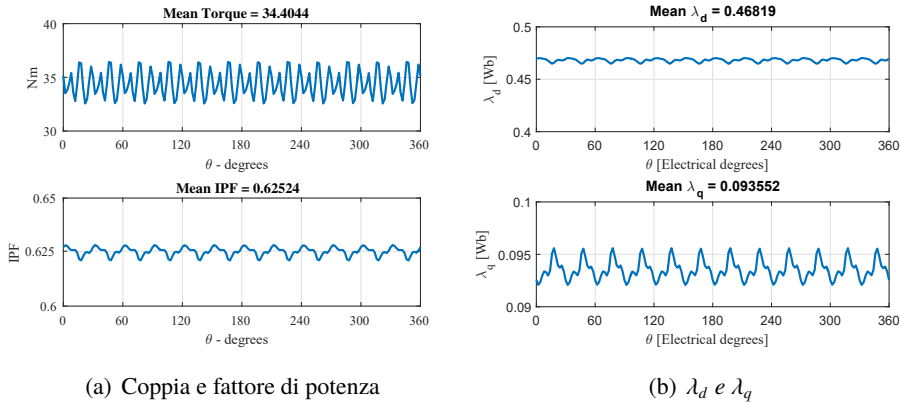


Figura 3.21: Macchina esafase: analisi punto di lavoro $i_0/50^\circ$

Lo step successivo è consistito, in maniera complementare a quanto fatto per la casistica precedente, nell'analisi della macchina volta ad individuarne il modello magnetico e le prestazioni meccaniche in un ampio range di funzionamento: nella finestra *Post Processing* della GUI è stato inserito un valore massimo di corrente

da simulare di $2.5i_0$, in questo caso pari a 12.72 A, mentre sono stati aumentati il numero di posizioni rotoriche da simulare, fissate a 15, ed il numero di punti in $[0 \ i_{max}]$ pari a 12, in modo da ottenere una griglia sufficientemente fitta ed analizzare con maggior precisione il comportamento della macchina.

In figura 3.22 sono riportate le relazioni tra i flussi e le correnti: anche in questo caso risulta molto evidente il comportamento non lineare della macchina, in particolare quella relativo all'accoppiamento fra i due assi (*cross saturation*); l'effetto smagnetizzante che una componente di corrente su un asse ha sulla componente di flusso opposta è molto accentuato.

Come nel grafico corrispettivo relativo al prototipo, nella parte alta si possono notare le curve relative al flusso diretto, in assenza (in alto) e con corrente di quadratura massima, nella parte bassa invece sono mostrate le relazioni fra il flusso in quadratura e le correnti.

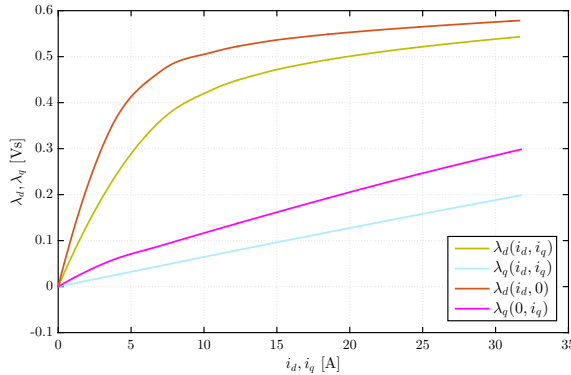


Figura 3.22: Caratteristiche flusso-corrente della macchina esafase

Per completezza in figura 3.23 si riportano le superfici parametrizzate rispetto alle coppie di valori di corrente (i_d, i_q) relative ai flussi in asse diretto e di quadratura ed ai valori medi e di ripple di coppia: sebbene non di immediata lettura, i dati a partire dai quali sono ottenute sono fondamentali per una caratterizzazione più approfondita della macchina attraverso le funzioni esterne alla GUI.

