

TESI DI LAUREA MAGISTRALE



Identificazione parametri cinematici e degli errori geometrici di un
robot collaborativo

Relatore:
Prof. Massimo Sorli
Correlatore:
Ing. Andrea Raviola

Candidato:
Gabriele Basile

Abstract

La prognostica permette di avere una stima della vita utile rimanente di un componente o di un sottosistema. Per fare ciò, è necessario avere dati di comportamenti del sistema in condizioni di funzionamento nominali e non. In quest'ottica, spesso vengono costruiti modelli *high-fidelity* di componenti in modo da generare set di dati simulati utili per allenare algoritmi *data-driven*. La calibrazione cinematica di un manipolatore industriale permette di identificare dei parametri utili alla costruzione di una replica digitale della macchina.

L'obiettivo di questo lavoro di tesi è quello di identificare i parametri cinematici di un robot manipolatore e di usare tali parametri per poterne individuare gli errori geometrici utili all'implementazione di un modello *high fidelity*.

Il caso studio sul quale si è lavorato è un robot manipolatore UR5 della Universal Robots presente all'interno del Dipartimento di Ingegneria Meccanica e Aereospaziale del Politecnico di Torino.

L'identificazione cinematica in letteratura è possibile attraverso un processo di calibrazione cinematica, possibile attraverso l'ausilio di un sistema di misura esterno, attrezzatura che non è stata disponibile per questo lavoro. All'interno della memoria del robot sono disponibili le modifiche dei parametri cinematici ottenuti in fase di calibrazione dal costruttore.

Questi parametri cinematici non hanno significato fisico e sono inutilizzabili per la cinematica inversa e la costruzione di un modello *high-fidelity*.

A seguito un'approfondita ricerca bibliografica sul processo di calibrazione si è cercato di identificare i parametri cinematici utilizzando diversi metodi di identificazione. Ipotizzando che il modello di rappresentazione del costruttore fosse il motivo dell'assenza di significato fisico, si sono riscritti i parametri cinematici, attraverso una proiezione geometrica, in una convenzione più adatta.

Non essendo possibile effettuare un processo di calibrazione si è proceduto con la simulazione della fase di misurazione; questa è stata approssimata utilizzando le

ABSTRACT

pose ottenute attraverso i parametri identificati dal costruttore, come l'output di un sistema di misura esterno. Per la fase di identificazione si sono utilizzati metodi di identificazione deterministici e stocastici quali il metodo di Gauss-Newton e gli Algoritmi genetici. I parametri cinematici ottenuti non sono frutto di una calibrazione, bensì di una identificazione cinematica; data l'assenza di un sistema di misura esterno. Infine, dai parametri cinematici, sono stati calcolati gli errori geometrici degli assi giunto del robot.

Sommario

Abstract.....	1
Indice delle figure	5
Indice dei grafici	7
Indice delle tabelle	8
1 Introduzione	9
1.1 Approccio PHM e Prognostica <i>Model-Based</i>	10
1.2 Calibrazione e modello <i>high fidelity</i>	11
1.3 Il robot collaborativo UR5	14
1.3.1 Posa di un manipolatore.....	17
2 Calibrazione: stato dell'arte	18
2.1 Classificazione degli errori	19
2.1.1 Ripetibilità ed accuratezza	19
2.1.2 Classificazione delle fonti di errore	22
2.2 Processo di calibrazione	23
2.2.1 Modellazione	24
2.2.2 Misurazione	28
2.2.3 Identificazione	42
2.2.4 Implementazione.....	56
3 Modello del problema cinematico diretto.....	57
3.1 Denavit Hartenberg Standard.....	58
3.1.1 Assegnazione dei parametri di DH	61
3.2 Convenzione di Hayati-Mirmirani e GDH.....	65
3.2.1 Confronto grafico tra DH Standard e GDH/Hayati	70
3.3 Proiezione geometrica SR UR5	75
3.3.1 Introduzione degli strumenti utili all'identificazione GDH.....	75
3.3.2 Identificazione dei parametri cinematici GDH.....	80
3.3.3 Verifica dei risultati e convalida dei dati	88
4 Identificazione dei parametri cinematici.....	95
4.1 Identificazione SVD e Gauss Newton.....	97
4.1.1 Identificazione SVD	100

SOMMARIO

4.1.2	Analisi SVD della matrice Jacobiana	103
4.1.3	Identificazione LSM o Gauss-Newton	107
4.2	Identificazione con algoritmi genetici.....	110
4.2.1	Algoritmo genetico semplice (GA).....	112
4.2.2	Algoritmo Genetico Multiobiettivo (Multi-GA).....	115
4.2.3	Algoritmo Genetico Multiobiettivo ibrido (Multi-GA Ibrido)	119
5	Errori geometrici.....	125
6	Conclusioni e sviluppi futuri.....	132
	BIBLIOGRAFIA	135

Indice delle figure

Figura 1.1: Robot collaborativo UR5 e classificazione dei nomi giunti [9].....	15
Figura 2.1: Legame tra ripetibilità ed accuratezza [8]	21
Figura 2.2: Rappresentazione di un teodolite [8].....	30
Figura 2.3: Schema del funzionamento di un interferometro semplice [8].....	32
Figura 2.4: Apparecchiature calibrazione ad interferometria laser [23].....	33
Figura 2.5: Misura distanza da diverse sorgenti [8].....	35
Figura 2.6: Esempio misurazione con visione stereoscopica [8].	37
Figura 2.7 Sistemi di misura CompuGauge [25]	37
Figura 2.8: Piastra forata e End Effector di precision [21]	38
Figura 2.9: Controllo dell'orientamento dell'EE all'interno del foro [35].....	39
Figura 2.10: <i>Ball Bar Gauge</i> e configurazioni nello spazio [27].....	40
Figura 2.11: Robot e cubo di granito [28].....	41
Figura 2.12: Calibrazione di due robot e catena cinematica chiusa [34].....	41
Figura 2.13: Fasi algoritmo filtro di Kalman	53
Figura 3.1: Esempio di due <i>link</i> contigui nella convenzione di Denavit Hartenberg [14]	59
Figura 3.2: Disposizione assi giunto convenzione DH Standard	61
Figura 3.3: Rappresentazione SR robot UR5 forniti dalla Universal Robots [43].....	63
Figura 3.4 Modello 3D del robot UR5 e rappresentazione dei SR dei <i>link</i> [43].....	64
Figura 3.5: Disposizione assi giunto convenzione di Hayati	66
Figura 3.6: Disposizione assi giunto convenzione GDH	69
Figura 3.7: Parametri di DH standard teorici del <i>link</i> 2 del UR5.....	71
Figura 3.8 Parametri di DH standard identificati dalla UR del <i>link</i> 2 del UR5.....	72
Figura 3.9: <i>Link</i> 2 con disallineamento di rotazione lungo l'asse y_2 , convenzione Hayati	72
Figura 3.10 : Esempio <i>Link</i> 2* con disallineamento di rotazione lungo l'asse y_2 , convenzione DH standard	73
Figura 3.11 : Esempio <i>Link</i> 2 con disallineamento di rotazione lungo l'asse y_2 , convenzione Hayati e DH standard	74
Figura 3.12: Riepilogo dei sistemi di riferimento intermedi e delle trasformazioni omogenee	77
Figura 3.13: Sistemi di riferimento proiezione geometrica.....	79
Figura 3.14 <i>Link</i> 2 sistemi di riferimento intermedi	82
Figura 3.15: Sistemi di riferimento passanti per lo stesso asse giunto.....	85

CAPITOLO 1 - INDICE DELLE FIGURE

Figura 3.16: Rappresentazione convenzione asse angolo per il TCP.....	89
Figura 5.1: Errori geometrici asse i-esimo [1]	125

Indice dei grafici

Grafico 3.1: Errori posa punti validazione da proiezione geometrica (convenzione GDH)	91
Grafico 4.1: Attuazione ai giunti usati in fase Identificazione e Convalida.....	96
Grafico 4.2: Errori posa punti validazione da identificazione SVD (convenzione GDH)	101
Grafico 4.3: Autovalori ed energia Cumulativa Jacobiana variabili linearmente indipendenti	104
Grafico 4.4: Autovalori ed En. Cumulativa Jacobiana variabili DH.....	106
Grafico 4.5: Confronto autovalori e Energia Cumulativa modello GDH e modello DH	107
Grafico 4.6: Riduzione parametro di convergenza nel metodo LSM.....	108
Grafico 4.7: Errori posa punti validazione da identificazione LSM (convenzione GDH)	109
Grafico 4.8: Errori posa punti validazione da identificazione GA (convenzione DH)	113
Grafico 4.9: Errori posa punti validazione da identificazione GA (convenzione GDH).....	114
Grafico 4.10: Diagramma di Pareto Multi-GA (convenzione GDH).....	116
Grafico 4.11: Errori posa punti validazione da identificazione Multi-GA (convenzione DH)	117
Grafico 4.12: Errori posa punti validazione da identificazione Multi-GA (convenzione GDH) ..	118
Grafico 4.13 Diagramma di Pareto Multi-GA Ibrido (convenzione GDH)	121
Grafico 4.14: Errori posa punti validazione da identificazione Multi-GA Ibrido (convenzione DH)	122
Grafico 4.15: Errori posa punti validazione da identificazione Multi GA Ibrido (convenzione GDH).....	123

Indice delle tabelle

Tabella 1.1: Capacità di carico modelli UR costruiti dalla Universal Robots.....	14
Tabella 1.2: Caratteristiche tecniche UR5 [5].....	16
Tabella 2.1 Risultati lavori di calibrazione	26
Tabella 3.1: Parametri di DH standard teorici forniti dalla Universal Robots del robot UR5 [43].	63
Tabella 3.2: Modifiche ai parametri di DH standard del robot UR5 fornite dalla Universal Robot	64
Tabella 3.3: Parametri di DH standard del robot UR5 calibrato dalla UR senza senso fisico	70
Tabella 3.4: Modifiche parametri cinematici da proiezione geometrica (convenzione GDH).....	87
Tabella 3.5: Confronto deviazione errori TCP usando i parametri cinematici GDH e robot non calibrato.....	92
Tabella 3.6: Deviazione errori convenzione GDH e robot non calibrati per 4 robot diversi	94
Tabella 4.1: Modifiche parametri cinematici identificazione SVD	100
Tabella 4.2: Confronto errori convalida dell'identificazione SVD e robot non calibrato	102
Tabella 4.3: Errori convalida identificazione SVD.....	103
Tabella 4.4 Modifiche ai parametri cinematici identificata mediante LSM.....	108
Tabella 4.5: Confronto errori convalida dell'identificazione LSM e proiezione geometrica.....	109
Tabella 4.6: Errori identificazione LSM per 4 robot diversi.....	110
Tabella 4.7: Parametri funzione fitness algoritmo genetico.....	112
Tabella 4.8: Modifiche ai parametri cinematici GA (Convenzione DH)	113
Tabella 4.9: Modifiche ai parametri cinematici GA (Convenzione GDH)	113
Tabella 4.10: Caratteristiche errori identificazione GA	115
Tabella 4.11 Parametri funzione fitness Multi-GA.....	116
Tabella 4.12: Modifiche ai parametri cinematici Multi-GA (Convenzione DH).....	117
Tabella 4.13: Modifiche ai parametri cinematici Multi-GA (Convenzione GDH).....	117
Tabella 4.14: Caratteristiche errori identificazione Multi-GA.....	118
Tabella 4.15: Parametri funzione fitness Multi-GA Ibrido	121
Tabella 4.16: Modifiche ai parametri cinematici Multi-GA Ibrido (Convenzione DH)	121
Tabella 4.17: Modifiche ai parametri cinematici Multi-GA Ibrido (Convenzione GDH)	122
Tabella 4.18: Caratteristiche errori identificazione GA Multiobbiettivo Ibrido	123
Tabella 5.1: Errori geometrici con senso fisico robot UR5.....	131

Capitolo 1

1 Introduzione

La tecnologia robotica è sempre più integrata con sistemi manifatturieri molto esigenti e complessi, questo rende indispensabile la presenza di sistemi robotici più affidabili.

Dal momento in cui un sistema robotico viene messo in servizio a collaborare in un processo manifatturiero, questo inizierà a degradare le sue componenti. Senza manutenzione, questa degradazione porterà a difetti e guasti che potranno andare ad intaccare il processo. Questi difetti e/o guasti possono generare tempi di fermo imprevisti e cali nella produzione se non risolti opportunamente.

Il degradarsi dell'accuratezza di posizionamento di un sistema robotico può portare ad una diminuzione della qualità e dell'efficienza della produzione. Il degradarsi dell'integrità del sistema durante il processo produttivo è più difficile da rilevare rispetto ad una interruzione completa del sistema. Dato che i sistemi robotici sono molto utilizzati in settori di alta precisione (ad es. aereospaziale, automobilistico), è importante comprendere come si degrada l'accuratezza nel posizionamento di un robot in modo tale da poter ottimizzare le strategie di manutenzione e controllo dei sistemi robotici [1].

La calibrazione rappresenta il primo passo necessario per poter ottimizzare l'utilizzo dei sistemi robotici, migliorando qualità e volume di produzione in tutti gli ambiti produttivi. La prognostica, che si occupa di prevedere in modo accurato e preciso la vita utile rimanente di un componente o di un sottosistema, può trarre vantaggio da una solida comprensione della calibrazione, in quanto questo permette di identificare le caratteristiche fisiche e geometriche utili ai fini predittivi di alcuni modelli prognostici.

1.1 Approccio PHM e Prognostica *Model-Based*

I tempi di fermo imprevisti e la perdita di produzione sono "punti deboli" per i produttori, soprattutto perché si traducono in perdite finanziarie.

Per ridurre al minimo questi punti deboli, i produttori stanno sviluppando una nuova tecnica di monitoraggio della salute, diagnostica, prognostica e manutenzione (nota come *prognostics and health management* (PHM)) per far progredire lo stato dell'arte nelle loro strategie di manutenzione.

La PHM è un approccio usato a supportare il ciclo di vita del sistema, cerca di ridurre/eliminare la manutenzione ordinaria effettuata ogni determinato intervallo di tempo. Attraverso un monitoraggio accurato, si cerca di rilevare i guasti sul nascere e di prevedere eventuali guasti imminenti [2].

La PHM può essere applicata sia a livello di componente che a livello di sistema. La PHM a livello di componente è in genere focalizzato sul monitoraggio dello stato di salute delle singoli componenti (ad esempio ingranaggi, motori e dispositivi elettronici) per determinare se lo stato del componente monitorato è degradato tenendo conto dei parametri ambientali, operativi e in funzione delle prestazioni utilizzate [3]

La PHM per l'intero sistema valuta lo stato di salute dell'intero apparato tenendo conto di funzione ed architettura del sistema e dei parametri del processo considerato [4]. Attraverso una buona attività prognostica è possibile ritardare la necessità di sostituire dei componenti che possono ancora non pregiudicare il corretto funzionamento del sistema.

Infatti, la sostituzione di un componente o piccole imprecisioni dei *link*, della base di appoggio o degli utensili montati al robot, possono causare imprecisioni nel posizionamento del *Tool Center Point* (TCP) del robot.

Nell'ambito della prognostica dei robot esistono degli schemi *model-based* che sfruttano modelli che descrivono il sistema fisico preso in considerazione. Questi forniscono un mezzo per definire degli indici, legati a componenti critici del sistema, tali da permettere di valutare gli effetti cumulativi dell'utilizzo del robot e tradurlo in termini di vita utile residua del componente/sistema.

Integrando tecniche di modellazione fisica è possibile ricercare quella funzione che definisce la vita utile residua dei componenti in funzione di diversi parametri, come l'errore di posizione sistematico del robot o eventuali fluttuazioni nella corrente del motore [5].

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.

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.

Per poter realizzare un modello high fidelity è necessaria una conoscenza approfondita del robot.

Tra gli elementi che bisogna modellare sintetizzando si hanno [6]:

- La struttura fisica e geometrica del robot: costituisce i corpi rigidi del robot in movimento;
- Motoriduttori: permette l'attuazione dei giunti, costituiti da motore e riduttore;
- Sensori: attraverso la retroazione del loro segnale dotano il robot della propriocezione del *Tool center point* e dei *link*;
- Unità di controllo: riceve ed invia i segnali, definendo la traiettoria del robot e ne controlla i movimenti.

1.2 Calibrazione e modello *high fidelity*

La calibrazione del robot è un processo attuato per migliorare l'accuratezza di un robot, questo avviene aggiornando il suo software di controllo [7].

Dato che i robot, come altri dispositivi meccanici, possono essere influenzati da lievi variazioni o derive causate da derive dimensionali, tolleranze e sostituzione

dei componenti, la calibrazione può ridurre al minimo il rischio di dover modificare i programmi applicativi a causa di lievi variazioni o derive causate dai fattori sopracitati nel sistema robotico [8].

La calibrazione permette di ottenere delle correzioni dei parametri geometrici come la geometrie asse-giunto, gli offset degli angoli giunto e dei parametri non geometrici come la *compliance* (elasticità del giunto e dei *link*), errori di forma degli ingranaggi (eccentricità ed errori di trasmissione), giochi ed espansioni degli organi di collegamento in funzione della temperatura.

Ottenute le correzioni, la calibrazione aggiorna il software di controllo del robot.

Il software di controllo, attraverso le correzioni dei parametri geometrici e non geometrici, permette di comandare il robot fisico in modo accurato.

Queste modifiche dei parametri possono essere utilizzate per creare un modello *high fidelity* che simuli il robot fisico che si vuole analizzare.

Senza le informazioni generate da una opportuna calibrazione, non è possibile creare ed analizzare una copia accurata del robot reale in un ambiente di calcolo. L'obiettivo del modello *high fidelity* è quello di poter descrivere il più fedelmente possibile il robot reale e le sue caratteristiche, la calibrazione può essere un ottimo strumento in grado di caratterizzare il modello del robot da prendere in analisi; e possibile dunque costruire un modello accurato del robot in modo da poter ottenere una replica digitale della macchina e, con questa, meglio valutare gli effetti di possibili guasti sul sistema.

PHM, modello *high fidelity* e calibrazione sono collegati e necessari per poter ottimizzare l'utilizzo dei sistemi robotici, migliorando qualità e volume di produzione in tutti gli ambiti produttivi.

L'obiettivo di questa tesi consiste nel calcolare i parametri cinematici, utili all'implementazione di un modello *high fidelity*.

Il robot di cui si vuole caratterizzare le geometrie è il robot UR5, costruito dalla Universal Robots è descritto al paragrafo 1.3.1. All'interno della memoria del robot, sono disponibili le modifiche dei parametri cinematici riscontrate a seguito della calibrazione effettuata dal costruttore. Come viene mostrato nel Terzo capitolo e in un altro lavoro di tesi [6], queste modifiche dei parametri cinematici non hanno senso fisico e non possono essere utilizzate all'interno di un modello *high fidelity*.

All'interno della tesi viene approfondito l'intero processo di calibrazione, il quale risulta particolarmente complesso e dispendioso. Nel capitolo 2 è stata effettuata una revisione dello stato dell'arte della calibrazione, si è cercato di approfondire come avviene il processo di calibrazione confrontando decenni di ricerche.

