

TESI DI LAUREA MAGISTRALE



**Sviluppo di un modello *high-fidelity* del robot UR5
a fini diagnostici e prognostici**

Relatore:

prof. Massimo Sorli

Correlatori:

prof. Stefano Mauro

ing. Andrea De Martin

ing. Andrea Raviola

Candidato:

Roberto Guida

Abstract

L'obiettivo del presente lavoro di tesi è quello di proporre una metodologia di modellazione dei robot manipolatori volta alla diagnostica ed alla prognostica *model based*.

I robot manipolatori presentano infatti, in pochissime occasioni, delle *failure* che ne causino l'arresto, ma in determinate applicazioni è necessario mantenere un certo livello di precisione, affidabilità e sicurezza, ad esempio nell'ambito della robotica collaborativa.

Tale penuria di dati spinge l'interesse verso la creazione di un modello del robot in modo da poter simulare e testare eventuali algoritmi di prognostica e diagnostica.

In particolare si è utilizzato come caso di studio il robot manipolatore UR5 della *Universal Robots*TM. Tale esemplare di robot è presente all'interno del Dipartimento di Ingegneria Meccanica e Aerospaziale del Politecnico di Torino.

Analizzando il sistema fisico reale è necessario suddividerlo in diversi sottosistemi all'interno dei quali iniettare i diversi guasti che possono presentarsi durante il funzionamento del robot. Per ciascun sottosistema è necessario definirne un modello matematico che presenti degli input ed output ben definiti in modo da rendere l'intero modello modulare e sempre migliorabile.

L'intero modello viene implementato all'interno di *MATLAB Simulink/Simscape Multibody*TM. Tutti i parametri, dinamici e cinematici, sono ricavati dalle informazioni fornite dall'azienda produttrice, dalla letteratura precedente e dal robot stesso.

In particolare sarà necessario utilizzare una prova come prova di identificazione dei parametri incogniti e delle ulteriori prove volte alla validazione del modello. In tal maniera è possibile verificare la contezza dei risultati forniti dal modello e quindi la sua effettiva applicabilità in campo diagnostico e prognostico.

Abstract

The aim of this thesis is to propose a modeling methodology of manipulator robots for model-based diagnostics and prognostics.

In fact, manipulator robots present, on very few occasions, a failure that causes a definitive break, but in certain applications, it is necessary to maintain a certain level of precision, reliability and safety, for example in the field of collaborative robotics.

This lack of data drives the interest in the creation of a robot model in order to simulate and test prognostic and diagnostic algorithms.

In particular, the UR5 robot manipulator of Universal Robots™, was used as a case study. This robot is in the Department of Mechanical and Aerospace Engineering of the Politecnico di Torino.

Analyzing the real physical system, it is necessary to divide it into different subsystems, into which to inject the various failures that may occur during the work of the robot. For each subsystem, it is necessary to define a mathematical model that presents well-defined inputs and outputs in order to make the entire model modular and always improvable.

The whole model is implemented in MATLAB Simulink/Simscape Multibody™. All dynamic and kinematic parameters are obtained from the information provided by the manufacturer, from previous literature and from the robot itself.

In particular, it will be necessary to use a test as an identification test of the unknown parameters. Further tests will be used to validate the model.

In this way, it is possible to verify the accuracy of the results provided by the model and therefore its effective applicability in the diagnostic and prognostic field.

Indice

1	Introduzione	10
1.1	La diagnostica e la prognostica nei sistemi ingegneristici . . .	11
1.1.1	La relazione tra diagnostica e prognostica	11
1.1.2	La diagnostica dei guasti	14
1.1.3	La prognostica dei guasti	14
1.2	La diagnostica e la prognostica nei robot	15
1.2.1	La prognostica <i>model-based</i> nei robot collaborativi . . .	17
2	Il robot collaborativo UR5	18
2.1	Il braccio robotico	20
2.1.1	I motoriduttori	23
2.2	L'unità di controllo	27
2.3	Il teach pendant	28
2.4	I sensori	31
3	Descrizione del modello dell'UR5	32
3.1	La cinematica inversa e la definizione delle leggi del moto . . .	36
3.2	Il modello dinamico in ambiente Simscape Multibody	38
3.2.1	Il sensore di coppia	46
3.3	Il motore monofase e la logica di controllo	48
3.3.1	Introduzione del motore trifase	57
3.3.2	I riduttori armonici	60
3.4	I sensori	65
4	Identificazione e validazione del modello tramite prove sperimentali	66
4.1	Prova di identificazione	66
4.2	Prove di validazione	74
5	Iniezione di guasti a scopo diagnostico e prognostico	96
5.1	Guasti nei motori elettrici	97

<i>INDICE</i>	4
5.2 Guasti agli <i>harmonic drive</i>	97
5.3 Gioco tra motore e riduttore	98
5.4 Guasti alla struttura meccanica	101
5.5 Guasti agli <i>encoder</i>	102
5.6 Analisi dei risultati a scopo prognostico	103
6 Conclusioni e sviluppi futuri	105

Elenco delle figure

1.1	Periodi di intervento della prognostica e della diagnostica nell'evoluzione temporale del guasto.	12
1.2	Schema generale di robot considerato	16
2.1	UR5 Robot.	19
2.2	Nomenclatura dei giunti del robot UR5.	21
2.3	Ingombri di massima, in millimetri, dell'UR5.	22
2.4	Spazio di lavoro dell'UR5.	23
2.5	Configurazione di allineamento del polso 2 con il piano in cui ruotano i giunti di base, spalla e polso 1 [1].	24
2.6	Componenti riduttore armonico.	25
2.7	Componenti dell'unità di controllo.	27
2.8	Teach pendant.	29
2.9	Esempio legge del moto con profilo di velocità trapezoidale.	30
2.10	<i>Encoder</i> rotativo magentico assoluto AskIM.	31
3.1	Schema a blocchi modello UR5.	33
3.2	Schema a blocchi modello UR5 in Simulink.	35
3.3	Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalla definizione delle pose del TCP.	36
3.4	Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalla definizione delle pose del TCP in funzione del tempo.	37
3.5	Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalle configurazioni angolari.	37
3.6	Sottosistema <i>Joint Space Trajectory</i>	38
3.7	Sottosistema " <i>Simscape UR5 Robot Dynamic Model</i> " espanso.	39
3.8	<i>Simscape UR5 Robot Dynamic Model</i>	40
3.9	Sottosistema " <i>Joint 1</i> ".	40
3.10	Parametri di Denavit-Hartenberg [2].	41

3.11	Parametri di errore di un asse di rotazione [3].	42
3.12	Modello all'interno dell'ambiente <i>Mechanics Explorer</i>	44
3.13	Sensore di coppia FT300.	46
3.14	Modifica del blocco di dinamica per inserire il sensore di coppia.	47
3.15	Modello del sistema di attuazione.	48
3.16	Modello motore monofase a corrente continua.	48
3.17	Schema a blocchi motore elettrico a corrente continua del primo giunto.	50
3.18	Schema a blocchi motore elettrico a corrente continua del primo giunto con coppia d'inerzia.	51
3.19	Andamento tipico della coppia d'attrito in funzione della velocità di rotazione [4].	52
3.20	Blocco " <i>C&V Friction</i> " del primo giunto.	53
3.21	Schema a blocchi riduttore.	54
3.22	Esempio tensione generata tramite tecnica PWM.	54
3.23	Logica di controllo in retrazione di posizione, velocità e corrente.	55
3.24	Modello <i>Simulink</i> TM per la taratura dell'anello di corrente.	56
3.25	Modello <i>Simulink</i> TM del controllore dell'anello di corrente per il motore trifase.	58
3.26	Modello <i>Simulink</i> TM per la conversione del segnale di tensione e della logica PWM per il motore trifase.	58
3.27	Modello <i>Simulink</i> TM dell'inverter per il motore trifase.	59
3.28	Modello <i>Simulink</i> TM del motore trifase col blocco di simulazione della temperatura.	60
3.29	Schema del meccanismo di traslazione del riduttore armonico [5].	61
3.30	Modello di <i>harmonic drive</i> considerando la geometria e le interazioni interne[5].	62
3.31	Modello dell' <i>harmonic drive</i> in <i>Simulink</i> TM	64
3.32	Blocco dei sensori di corrente e blocco con funzione di trasferimento del sensore di corrente del primo giunto.	65
4.1	Traiettoria nello spazio di lavoro del TCP per il " <i>test0</i> ".	67
4.2	Grandezze simulate e misurate per il giunto di base per il " <i>test0</i> ".	68
4.3	Grandezze simulate e misurate per il giunto di spalla per il " <i>test0</i> ".	69
4.4	Grandezze simulate e misurate per il giunto di gomito per il " <i>test0</i> ".	70
4.5	Grandezze simulate e misurate per il giunto di polso 1 per il " <i>test0</i> ".	71

4.6	Grandezze simulate e misurate per il giunto di polso 2 per il "test0".	72
4.7	Grandezze simulate e misurate per il giunto di polso 3 per il "test0".	73
4.8	Moto del TCP nello spazio di lavoro per il "test1".	75
4.9	Grandezze simulate e misurate per il giunto di base per il "test1".	76
4.10	Grandezze simulate e misurate per il giunto di spalla per il "test1".	77
4.11	Grandezze simulate e misurate per il giunto di gomito per il "test1".	78
4.12	Grandezze simulate e misurate per il giunto di polso 1 per il "test1".	79
4.13	Grandezze simulate e misurate per il giunto di polso 2 per il "test1".	80
4.14	Grandezze simulate e misurate per il giunto di polso 3 per il "test1".	81
4.15	Moto del TCP nello spazio di lavoro per il "test2".	82
4.16	Grandezze simulate e misurate per il giunto di base per il "test2".	83
4.17	Grandezze simulate e misurate per il giunto di spalla per il "test2".	84
4.18	Grandezze simulate e misurate per il giunto di gomito per il "test2".	85
4.19	Grandezze simulate e misurate per il giunto di polso 1 per il "test2".	86
4.20	Grandezze simulate e misurate per il giunto di polso 2 per il "test2".	87
4.21	Grandezze simulate e misurate per il giunto di polso 3 per il "test2".	88
4.22	Moto del TCP nello spazio di lavoro per il "test3".	89
4.23	Grandezze simulate e misurate per il giunto di base per il "test3".	90
4.24	Grandezze simulate e misurate per il giunto di spalla per il "test3".	91
4.25	Grandezze simulate e misurate per il giunto di gomito per il "test3".	92
4.26	Grandezze simulate e misurate per il giunto di polso 1 per il "test3".	93
4.27	Grandezze simulate e misurate per il giunto di polso 2 per il "test3".	94
4.28	Grandezze simulate e misurate per il giunto di polso 3 per il "test3".	95

5.1	Blocco di modellazione del gioco all'interno di <i>Simulink</i>	100
5.2	Condizioni verificabili per il <i>backlash</i>	101
5.3	Modellazione gradi di libertà dei giochi del giunto.	102

Elenco delle tabelle

2.1	Principali caratteristiche tecniche dei modelli della CB series [6].	18
2.2	Specifiche tecniche del braccio robotico dell'UR5 [7].	20
2.3	Coppie massime applicabili ai giunti dell'UR5 [8].	24
2.4	Specifiche tecniche dell'unità di controllo dell'UR5 [7].	28
2.5	Specifiche tecniche del <i>teach pendant</i> [7].	30
3.1	Parametri di Denavit-Hartenberg dell'UR5.	42
3.2	Massa e posizione del centro di massa dei <i>link</i> dell'UR5.	45
3.3	Limiti di posizione, velocità ed accelerazione dei giunti.	46
3.4	Grandezze motori UR5.	49
3.5	Parametri utilizzati all'interno dell'algoritmo di controllo.	57
5.1	Parametri tipici del riduttore armonico	98

Capitolo 1

Introduzione

Negli ultimi decenni l'automazione e l'utilizzo di sistemi automatici hanno assunto un ruolo sempre più di rilievo nell'ambito della produzione industriale in tutto il mondo, anche in termini di Industria 4.0. Essi forniscono la possibilità di far svolgere compiti di precisione, spesso ripetitivi, ad un robot, piuttosto che ad un operatore, presentando svariati vantaggi in termini di tempi, costi e qualità del lavoro svolto. Un ulteriore vantaggio è rappresentato, naturalmente, dalla possibilità di ridurre il carico da far svolgere alla forza lavoro, la quale può operare anche in collaborazione con la macchina, e la probabilità che possa verificarsi un infortunio.

Vista la crescente importanza assunta da tale tecnologia, la possibilità di effettuare manutenzione predittiva per la robotica industriale rappresenta una delle maggiori sfide nella nuova era dell'Industria 4.0 [9]. Lo scoglio da superare è rappresentato dalla mancanza di dati che permettano uno studio dello stato di salute dei robot di tipo, appunto, *data-driven* e per tale motivo risulta necessario convergere verso un approccio *model-based*.

L'obiettivo di questa trattazione è proprio quello di descrivere una possibile metodologia di approccio che permetta di effettuare uno studio diagnostico e prognostico tramite lo sviluppo di un modello software del robot. La possibilità di effettuare, infatti, delle simulazioni congruenti con il comportamento reale del robot garantisce un ampio database, altrimenti insufficiente, da cui attingere informazioni per l'individuazione, ad esempio, di indici dello stato di salute del robot e garantire così una manutenzione efficace ed efficiente.

1.1 La diagnostica e la prognostica nei sistemi ingegneristici

La diagnostica rappresenta l'insieme delle tecniche volte ad analizzare lo stato di salute di un sistema meccanico mediante l'acquisizione e l'elaborazione di dati relativi al funzionamento e ad eventuali guasti dello stesso. Fin dal suo sviluppo, la diagnostica, ha permesso di raccogliere un elevato numero di informazioni tale da permettere di sviluppare, in molti ambiti, tecniche di pianificazione della manutenzione basate sulle attuali condizioni del sistema, la *Condition Based Maintenance* (CBM).

Il monitoraggio delle condizioni permette di applicare una strategia di diagnosi in tempo reale ed è fortemente legato al concetto di prognostica e di studio della salute del sistema, *Prognostic and Health Management* (PHM).

L'obiettivo dello studio prognostico è, infatti, quello di massimizzare la vita utile del sistema e di ridurre al minimo i costi ed i tempi di manutenzione, attraverso l'introduzione di indici di salute misurabili, sostituendo così i vecchi schemi di manutenzione programmata con schemi di manutenzione intelligenti, riducendo il numero degli interventi al minimo, ma garantendo comunque il funzionamento del sistema.

A differenza della diagnostica, la prognostica, consiste nel prevedere il probabile esito di una determinata situazione ed è stata favorita dai recenti progressi nell'ambito del controllo computerizzato.

Lo stato dell'arte attuale, in ambito prognostico, prevede diversi metodi che possono essere più o meno complessi. Tipici esempi di metodi prognostici riguardano lo studio della degradazione delle prestazioni, l'individuazione della vita utile residua, la *Remaining Useful Life* (RUL), e lo studio dei guasti probabilistico [10].

Attraverso la filosofia e il concetto di gestione della salute, i moderni sistemi PHM combinano questi metodi fornendo una serie più ampia di vantaggi rispetto a qualsiasi singolo metodo da solo.

1.1.1 La relazione tra diagnostica e prognostica

Per comprendere il ruolo della prognostica predittiva è necessario comprendere la relazione tra diagnostica e capacità di prognostica. Per far questo si può far riferimento alla sequenza temporale, rappresentata in figura 1.1, che descrive l'evoluzione dello stato di salute di un componente nuovo in buono stato.

La diagnostica è solitamente applicata tra il rilevamento di *fault* e la *failure* del sistema completo.

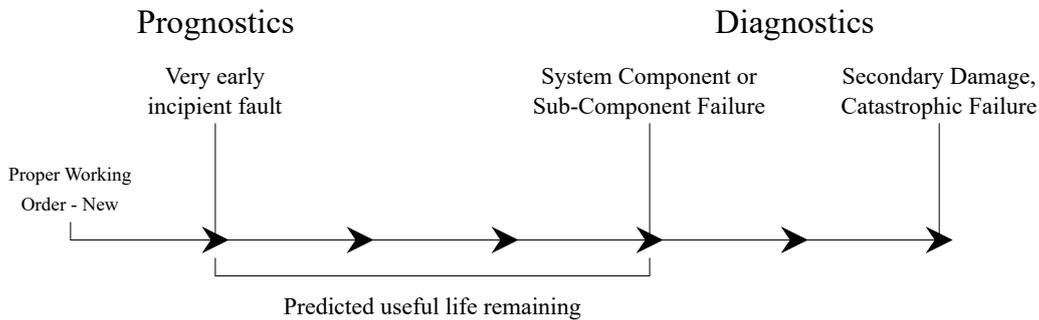


Figura 1.1: Periodi di intervento della prognostica e della diagnostica nell'evoluzione temporale del guasto.

È importante, infatti, nell'ambito dello studio dei guasti, definire la differenza che intercorre tra le parole *fault* e *failure*, che in italiano vengono tradotte entrambe con la parola "guasto". Il primo termine indica, però, un guasto che causa un decremento nella qualità delle operazioni svolte dal sistema, mentre il secondo indica un malfunzionamento tale da rendere il sistema non in grado di svolgere il compito che gli è stato assegnato.

Le tecnologie diagnostiche più recenti consentono di effettuare rilevazioni in stadi di guasto incipiente molto precoci. Al fine di massimizzare i vantaggi della durata operativa continua di un sistema o di un componente del sottosistema, la manutenzione spesso viene ritardata fino a quando il guasto non progredisce in uno stato più grave, ma prima di un evento di guasto effettivo, ed è in questo lasso di tempo che risulta necessario applicare le tecnologie prognostiche, poiché risulta necessario prevedere con una certa precisione la vita rimanente prima della progressione del guasto o della definitiva *failure* [10].

Una parte fondamentale dello sviluppo e dell'implementazione di tecnologie diagnostiche e prognostiche efficaci si basa sulla capacità di rilevare i guasti abbastanza rapidamente da poter effettuare isolamento e diagnosi degli stessi. Un eventuale errore dell'identificazione, e quindi diagnostico, di tali guasti può influire sulla vita di un componente critico e, quindi, dell'intero sistema.

I requisiti di accuratezza prognostica sono direttamente legati alle capacità diagnostiche ed in particolare dipendono dalla possibilità di definire:

- Il livello di degrado delle condizioni operative oltre il quale queste ultime siano considerate insoddisfacenti o inadatte;

- Il tempo minimo di preavviso per fornire all'operatore ed al manutentore le informazioni ed il tempo necessari ad agire sul sistema prima che si verifichi il guasto;
- Un valore minimo di vita utile residua che potrà essere minore o maggiore di un valore minimo di sicurezza.

Da un punto di vista teorico, la maggior efficienza dei sistemi di diagnostica e prognostica si ha quando l'implementazione degli stessi avviene già durante la progettazione del sistema. La realizzazione di tale approccio comporta l'implementazione sinergica di tecnologie di monitoraggio della salute dei componenti e di ragionamento per l'interpretazione degli output di rilevamento dei guasti.

La diagnosi e la prognosi della macchina per CBM coinvolge così un'architettura integrata costituita dai seguenti moduli principali:

- un insieme di sensori e strategie di rilevamento volti a raccogliere ed elaborare i dati legati alle variabili ed ai parametri critici di processo;
- un modulo di analisi dei modi, degli effetti e della criticità dei guasti, *Failure Mode, Effects and Criticality Analysis* (FMECA), che determina e assegna la priorità in base alla frequenza di occorrenza ed alla gravità dei possibili guasti;
- una routine di identificazione della modalità operativa che determina il corrente stato del sistema e le correlate caratteristiche della modalità di guasto in relazione alle condizioni operative;
- un estrattore di caratteristiche che seleziona ed estrae dai dati grezzi degli indicatori della condizione di salute del sistema che vengono poi utilizzati dal modulo diagnostico.
- un modulo diagnostico che valuta lo stato attuale dei componenti critici del sistema.
- un modulo prognostico che stima la rimanente vita utile di un componente prima del guasto;
- il modulo finale dell'architettura integrata, il pianificatore di manutenzione, il cui compito è pianificare le operazioni di manutenzione senza influire negativamente sulle funzionalità complessive del sistema, di cui il componente in questione è solo uno degli elementi costitutivi.

Appare chiaro, da questa architettura, come diagnostica e prognostica siano fondamentalmente interdipendenti.

1.1.2 La diagnostica dei guasti

La diagnostica, a differenza della prognostica, rappresenta un campo molto esplorato dai ricercatori, anche afferenti a diverse discipline. Questo ha permesso di sviluppare diverse metodologie per rilevare guasti o anomalie e decidere se intervenire in base al potenziale impatto causato dagli stessi.

Tale processo di sviluppo ha portato ai moderni algoritmi diagnostici, essi sfruttano dei database formati da una raccolta di dati tipicamente estratti sotto forma di indici. Tali indici vengono utilizzati come input per le routine diagnostiche il cui obiettivo è quello di verificare che non ci si trovi in condizioni di guasto imminente [10].

1.1.3 La prognostica dei guasti

La prognostica, come in parte anticipato in precedenza, si occupa di prevedere in modo accurato e preciso la vita utile rimanente di un componente o di un sottosistema. Rappresenta il passaggio critico del CBM/PHM, risulta, infatti, necessario fare delle previsioni a lungo termine dell'evoluzione del guasto tenendo conto della relativa incertezza, intrinseca a tale metodo e derivante da una pluralità di fonti. Inoltre, una prognosi accurata e precisa richiede buoni modelli probabilistici e campioni statisticamente sufficienti di dati di errore, che nella robotica sono spesso assenti, per convalidare e mettere a punto degli algoritmi prognostici. In robotica i *database*, legati ai guasti, sono piuttosto scarni per via di diversi fattori, legati innanzitutto alla vita utile dei robot piuttosto lunga e dal fatto che sia una tecnologia piuttosto recente.

I principali metodi di prognostica sono detti *data-driven*, in molti casi si dispone infatti di dati storici riguardanti *fault* e *failure*. Partendo da questi è possibile utilizzare degli algoritmi che forniscono gli indici atti a valutare la salute del robot, ad esempio attraverso l'utilizzo di sistemi a rete neurale e sistemi a logica fuzzy.

Il problema, per questo tipo di sistemi, si riduce, quindi, principalmente nell'utilizzare in maniera efficiente i dati disponibili per prevedere i guasti.

Negli ultimi tempi la tecnologia principalmente utilizzata è stata quella delle reti neurali artificiali, le *Artificial Neural Networks* (ANNs), che imparano dagli esempi forniti cercando di definire una relazione funzionale tra i vari dati. Per tale motivo le ANNs sono adatte per problemi pratici, dove è più facile avere dati che definire un modello del sistema oggetto di studio [11].

Una possibile applicazione delle reti neurali nel campo della robotica è rappresentata dall'analisi del rumore e delle vibrazioni che si verificano su

ciascun giunto. Per far questo si misurano preventivamente determinati parametri che servono da campione per analizzare successivamente lo stato di salute del robot. Tali parametri vengono utilizzati dalle reti neurali per prevedere la RUL del robot. Questo tipo di prognostica presenta lo svantaggio legato all'ulteriore costo dei sensori necessari a questo tipo di rilevazioni ed inoltre, nel caso in cui il robot cambi *task* tutta la raccolta di dati effettuata finora è superflua poiché lo spettro di vibrazioni del robot è diverso da quello del task precedente [10].

Gli schemi prognostici *model-based*, invece, sfruttano appunto dei modelli che descrivono il sistema. Tali modelli forniscono un mezzo per definire degli indici, legati ai componenti critici del sistema, e valutarne gli effetti cumulativi in termini di vita utile.

Integrando le tecniche di modellazione fisica e stocastica si può valutare la funzione che definisce la distribuzione della vita utile residua dei componenti in funzione di diversi parametri, quali possono essere l'errore di posizione sistematico del robot o eventuali fluttuazioni nella corrente del motore.

I risultati di tale modello possono quindi essere utilizzati per previsioni prognostiche dei guasti in tempo reale con limiti di confidenza specificati.

Da un punto di vista teorico, l'approccio *model-based* permetterebbe di ottenere un ampio *database* di guasti fornendo così capacità prognostiche complete per tutta la vita del componente, attraverso le quali dedurre le informazioni per la pianificazione dei componenti da ispezionare e dei periodi di revisione.

1.2 La diagnostica e la prognostica nei robot

In molti ambienti industriali, i robot rappresentano un passaggio chiave della catena produttiva ed un eventuale guasto rappresenterebbe una grave perdita in tempi di produttività e, quindi, di denaro.

Come visto in precedenza, la possibilità di adottare una strategia di controllo ed isolamento del guasto nei sistemi ingegneristici, ed in particolare nei robot industriali, è fondamentale per garantirne un'efficiente manutenzione, in maniera da ridurre al minimo eventuali danni e conseguenti perdite di produttività. Essi infatti sono utilizzati soprattutto nelle catene di montaggio ed un eventuale guasto o rallentamento, anche di un solo robot, comporta la comparsa di inefficienze su tutta la linea.

In particolare, per quanto riguarda i robot collaborativi, vi è anche un problema legato alla sicurezza dell'operatore. Essi condividono il loro spazio di lavoro anche con l'essere umano e nel caso di comportamento non nominale possono arrecare danno all'operatore.

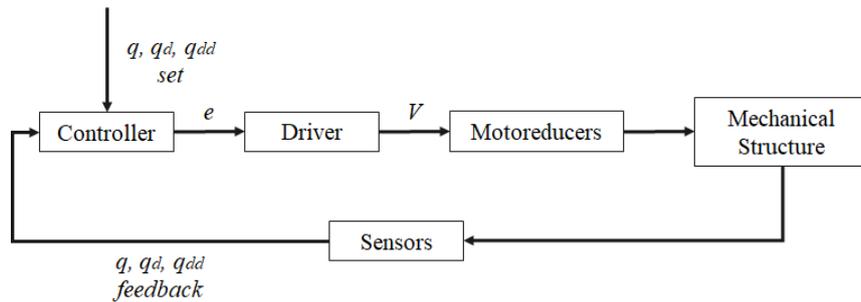


Figura 1.2: Schema generale di robot considerato

Ad oggi le due strade per ottenere dei risultati soddisfacenti, in ambito prognostico, sono rappresentate dall'approccio tramite reti neurali o modelli accurati. Il primo approccio necessita di un'elevata potenza computazionale ed il secondo anche di un modello il quanto più accurato possibile per evitare errori di valutazione della RUL sia in eccesso, portando ad un guasto imprevisto, che in difetto, riducendo in tal maniera l'efficienza del metodo [9].

I possibili guasti che possono verificarsi, in generale, nei robot industriali possono riguardare diversi componenti ed avere, di conseguenza, effetti differenti. Risulta quindi utile analizzare quali possono essere queste categorie di guasti [12]. Per classificarli si fa riferimento ad un generico robot industriale, il cui schema è rappresentato in figura 1.2, si considera ogni giunto azionato da un motore elettrico dedicato e dotato di un sensore di posizione, di velocità e di corrente.

Considerato tale schema i possibili guasti sono i seguenti:

- **Guasto ai sensori:** la posizione, la velocità dei singoli giunti e la corrente dei singoli motori, come detto in precedenza, sono acquisite mediante dei sensori, solitamente encoder o resolver per la posizione, tachimetro per la velocità ed un sensore effetto Hall per la corrente, che chiudono i relativi anelli di retroazione. Un guasto in questi componenti causa letture errate e di conseguenza errori nel controllo;
- **Guasto alla struttura meccanica:** i possibili guasti sono rappresentati dal blocco di un giunto, a causa ad esempio di un guasto ad un riduttore o ai freni, o alla collisione del robot con un oggetto non previsto all'interno dello spazio di lavoro;
- **Guasto al sistema di attuazione:** il sistema di attuazione include il sistema di controllo, i driver ed i motori elettrici. Dall'esperienza si

evincesce che un guasto al sistema di controllo o ai driver causa un valore di riferimento in ingresso al motore differente da quello effettivamente voluto, un guasto invece ai motori causa una coppia differente rispetto a quella comandata dal segnale in ingresso al motore.

Uno dei tipici utilizzi dei robot è rappresentato dai processi di verniciatura ed assemblaggio nell'industria dell'*automotive*, in questo campo sono richieste ai robot elevate precisione e ripetibilità [13]. Per tale motivo risulta importante avere cognizione del tempo medio al verificarsi di un guasto, il *Mean Time To Failure* (MTTF), e del tempo medio che intercorre tra due guasti, il *Mean Time Between Failure* (MTBF), infatti, nonostante ciascun componente abbia una probabilità di *failure* molto bassa, anche una minima *fault* causa un errato svolgimento del compito assegnato al robot.

1.2.1 La prognostica *model-based* nei robot collaborativi

Come visto in precedenza, gran parte delle soluzioni di manutenzione predittiva sono basate su un approccio *data-driven*. Tale metodo sfrutta un *database* che deve, però, essere voluminoso e contenere, oltre che informazioni riguardo al normale funzionamento del robot, anche informazioni riguardo guasti effettivamente avvenuti.