La prima elaborazione, eseguita con lo script `C_MtpaMtpvLut.m` all'esterno dell'interfaccia grafica, ha permesso di ottenere a partire dal modello magnetico completo i luoghi caratteristici di massima coppia per ampere *MTPA* e massima coppia per tensione *MTPV*, visibili in figura 3.24 rispettivamente nella parte destra e sinistra di singoli grafici: ricordando quanto visto in precedenza per la macchina benchmark, è possibile notare alcune somiglianze sulla forma di tali luoghi, che verranno approfondite nel seguito.

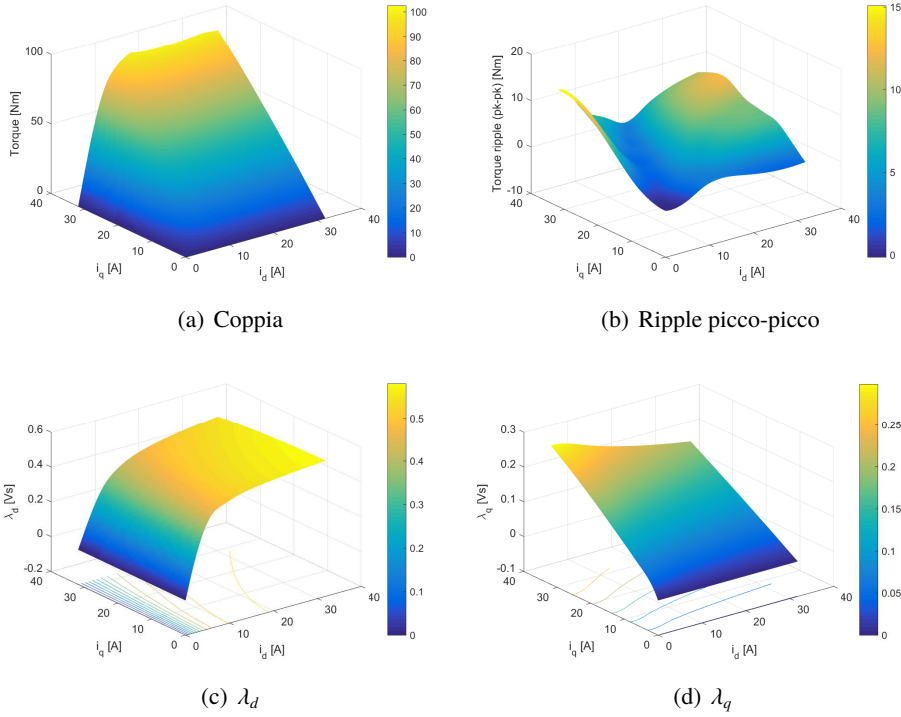


Figura 3.23: Modello magnetico della macchina esafase: superfici di flusso e coppia

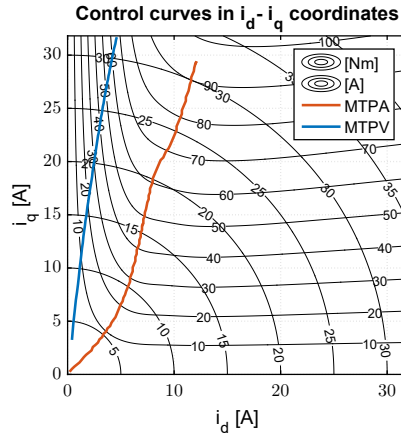


Figura 3.24: Macchina esafase: luoghi MTPA e MTPV

Attraverso la funzione `C_OperatingLimits.m` sono stati ottenuti i grafici delle performance meccaniche sulla base dei limiti di corrente e tensione massimi inseriti: nello specifico è possibile ottenere gli andamenti di coppia e potenza in

relazione alla velocità meccanica della macchina; nello stesso modo in cui ci si è mossi precedentemente, sono state impostate una tensione di 157 V e una corrente di 6.6 A, la metà della corrente nominale del prototipo descritto.

In figura 3.25(a) e 3.25(b) si riportano i grafici delle grandezze citate ottenuti per i due valori caratteristici di corrente i_0 e $2i_0$.