Nei laboratori del Politecnico non sono presenti strumenti di misura esterni idonei per poter effettuare un processo di calibrazione, in quanto certe attrezzature essendo molto precise e tecnologicamente avanzate, risultano assai costose. Data l'assenza delle attrezzature fondamentali per poter portare avanti una calibrazione completa, si sono cercate varie strade alternative per poter caratterizzare il manipolatore.

Nel capitolo 3 sono state descritte ed approfondite le convenzioni di rappresentazione dei parametri cinematici. Nello stesso si è ipotizzato che, descrivendo le geometrie non idonee in una diversa convenzione di rappresentazione, queste avrebbero potuto assumere significato fisico.

Data l'impossibilità di procedere con la fase di misura, questa è stata sostituita da una simulazione. Si sono utilizzate a questo scopo le informazioni disponibili nella memoria del manipolatore, quali i punti utilizzati dal costruttore per la calibrazione e le modifiche dei parametri cinematici senza senso fisico. Per simulare, si sono trattate le pose del TCP ottenute con i parametri corretti dal costruttore, come l'output di una macchina di misura virtuale. Questa simulazione porta con sé delle semplificazioni, in quanto il modello digitale non presenterà errori di ripetibilità né errori di natura non geometrica. Fatte queste semplificazioni della fase di misura, nel capitolo 4 si è proceduto con l'identificazione cinematica. Si sono seguiti diversi tipi di approcci, alcuni utilizzati in letteratura (Gauss-Newton), altri di tipo stocastico, quest'ultimi comodi per la risoluzione di sistemi complessi senza dover cercare una soluzione analitica per poter minimizzare le funzioni obiettivo (algoritmi genetici). Infine, nel capitolo 5 sono state utilizzate le modifiche dei parametri cinematici ottenuti per calcolare delle matrici di errore che descrivono la deviazione degli assi di rotazione dei giunti del robot [1]; queste matrici di errore sono utilizzate all'interno dei modelli high fidelity.

I risultati ottenuti non sono frutto di una calibrazione. Data l'impossibilità di effettuare la fase di misura, il lavoro si è occupato di affrontare due delle quattro fasi della calibrazione: modellazione ed identificazione. I risultati ottenuti sono

quindi il prodotto di una identificazione cinematica. Questo lavoro non può essere definito come un lavoro di calibrazione a causa delle semplificazioni portate dalla simulazione della fase di misura.

Le considerazioni fatte per la sola identificazione possono essere estese ad eventuali processi di calibrazione. Con la differenza che in un processo di calibrazione reale saranno presenti errori che non esistono all'interno della simulazione, quali: incertezza del sistema di misura, errori non geometrici e di ripetibilità del manipolatore.

1.3 Il robot collaborativo UR5

Il manipolatore del quale si vogliono identificare i parametri cinematici è il robot collaborativo UR5. La scelta è ricaduta su quest'ultimo in quanto è presente fisicamente nei laboratori del Politecnico di Torino. Il robot è costruito dalla compagnia danese Universal Robots che fornisce diverse taglie di robot collaborativi. Sono disponibili nel mercato la *CB-series* e la *E-series*, dove l'ultima rappresenta la più recente serie di produzione del costruttore, il robot presente nei laboratori fa parte della *CB-series*. La denominazione del robot si riferisce alla sua capacità di carico in kilogrammi. Per entrambe le serie oltre al modello UR5 esistono sul mercato il modello UR3 ed il modello UR10 che sono rispettivamente più piccolo e più grande rispetto ad UR5. Esiste inoltre, solo per la *E-series*, il modello UR16 con una capacità di carico ancora maggiore rispetto ai precedenti.

	Capacità di carico [kg]	Sbraccio [mm]
UR3	3	500
UR5	5	850
UR10	10	1300
UR16	16	900

Tabella 1.1: Capacità di carico modelli UR costruiti dalla Universal Robots

Tutti i robot UR indicati in tabella, incluso l'UR5, sono di tipo antropomorfo, realizzato per mezzo di sei giunti rotoidali, sono tutti di tipo collaborativo ed hanno caratteristiche comuni, in Figura 1.1 l'UR5.

I robot collaborativi presentano numerosi vantaggi tra i quali:

- **facilità di programmazione:** legata alla semplicità ed all'efficacia dell'interfaccia robot-utente, costituita da uno schermo touch screen detto *teach pendant*;
- **flessibilità:** questi robot sono semplici da programmare e possono passare rapidamente da un'attività all'altra. I manipolatori sono molto leggeri e possono essere montati su superfici verticali e orizzontali; infatti, come si può notare nella scheda tecnica nella Tabella 1.2 il modello UR5 pesa 18,4 kg;
- **collaboratività:** è possibile limitare le coppie al giunto, questo permette al robot di lavorare a diretto contatto con l'uomo, anche in assenza di gabbia protettiva;
- **compatibilità con varie periferiche:** possono essere installati sensori di forza, telecamere ed altri strumenti personalizzati;
- **buon rapporto tra massa del robot e carico massimo consentito:** il rapporto è di circa 3:1, ciò consente di movimentare un'ampia gamma di carichi con una macchina relativamente piccola.

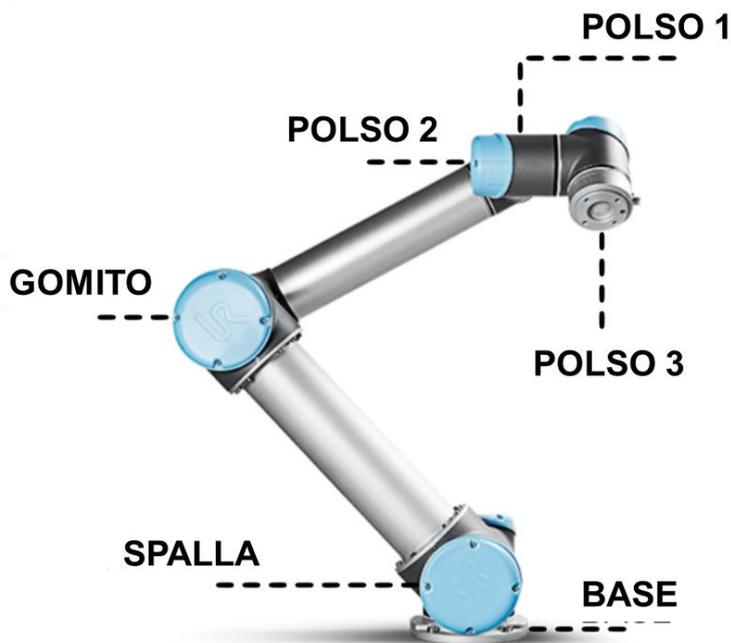


Figura 1.1: Robot collaborativo UR5 e classificazione dei nomi giunti [9]

In figura 1.1 il robot collaborativo UR5, le cui caratteristiche sono riportate in Tabella 1.2.

Materiale	Alluminio e Polipropilene
Peso	18,4kg
Sbraccio	850mm
Ripetibilità	$\pm 0,1$ mm
Raggio d'azione dei giunti	$\pm 360^\circ$
Velocità massima dei giunti	$\pm 180^\circ/s$
Range temperatura	0-50°C
Classificazione IP	IP54
Alimentazione elettrica	48V DC
Consumo energia elettrica	Min. 90 W Avg. 150W Max. 325W
Porte I/O	Ingressi digitali: 2 Ingressi analogici: 2 Uscite analogiche: 0 Uscite digitali: 2
Vita operativa	35000h

Tabella 1.2: Caratteristiche tecniche UR5 [5]

La struttura meccanica del robot è costituita dai *link* e dai giunti. Questi si alternano costituendo una catena cinematica aperta, costituita da sei giunti rotoidali che fungono da collegamento tra un *link* ed un altro. I robot costituiti da *link* di diversa geometria in serie vengono detti manipolatori di tipo seriale. Il primo *link* è connesso ad una superficie detta base, la combinazione della base e di tutti *link* arriva all'estremità distale del manipolatore detta *Tool Center Point*. Sul TCP è possibile montare un utensile o un attuatore, questo elemento finale viene detto *End Effector* (EE). La trattazione del modello e delle varie componenti del manipolatore seriale verrà approfondita nel capitolo 3.

Questo manipolatore viene detto antropomorfo, in analogia alla conformazione del braccio umano. Infatti, i primi tre giunti sono detti base, spalla e gomito, mentre i restanti tre si occupano invece dell'orientamento dell'organo terminale e sono detti polso. La struttura antropomorfa è comune nell'industria manifatturiera, in quanto risulta essere particolarmente versatile. A differenza di molti manipolatori, questi non presentano un polso sferico.

Tra le caratteristiche del robot spiccano il suo peso ridotto ed il raggio di azionamento dei giunti. La combinazione di queste fornisce a questo robot la caratteristica di essere particolarmente flessibile e permette di ottenere un'area di lavoro particolarmente ampia. L'UR5 è un robot pensato per l'uso collaborativo uomo-macchina, la velocità massima dei giunti e la sua ripetibilità risultano inferiori rispetto a quella dei robot industriali.

1.3.1 Posa di un manipolatore

L'obiettivo primario del manipolatore è di muovere l'EE o il suo TCP in una specifica posizione o lungo una traiettoria specifica. Dato che l'EE generalmente è un utensile adoperato per effettuare dei task, è opportuno che questo raggiunga una opportuna posizione desiderata nello spazio. Trattando l'EE come un corpo rigido si rende dunque necessario che il manipolatore sia in grado di garantire il corretto posizionamento di un punto significativo dell'EE nello spazio di lavoro, assicurandosi che il punto venga raggiunto attraverso il corretto orientamento dell'EE nello spazio. Il punto a cui si fa riferimento per la calibrazione ed in generale quando si parla della precisione di un robot è il TCP in quanto questo rappresenta l'estremità distale del robot, gli utensili da applicare su questa porzione possono variare e non devono influenzare la capacità del robot di essere accurato. La combinazione di posizione di un punto del TCP ed il suo orientamento viene detta posa ed è definita nello spazio di lavoro.

Per descrivere una posa nello spazio di lavoro, si rende necessario conoscere precisamente la geometria del manipolatore e successivamente specificare l'attuazione dei giunti nello spazio giunti per ottenere questa particolare posa \mathbf{P} .

$$\mathbf{P} = f(\boldsymbol{\theta}, \boldsymbol{\eta}) \quad (2.1)$$

Dove $\boldsymbol{\eta}$ è un vettore di costanti che descrive la geometria del manipolatore e la posizione del manipolatore nello spazio di lavoro e $\boldsymbol{\theta}$ è il vettore che rappresenta tutte le attuazioni di tutti i giunti per ciascuna posa, nello spazio giunti.

L'Eq. (2.1) è una funzione, per ora incognita, che permette il passaggio dallo spazio giunto allo spazio di lavoro del robot. Per poter descrivere questa funzione è necessario scegliere un opportuno modello, tra quelli indicati in letteratura.

Capitolo 2

2 Calibrazione: stato dell'arte

Ci sono due strade possibili per cercare di aumentare l'accuratezza robot. La prima consiste nel cercare di costruire ciascun robot in modo tale da far coincidere il più possibile ciascun parametro del robot reale con i valori "nominali". In altre parole, cercando di minimizzare le deviazioni dei *link* reali rispetto alle dimensioni nominali e mediante l'uso di strutture più rigide. La seconda invece, che è la soluzione che è stata adottata storicamente è quella di costruire delle macchine ad alto livello di ripetibilità e ovviare la mancanza di accuratezza compensando gli errori attraverso dei software scritti ad hoc per ciascuna macchina; questa procedura di compensazione viene detta calibrazione [8]. Chiaramente, la prima strada è difficilmente percorribile, ricercando solo tolleranze dimensionali più restrittive e materiali più performanti si avrebbero costi e scarti più elevati, inoltre sarebbero necessari processi molto più complessi.

Per raggiungere una certa posa definita nello spazio operativo, il controller del robot deve convertire il task definito nello spazio operativo in una attuazione dei giunti nello spazio giunti. Ogni giunto viene mosso individualmente e la configurazione desiderata viene ottenuta. La conversione da uno spazio all'altro è possibile grazie alla conoscenza del modello matematico del manipolatore. Questo modello matematico lega le attuazioni ai giunti alle pose del TCP e viceversa. Quando il modello matematico memorizzato dal controller del robot differisce dalla geometria del robot, la definizione dell'attuazione dei giunti a partire da una posa nello spazio operativo non sarà accurata.

Se non è possibile far corrispondere il robot con il modello teorico, basta cercare di identificare un modello matematico che rappresenti il robot reale preso in considerazione.

Questa è l'essenza della calibrazione. Semplicemente, è il processo di selezione di un appropriato modello matematico e quindi l'identificazione dei vari parametri che permettano un abbinamento al suo modello rappresentativo il più fedelmente possibile.

Due manipolatori teoricamente uguali presentano delle unicità ed hanno relazioni diverse tra spazio giunti e spazio operativo l'uno dall'altro. I programmi scritti per un robot, tali da raggiungere delle determinate pose nello spazio operativo, non possono essere trasferiti da una macchina all'altra; la calibrazione porta a dei risultati unici per ogni macchina.

La calibrazione in generale può essere suddivisa in due gruppi principali: statica e dinamica. La calibrazione statica è un'identificazione di quei parametri che influenzano principalmente le caratteristiche di posizionamento statico del robot (posizione e orientamento dell'end-effector), mentre la calibrazione dinamica viene utilizzata per identificare parametri che influenzano principalmente le caratteristiche del movimento (velocità e coppie) [10].

2.1 Classificazione degli errori

Per comprendere a fondo il processo di calibrazione bisogna introdurre e classificare opportunamente gli errori di posizionamento del manipolatore e le sue cause.

2.1.1 Ripetibilità ed accuratezza

I manipolatori, per poter raggiungere una determinata posa nello spazio, devono essere programmati. Esistono due principali metodi di programmazione: il '*teaching method*' e la '*programmazione off-line*'.

Nel primo metodo un operatore muove fisicamente il robot lungo le pose che si desidera raggiungere, per ciascuna posa raggiunta vengono memorizzate le configurazioni dei giunti assunte dal manipolatore. Nel secondo, le posizioni assunte dal robot vengono simulate in remoto.

Il primo metodo comporta necessariamente un fermo della macchina durante la fase di programmazione, è lungo e richiede l'uso di operatori specializzati. Il secondo può invece essere simulato ed è più flessibile, ma è fortemente dipendente dai parametri cinematici η adottati per la definizione della funzione dell'Eq.(2. 1) che rappresenta il modello cinematico del robot; infatti, piccole deviazioni dalla geometria del modello η rispetto a quella del robot reale possono produrre una sostanziale differenza tra la posizione teorica calcolata per il TCP e quella effettivamente raggiunta.

Per misurare l'abilità di un robot a raggiungere una determinata posa ci si avvale del concetto di errore. Gli errori che si incontrano nei due metodi di manipolazione sono detti ripetibilità ed accuratezza e sono utili a caratterizzare gli errori di posizionamento dei manipolatori.

La misura più comune dell'abilità di un robot a raggiungere una determinata posa è la ripetibilità. Semplicemente, la ripetibilità è la misura dell'abilità del robot di ritornare in una posa precedentemente raggiunta.

Se al manipolatore viene comandato di tornare in una posa precedentemente raggiunta, della quale è stata memorizzata la configurazione dei giunti, il TCP non ritornerà esattamente alla stessa posa. Il motivo di questa piccola deviazione può avere diverse cause, può includere piccoli errori nel controllo dell'attuazione dei giunti, nella flessibilità della struttura del robot, eventuali giochi ai giunti e numerosi altri fattori. Per descrivere geometricamente la ripetibilità di un robot, basti immaginare nello spazio operativo una sfera; questa sarà centrata nella posizione in cui si desidera collocare il TCP ed avrà raggio pari alla ripetibilità del robot considerato in Figura 2.1. Quando si cercherà di ritornare nella posizione identificata dal centro della sfera, il robot in realtà andrà a riposizionarsi all'interno di un punto qualunque entro i confini di questa sfera, rappresentante la ripetibilità del robot. L'orientamento è più complesso da rappresentare geometricamente, da un punto di vista numerico è meglio considerare la ripetibilità dell'orientamento nelle componenti della posa, queste varieranno in funzione del tipo di convenzione adottato per la descrizione dell'orientamento (RPY, Eulero, ecc.).

L'accuratezza invece è la capacità di muovere il manipolatore fino a portare il TCP ad assumere una posa specifica nello spazio di lavoro. La principale differenza tra

accuratezza e ripetibilità è che la ripetibilità indica l'abilità del robot a ritornare in una posizione precedentemente raggiunta, mentre l'accuratezza è l'abilità del manipolatore di muovere sino ad una posa specifica dello spazio di lavoro. Un altro modo di fare una distinzione tra accuratezza e ripetibilità è considerare il significato per cui la posa risultante viene specificata. Quando si considera la ripetibilità, la posa è stata precedentemente ottenuta e la configurazione ai giunti è nota. Parlando di accuratezza, la posa è definita nello spazio di lavoro, e deve essere determinato un particolare set di commutazione ai giunti per raggiungere tale posa.

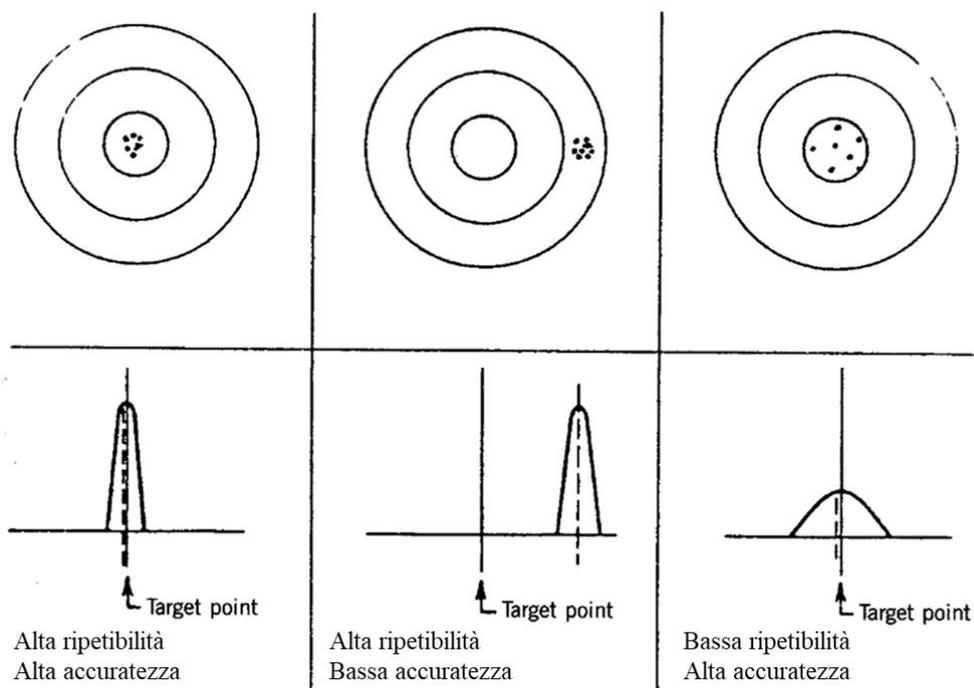


Figura 2.1: Legame tra ripetibilità ed accuratezza [8]

La Figura 2.1 rappresenta la distribuzione spaziale di un set di misurazioni lungo un piano. All'aumentare della distribuzione spaziale delle misurazioni e quindi della deviazione standard, ci sarà una diminuzione della ripetibilità dal robot. Mentre la mancata accuratezza sposta l'intera distribuzione spaziale in punto diverso rispetto al *target point*, il quale genera un errore medio all'intera distribuzione rispetto all'obiettivo del TCP. La circonferenza che racchiude l'errore di accuratezza racchiude al suo interno anche l'errore di ripetibilità. L'immagine è un buon esempio bi-dimensionale in grado di spiegare la relazione che intercorre tra ripetibilità ed accuratezza, ma il bersaglio del TCP in condizioni di utilizzo non è un punto su di un piano, bensì un punto nello spazio

tridimensionale. Inoltre, questo punto deve essere raggiunto con il TCP orientato correttamente, motivo per il quale non è sufficiente immaginare un bersaglio nello spazio, occorre considerare anche gli orientamenti e la convenzione adottata per la loro rappresentazione. A seguito dell'identificazione cinematica, nei capitoli successivi, verranno calcolati gli errori ottenuti mediante i vari metodi di identificazione, lo scopo di questo paragrafo permette di dare significato geometrico a questi errori e di poter meglio interpretare gli errori che verranno calcolati.