Tali condizioni non risultano rispettate, nel caso dei robot, per tale motivo è necessario sfruttare una metodologia che sostituisca allo storico dati una serie di prove effettuate mediante un modello matematico [9]. L'utilizzo di tecniche di prognostica *model-based*, nel caso dei robot collaborativi, presenta diversi vantaggi:

1. non sono necessari sensori addizionali esclusivi, gli unici segnali che si utilizzano sono, infatti, quelli di corrente, posizione e velocità già misurati dal robot;
2. è possibile tener conto di guasti di diversa natura e variabili nel tempo;
3. è possibile effettuare diagnostica e prognostica in tempo reale mentre il robot è in funzione.

Per ricavare il modello del robot è necessario suddividerlo in diversi sotto modelli descritti mediante equazioni differenziali che possono essere, ad esempio, implementate all'interno di un software per il calcolo numerico [14]. Attraverso tale software si effettuano diverse simulazioni per studiare il comportamento del braccio robotico in caso di guasti di diversa entità e per estrarre così, *features* a fini diagnostici e prognostici.

Capitolo 2

Il robot collaborativo UR5

Il robot che si è scelto di modellare, come caso di studio, è il robot UR5. Questo perché esso è presente all'interno dei laboratori del Politecnico di Torino, ma lo studio che si è effettuato ha, però, carattere generale e le considerazioni e le procedure effettuate possono essere applicate anche ad altri robot.

Il robot collaborativo UR5, riportato in figura 2.1, prodotto dalla Universal Robots, fa parte della CB-series [6]. La gamma UR, oltre che dall'UR5, è costituita da UR3 e UR10, tale differente denominazione fa riferimento alla capacità di carico, in chilogrammi, del robot stesso. In tabella 2.1 sono riportate le principali caratteristiche tecniche che differenziano i vari modelli della CB series.

	Capacità di carico [kg]	Sbraccio [mm]
UR3	3	500
UR5	5	850
UR10	10	1300

Tabella 2.1: Principali caratteristiche tecniche dei modelli della CB series [6].



Figura 2.1: UR5 Robot.

L'UR5 presenta delle caratteristiche [15], peculiari alla famiglia, oltre che al robot stesso, che ne motivano l'ampio impiego nel campo della robotica collaborativa:

- **facilità di programmazione:** legata alla semplicità ed all'efficacia dell'interfaccia robot-utente, il teach pendant;
- **flessibilità:** l'esiguo peso del robot, inferiore a 20 kg, facilita l'installazione sia su superfici orizzontali che verticali e la semplicità di programmazione rende possibile passare da un'attività all'altra velocemente;
- **collaboratività:** la capacità di lavorare a diretto contatto con le persone anche in mancanza di una gabbia di protezione. È possibile infatti applicare dei valori limite delle coppie al giunto per evitare che un'eventuale collisione con un operatore possa risultare fatale;
- **compatibilità con varie periferiche:** vi è la possibilità di installare diversi strumenti come telecamere e sensori di forza, o tool personalizzati;

- **basso rapporto tra massa del robot e carico utile massimo consentito:** pari a circa 3:1, permette l'uso di una macchina relativamente piccola per movimentare un'ampia gamma di carichi.

Materiale	Alluminio e polipropilene
Peso	18.4kg
Sbraccio	850mm
Ripetibilità	$\pm 0.1\text{mm}$
Raggio d'azione giunti	$\pm 360^\circ$
Velocità massima giunti	$\pm 180^\circ/\text{s}$
Range temperatura	$0 \div 50^\circ\text{C}$
Classificazione IP	IP54
Alimentazione elettrica	48V DC
Consumo energia elettrica	Min. 90W Avg. 150W Max. 325W
Porte I/O	Ingressi digitali: 2 Uscite digitali: 2 Ingressi analogici: 2 Uscite analogiche: 0
Vita operativa	35000h

Tabella 2.2: Specifiche tecniche del braccio robotico dell'UR5 [7].

2.1 Il braccio robotico

L'organo principale di un robot collaborativo, e quindi anche dell'UR5, è il braccio robotico, le cui caratteristiche sono riportate in tabella 2.2. La struttura meccanica di un robot è costituita dai *link* e dai giunti. Si tratta di una catena cinematica aperta costituita da sei giunti rotoidali che fanno da collegamento tra un *link* e l'altro. Ciascun *link* presenta una geometria differente e la combinazione di queste definisce le posizioni e le orientazioni che il *Tool Center Point* (TCP) può raggiungere.

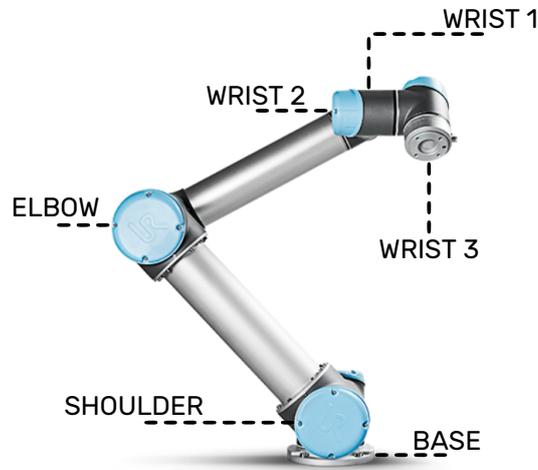


Figura 2.2: Nomenclatura dei giunti del robot UR5.

La geometria dell'UR5 è di tipo antropomorfo, realizzata per mezzo di sei giunti rotoidali, come mostrato in figura 2.2, i primi tre, per la somiglianza con il braccio umano, sono detti base, spalla (*shoulder*) e gomito (*elbow*). Questi si occupano del posizionamento del TCP mentre gli altri, che sono detti polso (*wrist*), si occupano dell'orientazione dell'organo terminale del manipolatore. Una particolarità dell'UR5 è quella di non possedere un polso sferico, cioè non esiste un punto nel quale si intersecano i tre assi del polso. Tale soluzione fa sì che ogni giunto abbia un proprio motore e non sia quindi necessario l'utilizzo di organi di trasmissione.

La struttura antropomorfa è quella che, tra tutte, risulta essere la più versatile e per tale motivo anche la più comune nell'industria manifatturiera [16]. Un ulteriore pregio di tale configurazione è la possibilità di raggiungere un punto, all'interno dell'area di lavoro, da ogni direzione e con qualsiasi orientamento dell'utensile, superando anche eventuali ostacoli presenti all'interno del *workspace*.

Tra le diverse caratteristiche tecniche dell'UR5 è possibile notarne immediatamente due, che risultano essere fondamentali: il peso irrisorio ed il raggio di azione dei giunti. La combinazione di queste due caratteristiche, oltre che favorire l'elevata flessibilità del robot, permette di ottenere un'area di lavoro piuttosto ampia, intorno al robot stesso. Essendo però l'UR5 un

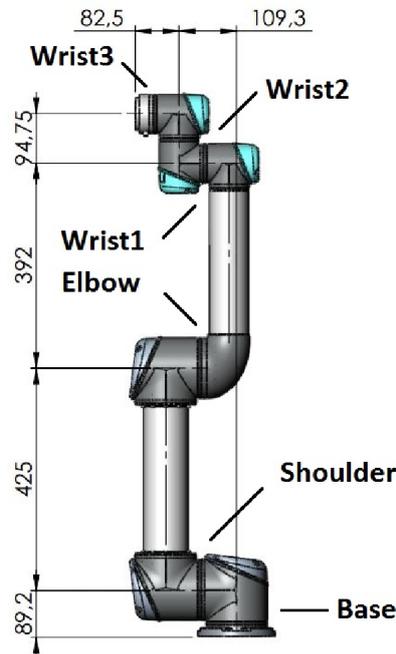


Figura 2.3: Ingombri di massima, in millimetri, dell'UR5.

robot pensato per l'utilizzo all'interno di aree di lavoro collaborative uomo-macchina la velocità massima dei giunti risulta inferiore rispetto a quella di un normale robot industriale, mentre la ripetibilità risulta essere inferiore rispetto a quella dei robot industriali.

Come visto ad inizio capitolo, i robot collaborativi sono caratterizzati da pesi ed ingombri molto contenuti, in particolare per l'UR5 sono riportati in figura 2.3 gli ingombri massimi in *HOME POSITION*¹.

Il peso ridotto è garantito dalla scelta dei materiali e dall'avere i motori direttamente incorporati nei giunti del robot.

Il volume di lavoro dell'UR5, rappresentato in figura 2.4, è di forma sferica, divisibile in due parti concentriche, la più interna, che rappresenta l'area di lavoro consigliata, ed un'altra, più esterna, che rappresenta invece l'area di lavoro massima. Il volume costituito dal cilindro costruito attorno al giunto di base e con raggio di 200mm presenta delle restrizioni, in quanto alcune posizioni ed orientamenti risultano fisicamente irraggiungibili a causa dell'ingombro stesso dei giunti.

¹La "HOME POSITION" è definita da Universal Robots e si ottiene ponendo i giunti rispettivamente a 0°, 90°, -90°, 0°, 0° e 0°

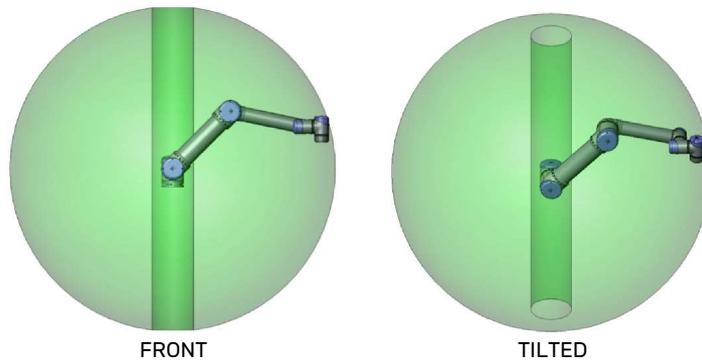


Figura 2.4: Spazio di lavoro dell'UR5.

Universal Robots fornisce la descrizione di tre situazioni principali in cui un robot UR può avere problemi a muoversi da un punto all'altro [17]. Tali situazioni sono denominate singolarità e sono classificate come:

- **limite esterno dello spazio di lavoro:** il robot può raggiungere qualsiasi punto con qualsiasi orientazione, purché si trovi all'interno dell'area consigliata, al contrario, all'esterno della stessa, è possibile raggiungere un certo numero di posizioni, legate all'estensione del robot, ma solo con determinati orientamenti;
- **limite interno dello spazio di lavoro:** tale limite è rappresentato dal volume, visto in precedenza, che si trova in corrispondenza del giunto di base;
- **allineamento del polso:** tale configurazione, come si vede in figura 2.1, si ha quando il giunto del polso 2 risulta allineato con il piano in cui ruotano i giunti di spalla, gomito e polso 1, portandolo ad un angolo di 0° o 180° limitandone i movimenti.

Non vengono forniti da Universal Robots strumenti per un controllo in tempo reale delle singolarità, ma esse sono evidenziate durante la programmazione tramite *teach pendant* e vengono risolte modificando la configurazione del robot o il punto da raggiungere.

2.1.1 I motoriduttori

I motori che monta l'UR5 sono motori brushless senza carter ad azionamento diretto. Si tratta di una versione modificata e pensata appositamente per Universal Robots dei motori della serie KBM di Kollmorgen [18].

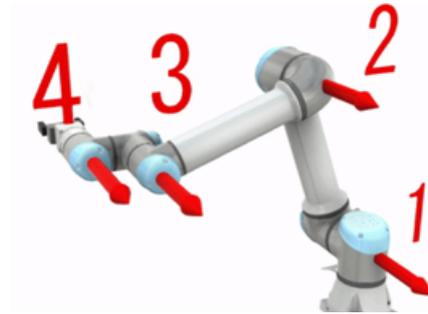


Figura 2.5: Configurazione di allineamento del polso 2 con il piano in cui ruotano i giunti di base, spalla e polso 1 [1].

Giunto	Coppia [Nm]
Base	150
Spalla	150
Gomito	150
Polso 1	28
Polso 2	28
Polso 3	28

Tabella 2.3: Coppie massime applicabili ai giunti dell'UR5 [8].

L'obiettivo di questi motori è quello di seguire la strategia minimalista adottata da Universal Robots, essi sono integrati direttamente nella carcassa del braccio robotico e presentano un'alta densità di potenza garantendo così un'elevata capacità di sollevamento del robot senza influire negativamente su peso e ingombri. I motori della serie KBM offrono un'avanzata progettazione elettromagnetica in modo da avere una coppia ottimizzata ed il minimo di coppia di riluttanza e distorsione armonica.

I motori utilizzati per l'UR5 sono di due taglie differenti, una per i giunti di base, spalla e gomito ed una per i tre giunti del polso. Tali coppie sono poi trasmesse attraverso un riduttore ed assumono i valori massimi riportati in tabella 2.3. Tali valori rappresentano gli unici dati forniti da Universal Robots per quanto riguarda i giunti, tutti gli altri sono stati assunti dal catalogo di Kollmorgen analizzando quali motori potrebbe montare l'UR5.

Un altro vantaggio garantito da questi motori è il funzionamento senza

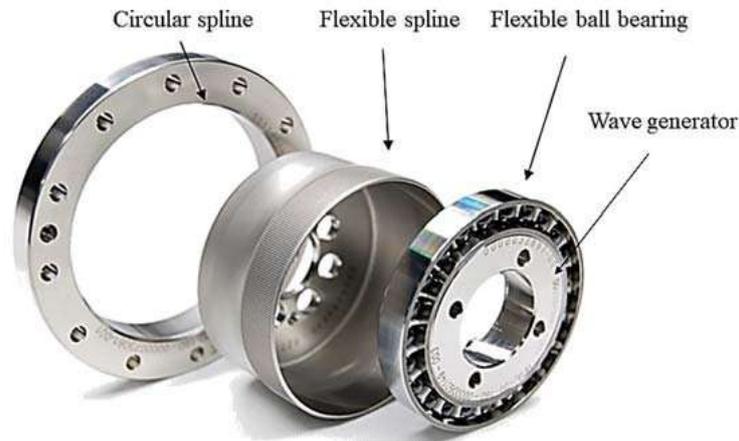


Figura 2.6: Componenti riduttore armonico.

rumore e a risparmio energetico. Direttamente legato a quest'ultimo vi è un ulteriore vantaggio dato dalla miglior efficienza, essa comporta minori perdite in termini di calore e per tale motivo i robot rimangono più freddi, riuscendo così a lavorare sempre sotto i valori nominali massimi di temperatura garantendo una vita di servizio più lunga.

Ciascun motore è accoppiato con un riduttore prodotto dalla Harmonic Drive AG, facente parte della serie HFUS-2SH [19].

In robotica, infatti, la velocità dei giunti è spesso inferiore rispetto a quella dei motori ed i riduttori sono necessari, oltre che per ridurre la velocità, per aumentare la coppia. Per questo tipo di applicazioni le configurazioni classiche dei riduttori, come ingranaggi seriali o planetari, non sono la giusta soluzione perché troppo pesanti, troppo grandi e con un'efficienza troppo bassa. Per tale motivo si utilizzano degli *harmonic drive*, quelli montati sull'UR5 sono caratterizzati da un rapporto di riduzione a singolo stadio 101:1 e forniscono coppie elevate pur pesando solo 1kg con un raggio di 850mm [15].

L'idea degli *harmonic drive* nasce nel 1955 in ambito aerospaziale, ma grazie all'elevato rapporto di trasmissione, il backlash molto ridotto e le dimensioni compatte sono stati gradualmente implementati in vari tipi di dispositivi.

Essi, come si può vedere in figura 2.6, sono costituiti da tre diversi componenti:

- *Wave Generator*: un cuscinetto ellittico, che deforma ellitticamente la *flexspline* facendola avanzare di un dente ad ogni giro, è direttamente

collegato all'albero motore;

- *Flexspline*: un componente con una notevole flessibilità messo in moto dal cuscinetto a sfera;
- *Circularspline*: un anello rigido dentato internamente.

Uno dei maggiori vantaggi dell'uso del riduttore armonico è che ha *backlash* molto ridotto [20], rendendolo ideale, appunto, per le applicazioni ad elevata precisione.

2.2 L'unità di controllo

L'unità di controllo si occupa di definire la traiettoria del robot ed, in generale, di controllarne i movimenti. Essa, come si può vedere in figura 2.7 è costituita da:

- **Scheda madre:** rappresenta il "cervello" effettivo del robot ed è dotata di connettività Ethernet ed USB;
- **Scheda di controllo di sicurezza:** gestisce tutti gli input e output dall'unità di controllo e la connessione alle apparecchiature periferiche, come i sensori dei dispositivi di sicurezza;
- **Modulo USB:** contiene tutto il software, incluso il sistema operativo Linux, l'interfaccia di programmazione Polyscope, brevettata da Universal Robots, e i programmi creati dall'utente.

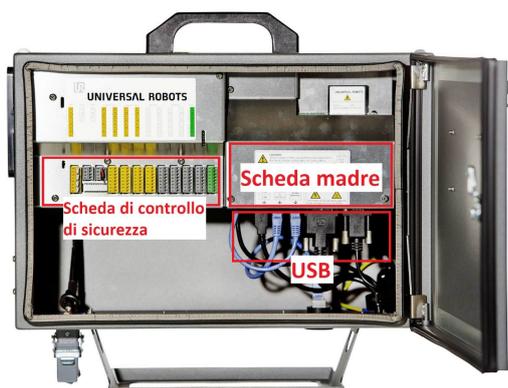


Figura 2.7: Componenti dell'unità di controllo.

Le sue specifiche sono riportate in tabella 2.4.

La comunicazione tra unità di controllo e braccio robotico avviene esclusivamente tramite cavo, in particolare per l'UR5 si utilizza l'UTITRONIC LIYCY (TP) con sezione 2x2x1mm e lunghezza di circa 6m.

Materiale	Alluminio
Peso	15kg
Dimensioni (LxHxP)	475x423x268mm
Classificazione IP	IP20
Range di temperatura	0÷50°C
Alimentazione elettrica	100-240V AC, 50-60Hz
PorteI/O	Ingressi digitali: 16 Ingressi analogici: 16 Uscite digitali: 2 Uscite analogiche:2
Alimentazione elettrica I/O	24V, 2A

Tabella 2.4: Specifiche tecniche dell'unità di controllo dell'UR5 [7].

2.3 Il teach pendant

Il *teach pendant*, rappresentato in figura 2.8, è costituito principalmente da un *touch screen* da 12" che ospita l'interfaccia utente di programmazione basata su sistema operativo Linux. Esso permette di programmare e comandare il robot mediante la scrittura di codici e verificare che il programma venga eseguito correttamente.

All'interno del *workspace* il robot si muove lungo una traiettoria ottenuta definendo un certo numero di *waypoints*, tali punti sono individuati attraverso la programmazione al *teach pendant*.

Attraverso il *teach pendant* si definiscono anche la velocità e l'accelerazione del TCP e dei singoli giunti. Una volta definiti i *waypoints*, l'unità di controllo si occupa di calcolare la traiettoria mediante un profilo di velocità trapezoidale.

Il profilo trapezoidale, di cui vi è un esempio in figura 2.9, è il più utilizzato nella pratica industriale e consiste in tre tratti differentemente definiti:

- **tratto ad accelerazione costante:** la velocità cresce in modo lineare, tramite la caratteristica forma a rampa, mentre la posizione ha un andamento parabolico;
- **tratto a velocità di crociera:** la velocità è costante e la posizione cresce in maniera lineare;



Figura 2.8: Teach pendant.

- **tratto a decelerazione costante:** la velocità decresce linearmente e la posizione assume, anche in questo caso, un andamento parabolico, generalmente ha una durata uguale a quella del tratto di accelerazione, ottenendo così un profilo simmetrico.

La programmazione mediante *teach pendant* può avvenire, oltre che tramite la definizione dei *waypoints*, anche grazie ad una porta USB attraverso la quale si possono importare o esportare file.

Oltre allo schermo è dotato di tre pulsanti fisici che si occupano di:

- **On/Off:** dedicato ad accensione, spegnimento e riavvio del sistema;
- **Emergenza:** aziona i freni del robot e scollega l'alimentazione elettrica;
- **Freedrive:** si trova sul retro del *teach pendant* e permette all'operatore di muovere manualmente il robot nelle posizioni desiderate. È necessario tenerlo premuto per tutto il tempo in cui si vuole utilizzare tale modalità ed è particolarmente utile durante la procedura di definizione dei *waypoints*.

Come detto in precedenza il *teach pendant* non è utile solo per programmare i movimenti del robot, ma anche nel caso in cui si verifichi un errore. Tale errore viene visualizzato a schermo e classificato in base alla tipologia.

L'ambiente di programmazione dell'UR5, PolyScope, è un ambiente di programmazione grafico che consente di configurare, controllare e gestire il robot.

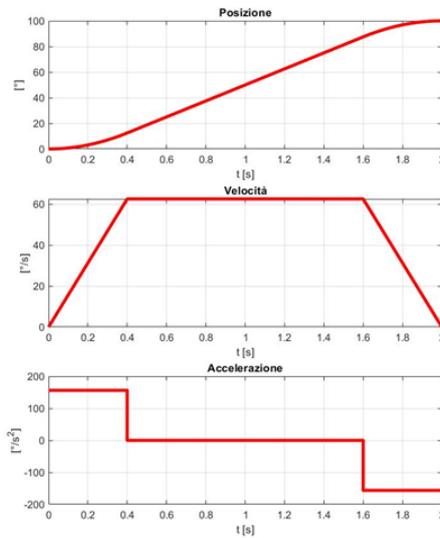


Figura 2.9: Esempio legge del moto con profilo di velocità trapezoidale.

Materiale	Alluminio e propilene
Peso	1,5kg
Lunghezza cavo	4,5m
Classificazione IP	IP20

Tabella 2.5: Specifiche tecniche del *teach pendant* [7].

La programmazione della traiettoria tramite *teach pendant* può avvenire in tre diverse modalità:

- **MoveL:** permette di spostare il TCP in modo lineare lungo la traiettoria. Di conseguenza, il movimento che ogni giunto esegue è più complicato per poter muovere il *tool* lungo una linea retta;
- **MoveJ:** muove i giunti in modo che questi raggiungano la posizione finale tutti nello stesso momento. Di conseguenza, l'utensile si muove secondo una traiettoria curvilinea. Questo movimento è da preferire quando non è importante il percorso seguito dall'utensile, bensì è necessario che il braccio robotico si muova rapidamente tra i vari *waypoints*;

- **MoveP**: fa in modo che il tool si muova con velocità costante, secondo traiettorie circolari. La dimensione del raggio di transizione è predefinita ed è un valore condiviso da tutti i *waypoints*.

2.4 I sensori

Come visto nel capitolo 1, i robot industriali montano tre diversi tipi di sensori volti a misurare la posizione angolare dei giunti, la velocità dell'albero motore e la corrente che scorre all'interno del motore.

La posizione angolare è misurata per mezzo di *encoder* magnetici assoluti, riportati in figura 2.10 prodotti da Renishaw™[21].



Figura 2.10: *Encoder* rotativo magnetico assoluto AskIM.

L'*encoder* è collegato a valle del riduttore per misurare direttamente la rotazione effettiva del giunto del robot. Questo approccio, a differenza di quello che prevede di montare il sensore a monte del riduttore, permette di eliminare l'errore sistematico causato dalla presenza del riduttore stesso fornendo al robot una ripetibilità di $\pm 0.1^\circ$ sufficiente a soddisfare la maggior parte delle esigenze del mercato.

Gli *encoder* sono specificatamente progettati in forma compatta per essere montati sul riduttore aumentando lo spessore del giunto di soli 7mm.

Un secondo *encoder* è montato tra motore e riduttore e si occupa di misurare la velocità dell'albero motore e chiudere il relativo anello di retroazione.

Capitolo 3

Descrizione del modello dell'UR5

Come detto in precedenza, per effettuare un tipo di prognostica *model-based*, è necessario definire un modello *high-fidelity* del robot che possa fornire indicazioni riguardo il comportamento del robot stesso in presenza di una certa *fault*.

Come visto nella sezione 1.2 è possibile suddividere il comportamento del robot mediante tre sottosistemi principali, rappresentati attraverso altrettanti modelli, a cui sono associati dei guasti specifici: i sensori, la struttura meccanica ed il sistema di attuazione.

L'ambiente di simulazione scelto per tale fase è MATLAB/Simulink, un software sviluppato dalla compagnia statunitense MathWorks e pensato per la modellazione, la simulazione e l'analisi di sistemi dinamici. Nel presente capitolo si illustreranno le modalità con cui si è giunti alla definizione di tali modelli e come essi interagiscono tra di loro.

Nella definizione del modello, oltre che dell'input bisogna tener conto di quali sono le grandezze effettivamente misurabili dall'UR5 presente nel laboratorio di Ingegneria Meccanica e Aerospaziale del Politecnico di Torino, senza l'aggiunta di ulteriori sensori. Ciò permetterà, in una fase successiva di analisi, di verificare qual è l'effetto di uno specifico guasto su queste grandezze ed andare così a definire lo stato di salute del robot in base alle stesse.

Per l'identificazione del modello sono stati utilizzati i dati di una prova di un test di *pick-and-place* effettuata sul robot presente in laboratorio. I dati che possono essere prelevati dal robot, oltre naturalmente agli input, sono le grandezze angolari di ciascun giunto e la corrente che scorre in ciascun motore. Tali grandezze possono essere prelevate dai sensori del robot con una frequenza di 125Hz, nel caso dei robot della *Universal Robots*TM della e-

series, invece, è possibile leggere le stesse grandezze con una frequenza di campionamento di 500Hz.

Non è possibile, invece, avere una misura diretta delle coppie ai giunti poiché non sono presenti sensori di coppia e per tale motivo esse sono ricavate a partire dall'intensità di corrente, fornita al motore, sapendo che vale la seguente equazione:

$$T = i \cdot k_c \cdot \tau \cdot \eta; \quad (3.1)$$

Dove i simboli utilizzati rappresentano:

- T_u , la coppia ai giunti;
- i , la corrente che scorre nel motore;
- k_c , la costante di coppia;
- τ , il rapporto di riduzione.

Il modello completo del robot deve occuparsi di simulare sia la componente software che quella hardware, infatti, mentre l'output del sistema è rappresentato, sostanzialmente, dal movimento del robot, l'input è dato da una certa traiettoria di *set* che i giunti del robot devono seguire, la quale può essere definita attraverso diversi comandi, come visto nel capitolo 2.

L'intero modello può essere avviato da uno script MATLAB, [UR5_main.m](#), che si occupa di richiamare le diverse funzioni necessarie alla simulazione e ad avviare la stessa, oppure direttamente all'interno del simulatore con gli appositi comandi.

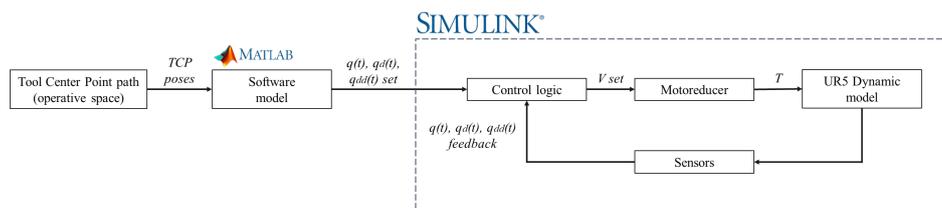
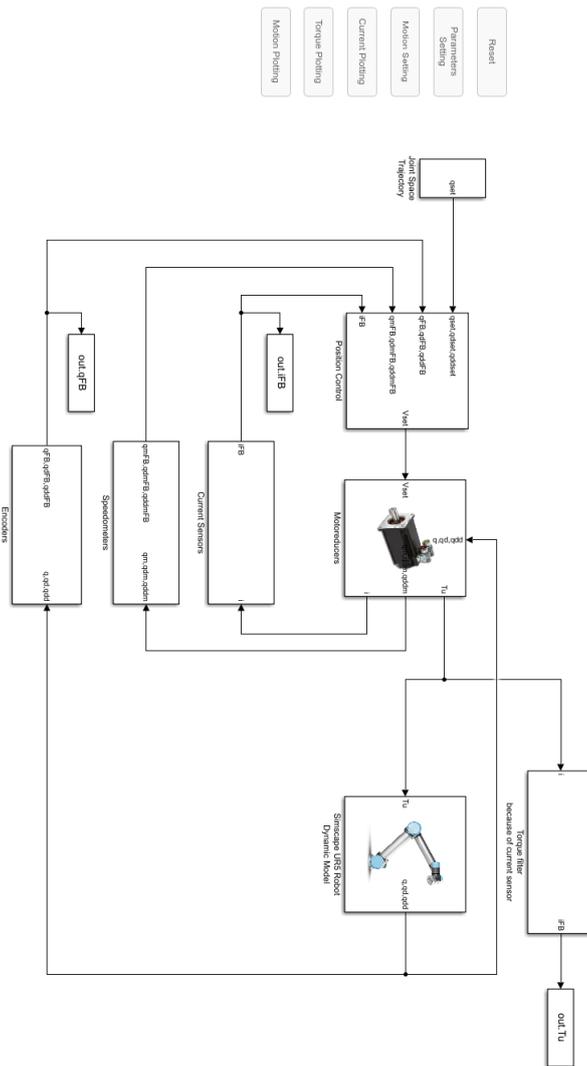


Figura 3.1: Schema a blocchi modello UR5.