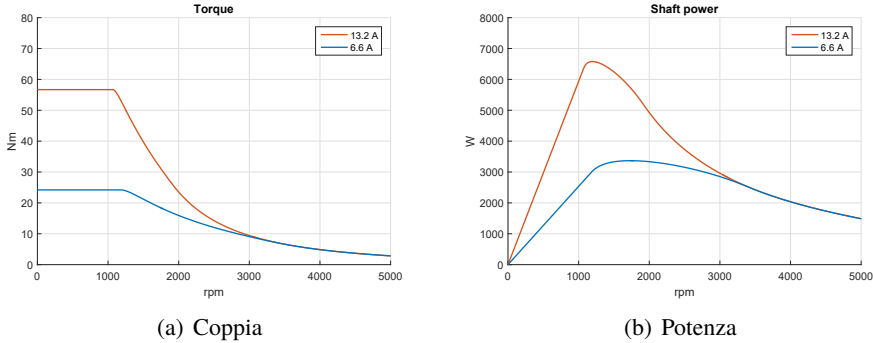


Figura 3.25: Caratteristiche meccaniche della macchina esafase

3.4.3 Confronti

Una volta analizzate separatamente le due macchine si avevano tutti gli elementi per procedere con un'analisi comparativa: come anticipato nella sezione precedente, i primi risultati ottenuti relativi alla macchina multirifase riguardavano l'entità della corrente nominale, risultata dimezzata rispetto al caso di partenza in accordo con 3.5.

La prima verifica effettuata è consistita nella simulazione delle due macchine in FEMM nelle stesse condizioni di funzionamento (precisamente $i_0/50^\circ$) per verificare i valori di densità di corrente, che in base alle considerazioni precedenti relative alla sezione dei conduttori e del numero di spire ci si aspettava fossero molto prossimi: in figura 3.26 si può notare come i limiti di tale grandezza nelle due macchine siano praticamente gli stessi, mentre i valori nella singola cava differiscono in quanto le trasformazioni di riferimento nelle due casistiche sono differenti.

Le prime analisi effettuate riguardavano tre punti di lavoro ben precisi: se i primi due, coincidenti con posizioni particolari in cui i risultati dovevano essere facilmente predicibili e d'interesse principalmente a scopo di verifica della correttezza del lavoro, quello caratterizzato da $i_0/50^\circ$, scelto arbitrariamente, permetteva invece di confrontare in maniera significativa le due macchine.

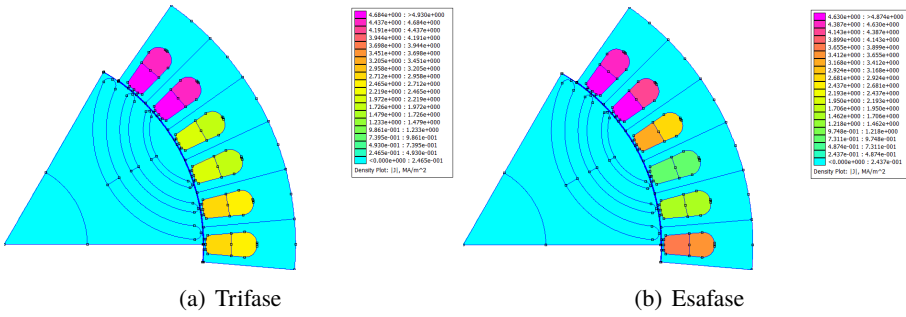


Figura 3.26: Densità di corrente in cava

Facendo riferimento a 3.27, si vuol far notare come la scelta di mantenere costante il numero di spire per non modificare la tensione d'avvolgimento, sulla quale è stato basato il calcolo del modulo di corrente, ha fatto in modo che i flussi di macchina non variassero in maniera sostanziale nel passaggio alla nuova casistica: si deve però segnalare un aumento non trascurabile per quanto riguarda i valori medi, in particolare per il flusso in asse di quadratura.