2.1.2 Classificazione delle fonti di errore

L'accuratezza e la ripetibilità dei robot è influenzata da numerosi fattori, questi possono essere distinti in cinque categorie [11]:

- **ambientali:** ad esempio cambiamenti di temperatura;
- **parametriche:** variazione dei parametri cinematici, influenza dei parametri dinamici, conformità della trasmissione, attriti ed altre non linearità, cicli d'isteresi ed urti;
- **misura:** risoluzione e non linearità dei sensori di posizione dei giunti;
- **computazionali:** errori di approssimazione del computer ed errori derivanti da condizioni transitorie dei servosistemi;
- **applicazione:** errori di installazione, e posizionamento ed errore geometrico delle componenti.

Le deviazioni geometriche, seppur piccole, comportano spesso errori di accuratezza non accettabili per molte applicazioni. L'analisi dell'influenza di ciascun contributo e la sua eliminazione è tuttora oggetto di un'intensa ricerca, guidata dalla necessità di incrementare accuratezza e ripetibilità.

Questa distinzione degli errori è molto utile ai fini dell'identificazione di tutte le cause che possono portare all'insorgenza di un errore, ma operativamente gli errori, nei lavori di calibrazione vengono distinti in due categorie [12]: errori geometrici e non geometrici. Gli errori geometrici sono definiti come errori nei parametri dimensionali, i quali definiscono le relazioni geometriche tra gli assi giunto del robot. In altre parole, non sono in funzione di eventuali carichi statici o dinamici applicati al manipolatore. Gli errori geometrici sorgono durante la costruzione del manipolatore e dipendono dalle tolleranze adottate ed per la costruzione e

l'assemblaggio. Gli effetti si tradurranno in una variazione delle lunghezze che descrivono il modello *link* e l'orientamento dei suoi assi giunti.

Le altre fonti di errori, quelle non geometriche, richiedono una conoscenza dei carichi ai quali è soggetto il robot e il suo tipo moto nello spazio e racchiudono inoltre tutte le altre fonti di errore classificate in precedenza.

Nel tempo ci sono state alcune controversie sullo stabilire quale tra i due gruppi di errore contribuisca in modo più significativo alla bassa accuratezza. Ad esempio agli estremi si hanno due ricerche [12] [13], nelle quali si hanno risultati contrastanti dove nella prima si mostra che sono più rilevanti gli errori non geometrici e nella seconda quelli geometrici.

A seguito di questi risultati agli estremi, si potrebbe affermare che la predominanza dei contributi di errore, geometrici o non geometrici, sull'accuratezza, dipenda fortemente dal design del robot che si sta analizzando [8]. Tuttavia, ci sono dei lavori successivi con risultati diversi rispetto alla ricerca che afferma che gli errori non geometrici abbiano carattere predominante [14]; con lo stesso manipolatore si sono ottenute delle cospicue riduzioni dell'errore di accuratezza a seguito della calibrazione solo degli errori geometrici.

Infatti, nelle revisioni più recenti si afferma che il 90% degli errori di accuratezza del TCP sono guidati dagli errori geometrici [15] e lavori più recenti [16][17][18] fanno riferimento a questi valori di riferimento. Data la tendenza a prediligere i contributi geometrici nelle calibrazioni più attuali, in questo lavoro di tesi si condividerà con l'affermazione tale per cui il 90% degli errori di accuratezza possono essere compensati dalla identificazione dei parametri cinematici.

2.2 Processo di calibrazione

Riprendendo la relazione tra lo spazio giunto e lo spazio operativo, l'obiettivo della calibrazione è definire una relazione funzionale dell'eq (2. 1) e determinare i coefficienti η , che permettano al modello teorico ed al robot reale di corrispondere il più possibile. Questo processo si riassume convenientemente in quattro

operazioni sequenziali: modellazione, Misurazione, Identificazione e Implementazione, ciascuna operazione è connessa all'Eq. (2. 1).

2.2.1 Modellazione

Il primo passo nel processo di calibrazione è scegliere il tipo di funzione con il quale esprimere la trasformazione (2. 1) che lega i parametri del robot (fisici e/o geometrici) e l'attuazione dei giunti alla posa del TCP. La modellazione si riferisce alla scelta di una relazione funzionale tra i parametri del robot e la posa del TCP. Bisogna scegliere la relazione funzionale che lega le variabili nello spazio giunti θ alla posa del TCP, attraverso il vettore η che racchiude le caratteristiche del robot che si vuole calibrare.

Esaminando la letteratura sulla modellazione cinematica sono presenti una vasta varietà di modelli usati dai vari ricercatori sul campo.

Il modello selezionato deve tenere conto di tutti i fattori ritenuti significativi nel contribuire all'accuratezza del robot. Il livello di dettaglio che si vuole raggiungere nel modellare il robot si decide durante la scelta delle variabili costituenti il vettore η .

In questa fase si sceglie infatti quali tipi di errori si desidera considerare nel modello del robot e quali si sceglie di mantenere trascurabili.

Come già detto, la distinzione più comune che viene effettuata tra gli errori consiste nel dividere gli errori geometrici da quelli non geometrici. Molti lavori di calibrazione, dato che considerando solo gli errori geometrici è possibile ottenere un'incremento dell'accuratezza fino al 90% , escludono gli errori non geometrici. La scelta di considerare o meno gli errori non geometrici deve comunque essere fatta in funzione del grado di accuratezza richiesta dalle operazioni che i robot andranno ad effettuare.

L'industria moderna, con l'avvento delle nuove tecnologie, guida l'esigenza di una maggiore accuratezza dei robot all'interno dei processi produttivi. Diventa sempre più necessario ottimizzare l'accuratezza dei robot, molti *paper* più recenti effettuano delle calibrazioni considerando i fattori non geometrici, cercando di ottimizzare quel 10% di accuratezza residuo.

Durante la fase di modellazione bisogna scegliere la complessità del modello e scegliere se optare per un modello contenente i soli parametri cinematici (geometrici) o se si vuole prendere in considerazione anche i parametri non geometrici.

Si classifica la calibrazione effettuata in tre distinti livelli, a seconda del modello che si decide di adottare per rappresentare il robot reale [11]:

1) **Calibrazione giunti o di primo livello:**

a questo livello l'obiettivo è quello di stabilire la corretta relazione tra i segnali dei sensori ai giunti e l'effettivo spostamento dei giunti.

2) **Calibrazione cinematica o di secondo livello:**

l'obiettivo di questa calibrazione è sviluppare un modello cinematico, ossia un modello con l'ipotesi che il robot sia composto idealmente da *link* perfettamente rigidi e da giunti non elastici.

3) **Calibrazione non geometrica (non cinematica) o di terzo livello:**

a questo livello, le deviazioni dal modello ideale cinematico subiscono anche gli effetti dovuti a fattori non geometrici.

Il secondo ed il terzo livello sono i livelli di calibrazione da adottare rispettivamente se si desidera compensare gli errori geometrici o entrambe i tipi di errore. Nella Tabella 2.1 una raccolta dei risultati di alcuni lavori di calibrazione analizzati.

Robot	Metodo Identificaz.	Punti misurati		Media errore posizionamento [mm]		
		Totale	Validaz.	Non Calibrato	Calibrazione Cinematica	Calibrazione Non Geometrica
6 GDL robot, lungo 2.5m [19]	Algoritmo HFTI (modifica del metodo di Newton)	200	100	5,65	0,91	0,4
Scara robot [19]		200	100	3,03	0,27	0,27
Hyundai HH800 [7]	Filtri di Kalman;	200	100	4,07	0,81	0,34
PUMA 560 [7] (*)	Reti Neurali (ANN) (compensare errori residui)	200	100	8,35	0,4	0,03
Hyundai robot HA006 [20]	Algoritmo genetico ibrido (GA+LSM)	45	45	3,99	0,16	0,09
PUMA 560 [21]	Minimi Quadrati (LSM)	-	-	4,8	-	0,3

Hyundai HA006 [22]	Minimi Quadrati (LSM)	56	56	3,66	0,13	-
Automatrix AID-900 [13]	Minimi Quadrati (LSM)	-	-	16,61	0,83	0,48
ABB IRB 1600 [23]	Minimi Quadrati (LSM)	52	1000	0,97	-	0,36
Fanuc LR Mate 200iC [24]	Minimi Quadrati (LSM)	124	10000	1,36	-	0,8
Yaskawa Motoman Robotics HP20D [25]	Levenberg- Marquardt (modifica Newton)			7,7	0,46	-
Manipolator e a tre gradi di mobilità [26]	Minimi Quadrati (LSM); Algoritmo genetico (GA);	-	-	13,28	1,57	-
KUKA KR- 270 [18]	Minimi Quadrati (LSM)	-	-	1,31	0,39	-
Sterling series FARO arm 6 DOF [27]	Levenberg- Marquardt (modifica Newton)	-	-	22,43	1,01	-
FANUC LR Mate 200iC [28]	Minimi Quadrati (LSM)	264	200	0,7	0,27	0,15
Staubli TX60L robot [29]	Minimi Quadrati (LSM)	14	30	1,41	0,52	0,15

Tabella 2.1 Risultati lavori di calibrazione

Come si può osservare dalla tabella, è possibile ottenere un'accuratezza che varia tra lo 0,5 e 1 millimetro attraverso la sola calibrazione cinematica [11], per cercare di ridurre ulteriormente il range di errore d'accuratezza è necessaria una calibrazione non geometrica; alcuni autori raccomandano che le fonti di errore non geometrico debbano essere incluse al modello solo se il desiderato livello di accuratezza non fosse possibile da raggiungere in altro modo [30].

Se si vuole effettuare una calibrazione cinematica bisogna selezionare uno tra i modelli messi a disposizione dalla letteratura.

I più utilizzati sono i modelli si basano su delle matrici di trasformazione omogenea. Il più utilizzato è il modello di Denavit Hartenberg (DH), dal quale poi molti ricercatori hanno proposto delle modifiche per risolvere eventuali problemi di singolarità in fase di identificazione.

La letteratura è molto ampia, sono presenti infatti altre alternative per il modello cinematico, quali il modello-s (*S-model*) ed il modello riferimento zero (*zero-reference model*) [14][15]. Questi modelli non sono stati approfonditi, in quanto il modello di DH e le sue variazioni sono molto più utilizzati nei lavori di calibrazione analizzati.

Per la modellazione cinematica del UR5 che verrà preso in analisi si tratteranno delle trasformazioni omogenee e piccole modifiche del modello di DH.

I modelli a matrice di trasformazione omogenea verranno spiegati e approfonditi nel capitolo 3.

Per una calibrazione di terzo livello bisogna aggiungere i contributi dati dai fattori non geometrici, bisogna tenere in considerazione che ci sarà una complicazione del modello e l'eventuale aggiunta di variabili al vettore η .

Gli errori non geometrici statici i più rilevanti sono:

2.2.1.1 Non linearità della trasmissione:

Gli effetti non lineari nella trasmissione dell'azionamento dei giunti è data dal fatto che per certi robot gli encoder non sono posizionati direttamente sul giunto, bensì sull'attuatore del giunto, i quali permettono il moto del giunto attraverso un sistema di trasmissione. Di conseguenza, l'influenza del gioco e degli effetti non lineari della trasmissione non vengono misurati dai sensori di posizione del giunto, dato che sono montati sull'attuatore, causando così un errore [31]. Questo tipo di non errore geometrico non appartiene a tutti i robot; il robot UR5 analizzato in questa tesi possiede un encoder sul motore e sul giunto, annullando gli effetti di non linearità di trasmissione.

2.2.1.2 Deformazione elastica di giunti e *link*:

I pesi dei *link* ed eventuali carichi aggiuntivi sul TCP provocano deformazioni elastiche dei *link* e sui cuscinetti dei giunti. Secondo quanto dimostrato dai alcuni risultati sperimentali [19], le deformazioni elastiche dei giunti hanno maggiore rilevanza rispetto alla deformazione elastica a quella dei *link* [19] [20], [16].

2.2.1.3 Gradiente Termico:

L'espansione termica può inficiare l'accuratezza dei manipolatori, spesso la calibrazione viene eseguita in ambienti a temperatura controllata per poter evitare di considerare questa fonte di errore. Questo errore non geometrico è meno rilevante

dei precedenti e non è solo funzione del campo di temperature, bensì dipende anche dal posizionamento del TCP [32].

2.2.1.4 Errori residui:

Nello sviluppo di un modello preciso è complesso modellare tutti i fenomeni rilevanti e determinare tutti i parametri di tale modello.

Al fine di includere eventuali ulteriori errori non geometrici non determinati nel modello del robot cinematico, è stato proposto un metodo di modellazione detto feedback residuo [33][11]. Il modello si basa sul presupposto che gli errori non modellati sono piccoli e che gli errori residui possono essere modellati come funzioni cicliche per ogni angolo giunto. Si può cercare di ottenere dunque, per ogni giunto, una funzione che descrive nel suo raggio d'azione, l'errore non modellato residuo.

In un lavoro si è scelto di trattare tutti gli errori non geometrici come errori residui, ricercando la soluzione attraverso delle reti neurali [7]. In questo modo si è bypassato il problema di creare un modello per gli errori non-geometrici, sfruttando la potenza di calcolo dei moderni calcolatori.

2.2.2 Misurazione

Il secondo step nel processo di calibrazione è la fase di misurazione.

La fase di misurazione consiste nel rilevamento, nello spazio di lavoro, della posizione del terminale del robot, determinando in modo accurato la posa del TCP, per un set di attuazioni ai giunti. Muovendo il robot in diverse pose, j , nello spazio di lavoro le pose reali misurate sono poi confrontate con le pose previste dal modello teorico, per ottenere i dati sull'incertezza nello spazio di lavoro. Vengono cioè registrate per ogni punto j le attuazioni ai giunti θ_j tali da portare il TCP nella posa prevista P_{pj} e mediante un sistema di misura, si determina la posa P_{mj} effettivamente raggiunta dal manipolatore. Questa fase è quella più difficile e che richiede maggior tempo nella calibrazione di un robot. La procedura di calibrazione deve mostrare i diversi parametri e il sistema di misurazione deve essere abbastanza accurato da misurare gli effetti di questi parametri.

Svariati metodi di misura e strumenti di misura sono stati utilizzati per la calibrazione dei robot. I metodi di misura e gli strumenti variano considerevolmente per precisione, velocità e facilità d'uso, quantità di informazioni ottenuta per

ciascuna posa del robot e costo, ma tutti condividono gli stessi inconvenienti: il processo di misurazione richiede molto tempo, è laborioso ed è incline ad errore umano.

Per questi motivi ci sono diversi vantaggi nel minimizzare il numero di misure da rilevare, senza compromettere il risultato finale del processo di calibrazione [15].

Per calibrare un robot, non è necessario eseguire una misura completa della posizione del TCP, sono sufficienti le misure di diverse pose incomplete [28][21].

Qualunque sia l'approccio scelto è sempre necessario un esteso intervento umano durante l'intero processo di calibrazione.

Ci sono due aspetti del processo di misurazione che è necessario considerare con particolare attenzione. Il primo è quale sistema di misura dovrebbe venire utilizzato, il secondo è come pianificare la strategia di osservazione correttamente.

La scelta del sistema di misura influenza in modo diretto la strategia di identificazione. Un modo per classificare la strategia di identificazione è scegliere tra la misura di una posa completa (6GDL) o una posa parziale. Dato che il TCP possiede 6 gradi di libertà (GDL) nello spazio, bisogna considerare fino sei informazioni utili per misurare completamente la posa. Generalmente consistono in tre coordinate di posizione e tre angoli per l'orientamento. La misura parziale della posa (che avrà meno di 6 GDL) viene spesso usata per calibrare i parametri del robot, in applicazioni più economiche e/o più rapide.

Oltre a numero di informazioni della posa, la calibrazione si può classificare in anello chiuso ed anello aperto. I metodi ad anello aperto sfruttano dei sistemi di misura esterni che vanno a rilevare la posa del TCP direttamente. L'approccio di calibrazione ad anello chiuso è previsto in letteratura per evitare l'uso di dispositivi di misura esterni [28][34]. Gli unici dati utilizzati provengono dagli encoder delle giunzioni del robot. La calibrazione ad anello chiuso si basa sul limitare il movimento del TCP. Questa tecnica prevede di forzare il terminale del robot in pose predefinite e note con precisione. In questo caso non si deve misurare la posizione del TCP, perché esso viene vincolato e la calibrazione è ottenuta solo dalle letture dei giunti tramite le equazioni di chiusura della catena cinematica. Ad esempio, il TCP può essere vincolato a raggiungere posizioni specifiche [21] [35] o a puntare verso lo stesso punto mentre le configurazioni dei giunti del robot cambiano [27],

oppure a giacere su una superficie specifica, come dei piani [28]. Le tecniche a catena chiusa possono essere viste come procedure di auto calibrazione, in cui non si utilizzano strumenti esterni per la misurazione della posa del terminale. L'auto-calibrazione permette inoltre di ottenere misure accurate nello spazio di lavoro in breve tempo, può essere automatizzata e permette la correzione dell'accuratezza direttamente in linea.

Tra gli strumenti di misura ad anello aperto più utilizzati si hanno:

2.2.2.1 Teodolite

Il teodolite consiste in uno strumento ottico simile ad un piccolo cannocchiale.