Considerando il modello nella sua interezza, esso può essere rappresentato, attraverso uno schema a blocchi, come in figura 3.1. Il blocco che riceve il primo input è il modello della parte software, essa, in maniera differente, a seconda del tipo di dati forniti in input, attraverso la cinematica inversa del manipolatore fornisce le leggi del moto dei giunti al modello Simulink. All'interno del modello Simulink la logica di controllo elabora tali leggi, confrontandole con il *feedback* che viene dai sensori e fornisce una tensione di *set* (V set) ai motoriduttori. Questi a loro volta forniscono una coppia (T) al modello dinamico della struttura meccanica, dal quale si ottengono in uscita i valori reali delle grandezze angolari di ogni giunto, permettendo così di chiudere i tre anelli di retroazione di posizione angolare del giunto e velocità e corrente del motore, attraverso i sensori.

In figura 3.2 è rappresentato il modello completo all'interno di *Simulink*TM. Esso rispecchia il modello presentato in figura 3.1 riportato nell'ambiente di simulazione. Oltre al modello stesso è possibile individuare sei pulsanti che permettono, rispettivamente, di resettare i parametri del modello ("*Reset*"), caricare i parametri inserite nell'apposito script ("*Parameters Setting*"), caricare la legge del moto angolare voluta ("*Motion Setting*") ed ottenere i grafici comparati di corrente , coppia e moto ("*Data Plotting*").



- Robot
- Parameters Setting
- Motion Setting
- Current Pickling
- Torque Pickling
- Motion Pickling

Figura 3.2: Schema a blocchi modello UR5 in Simulink.

3.1 La cinematica inversa e la definizione delle leggi del moto

La simulazione della componente di programmazione software del robot è demandata ad uno script, `UR5.simmot.m`, all'interno del quale è possibile programmare il robot in diversi modi a seconda di quali sono gli input di cui si è a disposizione, ottenendo comunque le variabili angolari di *set* da fornire al modello *Simulink*TM.

In figura 3.3 è rappresentato, sotto forma di schema a blocchi, l'algoritmo che permette di ottenere le leggi del moto di *set* a partire da un insieme di pose qualsiasi.

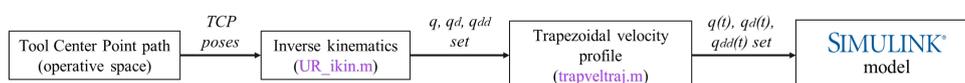


Figura 3.3: Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalla definizione delle pose del TCP.

In questo primo caso, che rappresenta la possibilità di fornire al robot un percorso qualsiasi, non necessariamente testato, le pose del TCP sono elaborate da una funzione, `posset.m`, che riceve in ingresso un vettore di matrici formato, appunto, dalle diverse pose, un vettore contenente le variabili di giunto al tempo iniziale, uno scalare che indica il numero di punti intermedi tra una posa e l'altra ed un vettore che contiene, invece, gli intervalli di tempo.

L'output della funzione è ricavato sfruttando altre due funzioni, in particolare, la funzione `UR_ikine.m`, rielaborazione di una funzione appartenente all'*UR Toolbox*TM, che si occupa della cinematica inversa e la funzione `trapveltraj.m`, del *Robotics System Toolbox*TM, che permette di generare traiettorie con profili di velocità trapezoidali [22][23][4].

La funzione che risolve la cinematica inversa fornisce, come risultati, otto possibili diverse soluzioni che risolvono tutte la medesima posa del TCP. Per ottenere un moto del TCP congruente è stato necessario introdurre un algoritmo di scelta che definisse le posizione angolari ad un certo istante t in modo che tale soluzione, tra le otto possibili, fosse quella più vicina alla configurazione angolare assunta dal robot all'istante $t - 1$.

Le diverse configurazioni angolari, ottenute per istanti di tempo discreti, sono invece utilizzate dalla seconda funzione per ricavare l'output, che è rappresentato da una struttura che contiene, per ciascuna variabile di giunto,

un vettore tempo ed una matrice formata dai valori di posizione, velocità ed accelerazione angolari per ogni istante di tempo.

Nel caso in cui si abbia a disposizione un test effettuato sul robot si può agire in due modi differenti a seconda se si fornisce allo script, per ogni istante di tempo, l'insieme delle pose del TCP o le configurazioni angolari.

Nel primo caso, rappresentato in figura 3.4, le pose vengono elaborate da una funzione `posset2.m` che mediante l'uso della funzione `UR_ikine.m` fornisce la posizione angolare, per ogni istante di tempo, corrispondente alle pose date in input.

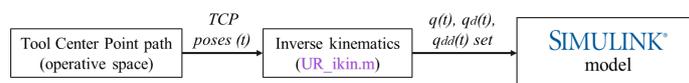


Figura 3.4: Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalla definizione delle pose del TCP in funzione del tempo.

Al contrario, come si vede in figura 3.5, se si fornisce come input l'insieme delle posizioni angolari, esse sono elaborate da una funzione `testlaw.m` e sono pronte per essere fornite direttamente al modello Simulink.



Figura 3.5: Schema a blocchi del modello del software per ricavare le leggi del moto dei giunti a partire dalle configurazioni angolari.

In ogni caso i valori di *set* sono poi forniti in ingresso al modello Simulink come valori di riferimento all'interno del blocco mostrato in figura 3.6.

Figura 3.6: Sottosistema *Joint Space Trajectory*.

3.2 Il modello dinamico in ambiente Simscape Multibody

La struttura meccanica dell'UR5, già descritta nella sezione 2.1, è simulata mediante un modello di natura dinamica che sfrutta le potenzialità dell'estensione Simscape Multibody di Simulink.

Simscape Multibody è un ambiente di simulazione *multibody*, la cui logica si basa sul principio di modellare il sistema come un insieme di corpi rigidi connessi tra loro mediante coppie cinematiche, tale ambiente è pensato appositamente per la simulazione di sistemi meccanici 3D come robot, sospensioni di veicoli, apparecchiature per l'edilizia e carrelli di atterraggio per aeromobili. Esso presenta inoltre il vantaggio di poterlo integrare con altri componenti della famiglia Simscape come, ad esempio, Simscape Electrical.

Tutto il sistema meccanico, come si può vedere in figura 3.7, è collegato ai tre blocchi fondamentali di Simscape, ovvero "*Solver*", "*World frame*" e "*Mechanism configuration*", questi blocchi definiscono l'ambiente e le sue proprietà, come ad esempio la gravità. Immediatamente dopo questi blocchi vi è un blocco "*Solid*" che costituisce il basamento sul quale è fissato il robot, tale collegamento è simulato mediante un "*Welding Joint*", il quale non fornisce alcun grado di libertà, ma può risultare utile per ricavare le interazioni, in termini di coppie e forze scaricate a terra, dal robot.

Il modello della struttura è poi racchiuso in unico blocco, rappresentato

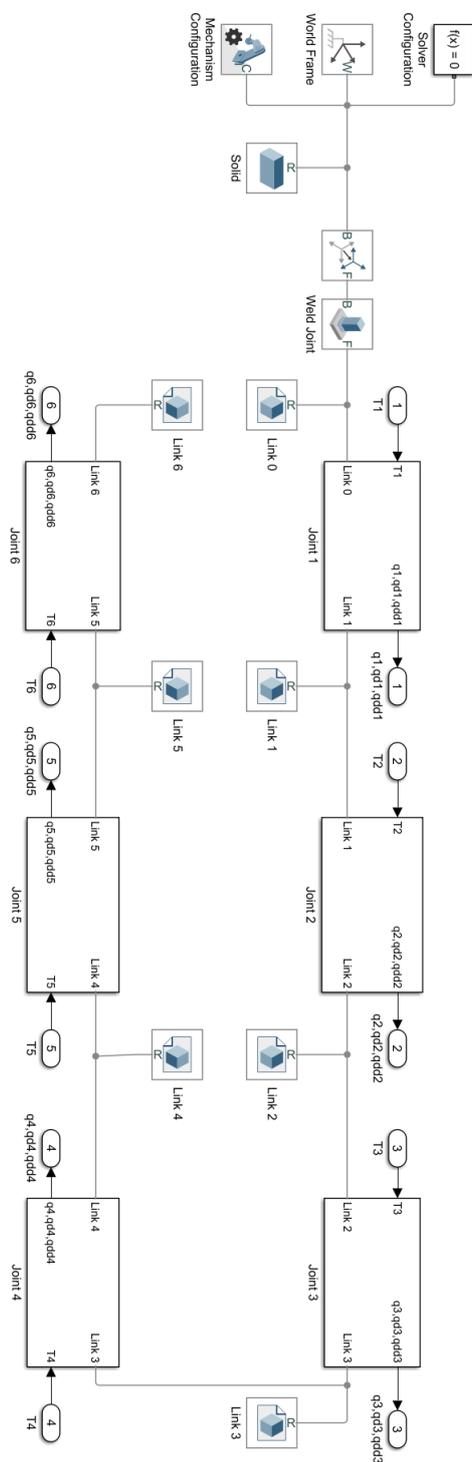


Figura 3.7: Sottosistema "Simscape UR5 Robot Dynamic Model" espanso.

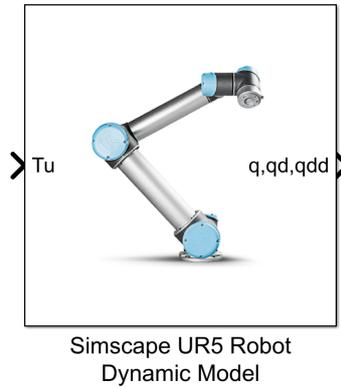


Figura 3.8: *Simscape UR5 Robot Dynamic Model*.

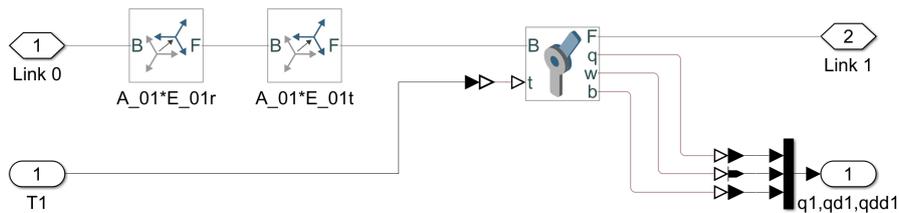


Figura 3.9: Sottosistema "Joint 1".

in figura 3.8. Il modello meccanico dinamico è costituito da sei sottosistemi, uno per ogni giunto, che si interpongono tra due *link* successivi. Come si può vedere in figura 3.9, ogni sottosistema presenta quattro porte: due ("Link 0" e "Link 1"), che rappresentano la connessione fisica, sono collegate al *link* precedente ed a quello successivo, una ("T1") che riceve la coppia in ingresso al giunto ed una ("q1,qd1,qdd1") che fornisce in uscita i valori reali di posizione, velocità ed accelerazione angolari del giunto [24][25].

All'interno del sottosistema di ciascun giunto, come si vede nell'espanso del primo giunto in figura 3.9, sono inserite due trasformazioni rigide, una rotazione ed una traslazione, definite attraverso i parametri di Denavit-Hartenberg (D-H) [2].

Il metodo di D-H è un metodo che permette di descrivere la cinematica del

robot assegnando ad ogni link quattro parametri che descrivono la posizione del sistema di riferimento di ciascun giunto del braccio robotico. L' *Universal Robots*TM utilizza la convenzione di D-H standard [26], secondo cui il giunto i -esimo collega il link i -esimo ed il successivo, mentre l'asse di rotazione (z_i) è allineato con l'asse del giunto $i + 1$ [2]. I quattro parametri di D-H, definiti anche in figura 3.10, sono:

- d_i : la distanza tra l'asse x_i e l'asse x_{i-1} , misurata lungo la direzione positiva di z_{i-1} ;
- a_i : la distanza tra l'asse z_i e l'asse z_{i-1} , misurata lungo la direzione positiva di x_i ;
- α_i : l'angolo tra l'asse z_{i-1} e l'asse z_i , è l'angolo antiorario, attorno all'asse x_i , di cui deve ruotare l'asse z_{i-1} per allinearsi con l'asse z_i ;
- θ_i : l'angolo tra l'asse x_{i-1} e l'asse x_i , è l'angolo antiorario, attorno all'asse z_i , di cui deve ruotare l'asse x_{i-1} per allinearsi con l'asse x_i .

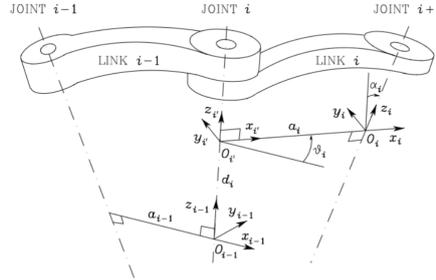


Figura 3.10: Parametri di Denavit-Hartenberg [2].

I parametri di D-H, forniti da *Universal Robots*TM per l'UR5, sono riportati in tabella 3.1.

A partire dai parametri così definiti è possibile costruire una matrice di rototraslazione omogenea per passare dal sistema di riferimento del giunto $i - 1$ a quello del giunto i -esimo:

$$A_{i-1}^i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \cdot \cos(\alpha_i) & \sin(\theta_i) \cdot \sin(\alpha_i) & a_i \cdot \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \cdot \cos(\alpha_i) & -\cos(\theta_i) \cdot \sin(\alpha_i) & a_i \cdot \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Joint	$d_i[m]$	$a_i[m]$	$\alpha_i[^\circ]$	$\theta_i[^\circ]$
Base	0.089159	0	90	q_1
Shoulder	0	-0.425	0	q_2
Elbow	0	-0.39225	0	q_3
Wrist 1	0.10915	0	90	q_4
Wrist 2	0.09465	0	-90	q_5
Wrist 3	0.0823	0	0	q_6

Tabella 3.1: Parametri di Denavit-Hartenberg dell'UR5.

Tenendo a mente l'obiettivo ultimo della definizione del modello, che è quello di poter iniettare guasti ben definiti all'interno del sistema, le matrici di trasformazione omogenea di D-H sono moltiplicate per delle matrici di errore che contengono quelli che sono errori geometrici di diversa natura, legati, ad esempio, ad eventuali errori di montaggio o di produzione [3], tale relazione è espressa nell'equazione 3.3, ed il significato dei parametri utilizzati è rappresentato in figure 3.11. Ogni coppia di matrici² viene espressa in funzione delle variabili angolari e dei parametri di D-H.

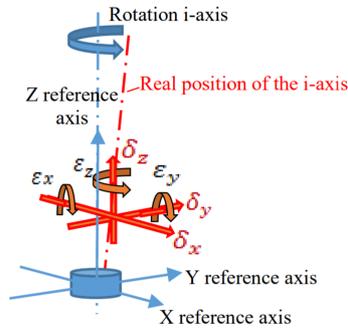


Figura 3.11: Parametri di errore di un asse di rotazione [3].

²Le notazioni s_θ e c_θ sono, rispettivamente, l'abbreviazione per $\sin\theta$ e $\cos\theta$, abbreviazioni adottate da qui in avanti.

$$H_0^6 = A_0^1 E_0^1 A_1^2 E_1^2 A_2^3 E_2^3 A_3^4 E_3^4 A_4^5 E_4^5 A_5^6 E_5^6 \quad (3.3)$$

$$A_{i-1}^i E_{i-1}^i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} \cdot c_{\alpha_i} & s_{\theta_i} \cdot s_{\alpha_i} & a_i \cdot c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} \cdot c_{\alpha_i} & -c_{\theta_i} \cdot s_{\alpha_i} & a_i \cdot s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\varepsilon_z(\theta) & \varepsilon_y(\theta) & \delta_x(\theta) \\ \varepsilon_z(\theta) & 1 & -\varepsilon_x(\theta) & \delta_y(\theta) \\ -\varepsilon_y(\theta) & \varepsilon_x(\theta) & 1 & \delta_z(\theta) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Attualmente questo tipo di approccio permette di introdurre delle variazioni alle matrici di rototraslazione legate a diversi tipi di errore non solo geometrici, come ad esempio backlash e carichi esterni. Tali errori sono espressi mediante delle funzioni delle variabili di giunto.

Nel caso oggetto di studio vengono inseriti nella matrice solo errori geometrici costanti per cui l'espressione diviene indipendente dalle variabili di giunto e cambia come di seguito:

$$A_{i-1}^i E_{i-1}^i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} \cdot c_{\alpha_i} & s_{\theta_i} \cdot s_{\alpha_i} & a_i \cdot c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} \cdot c_{\alpha_i} & -c_{\theta_i} \cdot s_{\alpha_i} & a_i \cdot s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\varepsilon_z & \varepsilon_y & \delta_x \\ \varepsilon_z & 1 & -\varepsilon_x & \delta_y \\ -\varepsilon_y & \varepsilon_x & 1 & \delta_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

Tale procedura adottata, non rappresenta però una semplificazione della modellazione degli errori, ma anzi evita di includervi gli errori non geometrici, dipendenti dalle variabili di giunto ed in generale dal tempo. Essi sono, infatti, analizzati mediante il restante modello *high-fidelity* del robot.

Attualmente, una approccio tipico nella modellazione dei guasti, è quello di approssimare gli errori, geometrici e non, mediante dei polinomi di Chebyshev di sesto grado, con relative approssimazioni.

Per tale motivo, uno degli obiettivi del lavoro è quello di valutare gli errori non geometrici alla loro fonte e descriverne così il comportamento in funzione della configurazione angolare dei giunti, confrontando poi tali risultati con le approssimazioni presenti in letteratura.

In linea di principio sarebbe possibile fornire traslazione e rotazione attraverso una sola trasformazione rigida, in tal maniera però il software elabora la trasformazione in maniera diversa rispetto a quanto avviene applicando la matrice di D-H, per tale motivo ci sarà una prima trasformazione rigida che applica una matrice di rotazione ed una seconda trasformazione rigida che applica un vettore di traslazione.

La matrice di rotazione è rappresentata dalla sottomatrice 3x3 della matrice di rototraslazione, mentre il vettore di traslazione è dato dalla sottomatrice 3x1.

Come detto, dopo le due trasformazioni, è presente il giunto rotoidale, il quale riceve in ingresso una coppia e fornisce in uscita la legge del moto del giunto stesso, sotto forma di posizione, velocità ed accelerazione angolari.

Ciascun *link* è costituito da un corpo solido le cui proprietà geometriche e grafiche sono estrapolate da un relativo file ".STEP", mentre le proprietà di inerzia, quali massa, centro di massa e matrice d'inerzia, sono fornite come variabili definite all'interno dello script che fornisce i parametri del sistema [UR5_simpar.m](#). Il modello così costituito permette di avere anche una simulazione grafica del robot, in figura 3.12 si può vedere il robot all'interno dell'ambiente di simulazione.

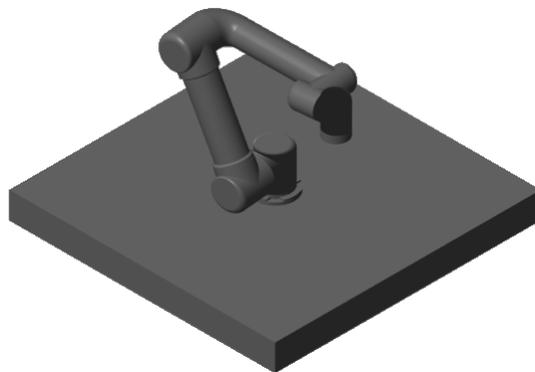


Figura 3.12: Modello all'interno dell'ambiente *Mechanics Explorer*.

*Universal Robots*TM fornisce alcuni parametri riguardanti l'inerzia, quali masse dei *link* e posizione del centro di massa rispetto al sistema di riferimento di D-H. Questi parametri sono riportati in tabella 3.2.

Gli altri parametri di inerzia necessari a definire in maniera completa il modello dinamico sono le matrici di inerzia [27].

Link	Massa [kg]	Posizione del centro di massa [m]
1	3.7	[0,-0.02561,0.00193]
2	8.393	[0.2125,0,11336]
3	2.33	[0.15,0,0.0265]
4	1.219	[0,-0.0018,0.01634]
5	1.219	[0,0.0018,0.01634]
6	0.1879	[0,0,0.001159]

Tabella 3.2: Massa e posizione del centro di massa dei *link* dell'UR5.

$$I_1 = \begin{bmatrix} 0.0067 & 0 & 0 \\ 0 & 0.0064 & 0 \\ 0 & 0 & 0.0067 \end{bmatrix}, I_2 = \begin{bmatrix} 0.0149 & 0 & 0 \\ 0 & 0.3564 & 0 \\ 0 & 0 & 0.3553 \end{bmatrix},$$

$$I_3 = \begin{bmatrix} 0.0025 & 0 & 0.0034 \\ 0 & 0.0551 & 0 \\ 0.0034 & 0 & 0.0546 \end{bmatrix}, I_4 = \begin{bmatrix} 0.0012 & 0 & 0 \\ 0 & 0.0012 & 0 \\ 0 & 0 & 0.0012 \end{bmatrix},$$

$$I_5 = \begin{bmatrix} 0.0012 & 0 & 0 \\ 0 & 0.0012 & 0 \\ 0 & 0 & 0.009 \end{bmatrix}, I_6 = \begin{bmatrix} 0.0001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{bmatrix}.$$

Risulta inoltre necessario fornire i limiti estremi delle variabili di giunti, anch'essi forniti da *Universal Robots*TM e riportati in tabella 3.3.

Link	$q[\text{rad}]$	$\dot{q}[\text{rad/s}]$	$\ddot{q}[\text{rad/s}^2]$
1	$\pm 2\pi$	$\pm \pi$	$\pm \pi$
2	$\pm 2\pi$	$\pm \pi$	$\pm \pi$
3	$\pm 2\pi$	$\pm \pi$	$\pm \pi$
4	$\pm 2\pi$	$\pm \pi$	$\pm \pi$
5	$\pm 2\pi$	$\pm \pi$	$\pm \pi$
6	$\pm 2\pi$	$\pm \pi$	$\pm \pi$

Tabella 3.3: Limiti di posizione, velocità ed accelerazione dei giunti.

3.2.1 Il sensore di coppia

L'esemplare di UR5 presente all'interno del laboratorio presenta un sensore di coppia montato al posto dell'utensile.



Figura 3.13: Sensore di coppia FT300.

Tale sensore di coppia è l'FT 300 rappresentato in figura 3.13, anche montato sull'UR5. Per modellarne la presenza all'interno del blocco della dinamica si è inserito un blocco solido a valle dell'ultimo link, mediante una trasformazione rigida che sposta il sistema di riferimento, tale variazione del blocco che elabora la dinamica è rappresentata in figura 3.14.

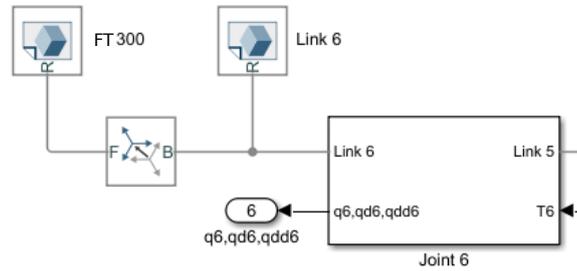


Figura 3.14: Modifica del blocco di dinamica per inserire il sensore di coppia.

Per modellare il sensore è necessario definire i parametri che riguardano la sua massa, 0.3kg, e la sua inerzia, rappresentata dalla seguente matrice.

$$I_1 = \begin{bmatrix} 0.000262 & 0 & 0.000001 \\ 0 & 0.000265 & 0.000001 \\ 0.000001 & 0.000001 & 0.000219 \end{bmatrix},$$

3.3 Il motore monofase e la logica di controllo

Il sistema di attuazione è costituito dai motoriduttori e dalle logiche di controllo degli stessi. Come detto nella sezione 2.1.1 l'UR5 monta dei motori brushless a corrente alternata e dei riduttori armonici.

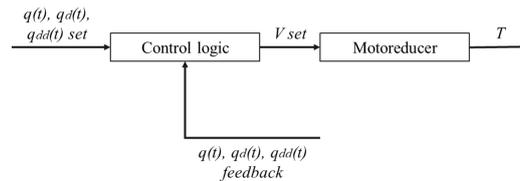


Figura 3.15: Modello del sistema di attuazione.

A prescindere dal livello di dettaglio della modellazione il sistema di attuazione, rappresentato in figura 3.15, riceve in ingresso i segnali di *set* e *feedback* angolari e fornisce in uscita le coppie ai giunti.

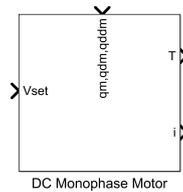


Figura 3.16: Modello motore monofase a corrente continua.

In questa prima fase si è sostituito il motore con il modello di un semplice motore a corrente continua ed il riduttore è stato modellato attraverso il rapporto di riduzione e l'efficienza del riduttore stesso, assunta, in questa fase preliminare, unitaria. Il motore a corrente continua, come si vede in figura 3.16, riceve in ingresso il *set* di tensione (" V_{set} ") proveniente dal controllore e le variabili angolari dell'albero del motore stesso (" q_m, q_{dm}, q_{ddm} ") provenienti dal riduttore. In uscita, invece, esso fornisce la corrente che lo attraversa (" i ") e la coppia a monte del riduttore (" T "). Il riduttore riceve, infatti, in ingresso, la coppia proveniente dal motore e le variabili angolari provenienti dal modello della struttura meccanica ed in uscita fornisce le variabili angolari dell'albero del motore e la coppia ai giunti. I parametri del motore, indicati in tabella 3.4, sono stati ricavati dal catalogo di Kollmorgen dopo aver ipotizzato, a partire dalla taglia, quali potessero essere i motori montati dall'UR5.

Size 3 (Base, Shoulder, Elbow)	
Momento d'inerzia	187.740kgmm ²
Resistenza equivalente	0.3Ω
Induttanza equivalente	0.83mH
Costante di coppia	Base: 0.1350Nm/A Shoulder: 0.1361Nm/A Elbow: 0.1355Nm/A
Size 1 (Wrist)	
Momento d'inerzia	20.767kgmm ²
Resistenza equivalente	1.65Ω
Induttanza equivalente	2.5mH
Costante di coppia	Wrist1: 0.0957Nm/A Wrist2: 0.0865Nm/A Wrist3: 0.0893Nm/A

Tabella 3.4: Grandezze motori UR5.

Le equazioni che descrivono il funzionamento di un motore elettrico DC sono le equazioni di una spira immersa in un campo magnetico prodotto da due magneti permanenti [28]. La coppia che subisce tale spira è legata alla forza di Lorentz e risulta pari a:

$$T_m = k_c \cdot i; \quad (3.5)$$

Ove si ha:

- T_m : la coppia in uscita dal motore;
- k_c : la costante di coppia;
- i : la corrente che scorre all'interno della spira

Per la legge di Lenz il conduttore vede un campo magnetico variabile e quindi risente di una tensione indotta, una forza contro-elettromotrice, essendo anch'esso in movimento.

$$e = k_v \cdot \omega; \quad (3.6)$$

Dove i diversi parametri assumono i seguenti significati:

- e : la forza contro-elettromotrice;
- k_v : la costante elettrica;
- ω : la velocità di rotazione della spira.

Per calcolare la corrente elettrica che scorre all'interno del motore si sfrutta un'equazione scritta sul circuito dell'armatura:

$$V(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t); \quad (3.7)$$

Dove le nuove grandezze rappresentano:

- V : la tensione di input del motore;
- R : la resistenza equivalente degli avvolgimenti del motore;
- L : l'induttanza equivalente degli avvolgimenti del motore.

Si ottiene così un sistema di tre equazioni che possono essere rappresentate mediante lo schema a blocchi in figura 3.17.

$$\begin{cases} T_m = k_c \cdot i \\ e = k_v \cdot \omega \\ V(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t) \end{cases} \quad (3.8)$$

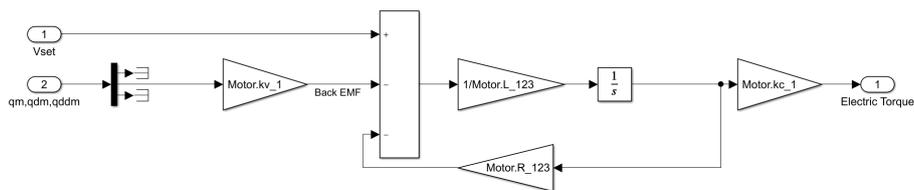


Figura 3.17: Schema a blocchi motore elettrico a corrente continua del primo giunto.