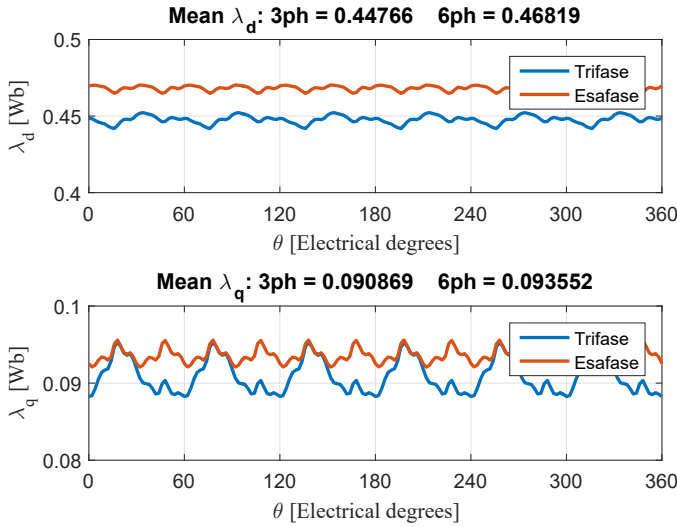


Figura 3.27: Confronto analisi punto di lavoro $i_o/50^\circ$: λ_d e λ_q

Nelle forme d'onda di coppia e fattore di potenza si trova riscontro di quanto appena emerso riguardo i flussi magnetici: il valore di coppia media della macchina esafase risulta maggiore del 5% rispetto alla caso di partenza, in maniera analoga a quanto riscontrato per il flusso in asse diretto.

D'altra parte il ripple picco-picco risulta invece diminuito, si passa infatti da un valore di 4.04 Nm del prototipo ai 3.35 della macchina esafase.

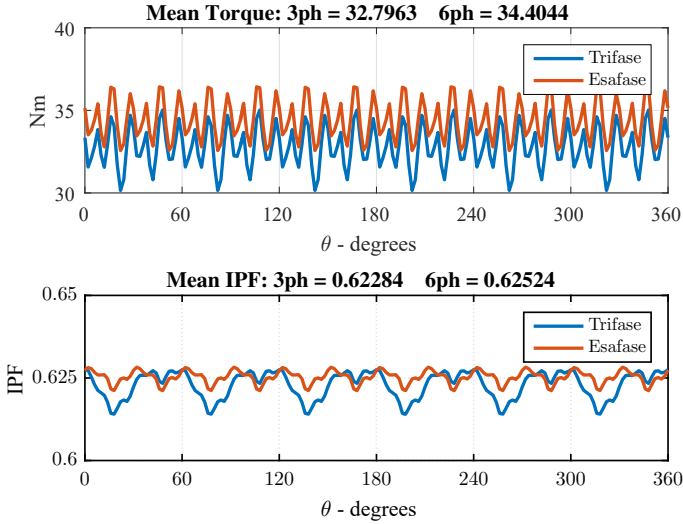


Figura 3.28: Confronto analisi punto di lavoro $i_o/50^\circ$: coppia e fattore di potenza

Le caratteristiche che legano i flussi di macchina alle correnti sono riportate in figura 3.29: in accordo con quanto notato precedentemente, rispetto alla macchina prototipo i valori di flusso appaiono, a parità di corrente, più elevati. Si può notare però come l'effetto della *cross saturation* sia meno marcato nella casistica esafase, ciò potrebbe essere legato all'entità minore delle correnti di fase, che producono un effetto di minor smagnetizzazione.