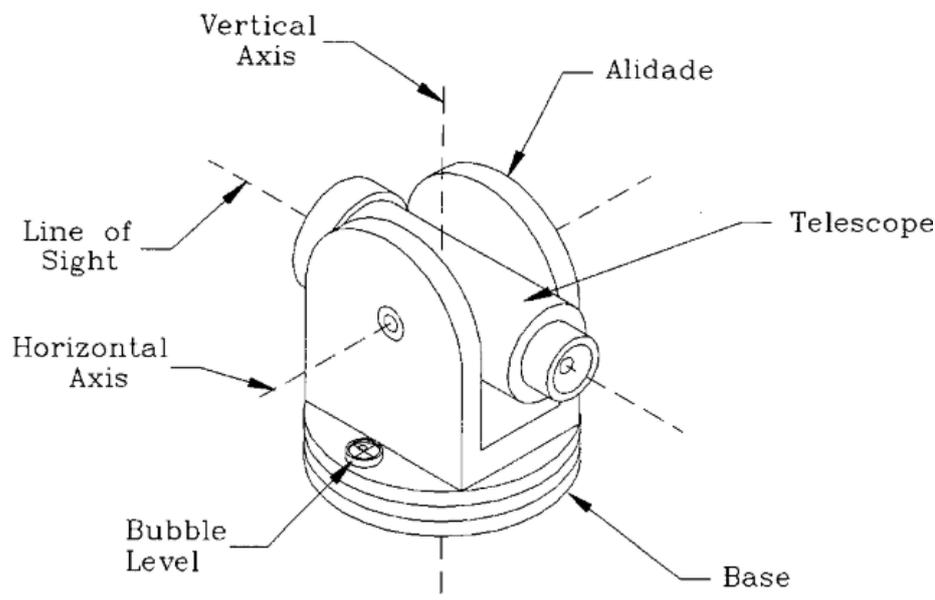


Figura 2.2: Rappresentazione di un teodolite [8]

La sua funzione è quella di allineare l'oggetto della misurazione con i due assi principali dello strumento, la cui intersezione genera *line of sight* o linea di visione. Questi strumenti possono misurare gli angoli di rotazione rispetto ad i due assi di rotazione verticale ed orizzontale. Operativamente durante l'utilizzo bisogna puntare attraverso il telescopio fino a quando non viene visualizzato il bersaglio. Il telescopio focalizzato sul bersaglio viene quindi allineato agli assi fino a centrare il punto target sul mirino del telescopio.

Si leggono quindi gli angoli verticale ed orizzontale attraverso degli encoder. La sensibilità dei moderni teodoliti raggiunge i $1.57 \mu\text{rad}$ ($0,1 \text{ mgon}$)[36], questo strumento non è in grado di misurare le distanze. Sebbene il costo di un teodolite

manuale sia relativamente basso, la localizzazione dei bersagli finali è un processo faticoso e lungo. Sono stati sviluppati sistemi con teodoliti automatizzati e sistemi di tracciamento laser che eliminano gran parte della fatica nel processo di raccolta dei dati. Questi sistemi, tuttavia, sono molto più costosi.

Diversi gruppi hanno sviluppato strategie di calibrazione basate su teodoliti. Whitney, Lozinski e Rourke [12] hanno utilizzato un singolo teodolite per acquisire i dati di calibrazione di un manipolatore. Per ogni posa, il teodolite misura un angolo verticale e un angolo orizzontale che definiscono una retta nello spazio. Il bersaglio si trova da qualche parte lungo la retta. Nota la retta dalla quale passa il bersaglio, la distanza di questo può essere determinata solo come rapporto con una costante arbitraria. La mancanza di una scala di lunghezza di base impone di effettuare almeno una misurazione aggiuntiva. Almeno due punti del manipolatore a distanza nota devono essere osservati con il teodolite e gli angoli devono essere registrati. In questo modo è possibile impostare la scala di lunghezza e determinare i valori assoluti dei parametri di lunghezza.

Judd e Knasinski [13] hanno sviluppato una procedura basata sulla triangolazione con le osservazioni di due teodoliti [19][13]. La triangolazione consiste nella determinazione di più rette che convergono nello stesso bersaglio e partono da punti di osservazione diversi. Calcolando l'intersezione di queste rette è possibile determinare la posizione del punto osservato. Per determinare la posizione di un punto è necessario misurare solo tre valori indipendenti. Poiché due teodoliti forniscono la misura di quattro angoli, è ragionevole aspettarsi una serie di equazioni sovradeterminate. Poiché qualsiasi sistema di misura reale contiene un certo errore, si usano tutti e quattro i valori cercando la posizione del punto osservato attraverso il calcolo dei minimi quadrati.

2.2.2.2 Interferometria laser

L'interferometro laser utilizza i principi di interferenza della luce per misurare con precisione lo spostamento lineare o la velocità di un corpo. Il funzionamento di un interferometro semplice è illustrato in Figura 2.3. Un laser produce un raggio di luce coerente monocromatico che viene diviso in due fasci di raggio. Parte del raggio viene riflessa verso uno specchio fisso e l'altra parte del raggio viene trasmessa verso uno specchio mobile. La luce degli specchi fissi e mobili viene

riflessa indietro verso il divisore di raggio, venendo ricombinata e trasmessi su di un fotorelevatore. I due fasci ricombinati, avendo la stessa frequenza, ma avendo seguito cammini ottici differenti, si ricombineranno sfasate portando ad una interferenza. L'onda risultante dalla combinazione delle onde separate può essere accuratamente misurata in termini di interferenza costruttiva o distruttiva, dai quali è possibile misurare distanze e spostamenti dello stesso ordine di grandezza della lunghezza d'onda utilizzata (visibile: 400-700 nm).

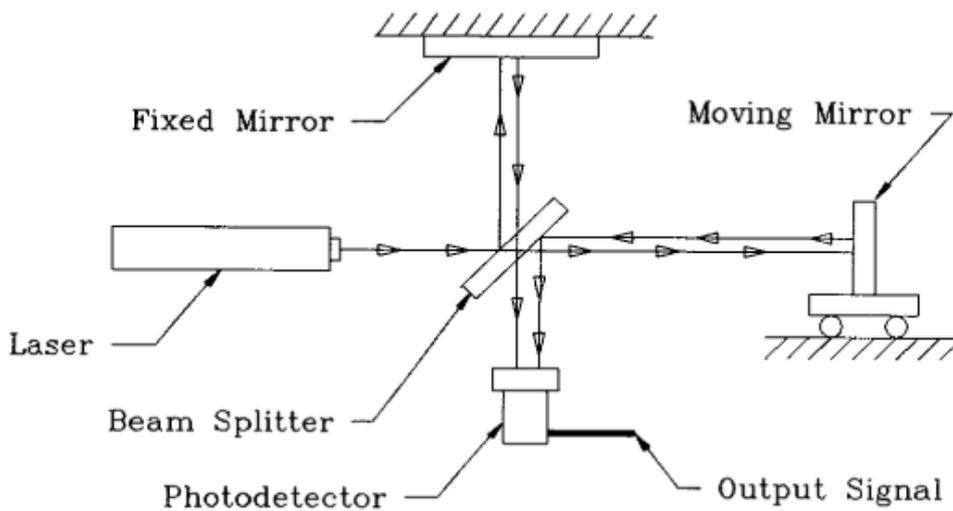


Figura 2.3: Schema del funzionamento di un interferometro semplice [8]

La precisione dell'interferometro viene influenzata dai fattori che alterano la lunghezza d'onda della luce. Quando la luce viaggia attraverso l'aria, la lunghezza d'onda è influenzata dalla pressione, dall'umidità e dalla temperatura. Poiché questi effetti sono spesso piccoli, in molti casi possono essere ignorati. Tuttavia, sono disponibili sistemi commerciali che compensano questi effetti.

I dispositivi di misurazione laser sono molto utilizzati in quanto permettono una elevata precisione di misurazione, seppur presentano alcuni svantaggi quali l'alto costo delle apparecchiature e la necessità di non avere ostacoli che interferiscano con le sorgenti laser. Per poter utilizzare questa tecnologia è necessario un dispositivo di misurazione laser in grado di emettere e ricevere la sorgente laser e di uno o più riflettori laser.

Le macchine di misura usate in commercio sono spesso in grado di fornire la coordinata spaziale del punto del riflettore laser dal quale viene riflesso il raggio

luminoso. A seconda della tecnica di misurazione è possibile ottenere la posizione oppure la posa completa del TCP.

Quando si utilizza un interferometro laser con un raggio orientabile, il fascio luminoso lascia il laser e viaggia fino a colpire il riflettore montato sul TCP. L'orientamento del riflettore (specchio mobile) viene controllato ruotandolo attorno a due assi orizzontale e verticale. Poiché gli angoli del riflettore e la distanza dal bersaglio sono noti con precisione, la posizione del bersaglio può essere determinata con accuratezza. È possibile, dunque, attraverso un solo riflettore ed un emettitore laser, esprimere la posizione del TCP.

In aggiunta, mediante l'uso di più riflettori, si può esprimere l'intera posa nello spazio. Non esiste un dispositivo in grado di misurare direttamente la posa completa di un oggetto nello spazio, la posa completa del TCP può essere determinata individuando almeno tre punti su di esso di cui si conosce la posizione relativa.

Nei lavori di calibrazione si montano sui TCP degli EE sui quali sono calettati più riflettori laser al duplice fine di ottenere la posa completa e di rendere possibile al laser di raggiungere i riflettori per delle porzioni di spazio di lavoro più ampie.

In figura un esempio di EE e i laser tracker montati su di un robot [23].

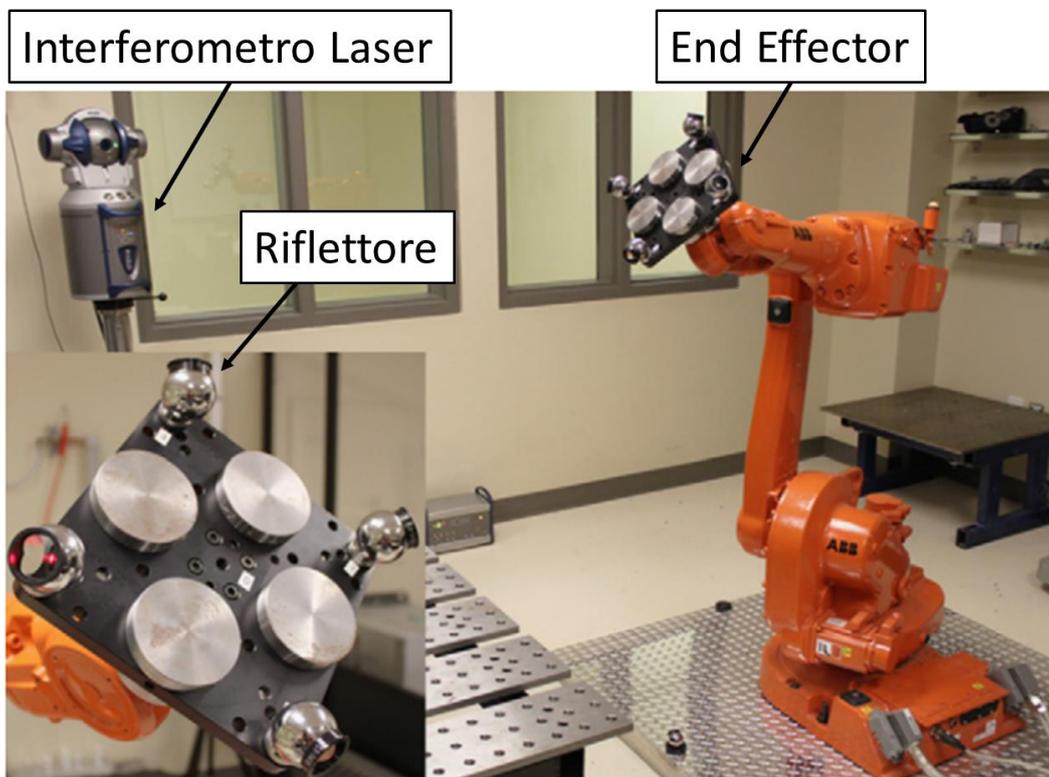


Figura 2.4: Apparecchiature calibrazione ad interferometria laser [23].

Macchina di misura a coordinate (CMM)

Storicamente una macchina di misura a coordinate (CMM) è un meccanismo a 3 DOF con tre assi prismatici ortogonali. Questi dispositivi sono realizzati per garantire un movimento preciso lungo gli assi desiderati e sono realizzate per determinare lo spostamento dei giunti con un elevato grado di precisione.

Le CMM sono disponibili in una vasta gamma di dimensioni e con vari livelli di precisione.

Nella "fascia bassa" di dimensioni e prestazioni ogni giunto è equipaggiato con un encoder lineare o un encoder rotativo azionato da una cremagliera, con una un'accuratezza intorno a 0,1 mm.

Ai livelli più alti di dimensioni e prestazioni, si possono avere CMM con assi altamente precisi che si muovono su cuscinetti ad aria, dotati di interferometri laser per la determinazione del movimento degli assi. Queste macchine possono raggiungere una precisione di 3 μm .

In genere, le macchine di misura a coordinate sono progettate per l'ispezione di pezzi e assiemi.

La strategia di misurazione più semplice per la quale può essere utilizzata una CMM è la misurazione dei punti per determinare la posizione di un target sul robot. La CMM è dotata di un tastatore sensibile che registra le coordinate dell'ultimo quando tocca un oggetto. Sul TCP viene montata una sfera e il tastatore viene manipolato in modo da toccare diversi punti sulla superficie della stessa. Il centro della sfera può quindi essere determinato dalle coordinate dei punti sulla superficie.

L'uso di una CMM per la misurazione dei punti è interessante perché è facile da usare, richiede solo un bersaglio e produce dati facilmente interpretabili. Purtroppo, la maggior parte delle CMM sufficientemente grandi per calibrare i tipici robot industriali sono troppo costose per giustificare la sola calibrazione dei robot.

2.2.2.3 Dispositivi a tempo di volo

Un metodo molto diffuso per misurare la distanza tra due corpi consiste nel far emettere a un corpo un segnale che viaggia a una velocità nota. Un sensore sul secondo corpo riceve il segnale e registra il tempo di percorrenza tra emettitore e ricevitore. Il tempo di volo, insieme alla velocità nota del segnale, indica la distanza percorsa. Attualmente, il segnale più diffuso per questo tipo di misurazione è

l'impulso sonico. A prima vista, sembrerebbe auspicabile utilizzare un impulso di luce per la misurazione del tempo di volo, poiché la velocità è ben nota e costante. Sfortunatamente, la luce viaggia troppo velocemente per rendere pratico il tempo di volo sulle distanze relativamente ridotte coinvolte nella calibrazione dei robot. Un approccio molto utilizzato per misurare il tempo di volo consiste nell'emettere un impulso sonoro nell'aria e misurare il tempo necessario all'impulso per attraversare la distanza tra l'emettitore e il ricevitore. Il problema principale di questo approccio è la variazione della velocità del suono nell'aria. La velocità di un impulso sonoro è una funzione della temperatura, dell'umidità e delle correnti di convezione nell'aria. Poiché tutte queste condizioni sono dinamiche, qualsiasi sistema di misura basato sul tempo di volo di un impulso sonoro nell'aria deve compensare continuamente questi effetti. Un'altra difficoltà che può sorgere con questo approccio è l'effetto che gli ostacoli nello spazio di misura possono avere sull'impulso sonoro. Un ostacolo tra l'emettitore e il rilevatore può influenzare in modo significativo il tempo di volo percepito.

Con questo tipo di tecnologia si utilizza una tecnica simile alla triangolazione, si basa sulla misurazione della distanza da un target da diversi punti situati nell'area di lavoro.

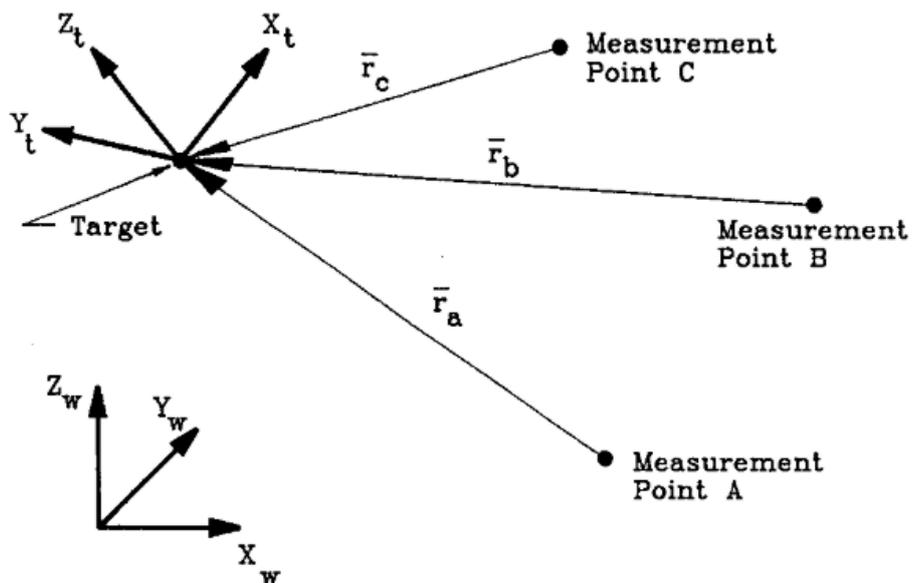


Figura 2.5: Misura distanza da diverse sorgenti [8]

Bisogna stabilire almeno tre punti di misura di cui si conosce con precisione la posizione nel sistema di coordinate globali e si misura la distanza di ciascuno da un bersaglio sul TCP del robot. Come nel caso del metodo della triangolazione, piccoli errori nella misura possono avere un effetto significativo sulla posizione stimata del punto osservato. Per minimizzare questo effetto, le misure possono essere effettuate da più di tre punti nell'area di lavoro. In questo caso, si ottiene un sistema sovradeterminato di equazioni non lineari che saranno risolte minimizzando il quadrato dell'errore totale. Queste misurazioni extra tendono a compensare l'effetto del rumore di misura sulla soluzione finale.

Stone, Sanderson e Neumann hanno misurato il tempo di volo di un segnale acustico verso tre microfoni. I tre tempi di volo sono stati utilizzati per determinare la distanza tra il bersaglio e ciascuno dei tre microfoni. Sebbene la risoluzione del sistema (0.2 mm) sia inferiore rispetto a quella ottenibile mediante altri sistemi, va notato che il sistema sonico è molto meno costoso.

2.2.2.4 Visione stereoscopica

La punta del robot viene equipaggiata con un reticolo di marcatori ottici o di sorgenti luminose, come dei led, che viene inquadrato da 2 telecamere fisse. Dalle immagini si possono estrarre le informazioni di posizione e assetto del reticolo ottico e quindi del TCP del robot. Gli svantaggi principali sono che il reticolo non deve essere oscurato da eventuali ostacoli e che l'illuminazione dell'ambiente durante la misura non deve saturare le videocamere. La precisione di questa tecnica dipende dalla risoluzione delle telecamere e dalla distanza tra queste ed il reticolo del robot. La tecnica di misurazione consiste nella triangolazione della posizione dei marcatori ottici, similmente a come avviene con la triangolazione fatta attraverso due teodoliti. Ogni telecamera restituisce letture equivalenti agli angoli orizzontali e verticali rispetto al target, dai quali è possibile calcolare la posizione nello spazio dei marcatori.