Lo schema a blocchi in figura 3.17³ può essere sostituito da un'equivalente funzione di trasferimento:

$$G(s) = \frac{V(s) - e(s)}{T_m(s)} = k_c \frac{1/R}{L/R \cdot s + 1} \quad (3.9)$$

Tale sistema riceve in ingresso la tensione al motore e la forza contro-elettromotrice fornendo in uscita la coppia elettrica sull'albero motore stesso. A tale coppia deve essere sottratta la coppia d'inerzia del motore ottenendo così la coppia a monte del riduttore, tale passaggio è illustrato in figura 3.18.

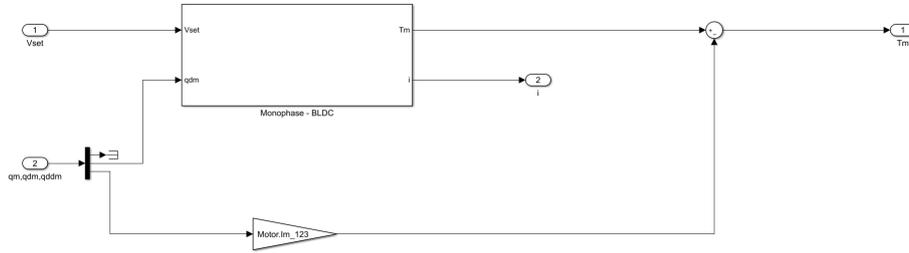


Figura 3.18: Schema a blocchi motore elettrico a corrente continua del primo giunto con coppia d'inerzia.

Attraverso il riduttore, in questo caso modellato semplicemente attraverso il rapporto di riduzione e l'efficienza, si ha una variazione della coppia secondo la legge:

$$T = T_m \cdot \tau \cdot \eta \quad (3.10)$$

Dove

- T : la coppia in uscita dal riduttore;
- τ : il rapporto di riduzione;
- η : l'efficienza del riduttore.

³Tutte le grandezze presenti all'interno del modello sono dotate di un pedice che indica a quale giunto sono riferite, ad esempio il pedice "1" indica le grandezze del primo giunto, mentre il pedice "123" indica le grandezze riferite ai primi tre giunti.

Ogni macchina, che prevede un moto relativo, è affetta da attrito, il quale causa la comparsa di una coppia che si oppone al moto. In questo caso si considerano tutti gli attriti del giunto concentrati nel riduttore ottenendo una coppia netta, in uscita dal motoriduttore che viene fornita in ingresso al modello della struttura meccanica pari a:

$$T_u = T - T_f \quad (3.11)$$

Ove T_f è la coppia dissipata per attrito. Questa è calcolata in funzione della velocità angolare del giunto come:

$$T_f = B \cdot q_d + C \quad (3.12)$$

In questa nuova equazione compaiono due parametri: B , il coefficiente di attrito viscoso, che, come si vede in figura 3.19, rappresenta la pendenza della coppia d'attrito in funzione della velocità, e C , l'offset rispetto allo zero, detto coefficiente di attrito coulombiano, esso dipende dal segno della velocità di rotazione ed è definito dalla seguente espressione:

$$C = \begin{cases} C_p & q_d > 0 \\ 0 & q_d = 0 \\ C_n & q_d < 0 \end{cases} \quad (3.13)$$

Figura 3.19: Andamento tipico della coppia d'attrito in funzione della velocità di rotazione [4].

In generale, il coefficiente d'attrito, dipende dal segno della velocità relativa ed è asimmetrico per valori di quest'ultima positivi e negativi. In figura 3.19 è rappresentato un tipico andamento dei coefficienti d'attrito in funzione della velocità di rotazione.

Nella zona delle velocità molto basse, evidenziata in grigio, si ha un effetto denominato *stiction*, contrazione di *static friction*. Per avere il moto la coppia applicata deve superare la coppia d'attrito, quando si verifica tale condizione si ha un fenomeno detto *stiction breaking*.

Una volta che è cominciato il moto, cioè la velocità non è più nulla, la forza d'attrito diminuisce rapidamente e l'attrito viscoso prevale su quello coulombiano. Una buona approssimazione di questo comportamento è quella che prevede di modellare l'attrito statico come l'*offset* della forza d'attrito e l'attrito viscoso come la pendenza di tale forza in funzione della velocità relativa [4].

Le cause di attrito all'interno del sistema fisico sono diverse, ad esempio, una causa è rappresentata dal motore stesso, per via della presenza dei cuscinetti calettati sull'albero.

La modellazione dell'attrito è avvenuta mediante un blocco "C&V Friction", visibile in figura 3.20, che riceve in ingresso la velocità di rotazione del giunto e fornisce in uscita la coppia d'attrito.

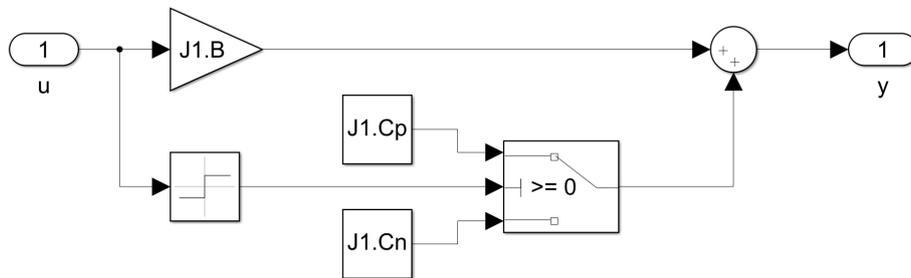


Figura 3.20: Blocco "C&V Friction" del primo giunto.

Tutto il modello del motoriduttore, comprensivo della coppia d'attrito, è visibile in figura 3.21.

Tale modello, come detto in precedenza, genera una coppia utile che viene fornita al corrispondente giunto. Tale coppia causa il moto del modello dinamico *Simscape* che, attraverso i sensori, fornisce il *feedback* al sistema di controllo.

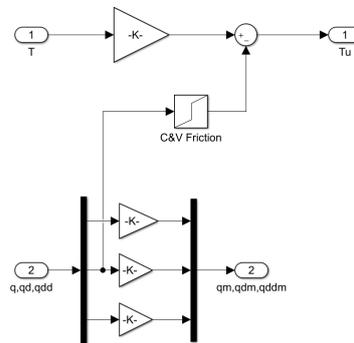


Figura 3.21: Schema a blocchi riduttore.

La logica di controllo è costituita da tre anelli di retroazione che governano la posizione angolare del giunto, la velocità del motore e la corrente che scorre all'interno del motore. Inoltre il valore di tensione di *set* viene fornito mediante modulazione di larghezza di impulso, anche detta tecnica *Pulse-width modulation* (PWM).

La tecnica del PWM sfrutta un segnale che oscilla tra un valore massimo ed un valore minimo in maniera continua. Quello che si va a variare è il tempo, rispetto al periodo di modulazione, per cui si mantiene il segnale in uscita al valore massimo. In tale maniera è possibile ottenere un valore medio nel tempo diverso da zero, che fa da segnale di tensione di riferimento del motore [28].

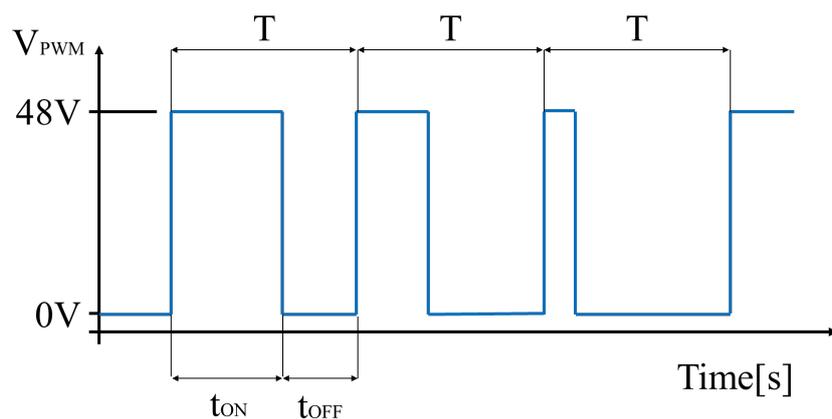


Figura 3.22: Esempio tensione generata tramite tecnica PWM.

Si definisce t_{ON} il tempo di durata di segnale alto, cioè tensione massima, e t_{OFF} , il tempo in cui il segnale di riferimento è nullo. La somma di questi due tempi rappresenta il periodo di modulazione, T , come si può vedere in figura 3.22.

Il rapporto tra t_{ON} e T è detto duty cycle. Nel caso oggetto di studio la tensione massima è rappresentata dalla tensione fornita dal *bus* di alimentazione di 48V.

$$dc = \frac{t_{ON}}{T} \quad (3.14)$$

La logica di controllo, così ottenuta, è raffigurata in figura 3.23.

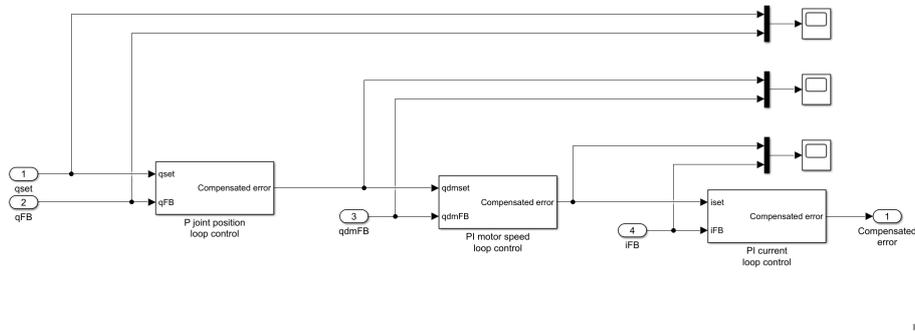


Figura 3.23: Logica di controllo in retroazione di posizione, velocità e corrente.

Non avendo alcuna nozione riguardo la tipologia di controllore utilizzata dall'azienda produttrice del robot si sono ipotizzati i controlli più frequenti per questo tipo di azionamenti, risulta però necessario tarare ciascun anello. In particolare, per tarare l'anello di corrente, si è estratto il motore dal modello, fornendo un input a gradino con un valore pari al 30% della corrente massima (2A) che può attraversare il motore. In figura 3.24 è possibile vedere il modello *Simulink*TM che si è utilizzato per tarare l'anello di corrente.

Tale operazione, cioè la taratura dell'anello di corrente del primo motore, ha fornito risultati validi per tutti e sei i motori, essendo tale taratura avvenuta estrapolando un sottomodello da modello intero, per cui si è scelto di adottare i medesimi parametri di controllo. La formula che esprime la tensione fornita dal controllo proporzionale integrativo è la seguente:

$$er_c = (i_{set} - i_{FB}) \cdot P + \int_0^t (i_{set} - i_{FB}) dt \cdot I \quad (3.15)$$

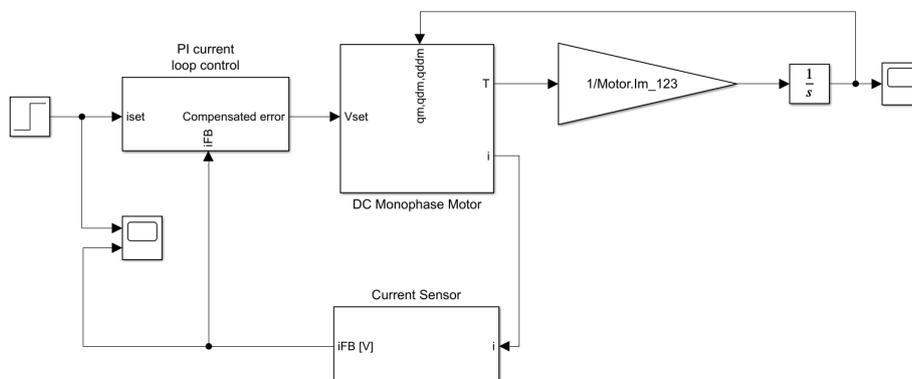


Figura 3.24: Modello *Simulink*TM per la taratura dell'anello di corrente.

In particolare l'anello di corrente presenta un guadagno proporzionale pari a $50V/A$ ed un guadagno integrativo pari a $10V \cdot s/A$.

Il passaggio successivo è rappresentato dalla taratura dell'anello di velocità, per far questo si è utilizzato l'intero modello del robot fornendo però, all'interno del controllore, come riferimento della velocità del motore, un gradino con valore pari al 30% del massimo della velocità raggiungibile dal motore ($3.3rad/s$). Per tarare l'anello di posizione si è utilizzato l'intero modello.

Una volta tarato il primo giunto risulta necessario verificare che i parametri siano validi anche per gli altri cinque, e verificare inoltre che il comportamento dei giunti successivi non influenzi il comportamento di quelli precedenti in maniera tale da non rendere più validi i parametri definiti in precedenza.

Per svolgere tale operazioni si sono tarati i parametri a partire dal motore che controlla il giunto di base e poi successivamente ci si è allontanati progressivamente dal sistema di riferimento fisso tarando, in successione, spalla, gomito, polso 1, polso 2 e polso 3, verificando ogni volta che il robot continuasse a comportarsi in maniera congrua.

In tabella 3.5 sono riportati tutti i parametri del controllo fin qui ricavati.

L'obiettivo è quello di ottenere un modello che descriva le grandezze fondamentali del sistema reale (posizione, velocità e corrente) in maniera piuttosto precisa in modo da poter capire, analizzando proprio queste grandezze, quando il comportamento del robot risulta anomalo e quindi la presenza o l'incipiente comparsa di un guasto.

	Proporzionale	Integrativo
Anello di corrente	$50V/A$	$10Vs/A$
Anello di velocità	$10As/m$	/
Anello di posizione	$5.5 * 10^3 As/m$	10

Tabella 3.5: Parametri utilizzati all'interno dell'algoritmo di controllo.

3.3.1 Introduzione del motore trifase

Il modello, così come è stato pensato, risulta essere modulare, tale scelta è volta alla possibilità di sostituire i modelli di prima approssimazione utilizzati con modelli sempre più accurati.

Per tale ragione è possibile sostituire l'attuale modello di motore monofase con un modello di motore trifase considerando quali sono gli input e gli output necessari.

La principale differenza tra il motore monofase ed il motore trifase è data dalla tipologia del segnale di corrente, infatti mentre per il primo il valore della corrente è dato istantaneamente da una grandezza scalare per il secondo motore si ha un vettore di tre componenti. Tale differenza non incide però su come è pensato il modello poiché il valore di corrente è prelevato dal motore come vettore e dato in ingresso alla logica di controllo sempre come vettore.

Il modello illustrato in figura 3.25 riceve in ingresso il segnale di corrente di *set* e di *feedback* che vengono confrontati all'interno ottenendo così un errore compensato tramite logica proporzionale-integrativa.

Il segnale in uscita dal controllore è un tensore in *direct-quadrature-zero* per cui deve essere convertito in modo che faccia da *input* al motore mediante la logica PWM.

In figura 3.26 è rappresentato il modello che riceve in ingresso il segnale della tensione in *direct-quadrature-zero* (DQ0), esso viene convertito in segnale di tensione trifase e per ciascuno dei tre segnali si ha una conversione in segnale PWM così come mostrato, per il segnale singolo, nella sezione 3.3.

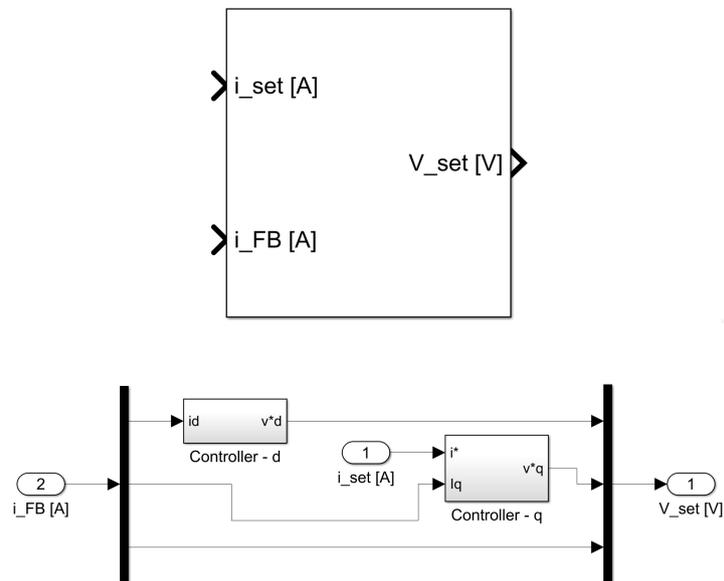


Figura 3.25: Modello *Simulink*TM del controllore dell'anello di corrente per il motore trifase.

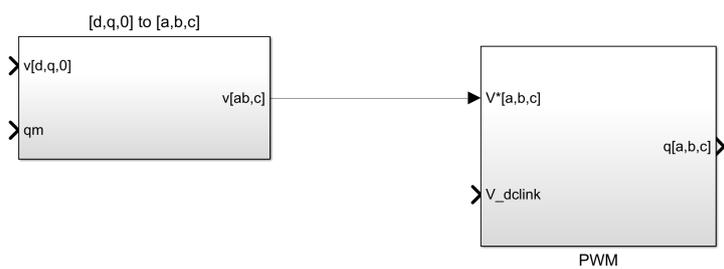


Figura 3.26: Modello *Simulink*TM per la conversione del segnale di tensione e della logica PWM per il motore trifase.

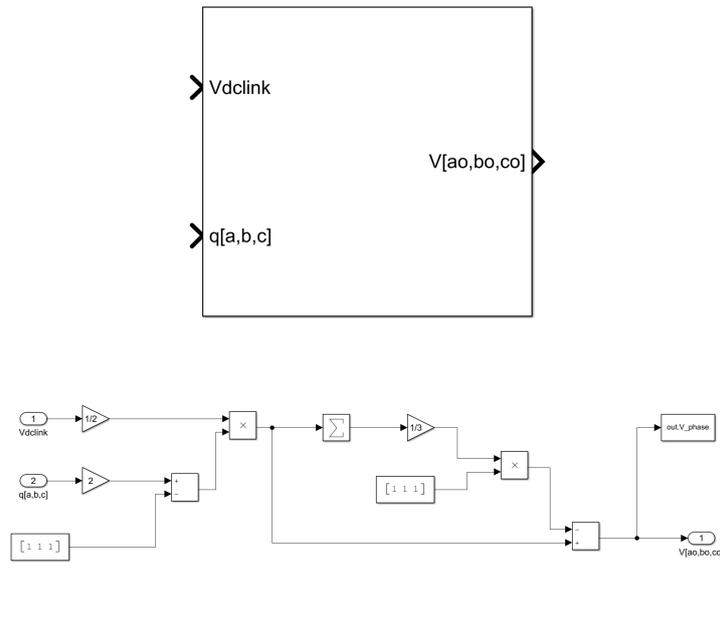


Figura 3.27: Modello *Simulink*TM dell'inverter per il motore trifase.

Il segnale PWM entra all'interno del blocco dell'inverter, il quale riceve in ingresso, oltre alla tensione di *set* anche la tensione massima di alimentazione. Come si può vedere in figura 3.27 esso da come output l'effettivo segnale in ingresso al motore.

Come si può vedere in figura 3.28 il modello del motore trifase riceve in ingresso, oltre al segnale di tensione, la posizione e la velocità, normalizzata del motore stesso, e la temperatura degli avvolgimenti, mentre in uscita fornisce la coppia a monte del riduttore, la corrente e la potenza dissipata dagli avvolgimenti.

Una volta inserito il nuovo modello sarà necessario effettuare una nuova taratura dei parametri del controllore i quali dipendono dalla tipologia di motore e dalle sue caratteristiche.

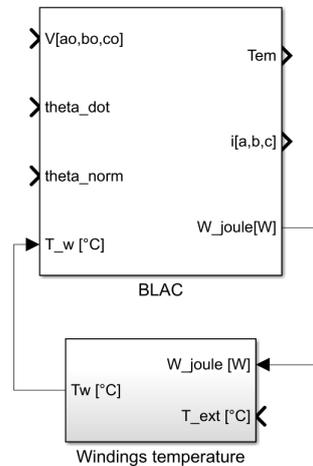


Figura 3.28: Modello *Simulink*TM del motore trifase col blocco di simulazione della temperatura.

3.3.2 I riduttori armonici

Come visto nella sezione 2.1.1 la maggior parte dei robot industriali, ed in particolare anche l'UR5, non montano semplici riduttori, bensì degli *harmonic drive*, tale componente è stato descritto ampiamente nella sezione 2.1.1. Per tale motivo, nell'ottica di avere un modello il più vicino possibile al sistema fisico reale si è sostituito il modello di riduttore visto in precedenza con un modello più complesso che permetta di andare a simulare la presenza di eventuali guasti agendo direttamente sui parametri del riduttore. Ad esempio la presenza di una cricca nella *flexspline* può essere modellata andando a modificare la rigidità torsionale della stessa.

Il modello di *harmonic drive*, qui presentato, è pensato per considerare la geometria, le interazioni interne ed eventuali errori di montaggio del riduttore [5].

L'introduzione di un modello dell'*harmonic drive* risulta significativa in quei sistemi in cui è particolarmente importante la precisione, cosa che si verifica in diverse applicazioni della robotica industriale e collaborativa.

Per descrivere il comportamento dell'*harmonic drive* molti modelli proposti sono basati su proprietà macroscopiche come la rigidità totale e la coppia d'attrito totale modellando gli errori in base ai dati sperimentali. Per costruire invece un modello più utile allo scopo prognostico è necessario descrivere le interazioni e gli scambi di forze interni al riduttore.

Quando il riduttore è in funzione, il *wave generator* ruota e costringe la

flexpline a deformarsi ellitticamente spingendo verso l'esterno i denti vicini all'asse maggiore dell'ellisse che vanno in contatto con i denti interni della *circularspline* causando così una rotazione.

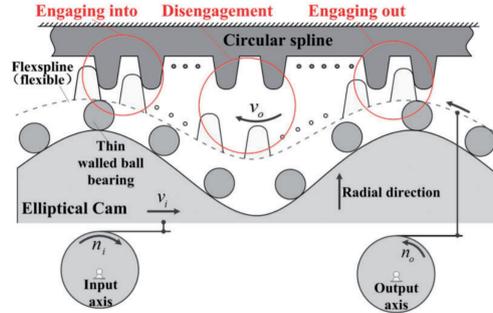


Figura 3.29: Schema del meccanismo di traslazione del riduttore armonico [5].

In figura 3.29 è rappresentato il modello che descrive la traslazione dell'*harmonic drive*. Il *wave generator*, forzando la *flexpline* a deformarsi verticalmente, mette in contatto un certo numero di coppie di denti che causano il moto della *flexpline* portando la coppia in presa a non essere più in contatto. Un possibile modello sostituisce l'impegno di una coppia di denti in presa con l'intero comportamento dell'*harmonic drive* [29].

In figura 3.30 è riportato il modello che tiene conto della geometria e delle interazioni interne del riduttore armonico. L'angolo di input del riduttore θ_i causa il moto dell'elemento che sostituisce il *wave generator*, che a sua volta produce il moto radiale x_1 dell'anello interno. Il dente della *flexpline*, che modella tutto il contatto, ha un moto radiale x_2 , il contatto causa l'avanzamento tangenziale del dente y_1 . La rotazione della *flexpline* causa il movimento tangenziale y_2 .

Le diverse interazioni tra gli elementi del riduttore sono riassunti dal seguente sistema:

$$\begin{cases} x_1 = \theta_i r_g \tan \alpha_n \\ K_b(x_1 - x_2 + e) + C_b(\dot{x}_1 - \dot{x}_2 + \dot{e}) \\ \quad = J_g \ddot{x}_2 / r_g^2 + K_m(x_2 \sin \alpha_t - y_1 \cos \alpha_t) \sin \alpha_t + F_{fm} \cos \alpha_t \\ K_m(x_2 \sin \alpha_t - y_1 \cos \alpha_t) \cos \alpha_t \\ \quad = J_g \ddot{y}_1 / r_g^2 + K_t(y_1 - y_2) / r_g^2 + F_{fb} + F_{fm} \sin \alpha_t \\ K_t(y_1 - y_2) / r_g^2 = J_o \ddot{y}_2 / r_g^2 \\ \theta_o = y_2 / r_g \end{cases} \quad (3.16)$$

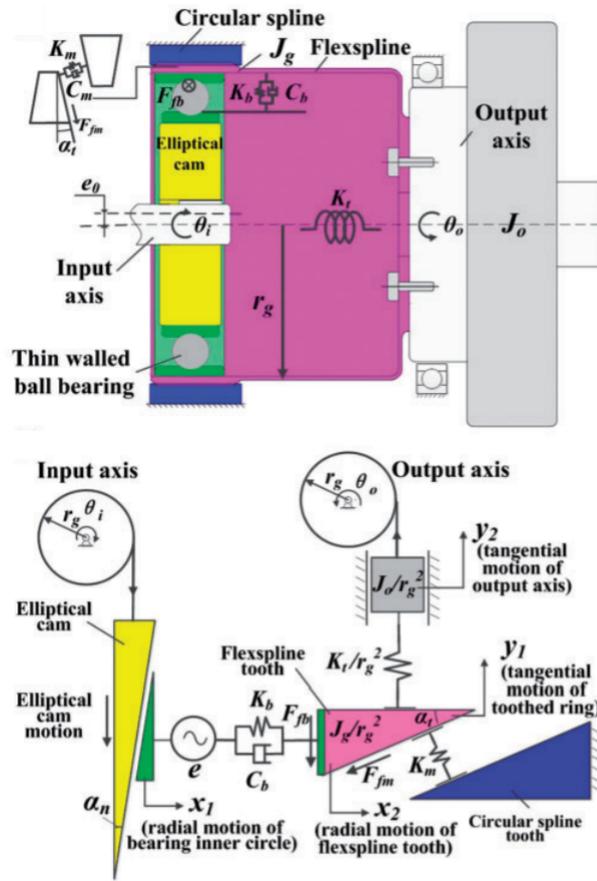


Figura 3.30: Modello di *harmonic drive* considerando la geometria e le interazioni interne[5].

- K_b e C_b rappresentano la rigidità e lo smorzamento della *flexspline*;
- e è il valor medio dell'errore di eccentricità che causa l'errore cinematico;
- F_{fb} e F_{fm} rappresentano gli attriti causati dal cuscinetto a sfere e dall'ingranamento dei denti;
- K_m è la rigidità equivalente d'ingranaggio;
- J_g è il momento d'inerzia della *flexspline*;
- r_g è il raggio dello strato neutro della *flexspline*;

- α_t è l'angolo del dente;
- J_o è il momento d'inerzia equivalente dell'asse d'uscita;
- K_t è la rigidezza torsionale della *flexspline*;
- θ_o è l'angolo di rotazione dell'asse d'uscita.

L'angolo α_n assicura che in assenza d'attrito, cedevolezza o errore cinematico, il modello di *harmonic drive* è dato semplicemente dal rapporto di riduzione e dall'angolo dei denti d'ingranaggio. Tale angolo è espresso dalla seguente equazione:

$$\alpha_n = \arctan\left(\frac{1}{N \tan \alpha_t}\right) \quad (3.17)$$

Dove N rappresenta il rapporto di riduzione dell'*harmonic drive*.

Risulta ora necessario definire i diversi parametri introdotti con la precedente modellazione, dipendenti dal riduttore armonico che si vuole simulare [5].

L'*harmonic drive* presenta contemporaneamente diverse coppie di denti in presa, la rigidità equivalente della singola coppia in presa è determinata dal numero di denti impegnati Z_R e dalla rigidezza di una coppia in presa:

$$K_m = Z_R K_{ms} \quad (3.18)$$

Il numero di denti in presa a sua volta è calcolato a partire dal numero di denti della *circularspline* Z_{CS} , legato a sua volta al numero di denti della *flexspline* Z_{FS} .

$$Z_R = 0.3 \cdot Z_{FS} / 2 = 0.3 \cdot (Z_{CS} - 2) / 2 \quad (3.19)$$

I risultati sperimentali mostrano che la coppia d'attrito segue un andamento dato dalla curva di Stribeck, quindi con una componente costante ed una dipendente dalla velocità. Come detto in precedenza le fonti d'attrito sono sostanzialmente due: il cuscinetto a sfere ed il contatto tra i denti.

Il moto relativo tra il dente della *flexspline* e quello della *circularspline* è simile al moto tra due superfici piane. Mentre l'attrito sul cuscinetto può essere rappresentato come una relazione attrito-velocità non lineare.

L'attrito con il cuscinetto è fornito dalla seguente relazione empirica di Palmgren:

$$F_{fb} = 10^{-7} \frac{f_0 (\nu_0 \dot{\theta}_i)^{\frac{2}{3}} d_m^3}{r_g} \operatorname{sgn}(\dot{\theta}_i) \quad (3.20)$$

Dove f_0 è il coefficiente determinato dal tipo di lubrificazione del cuscinetto, ν_0 è la viscosità del lubrificante e d_m è il diametro medio del cuscinetto.