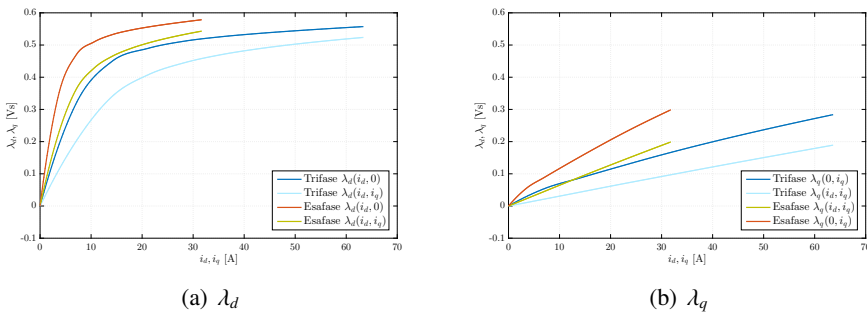


Figura 3.29: Confronto caratteristiche flusso-corrente

La prima elaborazione, eseguita con lo script `C_MtpaMtpvLut.m`, ha permesso di ottenere i luoghi caratteristici di massima coppia per ampere *MTPA* e massima

coppia per tensione *MTPV*, visibili in figura 3.30 rispettivamente nella parte destra e sinistra di singoli grafici: si noti come, a meno della differente scala degli assi, i due andamenti siano molto simili fra loro, sebbene si possano notare delle diversità per quel che riguarda le curve di livello a coppia costante, nello specifico nel caso della macchina esafase ne sono visibili alcune con valori di coppia più elevati.

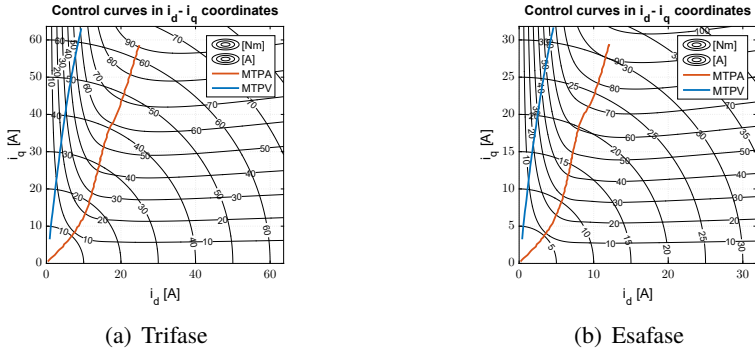


Figura 3.30: Confronto luoghi MTPA e MTPV

Tale similitudine viene confermata ulteriormente dalla figura 3.31, in cui sono riportati i valori delle componenti diretta e di quadratura della corrente lungo i luoghi a coppia massima MTPA e MTPV in relazione alla coppia desiderata: come era lecito aspettarsi in base al modello presentato, l'effetto più rilevante della divisione dell'avvolgimento trifase di partenza in più sets riguarda proprio la diminuzione dell'ampiezza del vettore corrente e questo comportamento è giustificato dal fatto che ogni terna contribuisce alla generazione della coppia meccanica in egual maniera.

Il carico risulta distribuito fra più terne in contemporanea, è quindi sensato che allo stesso riferimento di coppia corrispondano valori di corrente differenti in ampiezza, ridotti in maniera proporzionale al numero di terne disponibili: la coppia è una proprietà globale di riferimento per il motore, ma le grandezze effettivamente controllabili sono quelle delle singole terna, eventualmente anche in maniera indipendente.

Infine attraverso la funzione `C_OperatingLimits.m` sono state ottenute le caratteristiche meccaniche della macchina esafase e, in linea con quanto fatto in precedenza, sono stati utilizzati i dati di targa del prototipo: la tensione di alimentazione è stato impostata a 157 V, mentre la corrente di fase è stata fissata questa volta al valore corretto di 13.2 A, il doppio della corrente nominale che nel caso in questione risultava dimezzata.