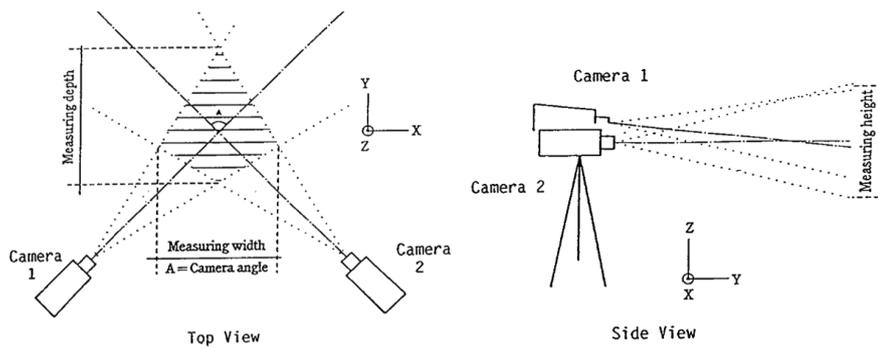


Figura 2.6: Esempio misurazione con visione stereoscopica [8].

2.2.2.5 Sistemi meccanici

Dei fili metallici vengono fissati al TCP che scorrono all'interno di rocchetti fissati al basamento. Un encoder misura la quantità di filo svolto e tramite tecniche di triangolazione è possibile risalire alla sua posizione. Anche in questo caso il problema maggiore è dato dalla presenza di eventuali ostacoli nello spazio di lavoro.

In un lavoro di Messay [25] attraverso delle misure parziali della posa del TCP (3D e 1D) è stato calibrato con buoni risultati un manipolatore industriale.

La calibrazione è stata eseguita mediante due attrezzature distinte rappresentate in Figura 2.7 :

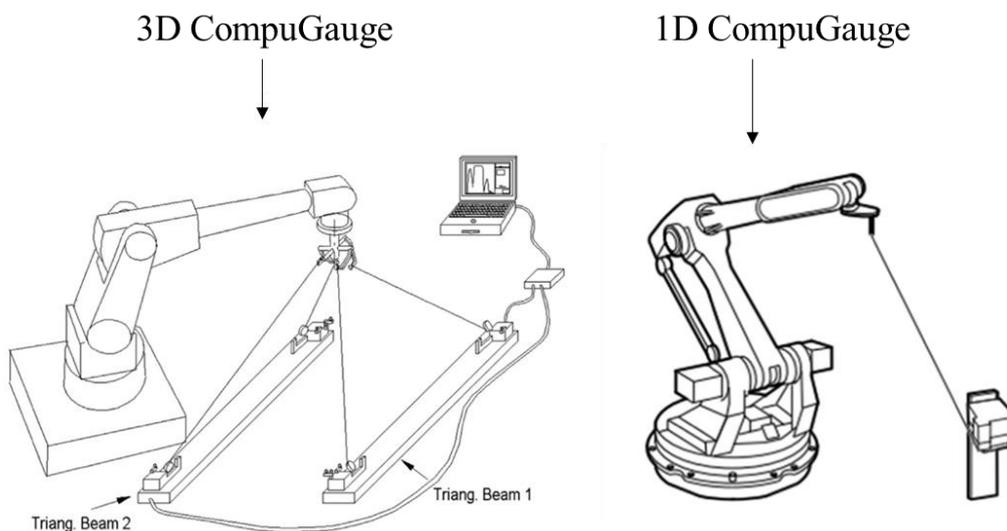


Figura 2.7 Sistemi di misura CompuGauge [25]

- **3D Coumpugauge:** sistema tridimensionale che triangola la posa del TCP mediante quattro cavi;

L'utensile non viene inserito fino in fondo, ma si misura con l'uso di uno spessimetro quanto è penetrato all'interno dei fori. Passando l'utensile per i vari fori e misurandone la profondità di penetrazione, si riesce a forzare l'EE a delle posizioni note rispetto alla base del robot. Da notare l'orientamento dell'asse dell'utensile è condizionato dal gioco disponibile all'interno dell'accoppiamento. In questo paper il gioco era molto ridotto e si è lasciato libero l'orientamento dell'utensile nel foro. Altri autori hanno adottato dimensioni meno restrittive per l'accoppiamento ed hanno controllato la posizione dell'EE rispetto al foro ed il suo orientamento attraverso l'uso di due telecamere [35].

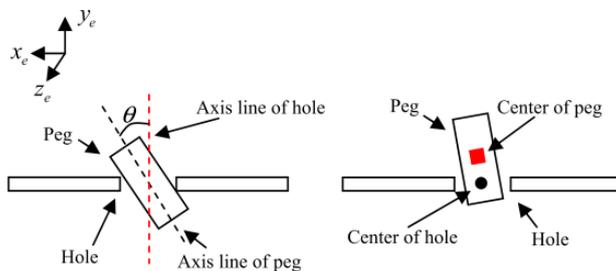


Figura 2.9: Controllo dell'orientamento dell'EE all'interno del foro [35]

L'uso della piastra forata è una delle possibili tecniche di autocalibrazione indicata dalla Universal Robot per l'autocalibrazione dell'UR5 [9]; questo metodo non è quello più consigliato dal costruttore infatti esso presenta diversi svantaggi.

Come dimostrato dai risultati di Veitschegger e Wu, mantenendo il robot con il gomito sempre nella stessa configurazione, inserendolo nei vari fori, si potrebbe ottenere la sovrapposizione del contributo gravitazionale a quello dei parametri geometrici.

Inoltre non è possibile ottenere informazioni complete sul *link* finale, in quanto, seguendo la procedura illustrata dalla Universal Robot, la rotazione lungo θ_6 non cambia il posizionamento del TCP.

2.2.2.7 Ball bar gauge

Un'altro modo per identificare una posizione cartesiana precisamente nello spazio è attraverso un dispositivo chiamato *ball bar gauge* [27]. Questo dispositivo consiste in una barra sulla quale sono calettate delle sfere, essa è composta da un telaio in fibra di carbonio e da 15 sfere in ceramica di 22 mm di raggio, con una incertezza sul raggio di 1,5 μm . Questa barra può essere disposta nello spazio in diverse configurazioni e per ogni configurazione è nota la posizione di tutte le sfere

nello spazio di lavoro. Utilizzando un'opportuno EE composto da tre sfere disposte a 120° , è possibile ottenere la posizione del centro della sfera montata sulla barra mettendo in contatto tutte e tre le sfere montate sull'EE. E' possibile raggiungere le stesse sfere attraverso diverse configurazioni ai giunti.

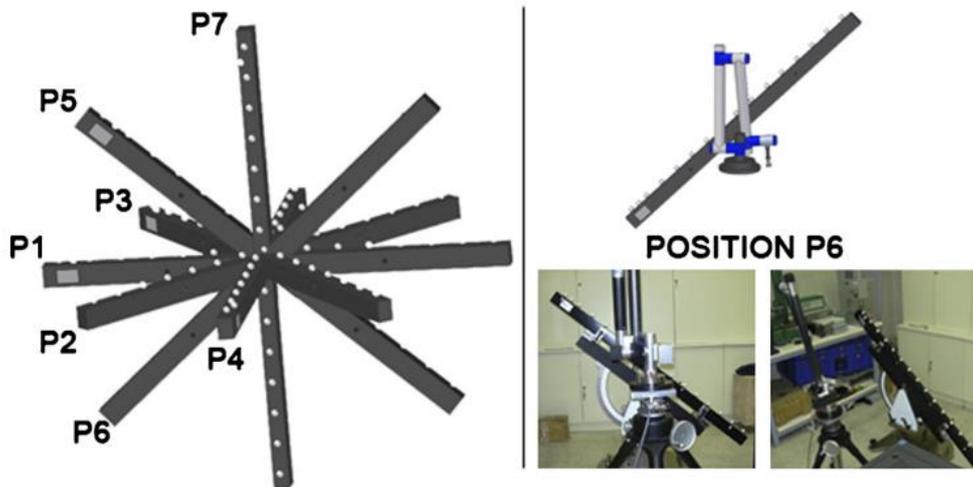


Figura 2.10: *Ball Bar Gauge* e configurazioni nello spazio [27]

2.2.2.8 Vincoli planari

Attraverso l'uso di almeno 3 superfici ortogonali nello spazio è possibile ottenere delle informazioni sulla posa sufficienti ad effettuare una calibrazione cinematica e non geometrica [28].

Per ottenere delle valide superfici di riferimento Joubair e Bonev si avvalgono di un cubo di granito per poter calibrare un manipolatore industriale.

Le sue facce opposte sono perfettamente parallele, le facce adiacenti perfettamente ortogonali ed ogni faccia rispetta una planarità entro gli 0,002 mm.

Sull'EE è montata una sonda a contatto, quando il sensore è sollecitato da una piccola forza, questa restituisce il feedback al manipolatore e vengono registrate le attuazioni ai giunti tali per cui il sensore ha incontrato il cubo di granito.

Si riescono dunque ad identificare delle informazioni parziali della posa appena ottenuta, in quanto la collocazione del piano di riferimento è nota rispetto al sistema di riferimento globale.

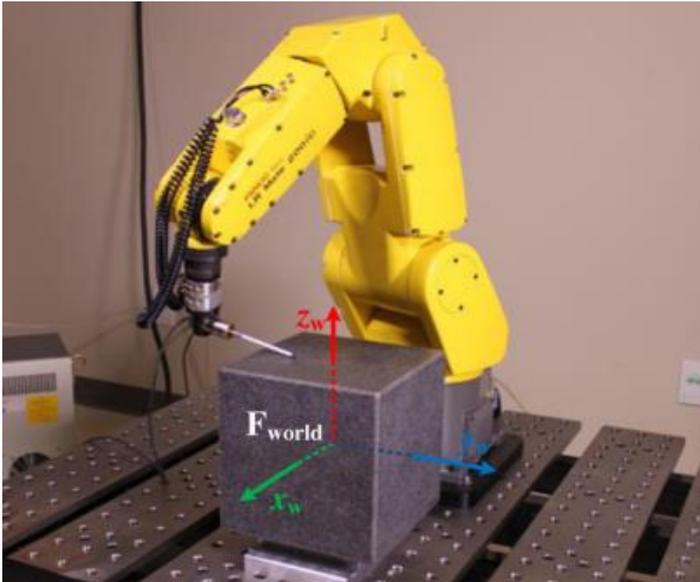


Figura 2.11: Robot e cubo di granito [28]

2.2.2.9 Calibrazione con due robot

La letteratura al riguardo è varia, in questo lavoro si è preso solo in considerazione il lavoro brevettato dalla Universal Robots e consigliato come metodo di calibrazione del robot UR5 [34].

Il metodo consiste nella misurazione ad anello chiuso attraverso l'utilizzo di due robot, i quali condividono la stessa base ed entrambi sono connessi ad una flangia aventi dimensioni note e precise.

Il modello a catena chiusa è costituito dai *link* dei due robot e dalla base.

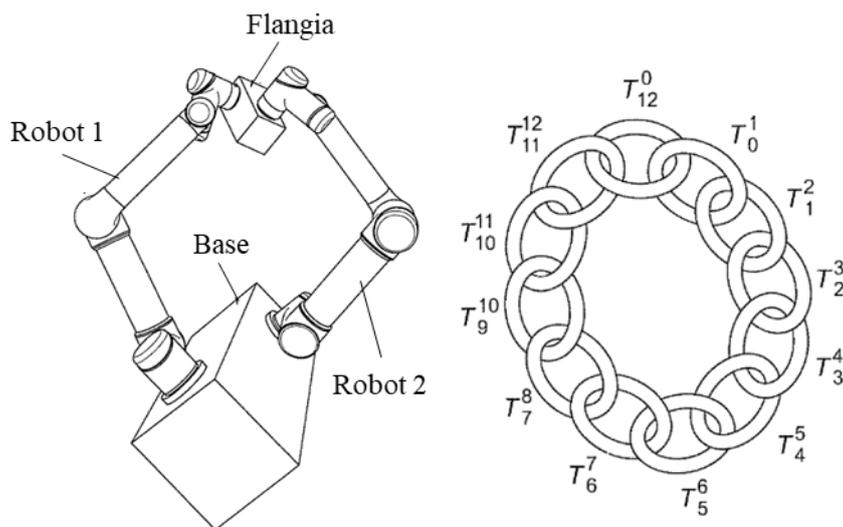


Figura 2.12: Calibrazione di due robot e catena cinematica chiusa [34]

Un robot vincola il posizionamento dell'altro, conoscendo l'attuazione ai giunti dei due robot e le dimensioni precise della base e della flangia è possibile identificare i parametri cinematici di entrambi i robot.

Questo è il metodo di identificazione che è stato utilizzato dalla Universal Robots per calibrare il robot UR5. Alcuni parametri cinematici del *link* finale non vengono identificati, in quanto questo sarà solidale alla flangia; come si vedrà nella Tabella 3.2 del capitolo successivo, due modifiche dei parametri del sesto *link* saranno pari a zero.

2.2.3 Identificazione

Una volta sviluppato un modello di manipolatore valido e raccolto un insieme di dati di misura, il passo successivo è quello di determinare i parametri del modello. Quell'insieme di parametri i quali permettano alle pose calcolate dal modello di corrispondere il più possibile alle pose effettivamente misurate. Questo processo viene detto fase di identificazione.

I dati fisici raccolti dalle misurazioni effettuate sul robot che deve essere calibrato contengono informazioni che permetteranno di ottenere una relazione accurata tra lo spazio giunti (le letture dei trasduttori) e lo spazio operativo reale del robot (la posa del robot).

Una considerazione importante durante l'identificazione è l'errore previsto nei coefficienti identificati a causa del rumore nel processo di misurazione.

L'esame della letteratura sulla calibrazione dei robot mostra che sono stati utilizzati diversi metodi numerici per identificare i parametri geometrici e non geometrici. Una delle considerazioni più importanti durante l'identificazione riguarda l'errore previsto nei coefficienti identificati a causa del rumore nel processo di misurazione. Esistono due strategie per mettere in relazione il modello reale con il modello nominale. Una è di tipo parametrico e l'altra è di tipo geometrico [8].

Il modello cinematico del robot, indipendentemente dalla convenzione di modellazione, come verrà approfondito nel capitolo 3, consiste fondamentalmente nello specificare le rette degli assi dei giunti del robot in una configurazione arbitraria. Alla base dell'approccio geometrico, i parametri di collegamento, come le normali comuni e gli angoli *twist* e giunto (descritti nel capitolo 3), vengono determinati dall'analisi della posizione relativa degli assi giunto adiacenti. La

filosofia dell'identificazione geometrica, identificando le equazioni delle linee 3D che costituiscono gli assi dei giunti del robot, permette che il modello cinematico vero e proprio possa essere costruito secondo qualsiasi convenzione di modellazione. L'approccio di identificazione parametrica, invece, prevede l'assunzione di errori additivi nei parametri dei collegamenti del robot. Il problema di identificazione consiste quindi nel trovare il vettore degli errori dei parametri cinematici. Questo approccio di identificazione può essere analizzato in due modi, scegliendo o meno di linearizzare il problema. Linearizzare il problema consiste nel trovare un sistema lineare che approssimi il comportamento della funzione (2. 1) non lineare in un certo intorno. Il metodo di linearizzazione utilizzato in letteratura avviene espandendo la funzione (2. 1) e fermandosi al primo ordine di approssimazione, trascurando i termini di secondo ordine o successivo [8], ottenendo il jacobiano di identificazione. Nello spiegare i metodi di identificazione lineari si procederà descrivendo questi indipendentemente dal metodo scelto per linearizzazione del sistema.

Il secondo approccio per trovare gli errori dei parametri cinematici consiste nell'applicare un modello di regressione non lineare.

In entrambi i tipi di approccio lineare e non lineare è necessario introdurre il concetto di funzione di costo.

La funzione costo che verrà approfondita in questo capitolo è il quadrato dell' errore medio della posa (2. 2). Minimizzando questo errore vorrà dire che si saranno identificati i parametri che attraverso la trasformazione (2. 1) approssimano meglio il comportamento del manipolatore reale .

Il processo di identificazione parte dalle misurazioni che hanno determinato l'errore di posa del TCP:

$$\mathbf{e}_j = \mathbf{P}_{mj} - \mathbf{P}_{pj} \quad (2. 2)$$

Dove \mathbf{P}_{mj} è la posa misurata e \mathbf{P}_{pj} è la posa prevista dal modello alla misurazione della posa j-esima.

Il vettore \mathbf{e}_j fornisce la differenza tra la posa misurata e la posa predetta dal modello per una certa configurazione j.

Si prenda in considerazione la posa introdotta nell'eq (2. 1). La posa \mathbf{P} dipende dai due vettori $\boldsymbol{\eta}$ e $\boldsymbol{\theta}$ e la funzione che lega la posa ad i due vettori dipenderà dal tipo di modello adottato.

Trattando il modello cinematico, la posa raggiunta dal TCP può essere espressa attraverso un vettore \mathbf{x}_j per una certa configurazione j .

$$\mathbf{x}_j \cong f(\boldsymbol{\theta}_j, \boldsymbol{\eta}) \quad j = 1, 2, \dots, m \quad (2. 3)$$

$\boldsymbol{\theta}_j$ rappresenta invece la corrispettiva variabile giunto per ciascuna configurazione j .

L'Eq. (2. 2) si può riscrivere come:

$$\mathbf{e}_j = \mathbf{x}_j - f(\boldsymbol{\theta}_j, \boldsymbol{\eta}) \quad (2. 4)$$

La posa misurata \mathbf{P}_{mj} viene identificata attraverso il vettore \mathbf{x}_j e la posa prevista \mathbf{P}_{pj} viene identificata dalla funzione $f(\boldsymbol{\theta}_j, \boldsymbol{\eta})$ (2. 3).

In condizioni ideali la funzione $f(\boldsymbol{\theta}_j, \boldsymbol{\eta})$ dovrebbe coincidere con la posa effettiva del TCP.

I vettori $\boldsymbol{\eta}$ e $\boldsymbol{\theta}_j$ di un approccio parametrico possono essere espressi:

$$\boldsymbol{\eta} = \boldsymbol{\eta}_0 + \delta\boldsymbol{\eta} \quad (2. 5)$$

$$\boldsymbol{\theta}_j = \boldsymbol{\theta}_{j0} + \delta\boldsymbol{\theta} \quad (2. 6)$$

Dove $\boldsymbol{\eta}_0$ è il set dei parametri cinematici nominali e $\delta\boldsymbol{\eta}$ è il vettore incognito degli errori ai parametri cinematici, escludendo gli offset dei giunti. Il vettore $\boldsymbol{\theta}_j$ contiene i valori nominali di attuazione ai giunti per una certa configurazione $\boldsymbol{\theta}_{j0}$ assieme a $\delta\boldsymbol{\theta}$ che è il vettore degli offset ai giunti. I vettori $\delta\boldsymbol{\eta}$ e $\delta\boldsymbol{\theta}$ sono le variazioni dei parametri cinematici del robot.

Lo scopo dello step di identificazione è quello di individuare il vettore del modello dei coefficienti $\delta\boldsymbol{\eta}$ e $\delta\boldsymbol{\theta}$ tale da minimizzare \mathbf{e}_j per il set di pose misurate.