L'attrito tra i denti è invece modellato come segue:

$$F_{fm} = (F_C + (F_S - F_C)\exp(-|\dot{\theta}_i/v_S|^\delta) + c\dot{\theta}_i)\text{sgn}(\dot{\theta}_i) \quad (3.21)$$

Dove F_C è l'attrito coulombiano che rappresenta la parte costante, F_S è l'attrito statico, v_S è la velocità di Stribeck, δ è il coefficiente di forma e c è la viscosità.

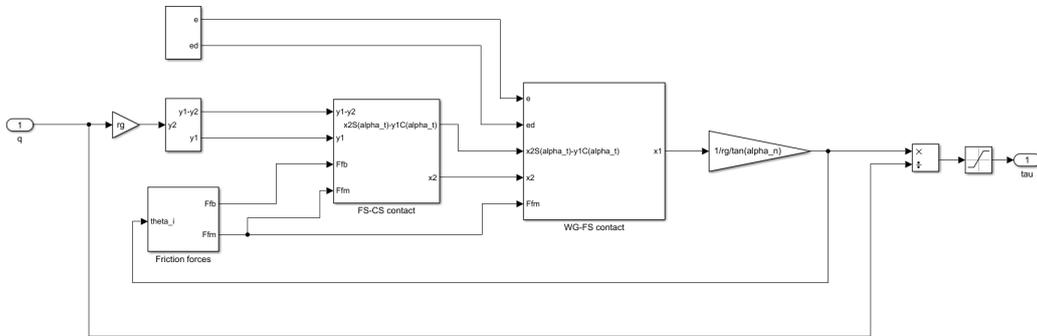


Figura 3.31: Modello dell'harmonic drive in Simulink™.

Tutte le relazioni espresse fin qui vengono introdotte all'interno di un modello *Simulink*™ apposito, in figura 3.31, che permette di calcolare il rapporto di trasmissione a partire dalla posizione angolare del giunto. Gli output del modello non risultano essere influenzati dalla presenza del modello dell'harmonic drive, questo perché in assenza di guasti o difetti all'interno del riduttore il rapporto di trasmissione risulta essere costante e pari a quello di progetto. L'utilità dell'aggiunta di questo modello sarà evidente nel momento in cui risulterà necessario iniettare guasti in questo specifico componente e studiare il conseguente comportamento del robot in condizioni di funzionamento non nominale.

3.4 I sensori

Come visto nel capitolo 2, i robot montano tre diversi tipi di sensori che servono per avere una misura diretta della corrente che attraversa il motore, della velocità del motore stesso e della posizione angolare dei giunti.

Ciascun modello di sensore riceve, in ingresso, il valore vero della variabile e fornisce, in uscita, il valore misurato della stessa variabile.

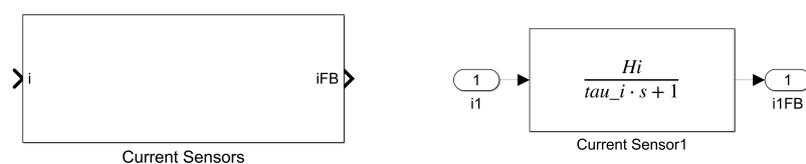


Figura 3.32: Blocco dei sensori di corrente e blocco con funzione di trasferimento del sensore di corrente del primo giunto.

Per quanto riguarda la modellazione di questi componenti si è sostituito ai sensori di corrente una funzione di trasferimento, rappresentata in figura 3.32, che induce una certa frequenza di taglio (2000Hz).

In particolare la grandezza H_i indica il guadagno statico del sensore di corrente, ciò significa che a regime il rapporto tra il valore reale ed il valore misurato è unitario. Invece τ_i indica la costante di tempo del sensore ed è scelta in modo da avere una frequenza di taglio di $2 \cdot \pi \cdot 2000 \text{ rad/s}$.

I sensori di posizione angolare dei giunti e di velocità dei motori sono stati considerati, inizialmente, ideali, per cui non inducono alcuna variazione sui segnali provenienti dal modello dinamico del robot.

Capitolo 4

Identificazione e validazione del modello tramite prove sperimentali

Nel capitolo 3 è stato definito il modello del robot UR5, in particolare i parametri del controllore sono stati tarati utilizzando una determinata prova sperimentale di *pick-and-place* denominata "test0", i risultati di tale identificazione sono stati riportati, precedentemente, in tabella 3.3. Per validare il modello si è scelto di utilizzare tre ulteriori prove, necessarie a verificare che il modello descriva in maniera corretta il comportamento del sistema fisico reale.

Per ciascuna delle tre prove saranno rappresentate diverse grandezze andandole a confrontare con le grandezze ottenute come output del sistema fisico reale.

In particolare tali grandezze sono posizione e velocità angolare di ciascun giunto, la corrente che scorre all'interno del motore elettrico e la coppia erogata per ciascun giunto.

4.1 Prova di identificazione

In questa prima sezione vengono illustrati i risultati ottenuti per il test di *pick-and-place* utilizzato per definire i parametri ottimali dei controllori.

In figura 4.1 è rappresentato il moto del TCP all'interno dello spazio di lavoro elaborato dal simulatore di Corke e dal modello introdotto nel capitolo 3.

Nelle successive figure 4.2, 4.3, 4.4, 4.5, 4.6, e 4.7 sono rappresentate tutte le grandezze che riguardano la suddetta prova sperimentale per quanto

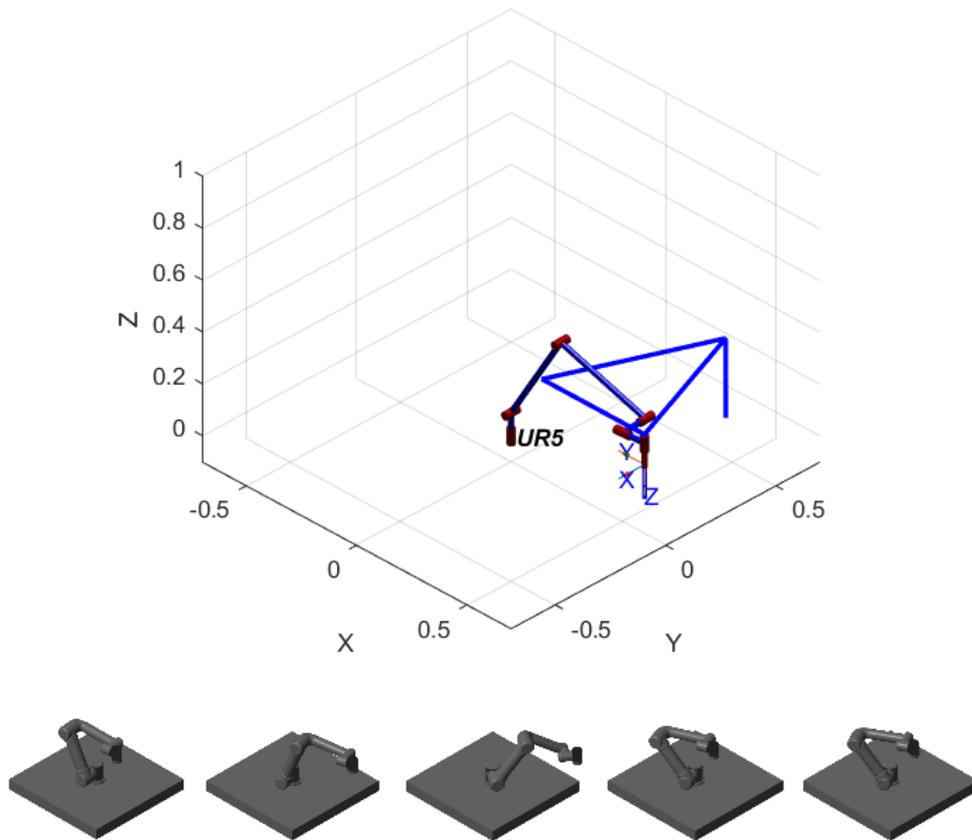


Figura 4.1: Traiettoria nello spazio di lavoro del TCP per il "test0".

riguarda i sei giunti del robot. In particolare le grandezze rappresentate nella colonna di sinistra rappresentano le grandezze cinematiche, cioè: la posizione di *set* del giunto (q_{set}), le posizioni di *feedback* del modello (q_{MODEL_FB}) e del sistema fisico reale (q_{UR5_FB}), la velocità di riferimento (qd_{target}), le velocità di *feedback* del modello (qd_{MODEL_FB}) e del sistema fisico reale (qd_{UR5_FB}) e l'errore di posizione e di velocità del modello. Nella colonna di destra sono rappresentate tutte le grandezze dinamiche riferite al giunto, quindi: il segnale di riferimento della corrente per il modello ($i_{MODEL_reference}$) e per il sistema fisico reale (i_{UR_target}), la corrente che scorre nel motore per il modello (i_{MODEL_FB}) e per il sistema fisico reale (i_{UR5_FB}) e le coppie erogate dal modello (T_{MODEL}) e dal sistema fisico reale (T_{UR5}).

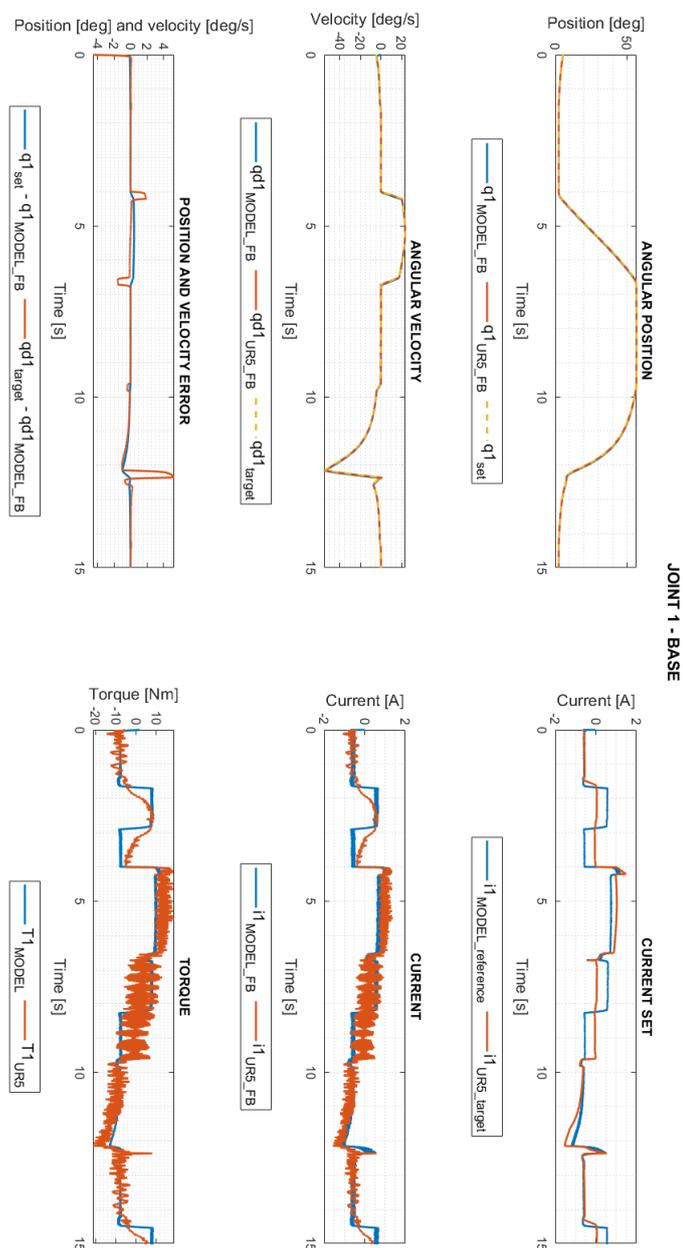


Figura 4.2: Grandezze simulate e misurate per il giunto di base per il "test0".

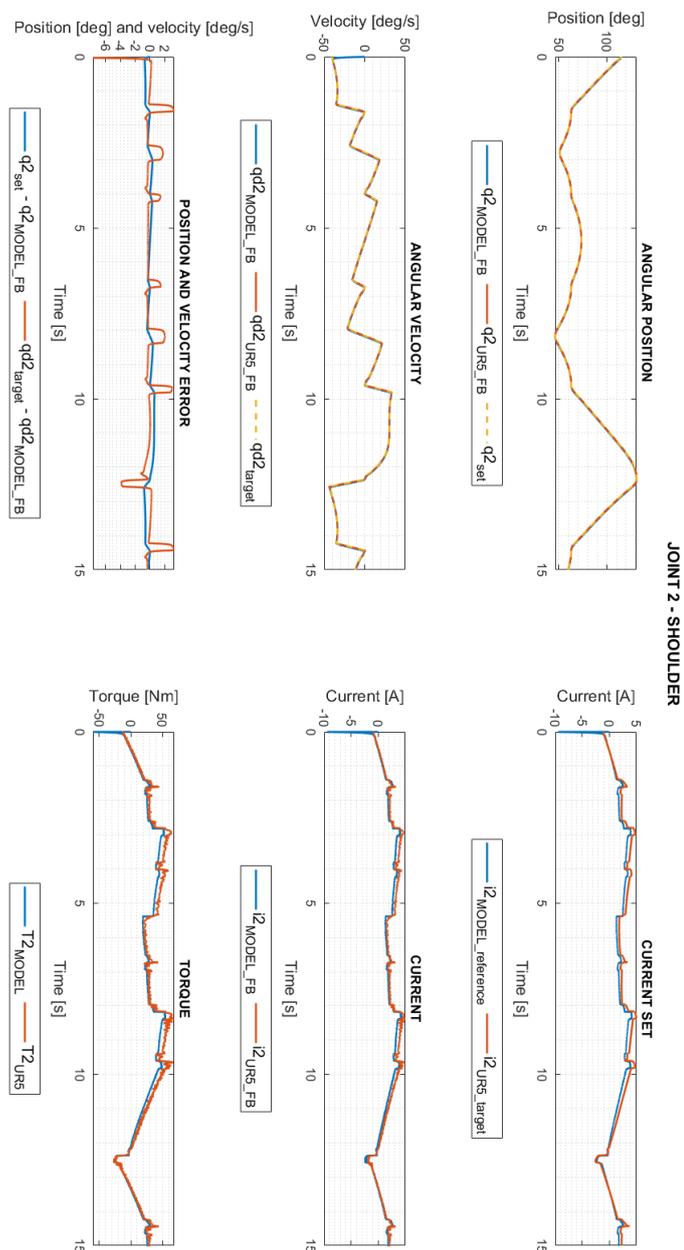


Figura 4.3: Grandezze simulate e misurate per il giunto di spalla per il "test0".

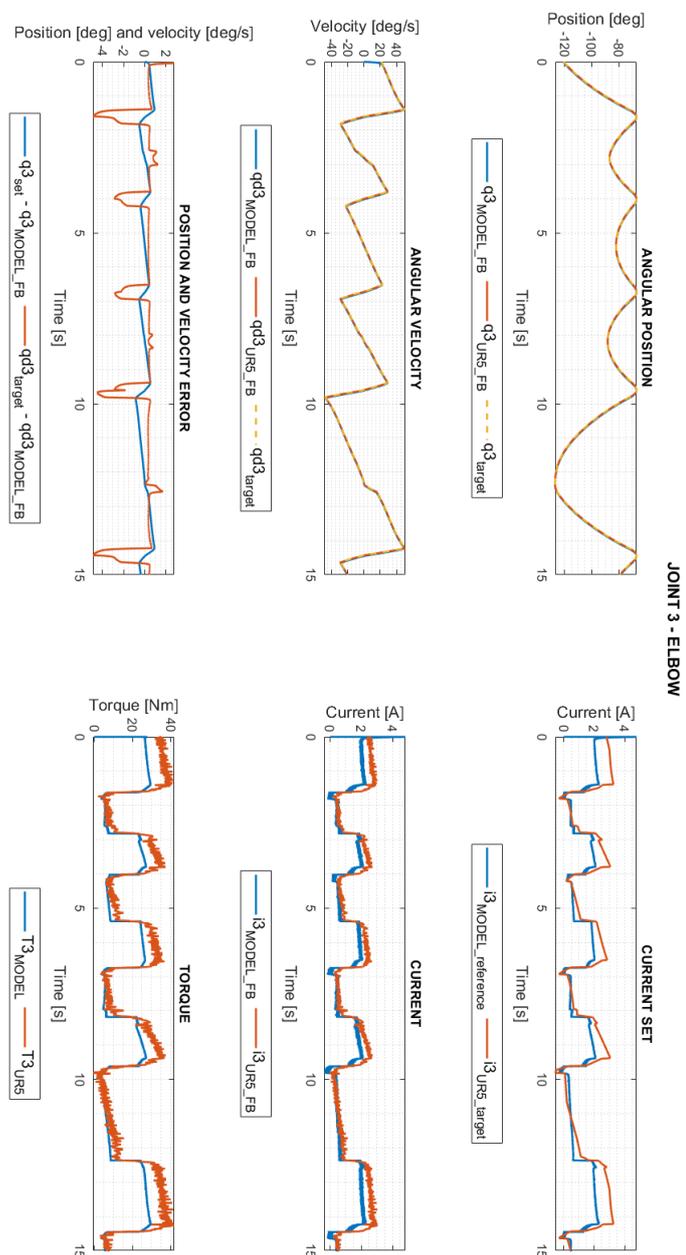


Figura 4.4: Grandezze simulate e misurate per il giunto di gomito per il "test0".

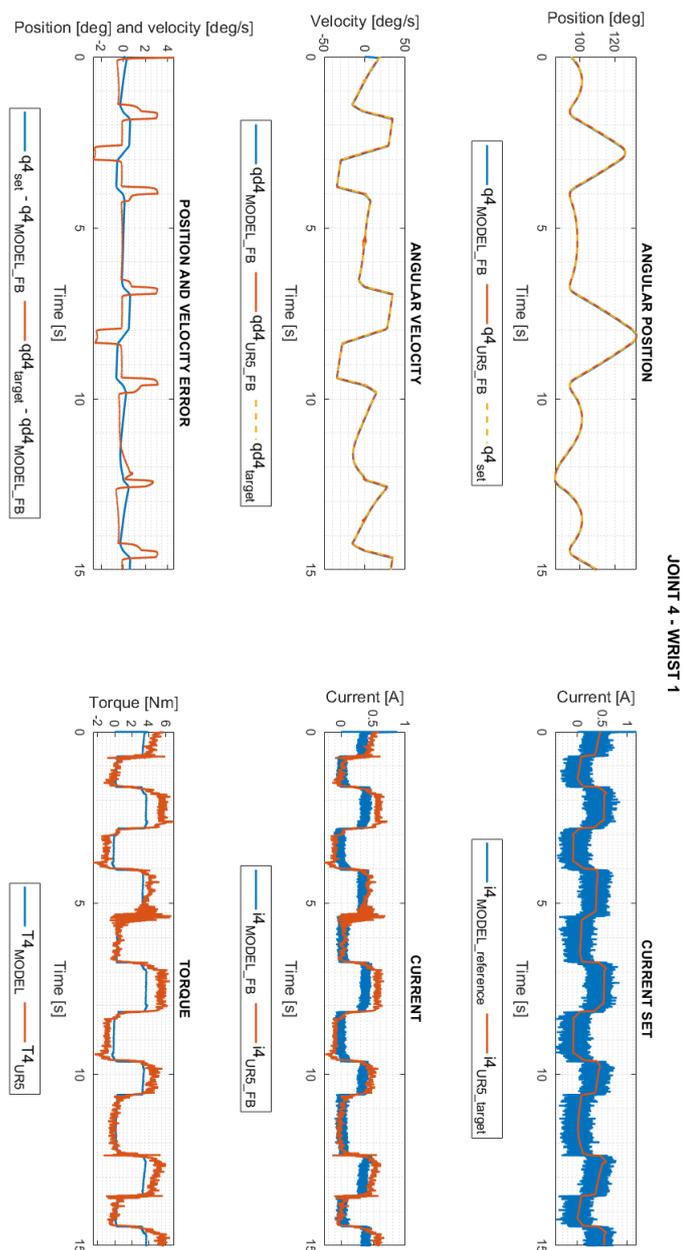


Figura 4.5: Grandezze simulate e misurate per il giunto di polso 1 per il "test0".

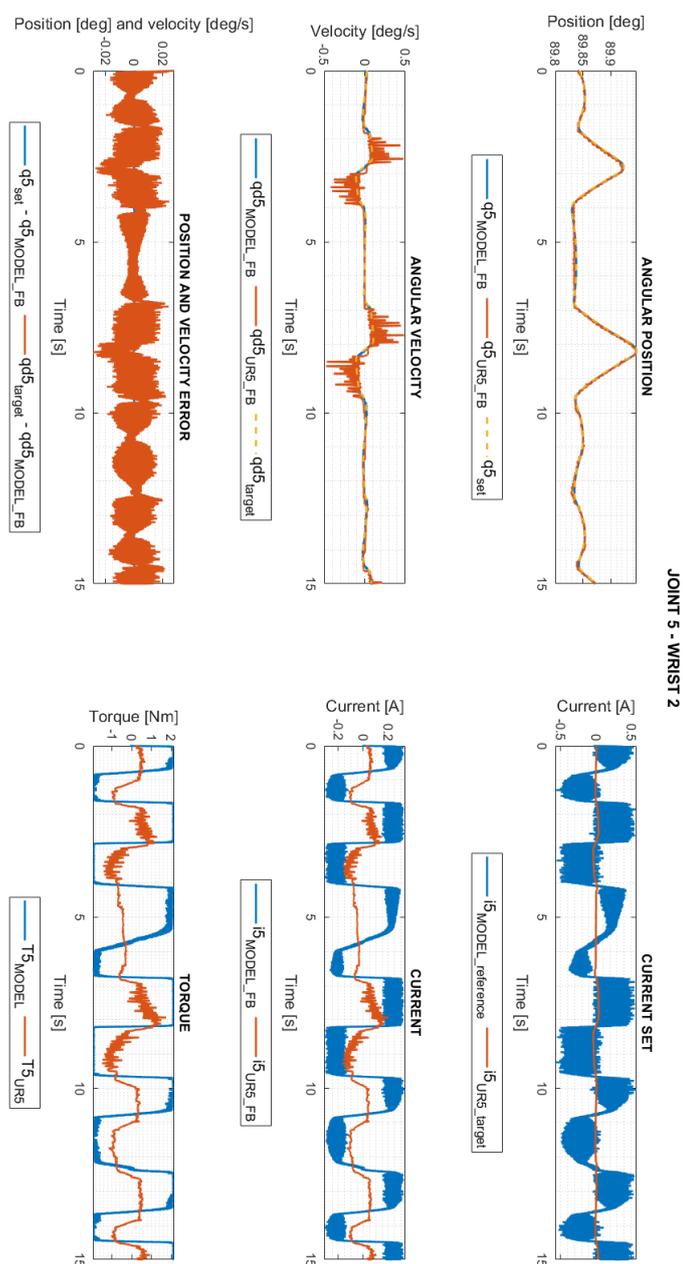


Figura 4.6: Grandezze simulate e misurate per il giunto di polso 2 per il "test0".

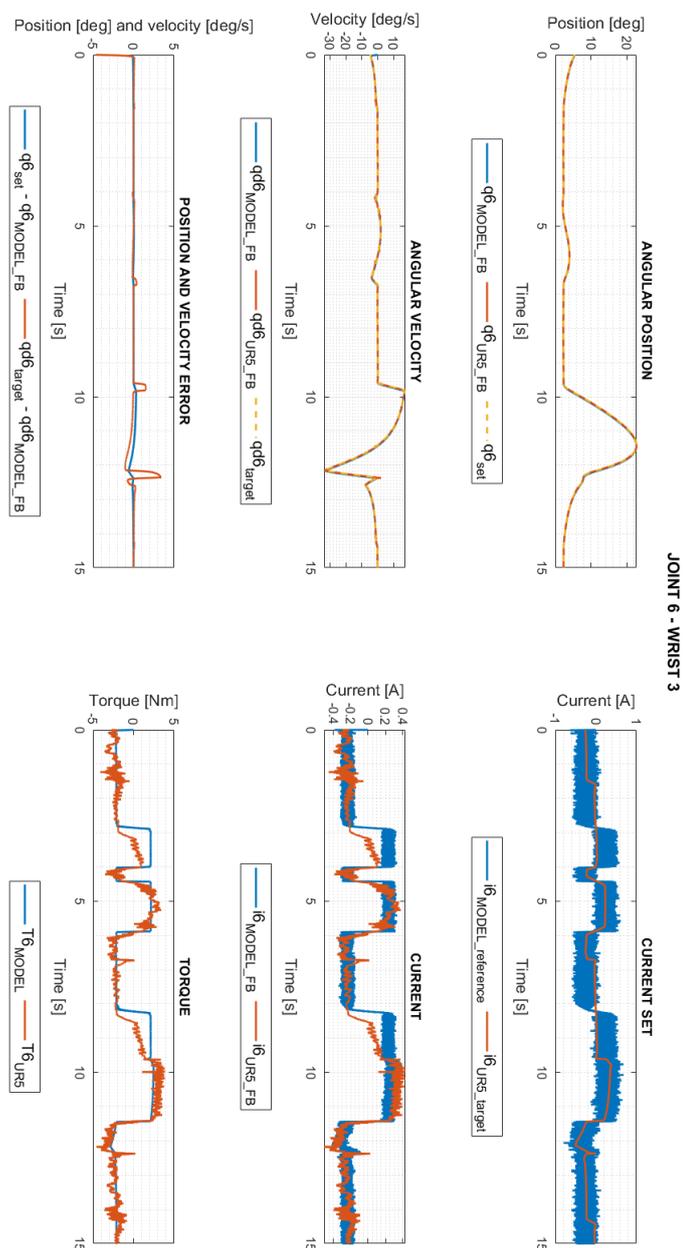


Figura 4.7: Grandezze simulate e misurate per il giunto di polso 3 per il "test0".

4.2 Prove di validazione

Nella seguente sezione vengono riportate, a titolo d'esempio, tre prove che permettono di confrontare le grandezze in uscita dal modello con quelle ricavate dal simulatore. I grafici rappresentano il moto del TCP e le grandezze estratte per ciascun giunto.

Attraverso queste prove è stata effettuata una prima validazione del modello. Analizzando tali grafici è possibile osservare come, in continuità con quanto visto per la prova di identificazione, il modello riesca ad emulare perfettamente il comportamento cinematico del sistema fisico reale, ma si abbiano delle discrepanze per quanto riguarda le variabili dinamiche. Tali discrepanze sono principalmente attribuibili alla mancanza di dati certi riguardo i parametri dinamici ed alla non conoscenza dell'effettiva logica di controllo del robot.

A seguito di tale campagna preliminare di prove sarà necessario effettuare una seconda che vada a provare maggiormente il robot, definendo delle leggi del moto che richiedano almeno un apporto di coppia pari al 50% del valore massimo erogabile dal motore studiando il comportamento dei giunti singolarmente, in modo da poter anche rapportare l'errore ad un range di variazione dei valori più ampio.

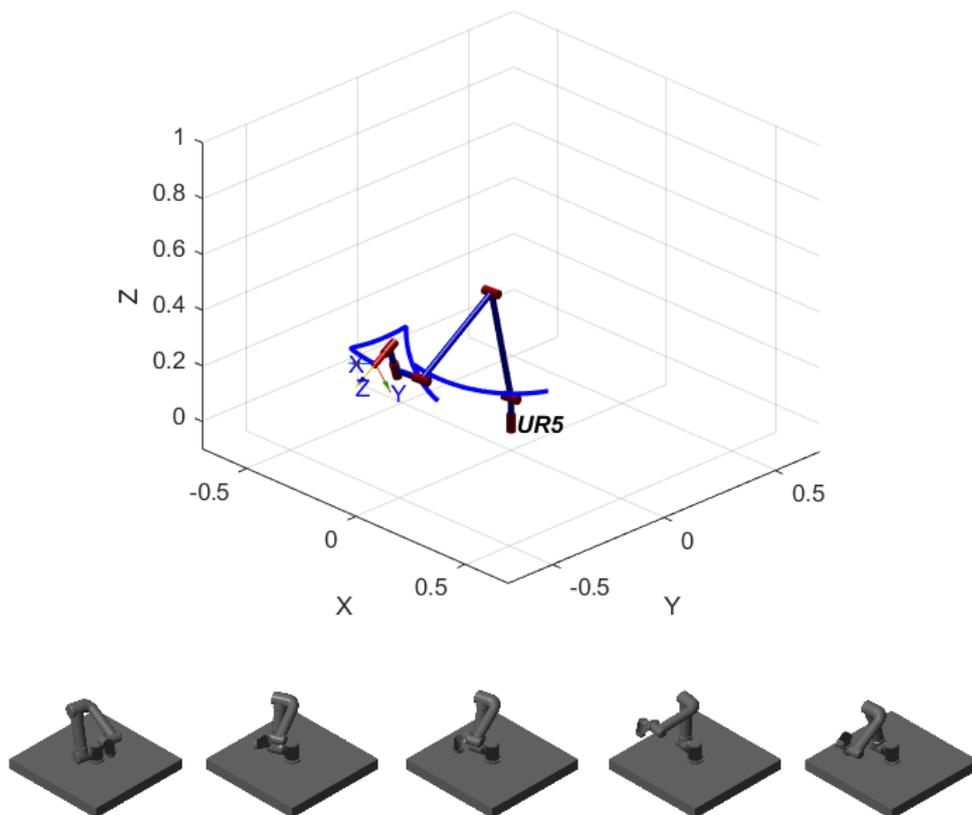


Figura 4.8: Moto del TCP nello spazio di lavoro per il "test1".