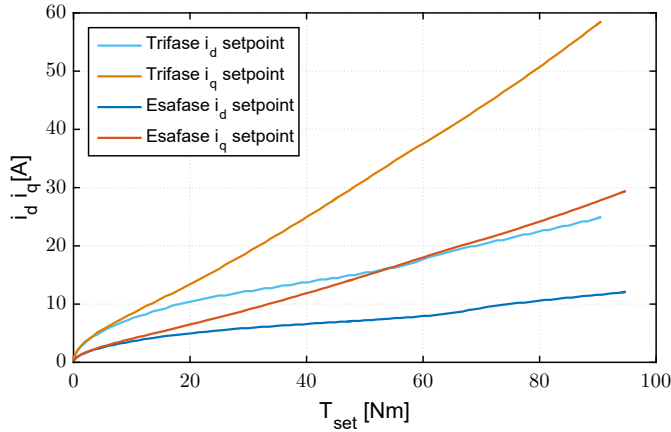


Figura 3.31: Confronto setpoint di corrente per MTPA e MTPV

In figura 3.32 sono riportate le caratteristiche di coppia delle due macchine: come evidenziato in precedenza nell'analisi del singolo punto di lavoro, la macchina riavvolta esafase mostra un valore di coppia massima più elevata ma, a causa del maggior flusso che presentava, incontra l'MTPA prima del prototipo (esattamente a 1233 rpm rispetto a 1281 nel caso del carico nominale), i valori di coppia risultano anch'essi leggermente differenti, rispettivamente 23.59 Nm nella macchina multitrifase e 22.27 in quella di partenza.

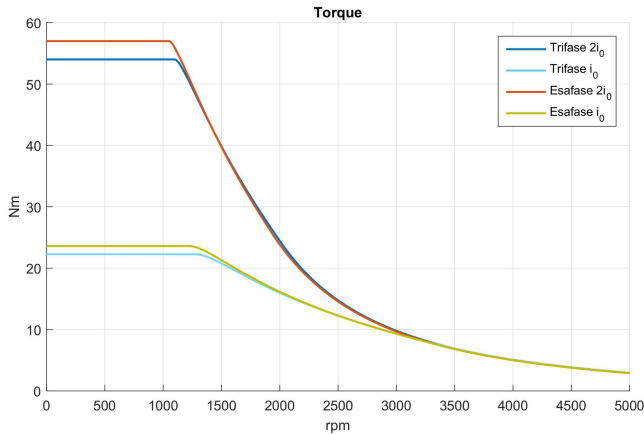


Figura 3.32: Confronto coppia all'albero

Per quanto riguarda invece le caratteristiche in sovraccarico trova conferma quanto appena evidenziato: la coppia erogata dalla macchina esafase risulta maggiore

(56.99 rispetto a 54 Nm), ma come è lecito aspettarsi le velocità base diminuiscono, passando rispettivamente 1050 e 1099 rpm, in quanto il limite massimo di tensione, pari a quella di alimentazione, viene eguagliato a velocità inferiori.

Le caratteristiche di potenza in funzione della velocità meccanica (figura 3.33) confermano quanto precedentemente evidenziato: nei grafici relativi al funzionamento alla corrente nominale alla velocità base il prototipo è caratterizzato da un valore di potenza pari a 2987 W, mentre alla sua velocità base la macchina esafase presenta un valore di potenza pari a 3047 W; nel funzionamento in sovraccarico tali valori passano rispettivamente a 6216 e 6269 W.

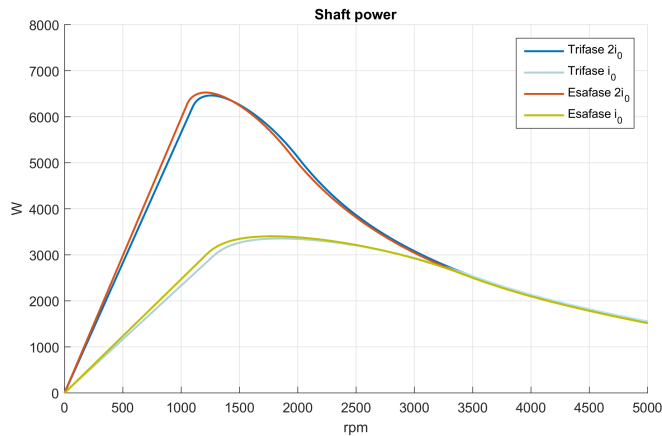


Figura 3.33: Confronto potenza

Una volta raggiunto il picco di potenza, pari a 3355 e 3403 W alla corrente nominale e 6526 e 6462 W in caso di sovraccarico, gli andamenti decrescono in maniera inversamente proporzionale con la velocità, più ripidamente per quanto riguarda la macchina esafase.