L'aggregazione delle pose misurate sarà \mathbf{x}_{ag}

$$\mathbf{x}_{ag} \cong \mathbf{f}_{ag}(\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m, \boldsymbol{\eta}) = \begin{bmatrix} f(\boldsymbol{\theta}_1, \boldsymbol{\eta}) \\ \vdots \\ f(\boldsymbol{\theta}_m, \boldsymbol{\eta}) \end{bmatrix} \quad (2. 7)$$

Dove $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m$ contengono tutte le attuazioni ai giunti effettuate sul manipolatore e gli offset $\delta\boldsymbol{\theta}$. Il numero di configurazioni assunte dal manipolatore in fase di calibrazione sarà pari ad m . Le pose assunte dal manipolatore devono essere

sufficienti da poter permettere l'identificazione e deve essere sufficientemente grande per poter calcolare le incognite $\delta\boldsymbol{\eta}$ e $\delta\boldsymbol{\theta}$; il vettore \boldsymbol{x}_{ag} non deve dare origini ad un sistema sottodeterminato.

Dato che ciascun vettore \boldsymbol{x}_i ha dimensione 6 (pari ai GDL della posa), la dimensione del vettore posa aggregato \boldsymbol{x}_{ag} sarà;

$$\dim \boldsymbol{x}_{ag} = \dim(\boldsymbol{x}_1^T, \dots, \boldsymbol{x}_m^T) = 6m > \dim \boldsymbol{\theta} + \dim \boldsymbol{\eta} = K \quad (2.8)$$

Per un manipolatore ad N gradi di libertà si ha che $K \geq 4N$ [8].

La posa misurata \boldsymbol{x}_{ag} non coincide perfettamente con la funzione che permette di stimare la posa del TCP \boldsymbol{f}_{ag}

Aggregando i vettori posa per tutte le configurazioni j si ottiene \boldsymbol{e}_{ag}

$$\boldsymbol{e}_{ag} = \boldsymbol{x}_{ag} - \boldsymbol{f}_{ag}(\boldsymbol{\theta}, \boldsymbol{\eta}) \quad (2.9)$$

Si introduce inoltre il vettore che indica gli errori sconosciuti nei parametri cinematici del robot come:

$$\delta\boldsymbol{k}_{ag} = (\delta\boldsymbol{\theta}^T, \delta\boldsymbol{\eta}^T)^T \quad (2.10)$$

Di seguito verranno spiegati alcuni metodi lineari e non lineari.

Per quanto riguarda i metodi lineari si tratteranno i minimi quadrati lineari [8] ed i filtri di Kalman[35][7]; invece, per i metodi non lineari, si tratteranno il *direct search*[8][37], i metodi a gradiente[8][38] e gli algoritmi genetici[33][26][37].

2.2.3.1 Minimi Quadrati Lineari

Si cerca una soluzione linearizzando il modello cinematico che esprime la differenza tra la posa prevista dal modello e la posa misurata, linearizzando \boldsymbol{e}_j introdotto nella (2.4) si ha:

$$\boldsymbol{e}_j = \boldsymbol{x}_j - \boldsymbol{f} = \boldsymbol{H}_j \delta\boldsymbol{k}_{ag} + \boldsymbol{v}_j \quad j = 1, \dots, m \quad (2.11)$$

In ogni posa di misura esiste un vettore di errore di misura inevitabile \boldsymbol{v}_j .

La matrice \boldsymbol{H}_j è definita dal modello cinematico del robot in ciascuna configurazione j.

Questa matrice cambia per ogni configurazione misurata del TCP

Aggregando i vettori posa si ottiene \boldsymbol{e}_{ag}

$$\boldsymbol{e}_{ag} = \boldsymbol{x}_{ag} - \boldsymbol{f}_{ag} = \boldsymbol{H}_{ag} \delta\boldsymbol{k}_{ag} + \boldsymbol{V}_{ag} \quad (2.12)$$

dove

$$\mathbf{H}_{ag} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_m \end{bmatrix} \quad (2.13)$$

e

$$\mathbf{V}_{ag} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_m \end{bmatrix} \quad (2.14)$$

Il vettore \mathbf{e}_{ag} è l'aggregato dei vettori errore della posa per le configurazione j. Ciascun vettore $\mathbf{e}_j(\delta\mathbf{k}_{ag})$ rappresenta l'errore della posa per la configurazione j esima, per ognuno di questi vettori esistono un numero di componenti pari alla dimensione della posa in una configurazione: $\dim(\mathbf{x}_j)$.

Esempio: $\dim(\mathbf{x}_j)=6$ allora \mathbf{e}_j è composto da sei elementi:

$$\mathbf{e}_j = \begin{bmatrix} e_{6j}(\delta\mathbf{k}_{ag}) \\ e_{6j+1}(\delta\mathbf{k}_{ag}) \\ \vdots \\ e_{6j+5}(\delta\mathbf{k}_{ag}) \end{bmatrix} \quad (2.15)$$

Il vettore \mathbf{e}_{ag} è dato quindi da:

$$\mathbf{e}_{ag} = \begin{bmatrix} \mathbf{e}_1(\delta\mathbf{k}_{ag}) \\ \vdots \\ \mathbf{e}_m(\delta\mathbf{k}_{ag}) \end{bmatrix} = \begin{bmatrix} e_1(\delta\mathbf{k}_{ag}) \\ \vdots \\ e_K(\delta\mathbf{k}_{ag}) \end{bmatrix} \quad (2.16)$$

Quando si risolve un insieme di equazioni algebriche lineari le cui incognite dipendono da dati misurati contenenti errori, per ridurre l'influenza degli errori sui risultati del calcolo, è opportuno studiare un sistema sovradeterminato. È necessario cioè prendere un numero sufficiente di dati misurati in modo che il numero di equazioni sia maggiore del numero di incognite. Nell metodo standard dei minimi quadrati, si tratta un modello non stocastico, ossia il vettore rumore \mathbf{V} è trattato come una variabile sconosciuta.

L'obbiettivo è trovare un valore $\Delta\mathbf{k}_{ag}$ approssimazione del vettore $\delta\mathbf{k}_{ag}$.

A questo proposito si crea la funzione obbiettivo scalare L sarà pari a

$$L = (\mathbf{e}_{ag} - \mathbf{H}\Delta\mathbf{k}_{ag})^T (\mathbf{e}_{ag} - \mathbf{H}\Delta\mathbf{k}_{ag}) \quad (2.17)$$

Si cerca di ottenere $\Delta\mathbf{k}_{ag}$ che minimizzi la funzione obbiettivo L somma dei quadrati degli elementi del vettore errore: $\mathbf{x}_{ag} - \mathbf{H}\Delta\mathbf{k}_{ag}$.

Questa funzione obiettivo è proporzionale al quadrato degli elementi di $\Delta\mathbf{k}_{ag}$. Si ottiene un minimo di L quando

$$\frac{\partial L}{\partial(\Delta\mathbf{k}_{ag})} = 0$$

assieme alla condizione che la matrice hessiana di L sia definita semipositiva

$$\frac{\partial^2 L}{\partial(\Delta\mathbf{k}_{ag})} \geq 0 \quad (2.18)$$

Derivando la (2.17) si ottiene

$$\frac{\partial L}{\partial(\Delta\mathbf{k}_{ag})} = -\mathbf{H}^T(\mathbf{e}_{ag} - \mathbf{H}\Delta\mathbf{k}_{ag}) = 0 \quad (2.19)$$

Da cui

$$\mathbf{H}^T \mathbf{e}_{ag} = \mathbf{H}^T \mathbf{H} \Delta\mathbf{k}_{ag} \quad (2.20)$$

Dai quali è possibile ottenere le modifiche ai parametri cinematici incogniti:

$$\Delta\mathbf{k}_{ag} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{e}_{ag} \quad (2.21)$$

Inoltre derivando ulteriormente la (2.20) si ha che la L è semidefinita positiva e quindi la (2.21), quando $(\mathbf{H}^T \mathbf{H})^{-1}$ è non singolare, i valori di $\Delta\mathbf{k}_{ag}$ convergeranno ad un minimo.

I casi singolari, quando la matrice $\mathbf{H}^T \mathbf{H}$ è singolare, indicano un modello di misura mal condizionato. Questo è il caso in cui alcune delle equazioni algebriche linearmente dipendenti da altre equazioni nelle equazioni di misura. Il modello di misura dovrebbe quindi essere modificato eliminando tutte queste equazioni ridondanti.

In letteratura per la matrice H viene utilizzata la matrice Jacobiana di L, infatti tutti i lavori citati nella Tabella 2.1 che usano come metodo di identificazione i minimi quadrati utilizzano la matrice Jacobiana per linearizzare il sistema.

La matrice Jacobiana viene introdotta nell'analisi dei metodi a gradiente. Il requisito di "piccola perturbazione" dei parametri cinematici è importante per garantire la convergenza dell'algoritmo dei minimi quadrati; ossia, in presenza di modifiche dei parametri cinematici troppo grandi l'algoritmo potrebbe non convergere oppure potrebbe farlo ma molto più lentamente[38]; infatti, come si

vedrà nel paragrafo dei metodi a gradiente, con questo metodo non si può garantire la convergenza, in quanto non si hanno informazioni sulla matrice Hessiana. Per convergere ad una soluzione conviene avere un numero di condizionamento della matrice Jacobiana basso e delle piccole modifiche dei parametri cinematici[8][39].

2.2.3.2 Metodi a Gradiente

I metodi non lineari, a differenza di quelli lineari, sviluppano l'identificazione come un problema di ottimizzazione non lineare non vincolato.

Il problema di ottimizzazione deve selezionare $\delta\boldsymbol{\eta}$ e $\delta\boldsymbol{\theta}$ tali da minimizzare la funzione costo L definita come:

La funzione costo L rappresenta l'errore quadratico della posa reale rispetto alla

$$\begin{aligned} L &= \|\mathbf{x}_{ag} - \mathbf{f}_{ag}(\boldsymbol{\theta}_1 + \delta\boldsymbol{\theta}, \dots, \boldsymbol{\theta}_m + \delta\boldsymbol{\theta}, \boldsymbol{\eta} + \delta\boldsymbol{\eta})\|_2 \\ &= \mathbf{e}_{ag}^T(\delta\mathbf{k}_{ag})\mathbf{e}_{ag}(\delta\mathbf{k}_{ag}) \end{aligned} \quad (2.22)$$

posa prevista per il TCP del manipolatore, il problema consiste nella minimizzazione di $L(\delta_1, \dots, \delta_K)$.

Affinchè un punto $(\delta\mathbf{k}_{ag})$ sia un minimo, sono necessarie due condizioni:

il gradiente di L, \mathbf{g} valutato su $\delta\mathbf{k}_{ag}$ deve essere pari a zero

$$\mathbf{g}[(\delta\mathbf{k}_{ag})] = \nabla L(\delta\mathbf{k}_{ag})|_{(\delta\mathbf{k}_{ag})^*} = 0 \quad (2.23)$$

e la matrice Hessiana, G, deve essere definita positiva.

$$\mathbf{G}[(\delta\mathbf{k}_{ag})^*] = \nabla^2 L(\delta\mathbf{k}_{ag})|_{(\delta\mathbf{k}_{ag})^*} > 0 \quad (2.24)$$

La maggior parte dei metodi dei minimi quadrati non lineari sono iterativi. Dove l'iterazione j-esima si compone in questo modo:

$$(\delta\mathbf{k}_{ag})_{j+1} = (\delta\mathbf{k}_{ag})_j + \alpha_j \mathbf{p}_j \quad (2.25)$$

dove α_j , è uno scalare che varia da un metodo di ottimizzazione all'altro, mentre \mathbf{p}_j si dice vettore di ricerca, dai quali è possibile effettuare la nuova stima $(\delta\mathbf{k}_{ag})_{j+1}$.

Si ha che i primi due termini dell'espansione in serie di Taylor del gradiente

$\mathbf{g}[(\delta\mathbf{k}_{ag})_j + \mathbf{p}_j]$ sono:

$$\begin{aligned} \mathbf{g} [(\delta \mathbf{k}_{ag})_{j+1}] &= \mathbf{g} [(\delta \mathbf{k}_{ag})_j + \mathbf{p}_j] \\ &\cong \mathbf{g} [(\delta \mathbf{k}_{ag})_j] + \mathbf{G} [(\delta \mathbf{k}_{ag})_j] \mathbf{p}_j \end{aligned} \quad (2.26)$$

Dove \mathbf{G} rappresenta la matrice Hessiana di L

Ponendo il gradiente pari a zero, per valutare \mathbf{p}_j bisogna risolvere la seguente equazione matriciale:

$$\mathbf{G} [(\delta \mathbf{k}_{ag})_j] \mathbf{p}_j = -\mathbf{g} [(\delta \mathbf{k}_{ag})_j] \quad (2.27)$$

Per risolvere l'equazione si usa come strumento la matrice Jacobiana

La definizione analitica della matrice Jacobiana è:

$$\mathbf{J}_j = \left[\frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \delta_1}, \dots, \frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \delta_K} \right] \Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_j+\delta\boldsymbol{\theta}, \boldsymbol{\eta}=\boldsymbol{\eta}_0+\delta\boldsymbol{\eta}} \quad (2.28)$$

Dove $\delta_1, \dots, \delta_K$ sono le componenti di $\delta \mathbf{k}_{ag}$.

Aggregando tutte le configurazioni misurate, $j = 1, \dots, m$, si ottiene la matrice giacobiana \mathbf{J} corrispondente per le configurazioni $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m$.

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_m \end{bmatrix}$$

La derivata parziale di L lungo una delle variabili scalari di $\delta \mathbf{k}_{ag}$ è

$$\mathbf{g}_j = \frac{\partial L(\delta \mathbf{k}_{ag})}{\partial \delta_j}$$

Derivando L definita nella (2.22) lungo tutte le componenti di $\delta \mathbf{k}_{ag}$ si può dimostrare che:

$$\mathbf{g}(\delta \mathbf{k}_{ag}) = -2\mathbf{J}^T \mathbf{e}_{ag} \quad (2.29)$$

Dove \mathbf{J} ed \mathbf{e}_{ag} dipendono da $\delta \mathbf{k}_{ag}$

La matrice Hessiana di L può essere ottenuta a partire dal gradiente e sarà composta da due componenti:

$$\mathbf{G} = 2\mathbf{J}^T \mathbf{J} + 2\mathbf{S} \quad (2.30)$$

Una componente ricavata da termini di derivazione al primo ordine $2\mathbf{J}^T \mathbf{J}$ ed un'altra non lineare \mathbf{S} ricavabile da termini derivati al secondo ordine.

$$\mathbf{S} = \sum_{i=1}^{6m} (\mathbf{e}_{ag})_i \mathbf{T}_i = \sum_{i=1}^{6m} (\mathbf{e}_{ag})_i \nabla^2 [(\mathbf{e}_{ag})_i] \quad (2.31)$$

La matrice Hessiana \mathbf{G} dipende da \mathbf{J} ed \mathbf{S} ed entrambi dipendono da $\delta\mathbf{k}_{ag}$.

Nota la matrice Jacobiana è possibile sostituire all'interno dell'Eq. (2. 27) le Eq333.

(2. 29) e (2. 30) ottenendo:

$$(\mathbf{J}^{(j)T}\mathbf{J}^{(j)} + \mathbf{S}^{(j)})\mathbf{p}_j = \mathbf{J}^{(j)T}\mathbf{e}_{ag}^{(j)} \quad (2. 32)$$

Da questo sistema lineare, si ottiene \mathbf{p}_j e si aggiorna $\delta\mathbf{k}_{ag}$:

$$(\delta\mathbf{k}_{ag})_{j+1} = (\delta\mathbf{k}_{ag})_j + \mathbf{p}_j \quad (2. 33)$$

Questa soluzione viene detta metodo di Newton ed è alla base di molti metodi di identificazione usati in letteratura.

Trascurando il termine $\mathbf{S}[(\delta\mathbf{k}_{ag})_j]$ nell'equazione di Newton (2. 32), l'algoritmo di identificazione si riduce all'algoritmo iterativo dei minimi quadrati, noto anche come metodo Gauss-Newton (2. 21):

$$(\mathbf{J}^{(j)T}\mathbf{J}^{(j)})\mathbf{p}_j = \mathbf{J}^{(j)T}\mathbf{e}_{ag}^{(j)} \quad (2. 34)$$

La direzione \mathbf{p}_j ottenuto attraverso la (2. 34), combinata con la (2. 33) porta alla soluzione del problema ai minimi quadrati lineari qualora nella (2. 21) si scegliesse come matrice \mathbf{H} la matrice Jacobiana valutata lungo le componenti di $\delta\mathbf{k}_{ag}$. L'Eq. (2. 34) non garantisce necessariamente che \mathbf{p}_j sia una direzione di discesa poiché il coefficiente scalare α_j che si assume pari ad 1, potrebbe essere un passo troppo grande lungo il vettore di ricerca. Infatti, si garantisce che \mathbf{p}_j è una direzione di discesa se $[\mathbf{J}^{(j)T}\mathbf{J}^{(j)} + \mathbf{S}^{(j)}]$ è una matrice positiva, (ossia \mathbf{G} positivo). Ignorare il termine $\mathbf{S}[(\delta\mathbf{k}_{ag})_j]$ è utile per i casi in cui $\|\delta\mathbf{k}_{ag}\|$ è piccolo. L'algoritmo di Gauss-Newton si interrompe in corrispondenza delle singolarità della jacobiana di identificazione e converge molto lentamente in prossimità di tali singolarità.

Oltre a Newton e Gauss-newton esistono metodi più popolari, come l'algoritmo di Levenberg-Marquardt [20][23].

Il metodo Levenberg-Marquardt (LM) è stato introdotto come miglioramento del metodo Gauss-Newton. La tecnica è stata progettata per superare i problemi legati alla singolarità della matrice $\mathbf{J}^{(j)T}\mathbf{J}^{(j)}$, pur mantenendo le proprietà di convergenza di Gauss-Newton in prossimità della soluzione ottimale [8]. L'idea è quella di

modificare l'Eq. (2. 34) aggiungendo un coefficiente scalare non negativo variabile nel tempo, cercando di approssimare $\mathbf{S} [(\delta \mathbf{k}_{ag})_j]$:

$$(\mathbf{J}^{(j)T} \mathbf{J}^{(j)} + \mu_j \mathbf{I}) \mathbf{p}_j = \mathbf{J}^{(j)T} \mathbf{e}_{ag}^{(j)} \quad (2. 35)$$

dove \mathbf{I} è la matrice identità. Per un valore sufficientemente grande di μ_j , la matrice $[\mathbf{J}^{(j)T} \mathbf{J}^{(j)} + \mu_j \mathbf{I}]$ è definita positiva e \mathbf{p}_j può essere trasformata in una direzione di discesa. Man mano che la soluzione si avvicina all'ottimo, lo scalare μ_j viene regolato in modo da avere valori sempre più piccoli, in modo che il metodo acquisisca il tasso di convergenza asintotico del metodo Gauss-Newton.

2.2.3.3 Direct search

Quando non è possibile ottenere una derivata di \mathbf{f}_{ag} della eq (2. 7) rispetto a $\boldsymbol{\theta}$ o $\boldsymbol{\eta}$, l'ottimizzazione di L dell'eq (2. 22) può essere effettuata attraverso il cosiddetto direct search, variando le componenti di $\delta \mathbf{k}_{ag}$ una alla volta.