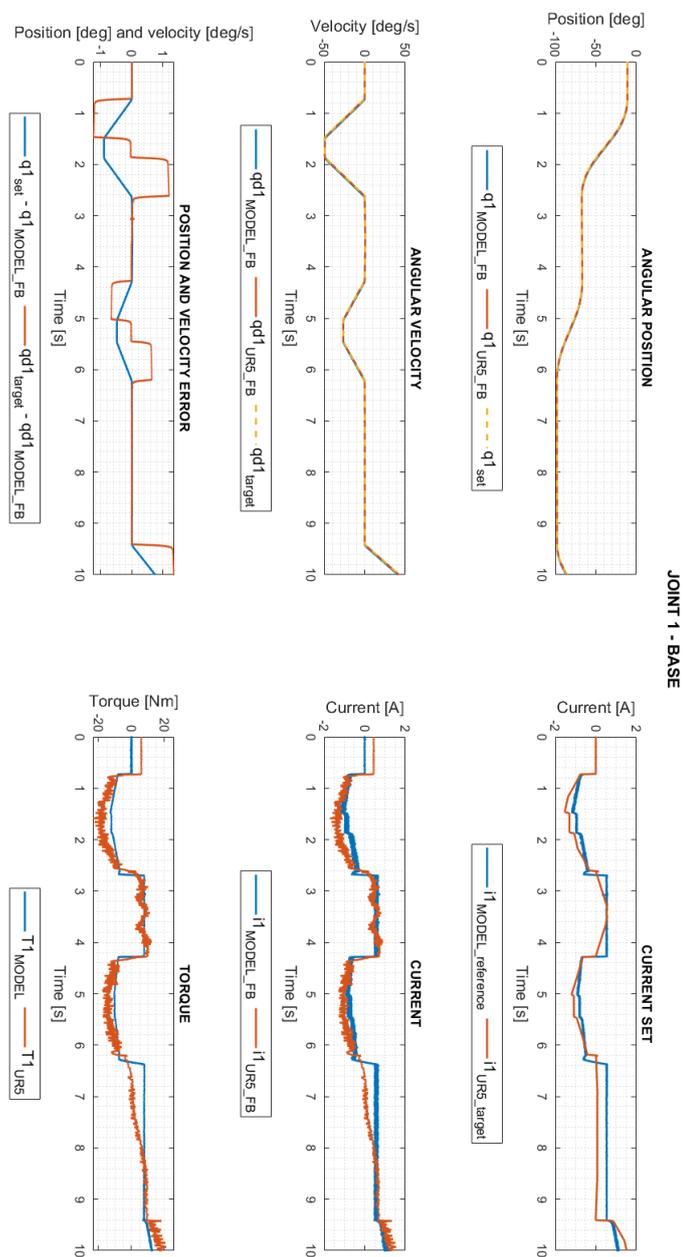


Figura 4.9: Grandezze simulate e misurate per il giunto di base per il "test1".

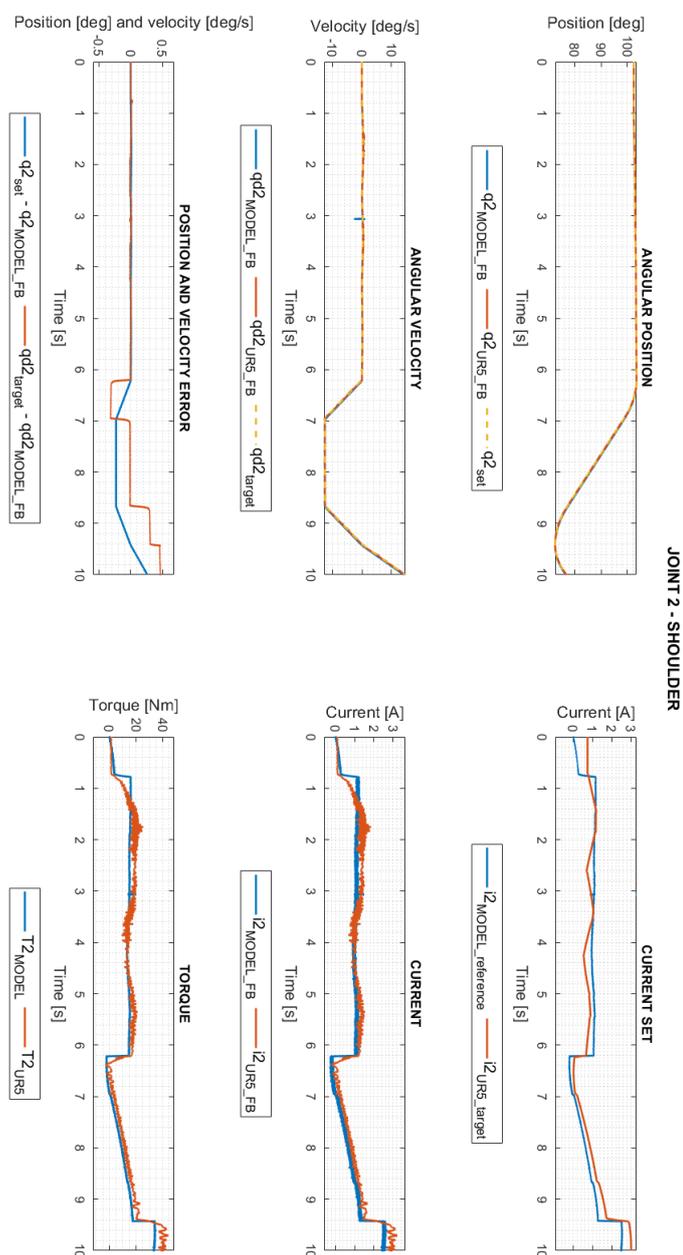


Figura 4.10: Grandezze simulate e misurate per il giunto di spalla per il "test1".

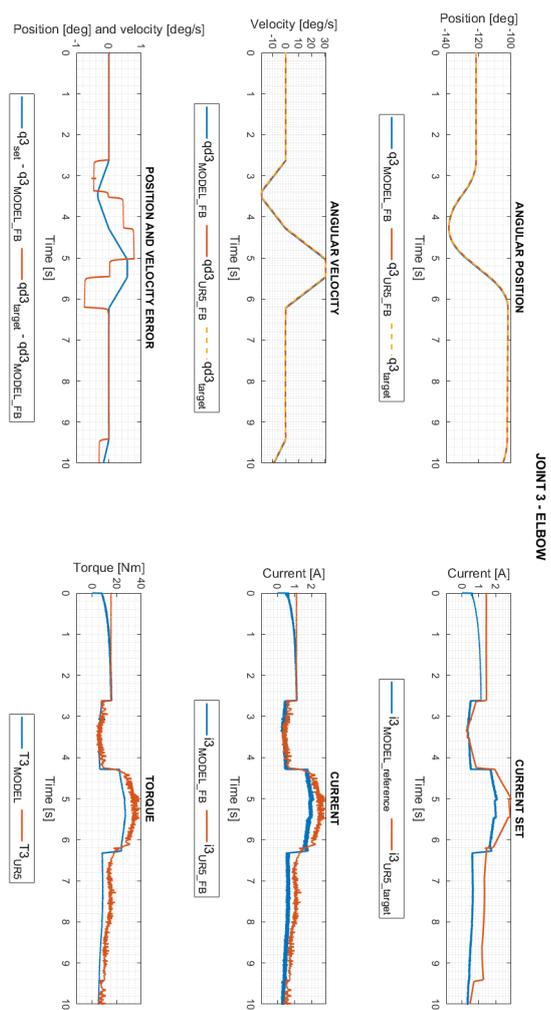


Figura 4.11: Grandezze simulate e misurate per il giunto di gomito per il "test1".

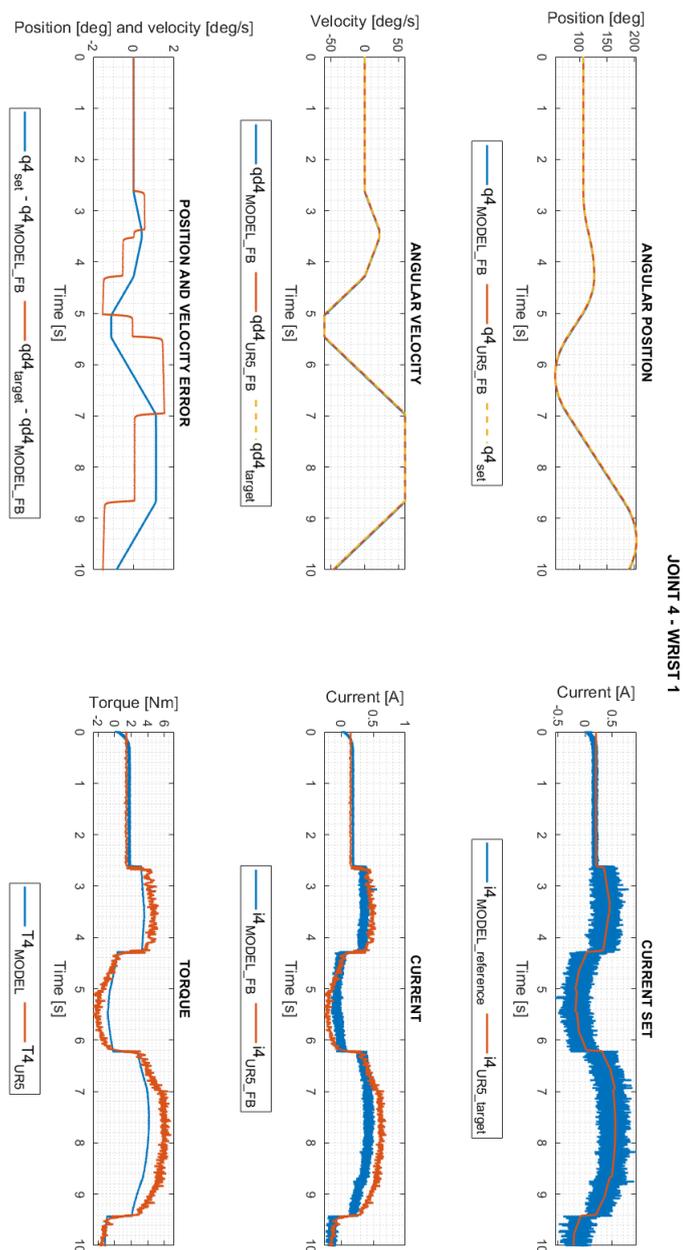


Figura 4.12: Grandezze simulate e misurate per il giunto di polso 1 per il "test1".

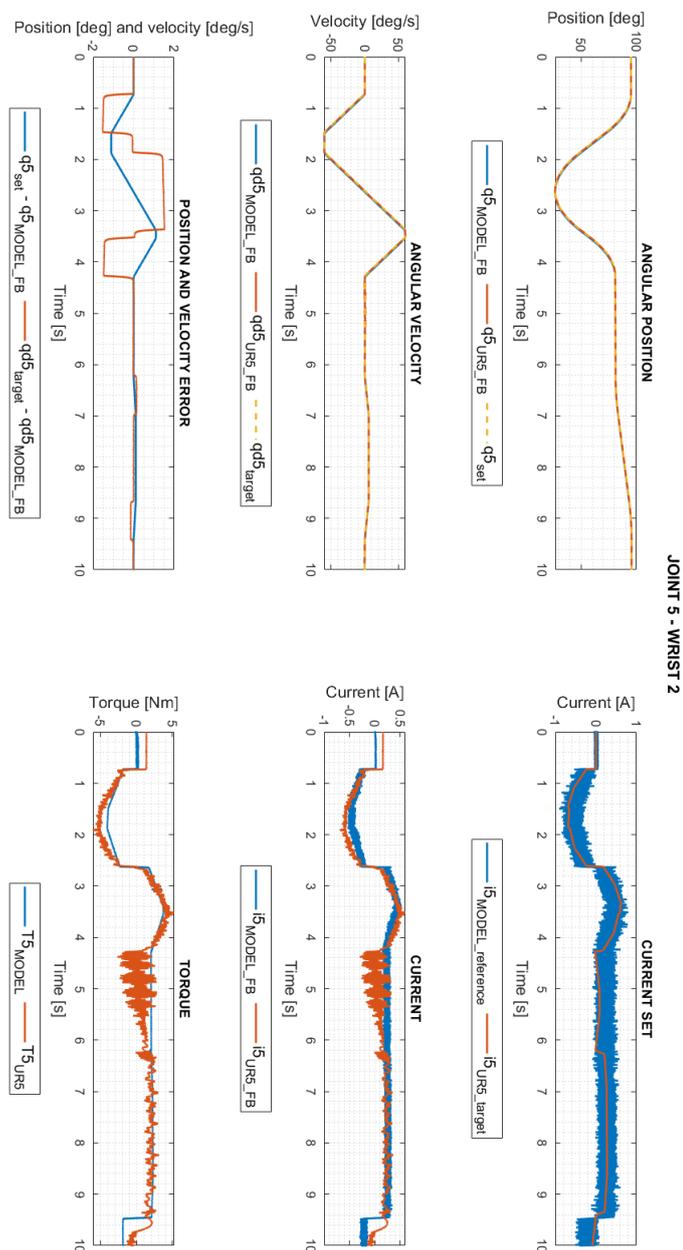


Figura 4.13: Grandezze simulate e misurate per il giunto di polso 2 per il "test1".

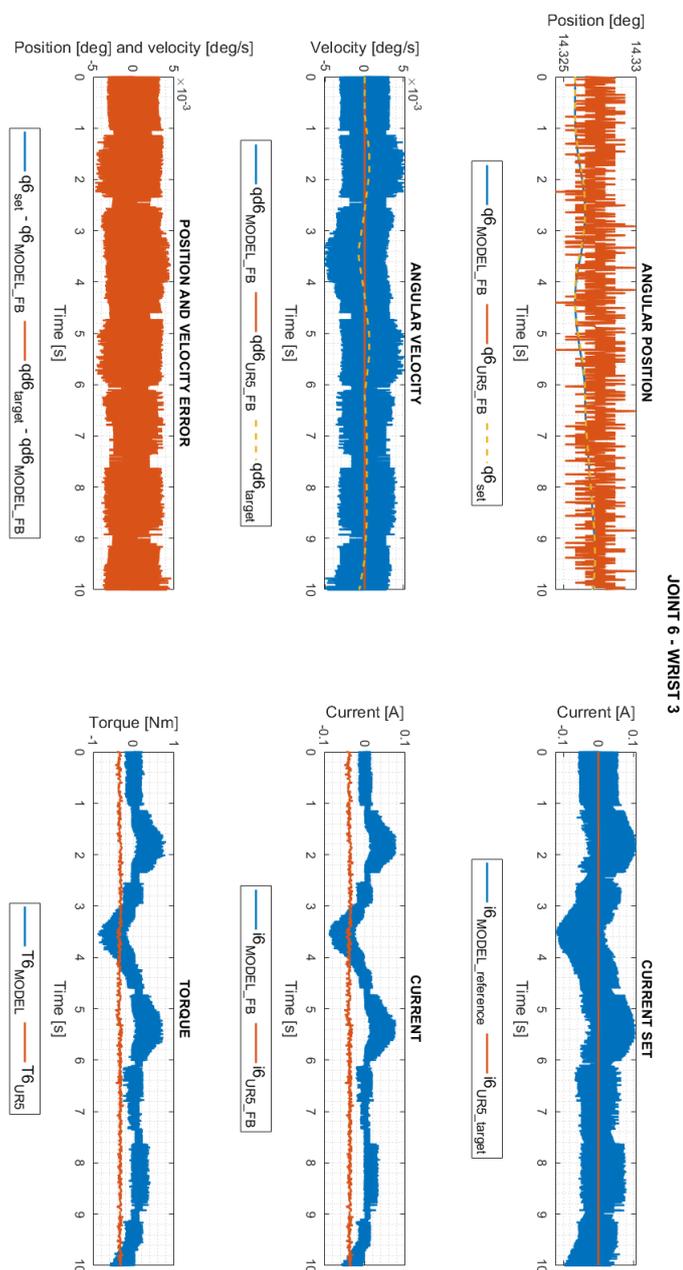


Figura 4.14: Grandezze simulate e misurate per il giunto di polso 3 per il "test1".

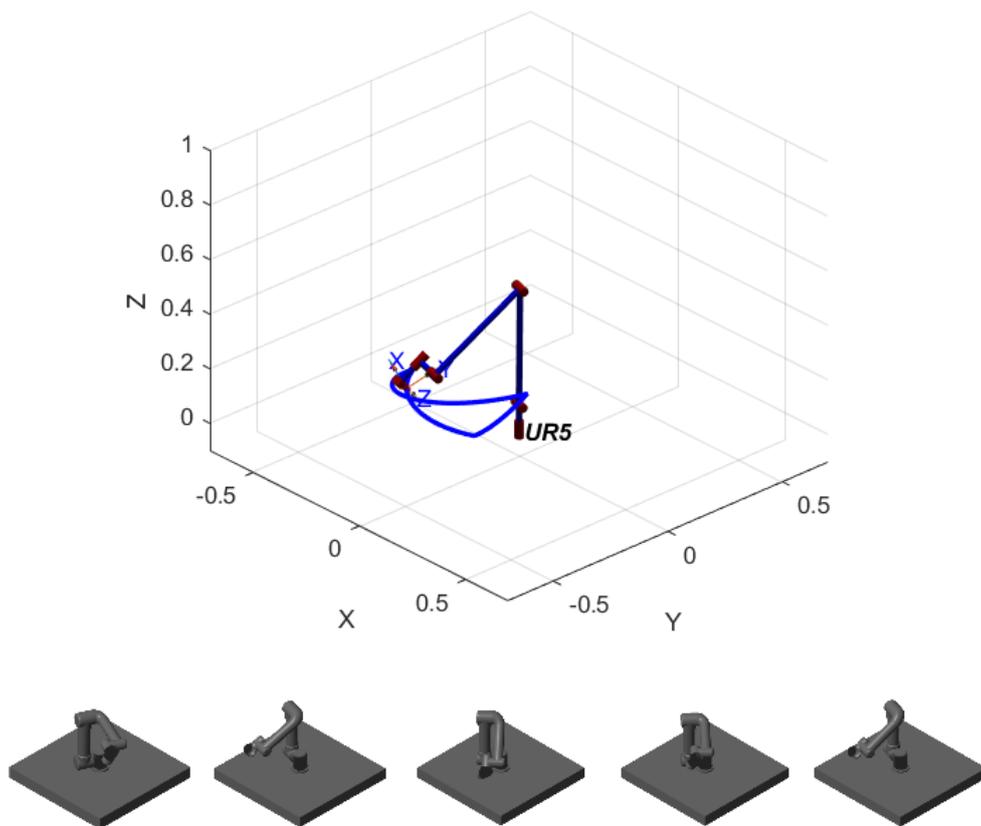


Figura 4.15: Moto del TCP nello spazio di lavoro per il "test2".

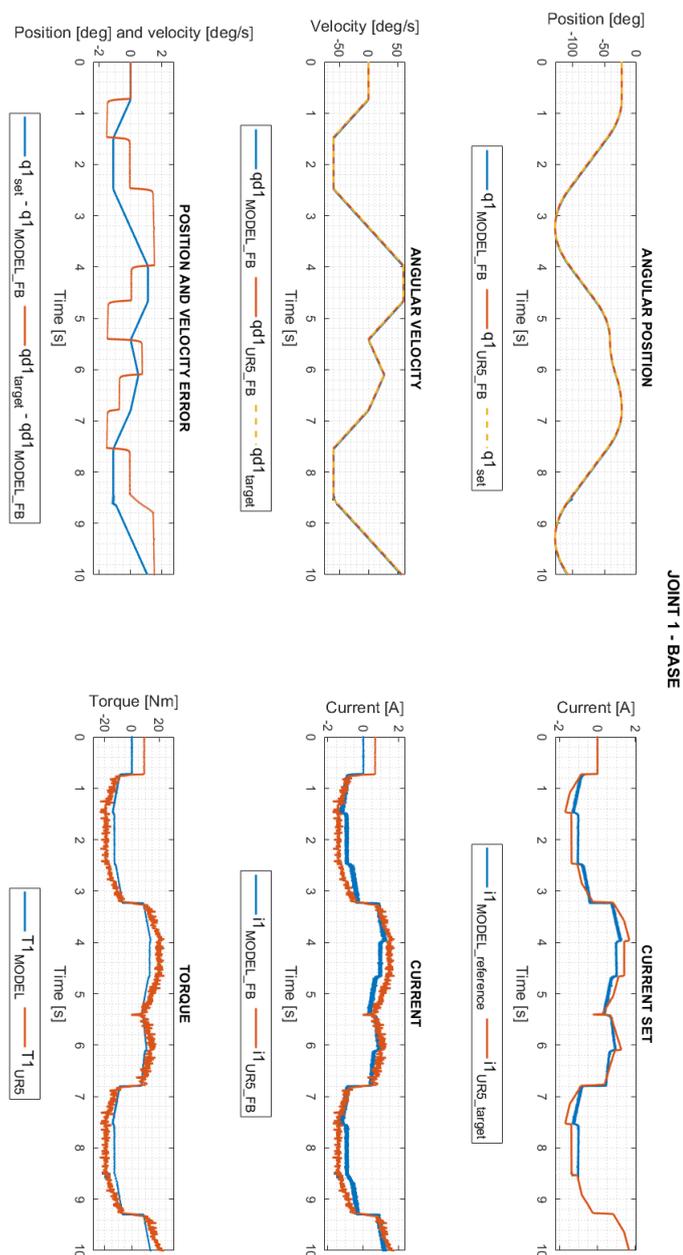


Figura 4.16: Grandezze simulate e misurate per il giunto di base per il "test2".

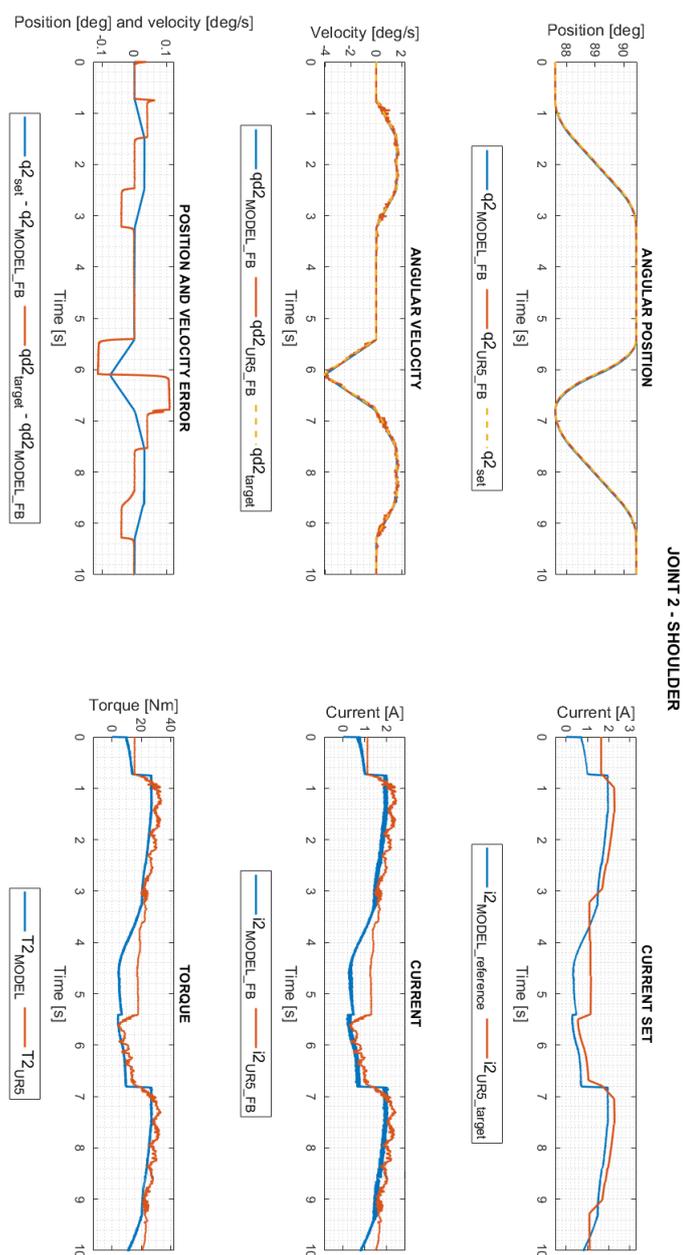


Figura 4.17: Grandezze simulate e misurate per il giunto di spalla per il "test2".

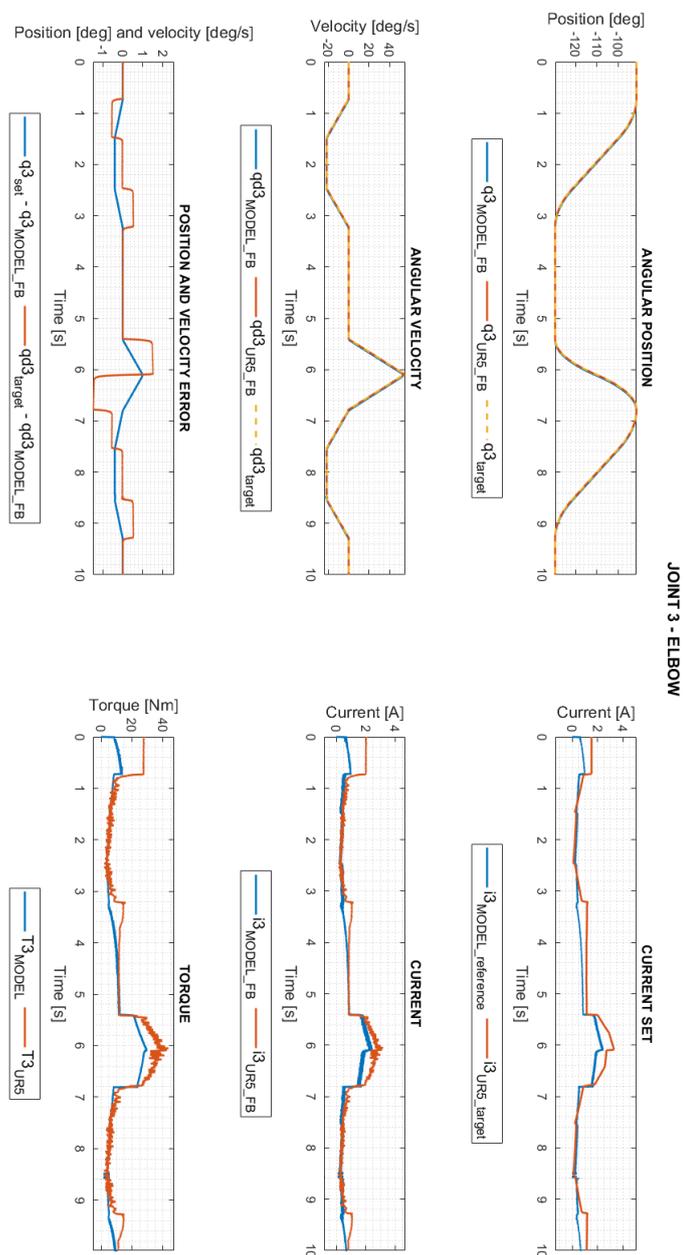


Figura 4.18: Grandezze simulate e misurate per il giunto di gomito per il "test2".