3.5 Conclusioni

In questo capitolo sono state presentate le modalità con cui è stato affrontato il problema dell'implementazione all'interno del codice di SyR-e di un'ulteriore tipologia di macchine: è stato innanzitutto introdotto il mondo delle macchine multifase, evidenziandone caratteristiche e configurazioni principali, si è passati poi a descrivere le modifiche apportate all'interfaccia grafica ed i fondamenti teorici sulla base dei quali si è proceduto con la modifica dei parametri delle grandezze di macchina interessate e delle trasformazioni di riferimento.

Le simulazioni effettuate hanno evidenziato come l'esito del lavoro possa ritenersi positivo, i risultati sono apparsi pressochè in linea con quanto ci si poteva attendere sulla base della letteratura acquisita.

Risulta però necessario far notare come il lavoro svolto rappresenti solo il primo passo verso la completa integrazione della casistica multifase all'interno del codice, in quanto al momento, oltre a non essere stato aggiornato lo strumento di valutazione e generazione automatica dell'avvolgimento, solo la famiglia delle macchina asimmetriche trifasi è stata tenuta in considerazione: in tal senso possibili sviluppi possono essere rappresentati dall'implementazione di ulteriori tipologie di avvolgimenti multifase, come ad esempio quelli asimmetrici composti da sets di fasi maggiori (5, 7, 11..), ed ovviamente dall'automatizzazione del processo di definizione dello schema d'avvolgimento.

Bibliografia

- [1] G. Pellegrino, A. Vagati, P. Guglielmi e B. Boazzo, “Performance Comparison Between Surface-Mounted and Interior PM Motor Drives for Electric Vehicle Application”, *IEEE Transactions on Industrial Electronics*, vol. 59, n. 2, pp. 803–811, feb. 2012, ISSN: 0278-0046. DOI: 10.1109/TIE.2011.2151825.
- [2] G. Pellegrino e F. Cupertino, *SyR-e User Manual*, v1.6, mar. 2017. indirizzo: <https://sourceforge.net/projects/syr-e/>.
- [3] R. Storn e K. Price, “Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces”, vol. 11, pp. 341–359, gen. 1997.
- [4] E. Levi, R. Bojoi, F. Profumo, H. A. Toliyat e S. Williamson, “Multiphase induction motor drives - a technology status review”, *IET Electric Power Applications*, vol. 1, n. 4, pp. 489–516, lug. 2007, ISSN: 1751-8660. DOI: 10.1049/iet-epa:20060342.
- [5] R. Bojoi, S. Rubino, A. Tenconi e S. Vaschetto, “Multiphase electrical machines and drives: A viable solution for energy generation and transportation electrification”, in *2016 International Conference and Exposition on Electrical and Power Engineering (EPE)*, ott. 2016, pp. 632–639. DOI: 10.1109/ICEPE.2016.7781416.
- [6] Y. Zhao e T. A. Lipo, “Space vector PWM control of dual three-phase induction machine using vector space decomposition”, *IEEE Transactions on Industry Applications*, vol. 31, n. 5, pp. 1100–1109, set. 1995, ISSN: 0093-9994. DOI: 10.1109/28.464525.
- [7] R. H. Nelson e P. C. Krause, “Induction Machine Analysis for Arbitrary Displacement Between Multiple Winding Sets”, *IEEE Transactions on Power*

Apparatus and Systems, vol. PAS-93, n. 3, pp. 841–848, mag. 1974, issn: 0018-9510. DOI: 10.1109/TPAS.1974.293983.

- [8] D. Meeker, N. Bianchi, J. Gyselinck, R. V. Sabariego, L. Alberti, G. Pellegrino e F. Cupertino, “Electrical machine analysis using free software”, in *2017 IEEE Energy Conversion Congress and Exposition (ECCE)*, ott. 2017, pp. 1–289. DOI: 10.1109/ECCE.2017.8096985.