Esistono varie routine di direct search e differiscono tra loro per la strategia scelta con cui viene fatto variare $\delta \mathbf{k}_{ag}$.

Ad esempio per comprendere questo algoritmo si prende la prima componente di $\delta \mathbf{k}_{ag}$.

$$\delta \mathbf{k}_{ag}^{(1)} = (\delta_1, 0, \dots, 0)^T \quad (2. 36)$$

Si sceglie un intervallo entro i quali può variare la nostra componente

$$\delta_{1 \min} \leq \delta_1 \leq \delta_{1 \max} \quad (2. 37)$$

E si cerca di ridurre la dimensione di questo intervallo in modo tale da minimizzare L (metodo della sezione aurea, bisezione ecc), sino ad ottenere un intervallo ottimo δ_1^* sufficientemente piccolo, attraverso la scelta di opportuni criteri di convergenza per il minimo locale di L.

Ottenuto δ_1^* ottimo, si procede con il termine scalare successivo del vettore $\delta \mathbf{k}_{ag}$ calcolando $\delta \mathbf{k}_{ag}^{(2)} = (\delta_1^*, \delta_2, \dots, 0)^T$ allo stesso modo.

Si procede fino all'ultima variabile $\delta \mathbf{k}_{ag}^{(K)} = (\delta_1^*, \dots, \delta_K^*)^T$ e successivamente si ripete l'intero processo dal primo elemento del vettore ottenuto.

$$(\delta_1^{**}, \delta_2^*, \dots, \delta_{4N}^*) \rightarrow (\delta_1^{**}, \delta_2^{**}, \dots, \delta_{4N}^*) \rightarrow \dots \quad (2. 38)$$

Fino a quando non si converge ad un valore minimo di L.

La convergenza di tutti gli algoritmi direct search è naturalmente molto lenta, tuttavia, non è necessario calcolare le jacobiane e si perde il requisito di "piccola perturbazione".

2.2.3.4 Kalman filtering

Le tecniche di stima discusse finora sono tutte algoritmi di *batch processing*. In altri termini, prima si raccolgono tutti i dati e poi si elabora l'intero set di misure simultaneamente. Un approccio alternativo è la stima ricorsiva. In uno stimatore ricorsivo le misure arrivano in sequenza e non è necessario memorizzare le misure precedenti per calcolare le stime presenti. Mentre nella stima *batch* la dimensione del vettore delle misure deve essere maggiore della dimensione del vettore stimato, nella stima ricorsiva si verifica tipicamente il contrario.

Il filtro determina stime degli stati passati, presenti e futuri e tali risultati possono essere conseguiti persino quando la natura del sistema modellato non è conosciuta con precisione. Il filtro di Kalman è un algoritmo utilizzato per il filtraggio dei dati costruito sulla base di una media ragionata tra il prossimo valore predetto e il prossimo valore stimato. Questo filtro spesso viene utilizzato per ottenere una migliore valutazione di un dato ottenuto dalla lettura di più sensori, ognuno caratterizzato da un rumore di misura avente caratteristiche differenti nel tempo.

Il metodoso miglia a quello lineare, invece di assumere un vettore incognito fisso $\delta \mathbf{k}_{ag}$, si assume un vettore che evolve nel tempo:

$$\delta \mathbf{k}_j = \mathbf{A}_{j-1} \delta \mathbf{k}_{j-1} + \mathbf{w}_{j-1} \quad (2.39)$$

Dove $\delta \mathbf{k}_j$ è un vettore di dimensione n chiamato vettore di stato. Lo stato al tempo $t = t_j$ è detto $\delta \mathbf{k}_j$. Esso rappresenta quindi l'errore geometrico dei parametri cinematici del robot calcolati nell'istante j . Il vettore \mathbf{w}_j è il rumore sull'identificazione, e un vettore con rumore bianco, valore medio nullo e con covarianza $\mathbf{Q}_{j-1} = E(\mathbf{w}_{j-1} \mathbf{w}_{j-1}^T)$. È conveniente quando si vuole modellare giunti poco stabili. In pratica, il vettore $\delta \mathbf{k}$ degli errori dei parametri cinematici può fluttuare in modo casuale quando gli assi delle articolazioni del robot oscillano. Se lo stato ricercato $\delta \mathbf{k}_j$ è costante \mathbf{A}_{j-1} corrisponde alla matrice identità, l'Eq. (2.32) diventa:

$$\delta \mathbf{k}_j = \delta \mathbf{k}_{j-1} + \mathbf{w}_{j-1} \quad (2.40)$$

A differenza del dei minimi quadrati lineari eq (2. 11), il vettore errore \mathbf{e}_j viene identificato ad ogni istante di tempo $t = t_j$ e viene indicato come una combinazione lineare delle variabili di stato in quel momento, perturbate dal rumore di misura:

$$\mathbf{e}_j = \mathbf{H}_j \delta \mathbf{k}_j + \mathbf{V}_j \quad (2. 41)$$

Si tratta di un errore \mathbf{e}_j in un istante di tempo e non di un vettore aggregato degli errori. \mathbf{H}_k è una matrice che dipende dall'istante di tempo j e \mathbf{V}_j rappresenta l'errore di misura della posa ed è statisticamente indipendente da \mathbf{w}_j e lo stato iniziale $\delta \mathbf{k}_0$. L'obiettivo è identificare lo stato ricorsivo $\Delta \mathbf{k}_j$ dato dalla combinazione lineare delle misurazioni di \mathbf{e}_j , e delle stime degli stati misurati precedentemente $\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_{j-1}$, in modo da trovare un ottimo di $\Delta \mathbf{k}_j$.

Si cerca quindi una $\Delta \mathbf{k}_j$ che minimizzi la varianza dell'errore dei parametri geometrici \mathbf{P}_j

Ossia

$$\mathbf{P}_j = E[\delta \mathbf{e}_j \delta \mathbf{e}_j^T] \quad (2. 42)$$

Dove $\delta \mathbf{e}_j$ rappresenta l'errore dello stato $\Delta \mathbf{k}_j$ approssimato:

$$\delta \mathbf{e}_j = \delta \mathbf{k}_j - \Delta \mathbf{k}_j \quad (2. 43)$$

Per rendere ciò possibile è necessario seguire un algoritmo che segue determinate fasi.

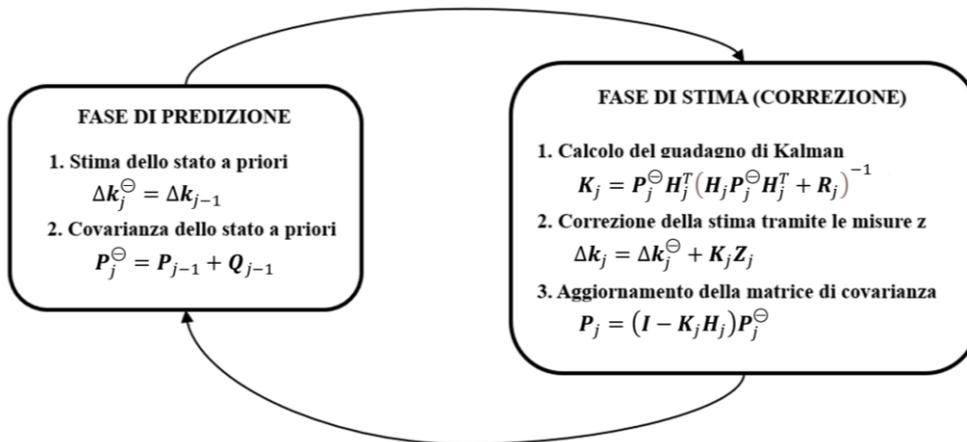


Figura 2.13: Fasi algoritmo filtro di Kalman

Il filtro di Kalman si divide in una prima fase di predizione dove date le letture precedenti si stima lo stato che si prevede per la misurazione al tempo k, Viene previsto lo stato al tempo $\Delta \mathbf{k}_j^\ominus$.

La previsione dello stato nell'istante di tempo k dipende dallo stato precedentemente identificato $\Delta \mathbf{k}_{j-1}$ al tempo j-1 e dalla matrice \mathbf{A}_{j-1} :

$$\Delta \mathbf{k}_j^\ominus = \mathbf{A}_{j-1} \Delta \mathbf{k}_{j-1}$$

Dato che si ricercano dei valori costanti dello stato, per questo caso specifico si avrà che lo stato all'istante di tempo precedente, corrisponderà con lo stato previsto per l'istante di tempo successivo.

$$\Delta \mathbf{k}_j^\ominus = \Delta \mathbf{k}_{j-1} \quad (2.44)$$

Si prevedono inoltre la covarianza \mathbf{P}_j^\ominus dell'errore stimato $\delta \mathbf{e}_j^\ominus$.

$$\mathbf{P}_j^\ominus = E[\delta \mathbf{e}_j^\ominus (\delta \mathbf{e}_j^\ominus)^T] \quad (2.45)$$

Dove l'errore $\delta \mathbf{e}_j^\ominus$ consisterebbe nell'errore stimato dello stato j:

$$\delta \mathbf{e}_j^\ominus = \delta \mathbf{k}_j - \Delta \mathbf{k}_j^\ominus \quad (2.46)$$

Chiaramente l'errore stimato non può essere calcolato in modo esatto in quanto il vettore di stato $\delta \mathbf{k}_j$ non può essere identificato, ma si cerca iterazione dopo iterazione di quantificare la covarianza di questo errore attraverso la eq__.

$$\mathbf{P}_j^\ominus = \mathbf{P}_{j-1} + \mathbf{Q}_{j-1}$$

Effettuata la fase di predizione si confrontano i valori predetti con le misurazioni effettuate, calcolando il guadagno di Kalman \mathbf{K}_j si ottiene lo stato aggiornato $\Delta \mathbf{k}_j$ e la covarianza aggiornata dell'errore nell'identificazione dello stato.

$$\mathbf{K}_j = \mathbf{P}_j^\ominus \mathbf{H}_j^T (\mathbf{H}_j \mathbf{P}_j^\ominus \mathbf{H}_j^T + \mathbf{R}_j)^{-1}$$

$$\Delta \mathbf{k}_j = \Delta \mathbf{k}_j^\ominus + \mathbf{K}_j \mathbf{Z}_j$$

$$\mathbf{P}_j = (\mathbf{I} - \mathbf{K}_j \mathbf{H}_j) \mathbf{P}_j^\ominus$$

Ottenuti i valori per l'istante di tempo j si ripete l'uso dell'algoritmo per l'istante di tempo successivo.

2.2.3.5 Algoritmi Genetici

L'algoritmo genetico (GA) è un tipo di algoritmo di ricerca globale adattativo e probabilistico. I GA imitano il processo evolutivo dei sistemi biologici ricercando

soluzioni sempre migliori a dei problemi numerici fino a quando non si ottiene una soluzione soddisfacente [40].

Il GA fornisce un quadro universale per la risoluzione di problemi di ottimizzazione complessi. Seppur ispirato al principio della selezione naturale ed all'evoluzione biologica, esso non è vincolato a specifici contesti e viene applicato in svariate discipline. Gli algoritmi genetici ereditano e riadattano dalla biologia alcune terminologie utili. La conoscenza di questa terminologia è utile se si vuole comprendere il funzionamento di questo tipo di algoritmo:

- **popolazione:** insieme di soluzioni relative al problema considerato;
- **cromosoma:** una delle soluzioni del problema;
- **gene:** parte di un cromosoma. Generalmente consiste in una o più parti del vettore rappresenta il cromosoma;
- **crossover:** generazione di una nuova soluzione mescolando delle soluzioni esistenti;
- **mutazione:** alterazione casuale di una soluzione;
- **funzione di fitness:** è la funzione obiettivo da ottimizzare e determina la direzione del GA.

In modo iterativo l'algoritmo genera una popolazione e per ciascun cromosoma si calcola il valore della funzione di fitness. Come avviene per la selezione naturale, le popolazioni successive saranno composte da cromosomi aventi quei geni tali da migliorare la funzione di fitness in direzione del risultato atteso. Per ciascuna iterazione verranno generate delle nuove generazioni di soluzioni attraverso crossover e mutazioni, prediligendo quei cromosomi per i quali si ha una funzione fitness migliore.

Questo algoritmo risulta molto vantaggioso: esso è capace di esplorare simultaneamente molte regioni dello spazio di ricerca, è poco sensibile ai minimi locali e non richiede la conoscenza approfondita dei problemi di cui si sta cercando una soluzione. Non è necessario infatti dover effettuare il calcolo delle derivate per risolvere i problemi [20].

Dato che si tratta di un metodo randomico, non è mai certo che il risultato ottenuto sia un ottimo globale del problema. I GA sono inevitabilmente più lenti dei metodi

basati sul calcolo dei problemi in cui possono essere utilizzati. Tuttavia, sono utili per i problemi altamente complicati, non lineari e discontinui.

Data la versatilità di questo algoritmo, è stato utilizzato come metodo di identificazione in alcuni lavori di calibrazione più recenti, con il vantaggio di richiedere informazioni minime sulla funzione obiettivo da ottimizzare [41][33][26]. Nei tre lavori citati nei quali vengono utilizzati gli algoritmi genetici vengono adottate la stessa funzione obiettivo definita nell'Eq. (2. 22) o la sua radice quadrata. Esistono anche varianti ibride del GA, ad esempio il GA-PSO [42] dove l'acronimo PSO sta per "Particle Swarm Optimization" il quale anch'esso è un algoritmo di ottimizzazione che considera una popolazione. Esistono inoltre degli algoritmi genetici ibridi, nei quali a seguito del avvicinamento della funzione di fitness a ridosso di un minimo globale, affina la soluzione ricercando un minimo locale. Un metodo per la ricerca del minimo locale può essere una variante dei metodi a gradiente. Questo tipo di algoritmo viene utilizzata nel quarto capitolo per identificare i parametri cinematici del manipolatore.

2.2.4 Implementazione

L'implementazione consiste nell'uso delle informazioni scaturite dalle tre fasi precedenti della calibrazione per incrementare la performance del manipolatore. La calibrazione dovrebbe aver fornito un modello con dei parametri cinematici accurati, tali da permettere una relazione accurata tra le variabili giunto e la posa del TCP. La fase di implementazione è la fase meno generalizzata del processo di calibrazione, in quanto la sua implementazione dipende dalla macchina presa in considerazione.

Tutti i manipolatori che sono in grado di accettare come input una posa nello spazio operativo, devono essere in grado, attraverso la cinematica inversa, di ottenere la relativa configurazione nello spazio giunti tale da poter collocare TCP nella posa dello spazio operativo corretta. Affinchè ciò sia possibile questa fase coinvolge la modifica del modello nel modello embedded del controller del robot. Vengono cioè aggiornati i parametri cinematici (ed eventualmente quelli non geometrici) nel software del robot.

Capitolo 3

3 Modello del problema cinematico diretto

Riprendendo la struttura meccanica introdotta nel primo capitolo, si cercherà di descrivere e modellare le due componenti principali del manipolatore: giunti e *link*. I giunti di un robot possono essere molto semplici, come un giunto di rivoluzione o prismatico, con un solo grado di libertà oppure possono essere più complessi e presentare un numero maggiore di GDL.

Dato che questa tesi si tratterà un manipolatore con giunti rotoidali ad un solo grado di libertà, l'azione di ciascuno di questi potrà essere descritta da un singolo parametro o grado di mobilità (GDM). La configurazione del singolo giunto sarà espressa mediante un angolo.

Il modello cinematico di un robot permette la descrizione analitica delle relazioni che intercorrono tra la posizione dei giunti e la configurazione del TCP, posto all'estremità della catena di corpi e giunti che formano il braccio.

Il problema cinematico diretto riguarda la determinazione della posa del TCP note le configurazioni dei giunti.

Esistono numerosi approcci per sviluppare il modello cinematico di un robot manipolatore, il metodo più popolare è stato definito da Denavit e Hartenberg e questo è basato sull'utilizzo di matrici di trasformazione omogenee.

Le trasformazioni omogenee tra configurazioni di corpi rigidi sono lo strumento che verrà usato in questa tesi per approcciare il problema cinematico diretto dei bracci manipolatori seriali. Si può procedere infatti a descrivere la configurazione di ogni membro, cui si associa una terna di riferimento solidale, descrivendola relativamente al precedente, a partire dal membro fisso, o base.

Il set di possibili trasformazioni omogenee elementari è rappresentato da tre traslazioni (T) e tre rotazioni (R) nello spazio:

$$\begin{aligned}
 T_{x,a} &= \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & R_{x,\alpha} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\alpha & -s_\alpha & 0 \\ 0 & s_\alpha & c_\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{y,b} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & R_{y,\beta} &= \begin{bmatrix} c_\beta & 0 & s_\beta & 0 \\ 0 & 0 & 0 & 0 \\ -s_\beta & 0 & c_\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & (3.1) \\
 T_{z,c} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} & R_{z,\gamma} &= \begin{bmatrix} c_\gamma & -s_\gamma & 0 & 0 \\ s_\gamma & c_\gamma & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

È una pratica generalizzata quella di procedere in modo sistematico nella scelta di sistemi di riferimento che permettano di scrivere le catene di trasformazioni omogenee nel modo più semplice possibile.

Tra le rivisitazioni del modello di DH più usate, è presente quella di Hayati [39]. Questa è molto importante perché ha permesso di ridurre i problemi di singolarità e mal condizionamento che si ottenevano adottando la convenzione di DH standard, in presenza di link con giunti ad assi paralleli.

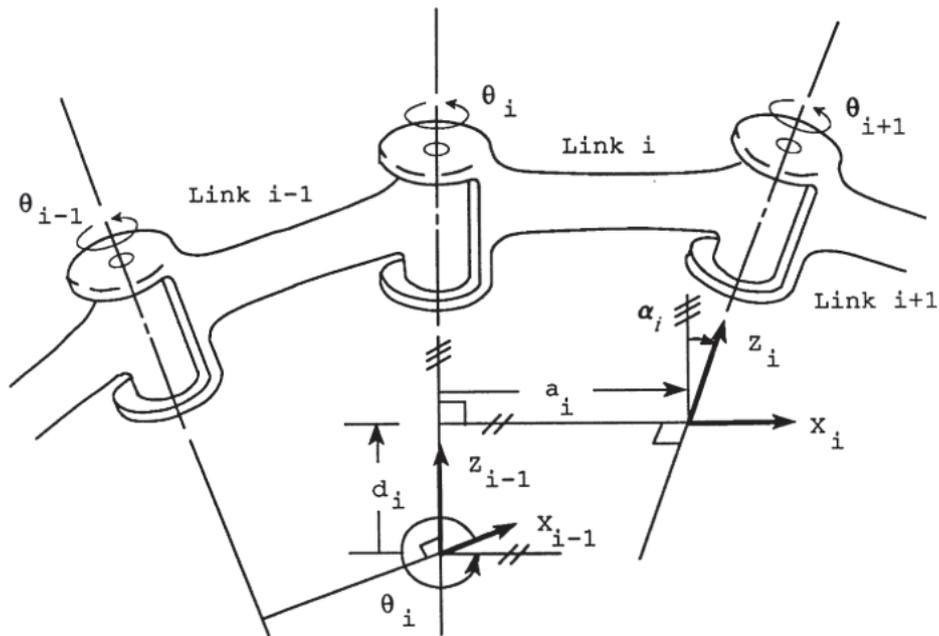
Queste convenzioni ed altre si differenzieranno a seconda di come vengono selezionati i sistemi di riferimento solidali ai link del manipolatore, a seconda delle trasformazioni omogenee adottate e dall'ordine con la quale quest'ultime vengono adoperate.