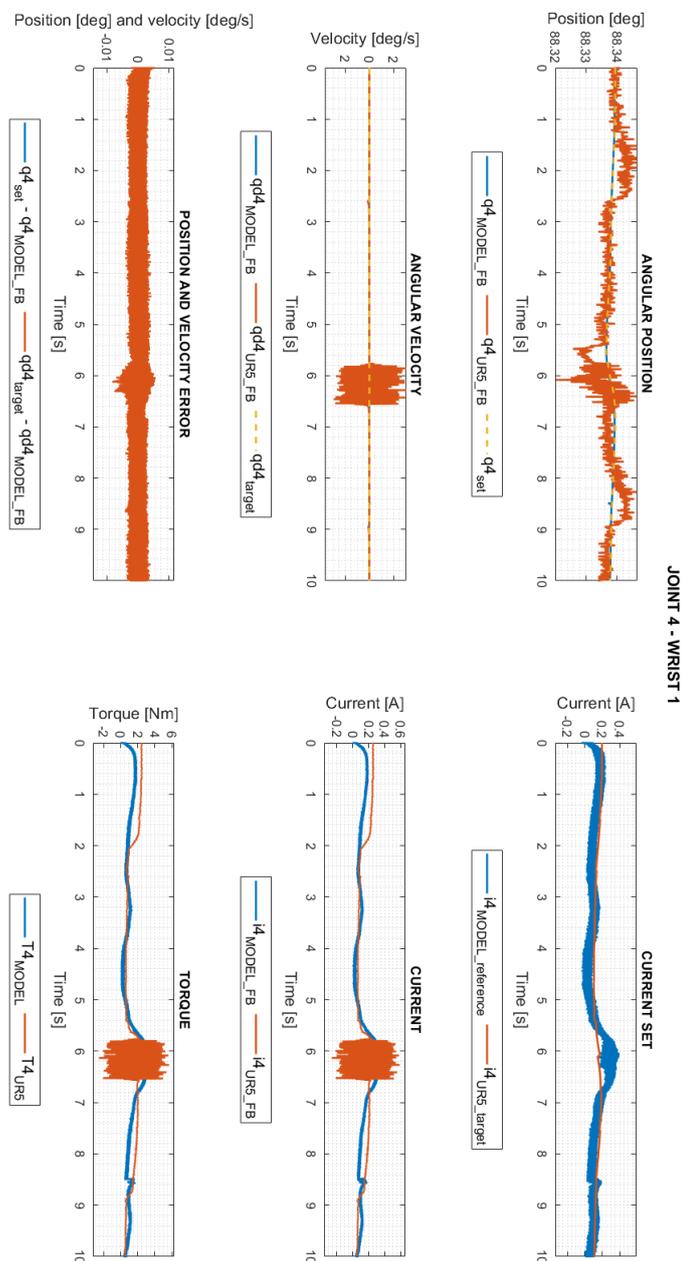


Figura 4.19: Grandezze simulate e misurate per il giunto di polso 1 per il "test2".

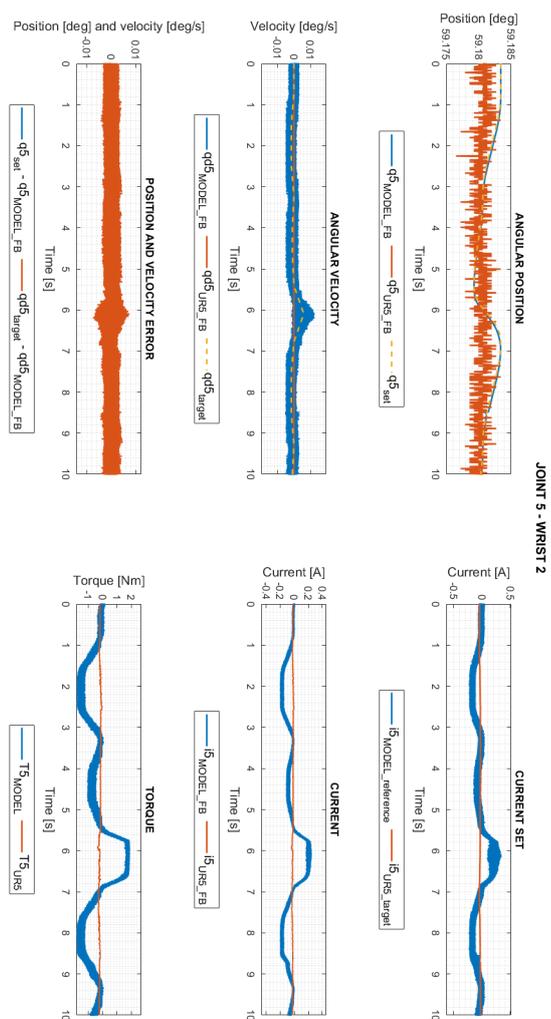


Figura 4.20: Grandezze simulate e misurate per il giunto di polso 2 per il "test2".

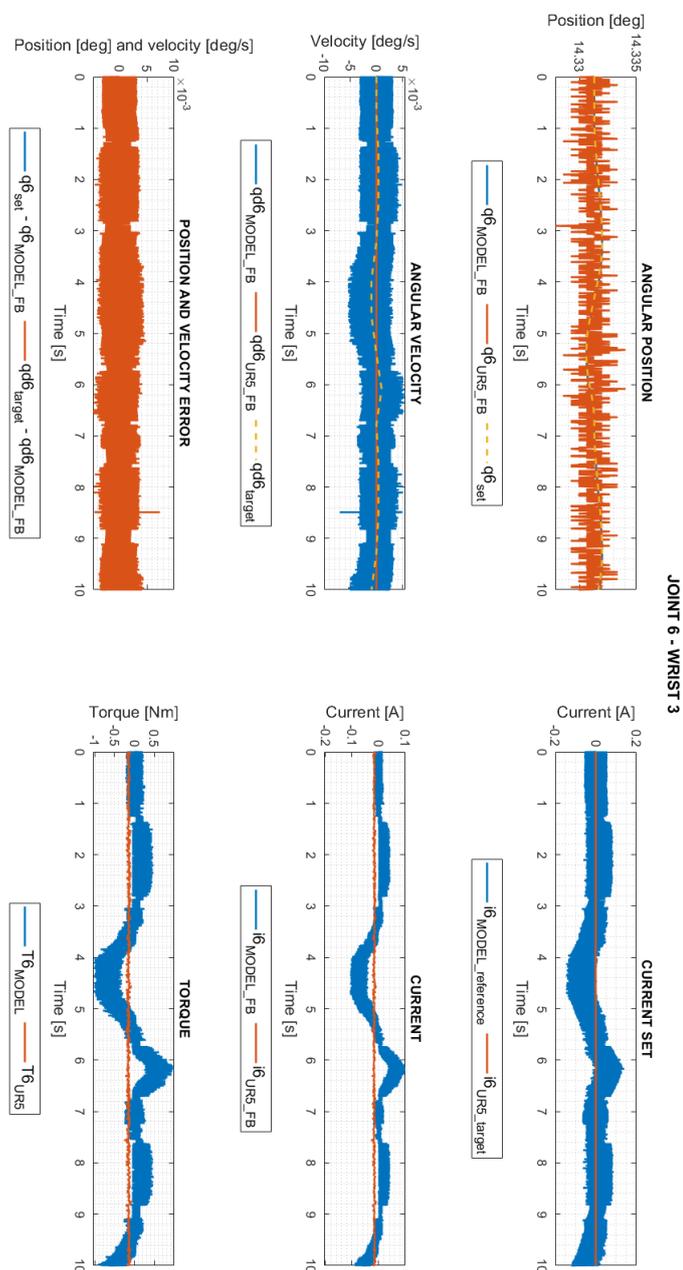


Figura 4.21: Grandezze simulate e misurate per il giunto di polso 3 per il "test2".

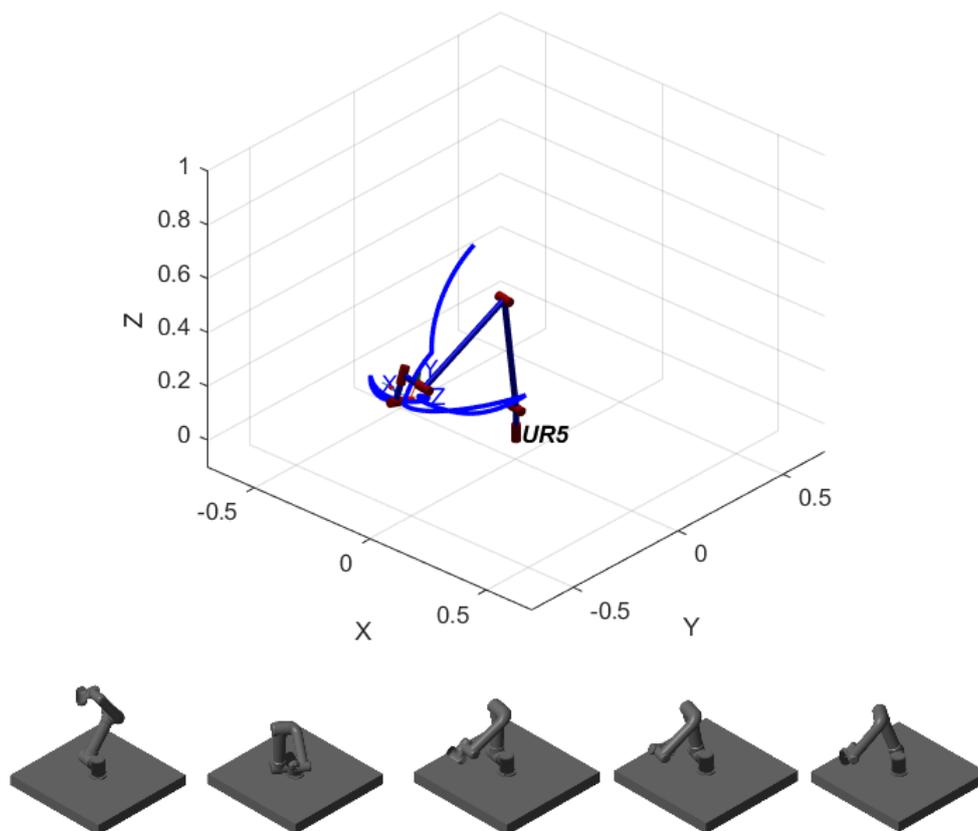


Figura 4.22: Moto del TCP nello spazio di lavoro per il "test3".

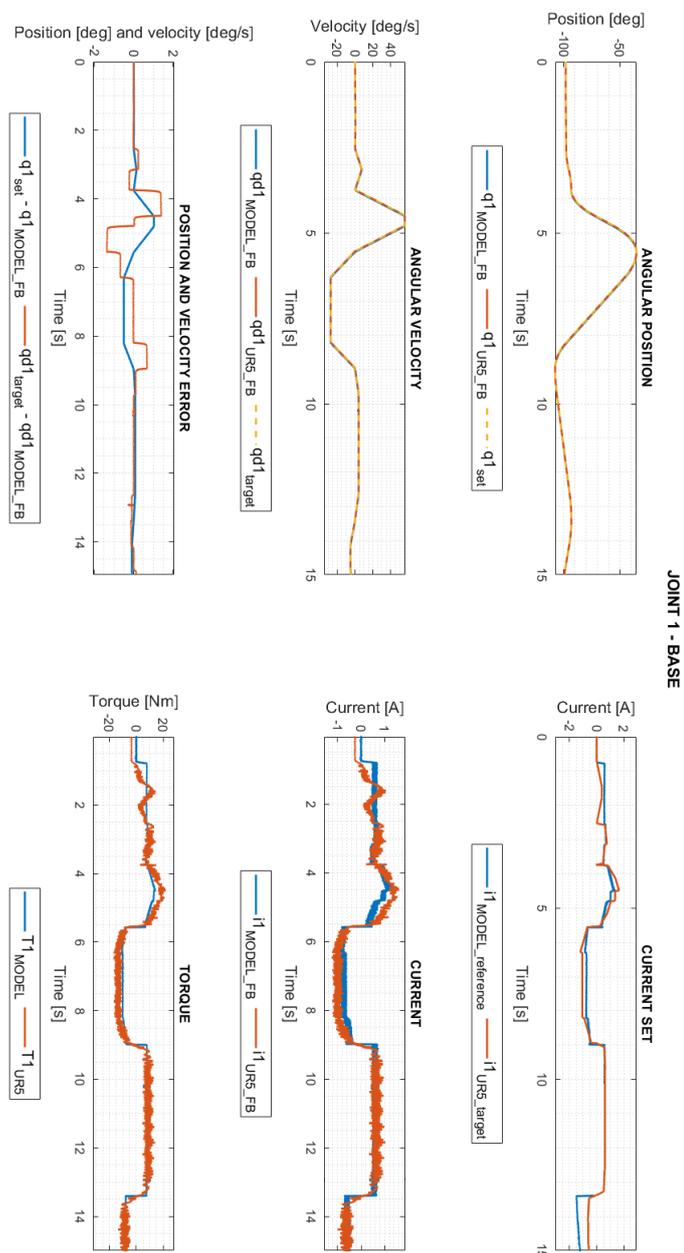


Figura 4.23: Grandezze simulate e misurate per il giunto di base per il "test3".

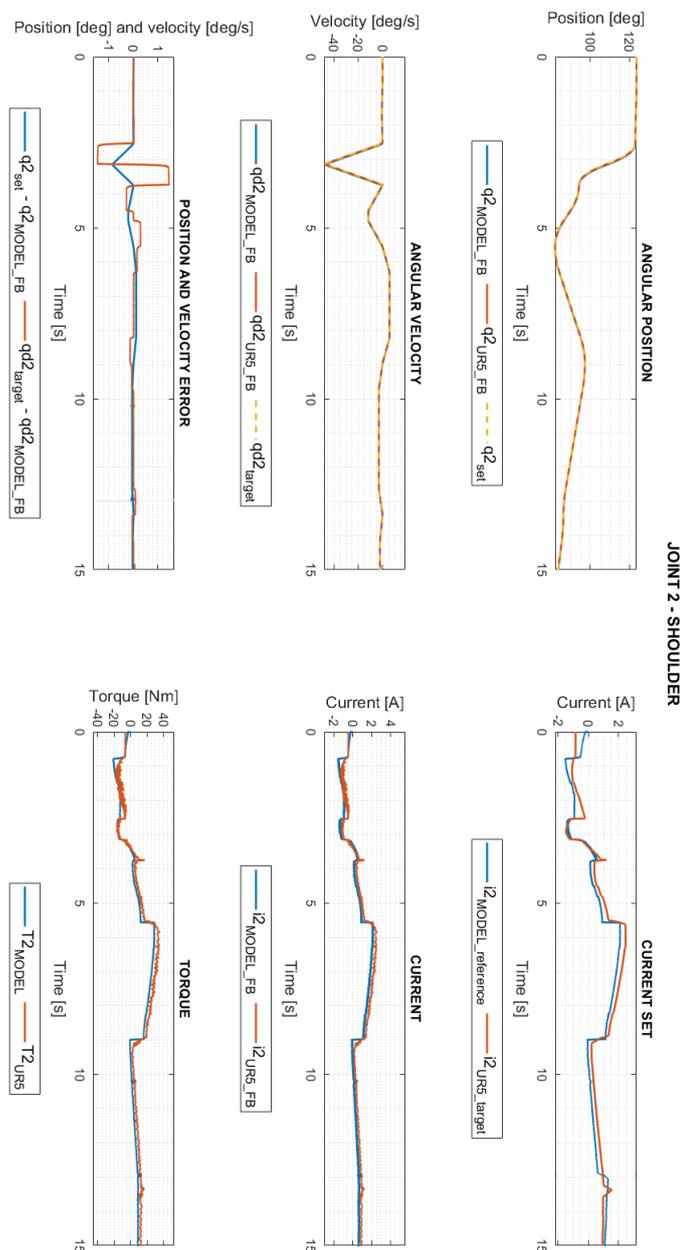


Figura 4.24: Grandezze simulate e misurate per il giunto di spalla per il "test3".

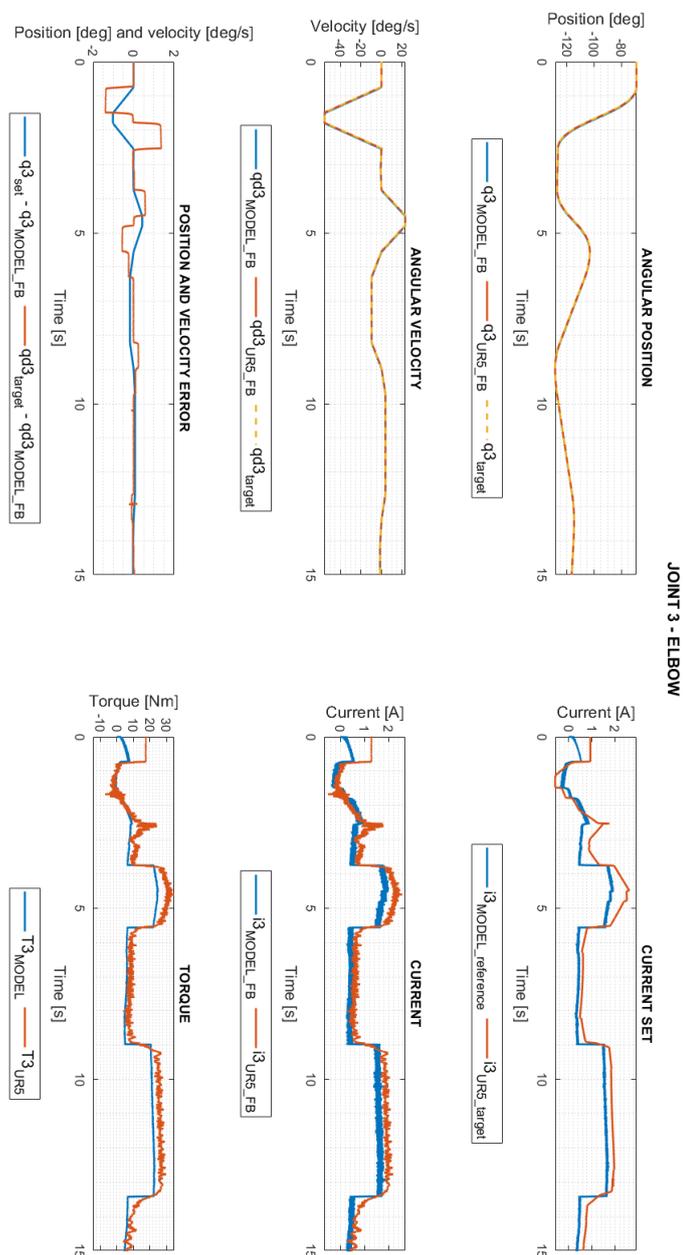


Figura 4.25: Grandezze simulate e misurate per il giunto di gomito per il "test3".

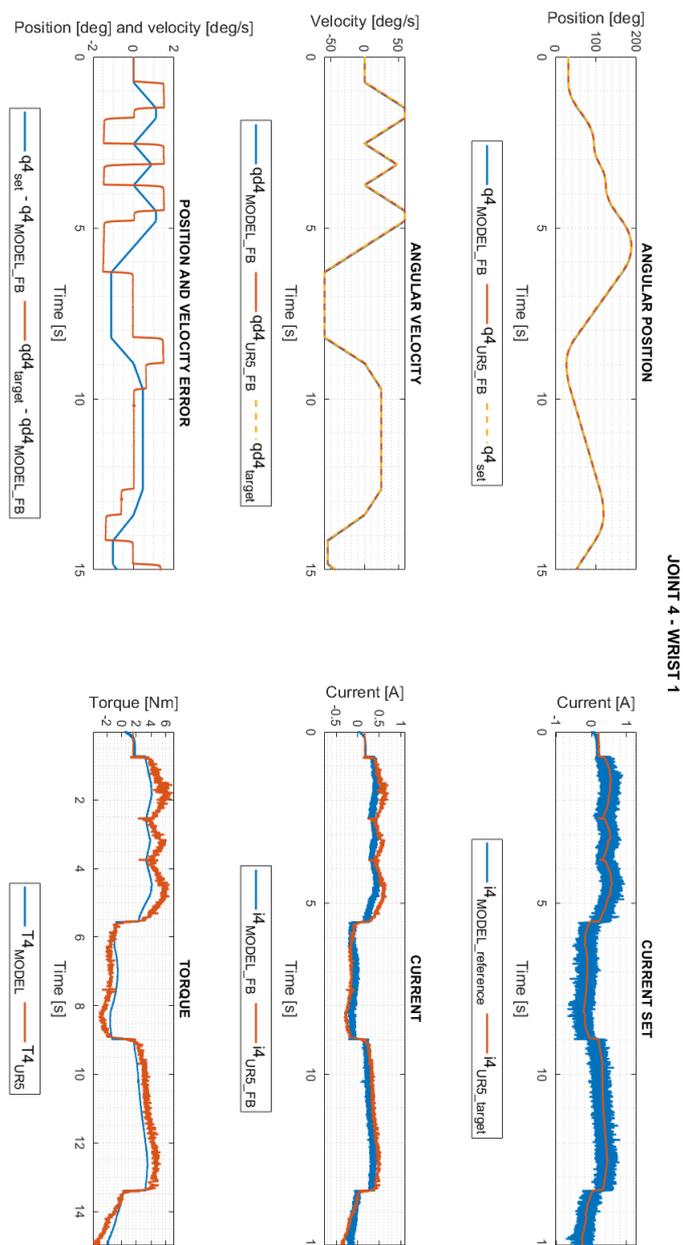


Figura 4.26: Grandezze simulate e misurate per il giunto di polso 1 per il "test3".

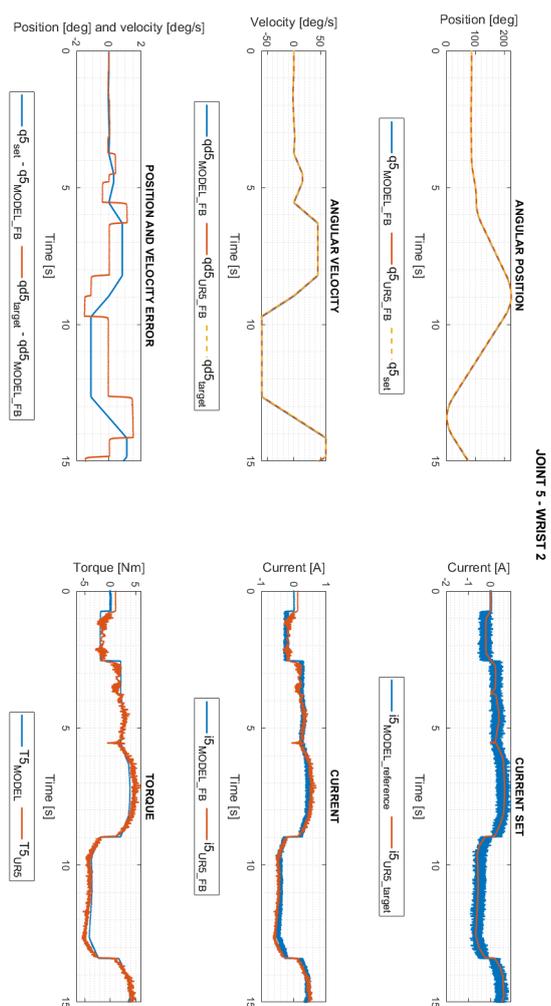


Figura 4.27: Grandezze simulate e misurate per il giunto di polso 2 per il "test3".

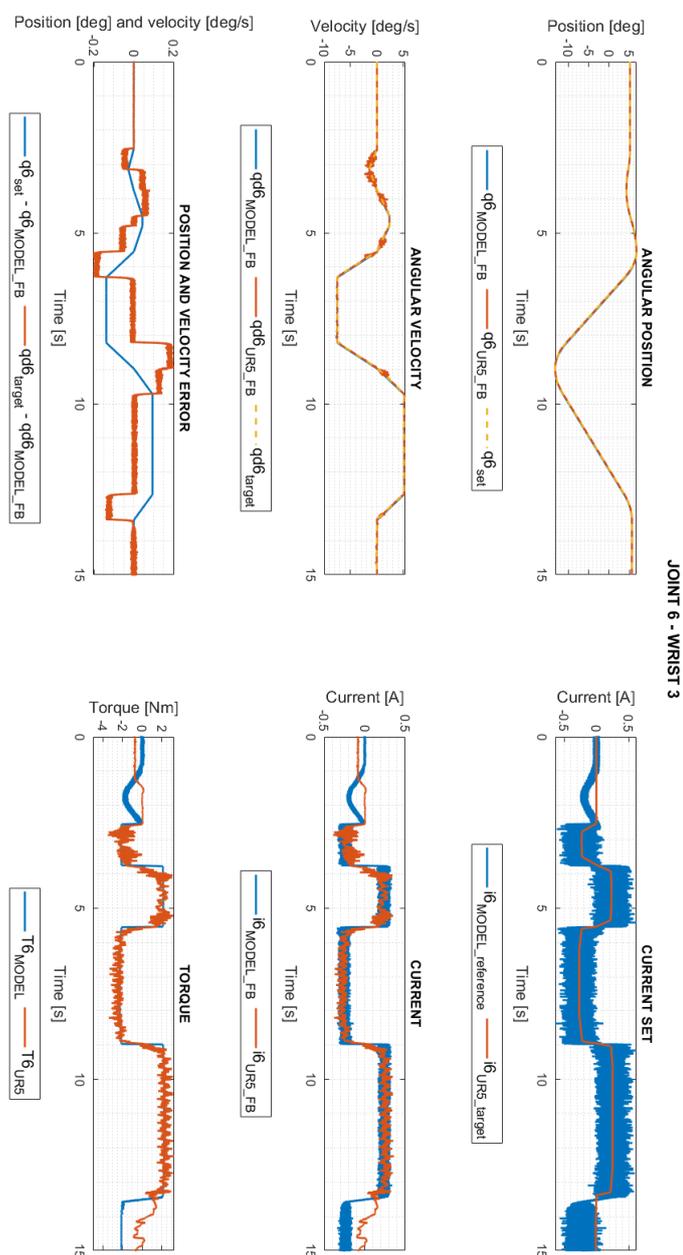


Figura 4.28: Grandezze simulate e misurate per il giunto di polso 3 per il "test3".

Capitolo 5

Iniezione di guasti a scopo diagnostico e prognostico

Nel seguente capitolo saranno analizzati più nello specifico quelli che sono i possibili guasti che possono essere iniettati nel modello, anche in vista di un'evoluzione dello stesso. Si introdurranno quelli che sono i guasti più frequenti, per i diversi sottosistemi di cui è composto il robot, e come essi possono essere modellati all'interno del modello *Simulink*TM che si è sviluppato nel capitolo 3.

Risulta infatti utile definire una classificazione di tutti i guasti più frequenti che si verificano nei robot manipolatori costruendo un quadro di riferimento dei guasti a supporto di tale classificazione. Tali guasti saranno il frutto di specifici studi volti alla definizione di una *Failure Mode, Effects, and Criticality Analysis* (FMECA) ed una *Fault tree analysis* (FTA). A seguito di tale classificazione è necessario definire un modello di iniezione di ogni determinato guasto all'interno del modello del robot sviluppato nel presente lavoro di tesi andando ad individuare a quale sottosistemi è riferito il guasto e come esso può essere modellato.

Oltre alle *fault* di seguito introdotti è necessario tener conto di guasti legati a componenti non ancora modellati ed inseriti. Tali guasti possono riguardare i cuscinetti presenti all'interno del robot, il lubrificante che viene utilizzato, la presenza dei cavi elettrici e l'utilizzo di ruote dentate.

Il modello fin qui descritto, seppur risulti valido nella descrizione del comportamento del sistema fisico reale, sia da un punto di vista cinematico che dinamico, è costituito da sottomodelli che presentano un livello di dettaglio non ancora tale da poter iniettare dei guasti direttamente all'interno dei suddetti sottomodelli. Ad esempio, per quanto riguarda il motoriduttore si ha necessità di introdurre un modello costituito da un motore trifase, e non monofase, e che all'interno del riduttore presenti una modellazione che descriva

le interazioni dinamiche tra i denti delle ruote dentate. Per far questo è infatti necessario andare a definire dei modelli per ciascun componente del robot in modo da poter descrivere la dinamica, dei singoli componenti, in maniera fedele con quello che avviene per il sistema fisico reale.

5.1 Guasti nei motori elettrici

Esistono diverse tipologie di guasti che possono presentarsi in un motore elettrico: esterni o interni.

I guasti interni, ad esempio un rotore sbilanciato, si verificano principalmente a causa di difetti di fabbricazione o in seguito a lunghi periodi di funzionamento, a causa di disallineamenti interni o flessioni dell'albero [30].

La simulazione dei guasti interni può avvenire modificando i parametri d'inerzia del motore oppure introducendo una coppia che simuli tale squilibrio sia statico che dinamico.

Ulteriori problemi possono nascere a causa di guasti sul circuito elettrico. Per simulare tali errori è necessario andare a variare i parametri elettrici inseriti all'interno del modello del motore elettrico, tenendo conto ad esempio dell'effetto che hanno la temperatura e l'usura degli avvolgimenti sulla resistenza.

Nelle macchine elettriche uno dei guasti più frequenti riguarda il cortocircuito dello statore [31]. Questo tipo di guasto può produrre un'intensità del campo magnetico superiore alla coercitività dei magneti, smagnetizzandoli in maniera permanente danneggiando il motore stesso.

I guasti da cortocircuito rappresentano il 21% dei guasti che si verificano nelle macchine elettriche [30] e solitamente si manifestano, nelle fasi iniziali, come guasti degli avvolgimenti. Per simulare tali guasti esistono diverse tecniche che vanno a modellarli matematicamente.

5.2 Guasti agli *harmonic drive*

A differenza dei riduttori ad ingranaggi, nei riduttori armonici la trasmissione dipende principalmente dall'interazione tra *flexspline* e *wave generator*, il primo in particolare rappresenta uno dei componenti più critici dell'intero sistema di trasmissione.

Nell'applicazione pratica la *flexspline* è sottoposta a stress di deformazione e carichi ciclici, per tale motivo è soggetta a cedimenti per fatica. Dopo l'uso per un lungo periodo di tempo si può avere la comparsa di cricche e l'usura dei denti portando, non solo ad una trasmissione del moto irregolare,

ma anche ad incidenti sulla linea di produzione, i quali assumono ancora più rilevanza nel campo della robotica collaborativa in termini di sicurezza per l'operatore.

Il motivo principale della comparsa di *failure* negli *harmonic drive* è legata alle *fault* che si verificano nella *flexspline*, sulla cui superficie, dopo diverse ore di funzionamento, compaiono crepe locali che si traducono in contatti anomali tra i denti e rotture locali. Tali guasti possono essere simulati andando ad agire sulle proprietà della *flexspline*, come ad esempio la rigidezza torsionale che a seguito della comparsa di cricche si riduce oppure a quella di contatto tra dente e dente nel caso di usura del dente stesso [32].

5.3 Gioco tra motore e riduttore

Uno dei possibili guasti che si possono osservare nei sistemi costituiti da motore e riduttore è il gioco che viene a crearsi all'interno del riduttore. Il riduttore è infatti costituito da una coppia di ingranaggi e a seguito dell'uso continuato si ha una parziale erosione del fianco dei denti che garantiscono la trasmissione del moto. A partire da tale erosione è possibile definire un certo valore di angolo di gioco da inserire all'interno del modello. Di seguito verrà considerato un riduttore armonico generico, con caratteristiche simili a quelle degli *harmonic drive* montati sull'UR5, le cui dimensioni sono riportate in tabella 5.1 [5].

Grandezza	Simbolo	Valore
Larghezza di base del dente	L_b	0.91 mm
Altezza del dente	H_t	0.76 mm
Raggio della <i>flexspline</i>	r_{FS}	35.25 mm
Eccentricità massima	ϵ	0.5 mm
Numero di denti della <i>flexspline</i>	Z_{FS}	240

Tabella 5.1: Parametri tipici del riduttore armonico

Si procede calcolando i due semiassi ed il valore del perimetro della *flexspline* deformata:

$$a = r_{FS} + \epsilon = 35.75mm$$

$$b = r_{FS} - \epsilon = 34.75mm$$

$$P = 2\pi\sqrt{\frac{a^2 + b^2}{2}} = 221.505mm \quad (5.1)$$

Si procede calcolando l'approssimativa larghezza del vano tra i due denti e da questo, mediante un calcolo geometrico si ricava l'angolo di gioco alla base del dente considerando una riduzione della larghezza del dente stesso del 5%.

$$space = \frac{P}{Z_{FS}} - L_b = 0.0129mm \quad (5.2)$$

$$\theta_{clearance} = \arctg\left(\frac{0.05space}{r_{FS} + \epsilon}\right) = 1.048^\circ \times 10^{-3} \quad (5.3)$$

Come detto in precedenza in realtà il gioco è legato all'usura del dente, per come funziona il riduttore armonico tale usura avviene all'altezza del diametro nominale della ruota dentata, per cui il valore del gioco deve essere riportato a tale altezza del dente, per far questo si opera una proporzione tra la base del dente e la sua altezza media.

$$\theta_{clearance,2} = \frac{\theta_{clearance}}{r_{FS} + \epsilon} \left(r_{FS} + \epsilon + \frac{H_t}{2}\right) = 1.059^\circ \times 10^{-3} \quad (5.4)$$

Per inserire tale gioco all'interno del modello è necessario inserire un blocco che si vada ad interporre tra il motore ed il riduttore.

Tale blocco di *backlash*, rappresentato in figura 5.1, riceve in ingresso la differenza di posizione ("Delta_q") e velocità ("Delta_{qd}") tra l'albero motore e l'albero in uscita dal riduttore. Tale differenza di posizione è causata appunto dalla presenza del gioco, infatti le posizioni angolari in ingresso ed in uscita dal riduttore sono differenti.

Per tener conto di tale condizione si individuano quattro diversi casi, rappresentati in figura 5.2, che possono verificarsi nell'interfacciamento all'interno del riduttore. Un primo caso (a) è quello per cui i denti dei due elementi che trasferiscono il moto sono a contatto tra di loro per cui il gioco tra i due elementi è nullo ed il trasferimento di coppia è totale, in particolare la coppia è calcolata mediante un modello molla-smorzatore. Il secondo ed il terzo caso (b e c) riguardano invece la condizione per cui il gioco non è nullo, ma i due elementi di trasmissione del moto sono comunque a contatto poiché il gioco è stato completamente recuperato. L'ultima condizione (d) riguarda invece il momento in cui il gioco tra i due elementi ha un effetto sul comportamento del sistema, e cioè quando si ha un inversione del moto ed il

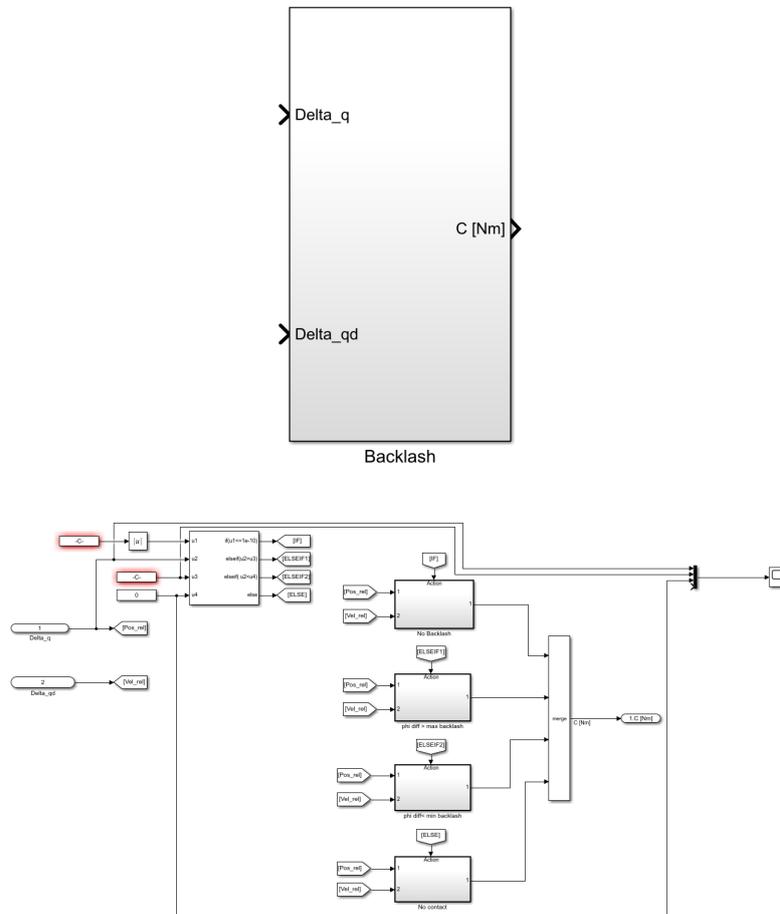


Figura 5.1: Blocco di modellazione del gioco all'interno di *Simulink*.

distacco tra i due denti che dovrebbero trasferire la coppia, in tale condizione la coppia trasmessa è nulla fino al momento in cui non si recupera il gioco in una o nell'altra direzione.

All'uscita dal blocco di *backlash* si ha la coppia scambiata tra il motore ("C") ed il riduttore che fa da input al blocco del riduttore stesso.

All'interno del modello presentato in precedenza non è stato effettivamente inserito il blocco di *backlash* poiché esso prevede un interfacciamento con un blocco dinamico non costruito in ambiente *Simscape* e per tale motivo potrà essere inserito solo in seguito allo sviluppo dello stesso, un modello cioè costituito solo da equazioni dinamiche legate al moto della struttura meccanica.

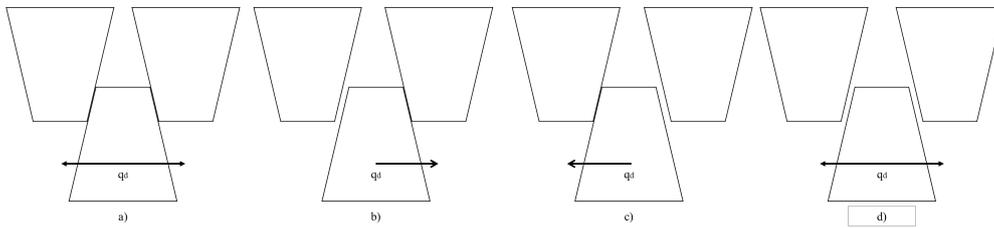


Figura 5.2: Condizioni verificabili per il *backlash*.

5.4 Guasti alla struttura meccanica

I guasti più frequenti che possono presentarsi, nel caso della struttura meccanica, sono già in parte stati analizzati nei capitoli precedenti.

Nel capitolo 3 è stato introdotto il concetto di errori geometrici, in questo caso non si tratta di guasti che compaiono durante il funzionamento del robot, bensì sono errori di montaggio o di produzione che rimangono costanti per tutta la vita utile del robot. Essi vengono introdotti mediante delle matrici di errore costanti che premoltiplicano le matrici di rototraslazione di D-H andando a modificare la posizione relativa tra due *links* successivi. Tale tipologia di errore introduce una variazione in termini di posizione ed orientazione del TCP variando infatti la posizione reciproca di due *links*. L'effetto che si avrebbe sulle correnti, e quindi sulle coppie, non può essere individuato questo perché tali errori geometrici introducono delle minime variazioni sul segnale di corrente che sono mascherate dal relativo rumore. Per tale motivo gli errori di montaggio e produzione possono essere rilevati esclusivamente mediante sensori esterni che vanno a misurare l'effettiva posizione ed orientamento del TCP.

Tali errori sono costanti e trattandosi di correzioni nell'ordine del decimo del millimetro, di solito, hanno un'influenza ridotta sulle coppie ai giunti, ma un effetto importante, nell'ordine del millimetro, sulla posa del TCP.

Un'ulteriore tipologia di errori di cui può essere affetta la struttura meccanica riguarda la presenza di giochi all'interno dei giunti.

Da un punto di vista teorico gli assi di rotazione di due corpi legati tra di loro, in questo caso il riduttore e il giunto del robot vero e proprio, risultano essere coassiali e si muovono alla stessa velocità. La presenza di un gioco radiale, rispetto ad un giunto rotoidale ideale, introduce due nuovi gradi di libertà all'interno del sistema meccanico.

Il gioco radiale può essere simulato introducendo un collegamento virtuale senza massa con lunghezza variabile. All'interno del *Simulink*TM può essere simulato mediante l'introduzione di un giunto che simuli i due nuovi gradi di

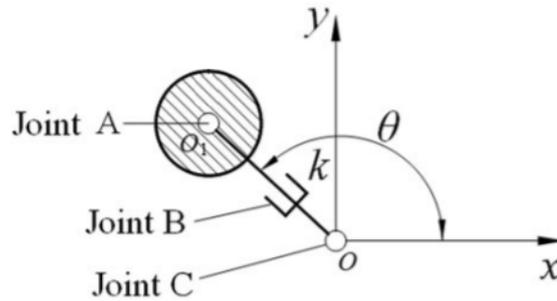


Figura 5.3: Modellazione gradi di libertà dei giochi del giunto.

libertà: una traslazione ed una rotazione, mostrate in figura 5.3, tra gli assi dei due alberi che idealmente dovrebbero essere coassiali[33].

Tale tipologia di guasto può essere inserita all'interno del modello andando a sostituire il *joint* che collega due *link* successivi. Si rimuove il normale giunto di rotazione con un giunto planare, tale giunto sostitutivo permette di inserire, non solo un gioco rispetto alla rotazione, ma anche un gioco in termini radiali. Naturalmente imponendo un gioco nullo il sistema assume un comportamento nominale del tutto uguale a quello descritto prima della modifica del giunto.

A seguito di tale modifica del modello è possibile introdurre tale gioco e valutarne gli effetti sulle grandezze misurabili dal robot.

5.5 Guasti agli *encoder*

Nelle applicazioni industriali è richiesta continuità di servizio dei sensori soprattutto nel caso in cui il sistema funzioni con una logica di controllo ad anello chiuso.

I guasti che possono verificarsi sono di diversa natura e riguardano, principalmente, la perdita totale o parziale delle informazioni sulla posizione, un eventuale offset di misura e la presenza di disturbi. Questi difetti portano all'instabilità del driver del motore o persino ad una *failure* del motore stesso. Le cause di perdita d'informazione sono diverse e legate principalmente a problemi di natura meccanica o elettronica. Le prime dipendono da diversi fattori tra i quali errori di montaggio, tolleranze assiali, velocità di rotazione dell'albero e temperature di funzionamento. Questi errori portano principalmente ad un disallineamento tra l'*encoder* e l'albero del motore[34].

Nel caso in cui i circuiti elettronici dell'*encoder* siano affetti da un guasto si ottiene un segnale in uscita diverso da quello che effettivamente descrive la posizione e la velocità dell'albero.

La *fault* considerata per questi componenti è rappresentata da un disturbo dato da un rumore con elevata ampiezza. Le possibili sorgenti di rumore sono diverse, e, nel caso di azionamenti a velocità regolabile, la principale è rappresentata dagli *inverter* [34]. Essi sono costituiti da semiconduttori di potenza che funzionano a frequenze di commutazione alte. Inoltre il rumore è generato anche dalla presenza di onde elettromagnetiche o da fili elettrici che alimentano i motori.

Per simulare la presenza di una *fault* sugli *encoder*, un possibile approccio prevede di introdurre un disturbo con un livello di rumore uguale o superiore a -28dB [35]. Tale valore è ricavato dalla precedente letteratura nella quale vengono proposti diversi test volti a determinare il livello di rumore necessario a distorcere in modo significativo i segnali di alimentazione del motore.

Tale rumore viene introdotto arbitrariamente aggiungendosi alla posizione misurata in un dato istante. Inoltre, è necessario considerare un livello di rumore di -50dB aggiunto ai segnali di corrente dello statore del motore in modo da tener conto del normale livello di rumorosità introdotto dai sensori [35].

La presenza di un disturbo sull'*encoder* genera vibrazioni e malfunzionamenti direttamente sul motore causando prestazioni insoddisfacenti.

Nel caso in cui il guasto sui sensori provochi solo un *offset* nella misura della posizione angolare non è possibile avere contezza di tale guasto a partire dalle variabili misurabili dal robot, ma solo analizzando la posa del TCP.

Ad esempio, con un offset positivo di 0.5° ed una posizione angolare di *set* di 90° un valore di 89.5° del giunto darebbe un errore nullo poiché il sensore leggerebbe un valore di 90° poiché il guasto è presente sul sensore e non nel giunto. Per avere contezza di tale guasto risulta necessario l'utilizzo di un sistema di misura esterno, assente sul robot, per avere una misura dell'esatta posa del TCP che attualmente è ricavata a partire dalle posizioni angolari dei sei giunti mediante un algoritmo di cinematica diretta. Alternativamente sarebbe necessario, a seguito di una modellazione più accurata dell'*encoder*, osservare eventuali oscillazioni nel segnale fornito dal sensore.

5.6 Analisi dei risultati a scopo prognostico

Come detto nel capitolo 1, attualmente, l'analisi prognostica si svolge sfruttando metodi *data driven*, i quali necessitano di un elevata mole di dati riguardo i guasti. La mancanza di tali dati, nel campo della robotica indu-

striale, fa sì che tale metodologia non sia utilizzabile per testare gli algoritmi di diagnostica e soprattutto di prognostica. Di conseguenza si è sviluppato un modello del robot all'interno del quale inserire i suddetti guasti ed ottenere in tal maniera dati utili per l'analisi dello stato di salute dei robot.

Una possibile limitazione a tale approccio è legato al fatto che la validazione è effettuata con un robot in buono stato di salute. Per ottenere una validazione completa del modello sarebbe però necessario effettuare delle analisi sperimentali mirate su robot con guasti noti e ben definiti in modo da validare il modello anche in caso di robot guasto. Queste analisi prevederebbe però di avere un robot guasto ottenuto o provocando il guasto, condizione naturalmente non fattibile, o a seguito di un eventuale campagna di analisi esterna all'università che permetta di studiare diversi esemplari con diverse tipologie di guasti.

Si può, però, affermare che, in assenza di incertezze riguardo la modellazione, qualsiasi comportamento che il robot reale assume differente rispetto a quello nominale del modello può essere attribuito alla presenza di un guasto del robot.

A seguito dell'ottenimento dei dati legati al comportamento non nominale del robot, a causa dell'iniezione di guasti, si possono utilizzare dei metodi di analisi che si basano su logica fuzzy e reti neurali per applicare metodologie basate su logica di soglia, test stocastici e riconoscimento di pattern [36], sfruttando così i dati ottenuti dal modello.

La metodologia illustrata non sostituisce i modelli *data-driven*, bensì ne rappresenta una tecnica diversa, ma complementare.

Per far questo è necessario effettuare un elevato numero di simulazioni introducendo, all'interno del modello, le diverse tipologie di guasti viste nel presente capitolo. In questo modo è possibile utilizzare i dati forniti dal modello per individuare delle *features*. Le *features* sono ottenute mediante la combinazione di indici, cioè segnali rilevati dal robot, osservandone i cambiamenti e come questi avvengono è possibile attribuirli ad un determinato tipo di guasto.

Tale mole di dati ottenuta dalle simulazioni viene confrontata con i dati rilevati dal robot. Per poter rilevare la presenza dei guasti uno degli approcci proposti in letteratura prevede di sfruttare le reti neurali per analizzare l'errore sulle grandezze misurate per ciascun giunto [37].

Capitolo 6

Conclusioni e sviluppi futuri

Dopo aver analizzato quelli che sono i concetti legati alla diagnostica ed alla prognostica, in particolare quella *model-based* dei robot industriali, si è descritta, in seguito, una possibile procedura atta ad ottenere un modello del robot che potesse descriverne il comportamento.

Alla descrizione dei componenti di cui è costituito il robot, è seguita l'introduzione dei sottosistemi nei quali lo stesso può essere suddiviso. Per ciascun componente si è applicata una modellazione matematica in maniera tale che il modello potesse essere introdotto in *Simulink*TM.

Le principali difficoltà, che si sono riscontrate in questa fase, hanno riguardato la mancanza di dati relativi ai parametri dinamici in termini di quantità ed affidabilità, per tale motivo, nel prosieguo dello sviluppo del modello, sarà necessario effettuare una identificazione di questi parametri.

Inoltre, la procedura, che ha richiesto diverso tempo, ha riguardato la definizione dei parametri del controllore. Non si hanno, infatti, informazioni riguardo la tipologia di controllo che viene utilizzata all'interno del robot reale e si è deciso, pertanto, di introdurre quella che è la logica di controllo più utilizzata in questo tipo di attuatori, cioè un anello più interno, con guadagno proporzionale-integrativo, che controlla la corrente che scorre nel motore, un secondo anello intermedio, con guadagno proporzionale-integrativo, che controlla la velocità di rotazione del motore ed un anello più esterno, con guadagno proporzionale, che controlla la posizione angolare del giunto.

Dai risultati che si sono mostrati all'interno del capitolo 4, è possibile affermare che, da un punto di vista cinematico, il modello descriva in maniera più che soddisfacente il comportamento del sistema fisico reale, mentre, da un punto di vista dinamico e di modellazione della parte di attuazione, il comportamento, seppur ben descritto, deve tener conto di una certa incertezza legata alla approssimazione della modellazione stessa.

Un ulteriore sviluppo del modello è rappresentato dalla modellazione più accurata dei *link*. Essi attualmente sono considerati come corpi con rigidità infinita, nella realtà essi non sono infinitamente rigidi, ma hanno un certo grado di flessibilità. All'interno del modello definito attraverso *Simscape Multibody* è possibile sostituire i blocchi che attualmente modellano i singoli membri della catena cinematica con corpi deformabili, i quali presentano determinate caratteristiche di rigidità e smorzamento. Allo stato attuale tali valori non sono noti e sarà necessario uno studio specifico in tal senso per individuarli.

L'introduzione della deformabilità dei corpi ha effetti anche sull'individuazione dei guasti. Tale modifica, oltre a migliorare l'accuratezza del modello, ha effetti anche in termini di *PHM*. Infatti, una struttura infinitamente rigida, nel caso ad esempio di piccole vibrazioni ad un giunto, porta a vedere tali vibrazioni anche sugli altri giunti, cosa che in un sistema con rigidità non infinita non sarebbe osservabile poiché le vibrazioni risulterebbero smorzate dal sistema stesso e nascoste dal rumore che si ha a valle dei sensori.

In futuro, sarà utile effettuare una ben definita campagna di prove che permetta di effettuare dei movimenti specifici che vadano ad eccitare i singoli giunti in modo da verificare quali siano effettivamente gli errori di modellazione commessi su ciascun giunto indipendentemente dal comportamento degli altri. Tali prove dovranno essere effettuate con *payload* differenti in modo da rendere più o meno dipendenti i risultati dai parametri di inerzia incogniti.

Al momento, le prove effettuate non raggiungono valori di coppia alti, ma anzi essi risultano piuttosto limitati all'interno del range nel quale il robot può effettivamente lavorare, il quale ha un valore massimo di 28Nm per i giunti del polso e 150Nm per i giunti di base, spalla e gomito. Per tale motivo sarà necessario effettuare anche delle prove che coinvolgano almeno il 50% della coppia massima esplicabile da ogni giunto.

Da un punto di vista puramente di modellazione, per come è stato concepito, il modello risulta essere modulare e sempre migliorabile. Conoscendo infatti input ed output dei diversi blocchi è possibile sostituire gli stessi con versioni più affidabili dei singoli modelli ottenendo un miglioramento complessivo delle prestazioni totali. Inoltre, nel caso in cui sia necessario concentrarsi su di un singolo sottosistema, è possibile sostituire i sottosistemi superflui, che inevitabilmente aumentano il tempo di compilazione del programma, con sistemi più semplici già introdotti nella presente trattazione. Ad esempio si può scegliere di utilizzare il modello di motoriduttore semplice piuttosto che quello con l'*harmonic drive* o il modello di motore monofase in controllo posizione, velocità e corrente con logica PWM piuttosto che il modello del motore trifase.

D'altro canto, è possibile migliorare il modello sostituendo il motore a cor-

rente continua con un motore asincrono a corrente alternata, che rappresenta il reale motore montato sui robot della *Universal Robots*TM.

Una volta ottenuto un modello il più possibile vicino, in termini di relazioni input-output, al sistema fisico reale, sarebbe necessario effettuare una campagna di prove che fornisca informazioni per diverse tipologie di guasti.

Analizzando tali prove, sarà possibile estrapolare degli indicatori, legati alle variabili misurabili direttamente dal robot, capaci di definire lo stato di salute del robot e non solo se questi è vicino ad una *failure* fatale, ma anche se qualche *fault* ha causato perdita di precisione, di ripetibilità, o comunque una variazione nelle prestazioni del robot.

La metodologia di modellazione presentata all'interno della trattazione ha una valenza universale nel campo della robotica industriale e può essere applicata anche a diverse tipologie di manipolatori. Tenendo conto, però, della mancanza di dati in questo campo, le possibilità che essa offre sono ampie e vantaggiose in termini di programmazione della manutenzione e, quindi, di aumento della sicurezza e di risparmio economico.

Bibliografia

- [1] Singularities of a typical collaborative robot.
- [2] Lorenzo Sciavicco and Bruno Siciliano. *Robotica industriale: modellistica e controllo di manipolatori*. McGraw-Hill Libri Italia, 2000.
- [3] Guixiu Qiao, Craig Schlenoff, and Brian A Weiss. Quick positional health assessment for industrial robot prognostics and health management (phm). In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1815–1820. IEEE, 2017.
- [4] Peter Corke. *Robotics, Vision and Control: Fundamental Algorithms In MATLAB® Second, Completely Revised*, volume 118. Springer, 2017.
- [5] Chuang Zou, Tao Tao, Gedong Jiang, Xuesong Mei, and Junhui Wu. A harmonic drive model considering geometry and internal interaction. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 231(4):728–743, 2017.
- [6] Universal Robots. *UR5 Manual*.
- [7] Universal Robots. *UR5 Technical Specification*.
- [8] Universal robots support: Maximum joint torques., 2020.
- [9] Tawfik Borgi, Adel Hidri, Benjamin Neef, and Mohamed Saber Naceur. Data analytics for predictive maintenance of industrial robots. In *2017 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, pages 412–417. IEEE, 2017.
- [10] George Z Vachtsevanos. *Intelligent fault diagnosis and prognosis for engineering systems*. John Wiley & Sons, 2006.

- [11] Ikbal Eski, Selcuk Erkaya, Sertaç Savas, and Sahin Yildirim. Fault detection on robot manipulators using artificial neural networks. *Robotics and Computer-Integrated Manufacturing*, 27(1):115–123, 2011.
- [12] Cesare Fantuzzi, Cristian Secchi, and Antonio Visioli. On the fault detection and isolation of industrial robot manipulators. *IFAC Proceedings Volumes*, 36(17):399–404, 2003.
- [13] Mohd Amin Abd Majid and Fauzi Fudzin. Study on robots failures in automotive painting line. *Asian Res. Publishing Netw.*, 12(1):62–67, 2017.
- [14] Bernd Freyermuth. An approach to model based fault diagnosis of industrial robots. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 1350–1351. IEEE Computer Society, 1991.
- [15] Cobotsguide: Universal robots, 2015.
- [16] Luca Iuliano. *Sistemi integrati di produzione: I robot industriali*. 2019.
- [17] Universal robots support: What is a singularity?, 2018.
- [18] Ian Young. Kollmorgen drives the lightweight helpers of universal robots, 2017.
- [19] Jacqueline Regnier. Universal robots given a helping hand, 2015.
- [20] Harmonic drive: Strain wave gear technology.
- [21] Aksim™ supports universal robots for smart factory automation.
- [22] Michael Kutzer. Urtoolbox. *GitHub*, 2020.
- [23] Peter Corke. Robot manipulator capability in matlab: A tutorial on using the robotics system toolbox. *IEEE Robotics & Automation Magazine*, 24(3):165–166, 2017.
- [24] R Guida, MC De Simone, P Dašić, and D Guida. Modeling techniques for kinematic analysis of a six-axis robotic arm. In *IOP Conference Series: Materials Science and Engineering*, volume 568, page 012115. IOP Publishing, 2019.

- [25] Parham M Kebria, Saba Al-Wais, Hamid Abdi, and Saeid Nahavandi. Kinematic and dynamic modelling of ur5 manipulator. In *2016 IEEE international conference on systems, man, and cybernetics (SMC)*, pages 004229–004234. IEEE, 2016.
- [26] M Granja, N Chang, V Granja, M Duque, and F Llulluna. Comparison between standard and modified denavit-hartenberg methods in robotics modelling. *vol*, 1:1–10, 2016.
- [27] Katharina Kufieta and Jan Tommy Gravdahl. *Force Estimation in Robotic Manipulators: Modeling, Simulation and Experiments. The UR5 Manipulator as a Case Study*. PhD thesis, Diploma Thesis, Department of Engineering Cybernetics, Norwegian University . . . , 2014.
- [28] Massimo Sorli and Giuseppe Quaglia. *Meccatronica*. Politeko, 2003.
- [29] Timothy D Tuttle and Warren P Seering. A nonlinear model of a harmonic drive gear transmission. *IEEE Transactions on Robotics and Automation*, 12(3):368–374, 1996.
- [30] SK Ahamed, S Karmakar, M Mitra, and S Sengupta. Diagnosis of induction motor faults due to broken rotor bar and rotor mass unbalance through discrete wavelet transform of starting current at no-load. *Journal of Electrical systems*, 6(3):442–456, 2010.
- [31] Arun Gandhi, Timothy Corrigan, and Leila Parsa. Recent advances in modeling and online detection of stator interturn faults in electrical motors. *IEEE Transactions on Industrial Electronics*, 58(5):1564–1575, 2010.
- [32] Jianlin Zheng and Wei Yang. Failure analysis of a flexspline of harmonic gear drive in stc industrial robot: Microstructure and stress distribution. In *IOP conference series materials science and engineering, Melbourne, Australia*, pages 15–16, 2018.
- [33] Quan Liu, Sheng-Nan Lu, and Xi-Lun Ding. An error equivalent model of revolute joints with clearances for antenna pointing mechanisms. *Chinese Journal of Mechanical Engineering*, 31(1):39, 2018.
- [34] M Bourogaoui, I Jlassi, S Khojet El Khil, and H Ben Attia Sethom. An effective encoder fault detection in pmsm drives at different speed ranges. In *2015 IEEE 10th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED)*, pages 90–96. IEEE, 2015.

- [35] M Bourogaoui, H Berriri, H Ben Attia Sethom, and I Slama Belkhodja. Wavelets and parity equations methods based position sensor fault detection in pmsm drives. In *Proc. Int. Conf. ELECTRIMACS2011, Cergy-Pontoise, France, 2011*.
- [36] M Dev Anand, T Selvaraj, and S Kumanan. Fault detection and fault tolerance methods for industrial robot manipulators based on hybrid intelligent approach. *Advances in Production Engineering & Management*, 7(4), 2012.
- [37] F Caccavale, P Cilibrizzi, F Pierri, and L Villani. Actuators fault diagnosis for robot manipulators with uncertain model. *Control Engineering Practice*, 17(1):146–157, 2009.