Nonostante la presenza di vari modelli, la convenzione di DH rimane il modello più adoperato per i modelli cinematici dei robot attualmente utilizzati.

3.1 Denavit Hartenberg Standard

Come già anticipato, il metodo si basa su delle trasformazioni omogenee. La procedura consiste nel definire dei sistemi di coordinate su ciascun asse giunto.

Ciascun sistema di coordinate è riferito alle successive attraverso uno specifico set di coefficienti all'interno delle matrici di trasformazione omogenea.


 Figura 3.1: Esempio di due *link* contigui nella convenzione di Denavit Hartemberg [14]

In Figura 3.1 sono rappresentati due *link* e tre giunti. Ogni *link* è connesso a due giunti, i giunti i -esimi sono connessi ai *link* i ed $i-1$.

Dato un robot manipolatore con n giunti ed $n+1$ *link*, i giunti vengono numerati da 1 ad n , e i *link* da 0 ad n , partendo dalla base. Il *link* $n+1$ è il TCP all'estremità del robot.

Si associa al giunto i -esimo, la variabile giunto denotata con q_i che rappresenta la configurazione giunto. Il posizionamento del giunto i -esimo è fisso rispetto al *link* $i-1$. Quando il giunto i viene attuato, attuando una particolare configurazione q_i , il *link* i si muoverà.

Il *link* 0 (base) è fisso e non si muove, il frame a lui solidale viene identificato come sistema di riferimento inerziale.

Ciascuna trasformazione omogenea A_i^{i-1} che esprime le coordinate del *link* i -esimo rispetto al precedente è composta da quattro trasformazioni omogenee.

$$A_i^{i-1} = R_{z,\theta_i} T_{z,d_i} T_{x,a_i} R_{x,\alpha_i} = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Dove le quattro quantità a_i , α_i , d_i e θ_i sono i parametri associati a *link* e giunto i -esimi, questi sono detti parametri di DH e rappresentano i parametri cinematici del *link* i -esimo del manipolatore.

I quattro parametri di DH descrivono la posizione e l'orientamento relativo del *link* i -esimo rispetto alla posizione del giunto i -esimo.

Dato che è presente un solo GDM, tre parametri di questi quattro assumono un valore costante per un dato *link*. Il quarto parametro, θ_i per un giunto di rivoluzione, andrà a rappresentare la variabile giunto.

I parametri di DH sono detti:

- θ_i angolo giunto: l'angolo da x_{i-1} a x_i misurato rispetto a z_{i-1} .
- α_i angolo di *twist*: l'angolo da z_{i-1} a z_i misurato rispetto a x_i .
- a_i lunghezza *link*: distanza da z_{i-1} a z_i misurato rispetto a x_i .
- d_i *link* offset: distanza da x_{i-1} a x_i misurato rispetto a z_{i-1} .

I nomi assegnati a questi parametri di DH derivano dalla relazione geometrica che intercorre tra i sistemi di coordinate dei due *link* adiacenti.

L'Eq. (3. 2) può essere intesa come un susseguirsi di step sequenziali dal frame solidale al *link* $i-1$, attraverso A_i^{i-1} passiamo dal frame $i-1$ al frame del *link* i -esimo. Ognuna delle quattro trasformazioni omogenee che compongono la trasformazione A_i^{i-1} sono uno step (Figura 3.1 e Figura 3.2)

- Rotazione attorno all'asse z_{i-1} dell'angolo θ_i
- Traslazione lungo l'asse z_{i-1} della distanza d_i
- Traslazione lungo il ruotato asse x_{i-1} della distanza a_i
- Rotazione intorno ad x_i dell'angolo α_i

Per conoscere l'ubicazione del TCP del manipolatore o posa si considerano n *link* ed n trasformazioni omogenee.

Queste trasformazioni determinano la relazione funzionale tra la posa ed il sistema di riferimento inerziale (SR applicato alla BASE).

$$A_6^0 = A_1^0 A_2^1 A_3^2 A_4^3 A_5^4 A_6^5 \quad (3. 3)$$

Dove A_6^0 rappresenta la trasformazione omogenea che porta dalla base alla posa, rappresenta cioè le coordinate del sistema di riferimento associato al TCP del robot.

Più in generale è possibile definire le coordinate di tutti i SR dei *link* non solo riferendoli alla base o al SR precedente, bensì riferendoli a qualunque *link* del sistema robotico.

$$\begin{cases} A_j^i = A_{i+1}A_{i+2}\dots A_{j-1}A_j & \text{se } i < j \\ A_j^i = I & \text{se } i = j \\ A_j^i = (A_i^j)^{-1} & \text{se } j > i \end{cases} \quad (3.4)$$

Dove A_j^i rappresenta la matrice omogenea che lega due *link* i e j ; la matrice I invece è la matrice identità. Per semplificare la notazione dell'Eq. (3.4), la generica trasformazione A_i^{i-1} è stata scritta come A_i per alleggerire la notazione.

3.1.1 Assegnazione dei parametri di DH

L'assegnazione di ciascun parametro di DH a ciascun *link* non è arbitraria, ma segue delle specifiche regole d'attribuzione. Per poter attribuire correttamente i parametri di DH bisogna identificare il SR_i a partire dal SR_{i-1} già noto, assumendo che siano note le direzioni degli assi giunti nello spazio, coincidenti con gli assi giunto $i+1$ ed i (assi z_i e z_{i-1}) (Figura 3.2).

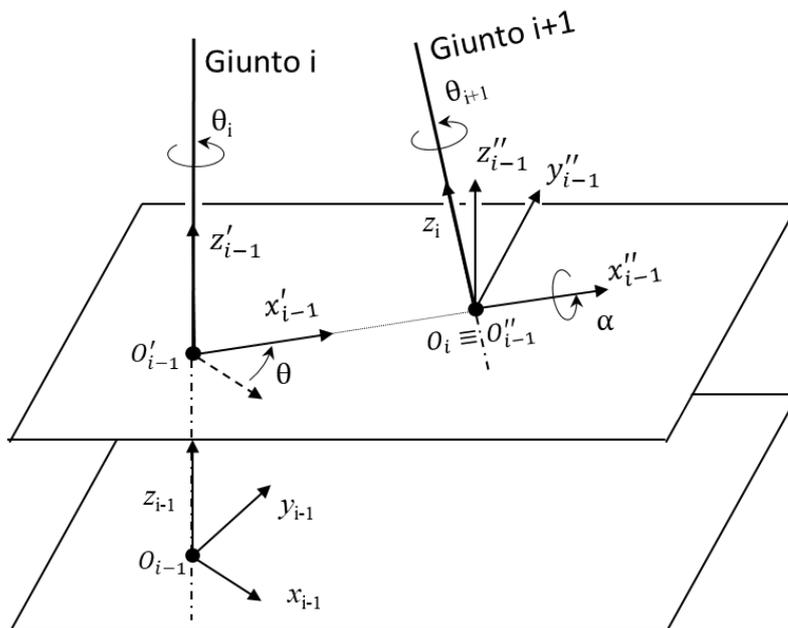


Figura 3.2: Disposizione assi giunto convenzione DH Standard

Il passaggio dal SR_i al SR_{i+1} avviene attraverso la trasformazione (3.2) composta da due rototraslazioni. I SR'_i e SR''_i sono due sistemi di riferimento intermedi.

- il SR'_i si ottiene attuando una rototraslazione del SR_i lungo l'asse z_{i-1} attraverso i parametri θ_i e d_i ;
- il SR''_i mediante la traslazione del SR'_i lungo l'asse x'_{i-1} della distanza a_i ;
- il SR_{i+1} infine sarà dato dalla rotazione del SR''_i lungo l'asse x''_{i-1} dell'angolo α_i .

Una volta stabilito il SR_0 , si segue un processo iterativo nei quali si vanno a identificare a cascata i successivi sistemi di riferimento.

Tenendo in considerazione che tutti i SR saranno destrorsi, per definire il SR_i , costruito a partire dal SR_{i-1} è necessario considerare tre tipi di configurazione.

1) z_{i-1} e z_i non sono complanari: in questo caso esiste un unico segmento perpendicolare ad entrambi gli assi z_{i-1} e z_i , questo segmento rappresenta la direzione lungo il quale viene minimizzata la distanza tra i due assi.

La linea contenente questa normale comune a z_{i-1} e z_i definisce x_i ed il punto che interseca z_i rappresenta l'origine O_i .

2) z_{i-1} è parallelo a z_i : in questo caso esistono infinite normali che minimizzano la distanza tra i due assi, questa condizione non specifica x_i completamente. In questo caso è possibile collocare l'origine O_i arbitrariamente lungo z_i , lo stesso vale per l'asse x_i .

O_i viene generalmente scelto in modo coerente con la geometria del *link* preso in considerazione mentre l'asse x_i spesso viene scelto parallelo all'asse x_{i-1} e verso scelto arbitrariamente.

3) z_{i-1} interseca z_i : in questo caso è scelto x_i ortogonale al piano generato da z_{i-1} e z_i , il suo verso può essere scelto in modo arbitrario. L'origine O_i sarà coincidente con il punto d'intersezione tra i due assi z_{i-1} e z_i .

In tutti e tre i casi l'asse y_i sarà vincolato a seguito della definizione di z_i e x_i in quanto la terna scelta per tutti i SR è destrorsa. Per sintetizzare, dati due assi giunto i ed $i+1$, si identifica l'origine del sistema di riferimento i , come quel punto appartenente all'asse $i+1$ avente distanza minima rispetto all'asse i . Nel caso di assi paralleli, esistendo infiniti punti appartenenti alla retta $i+1$ equidistanti dalla retta i , la scelta del punto diventa arbitraria.

Una volta identificati i sistemi di riferimento dei *link*, i parametri di DH a_i , α_i , d_i e θ_i , saranno misurabili come distanze ed angoli tra gli assi dei due sistemi di riferimento SR_i e SR_{i-1} .

Il robot preso in analisi, UR5 presenta degli specifici parametri di DH riportati in Tabella 3.1.

Parametri Cinematici	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
a [m]	0	-0.425	-0.39225	0	0	0
α [rad]	$\pi/2$	0	0	$\pi/2$	$-\pi/2$	0
d [m]	0.089159	0	0	0.10915	0.09465	0.0823
θ [rad]	0	0	0	0	0	0

Tabella 3.1: Parametri di DH standard teorici forniti dalla Universal Robots del robot UR5 [43]

I parametri di DH riportati rappresentano la geometria del robot teorico. In realtà ogni robot presenta piccole modifiche a questi parametri teorici, il processo produttivo porta con sé degli errori che non permettono l'ottenimento di un oggetto reale che combacia perfettamente con il modello teorico.

Sempre dal sito della Universal Robots è presente la rappresentazione dei sistemi di riferimento dei *link* dalla base al TCP (Figura 3.3).

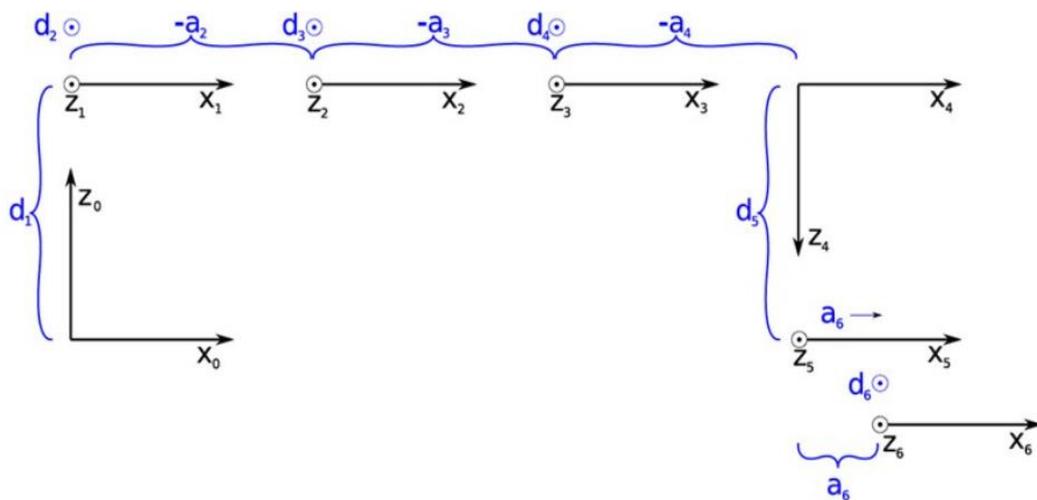


Figura 3.3: Rappresentazione SR robot UR5 forniti dalla Universal Robots [43]

Per comodità di rappresentazione la Universal Robots ha tracciato i sistemi di riferimento con gomito e spalla ruotati di 180°. La conformazione dei sistemi di

riferimento in assenza di attuazioni dei giunti è come espresso nella Figura 3.4 dove è rappresentato il modello 3D del robot, composto dai suoi vari *link* e sistemi di riferimento.

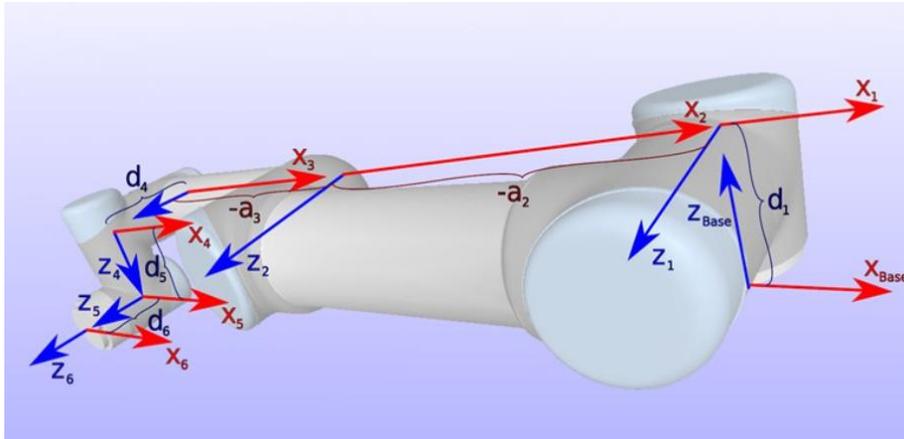


Figura 3.4 Modello 3D del robot UR5 e rappresentazione dei SR dei *link* [43]

I soli parametri di DH del robot non sono sufficienti a garantire un'accuratezza del TCP soddisfacente; infatti, ogni robot deve essere calibrato per poter compensare le piccole differenze geometriche che si presentano nel robot reale rispetto al modello teorico esente da imperfezioni.

A tal proposito il robot UR5 presente in laboratorio è stato calibrato dalla Universal Robots. È stato possibile estrarre dai file di sistema del manipolatore, le modifiche ai parametri cinematici del fornitore dal costruttore.

	Giunto 1*	Giunto 2*	Giunto 3*	Giunto 4*	Giunto 5*	Giunto 6*
Δa [mm]	0,110	315,002	17,123	0,025	-0,069	0
$\Delta \alpha$ [°]	-0,095	0,014	-0,243	-0,034	-0,013	0
Δd [mm]	-0,075	1731811,97	-1704890,05	-26921,15	0,229	0,194
$\Delta \theta$ [°]	0,003	74,987	-91,922	16,896	0,009	-0,006

Tabella 3.2: Modifiche ai parametri di DH standard del robot UR5 fornite dalla Universal Robot

Da come si può osservare vengono apportate delle modifiche completamente fuori scala rispetto ai parametri di DH originali. Ad esempio, le modifiche dei parametri d_2 e d_3 sono di quattro ordini di grandezza maggiori rispetto a d_4 che è il parametro

di DH più grande del robot non calibrato. Queste forti differenze di ordine di grandezza non hanno senso fisico, non permettono di utilizzare un modello di cinematica inversa e non sono molto utili a fini prognostici.

3.2 Convenzione di Hayati-Mirmirani e GDH

La convenzione di DH standard ha portato la possibilità di creare un modello di cinematica diretta semplice ed intuitivo, ma al suo interno esistono delle criticità in alcune particolari configurazioni.

L'esigenza di un modello alternativo nasce proprio per soddisfare le criticità esistenti in fase di calibrazione quando si cerca di identificare gli errori dei parametri di DH presenti sul robot reale che si discostano da quello teorico.

Sapendo come vengono assegnati i parametri di DH ad un *link* i , quando due assi sono tra loro paralleli la distanza di offset d_i diventa arbitraria e varierà in funzione di dove viene scelta l'origine del *link* i ; l'attribuzione arbitraria, porta con sé una serie di problematiche in fase di calibrazione.

Il modello di DH in fase di calibrazione, quando si considerano giunti paralleli, porta con sé una forte instabilità numerica in fase di identificazione dei parametri cinematici; infatti, la stragrande maggioranza delle ricerche elencate in Tabella 2.1 cita o utilizza il modello che poggia sul lavoro proposto da Hayaty [7] [20] [22] [25]. Alcuni autori in presenza di *link* paralleli, citano Hayati ma procedono senza l'ausilio di questo modello, cercando di aggirare l'instabilità, non riuscendo a identificare tutte le modifiche ai parametri cinematici del sistema, in quanto si ottiene un sistema sovradeterminato e ridondante [24][28][25]. Nel paragrafo 3.2.1 viene analizzata geometricamente questa instabilità, mentre nel paragrafo 4.1.2 si mostrano gli effetti numerici di un sistema numerico sovradeterminato.

Hayati spiega che piccoli errori del TCP non possono essere modellati da piccoli errori nei parametri di DH del *link* in presenza di due *link* paralleli o quasi paralleli, questo causa instabilità nel processo di identificazione.

Il modello presentato da Hayati prevede la modifica delle trasformazioni omogenee usate nel modello di DH standard per descrivere i *link* paralleli. Rappresentato nella Figura 3.5 il modello di Hayati presenta la seguente trasformazione omogenea:

$$A_i^{i-1} = R_{z,\theta_i} T_{x,a_i} R_{x,\alpha_i} R_{y,\beta_i}$$

$$= \begin{bmatrix} c_{\theta_i} c_{\beta_i} - s_{\theta_i} s_{\alpha_i} s_{\beta_i} & -s_{\theta_i} c_{\alpha_i} & c_{\theta_i} s_{\beta_i} + s_{\theta_i} s_{\alpha_i} c_{\beta_i} & a_i c_{\theta_i} \\ s_{\theta_i} c_{\beta_i} + c_{\theta_i} s_{\alpha_i} s_{\beta_i} & c_{\theta_i} s_{\alpha_i} & s_{\theta_i} s_{\beta_i} - c_{\theta_i} s_{\alpha_i} c_{\beta_i} & a_i s_{\theta_i} \\ -c_{\alpha_i} s_{\beta_i} & s_{\alpha_i} & c_{\alpha_i} c_{\beta_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

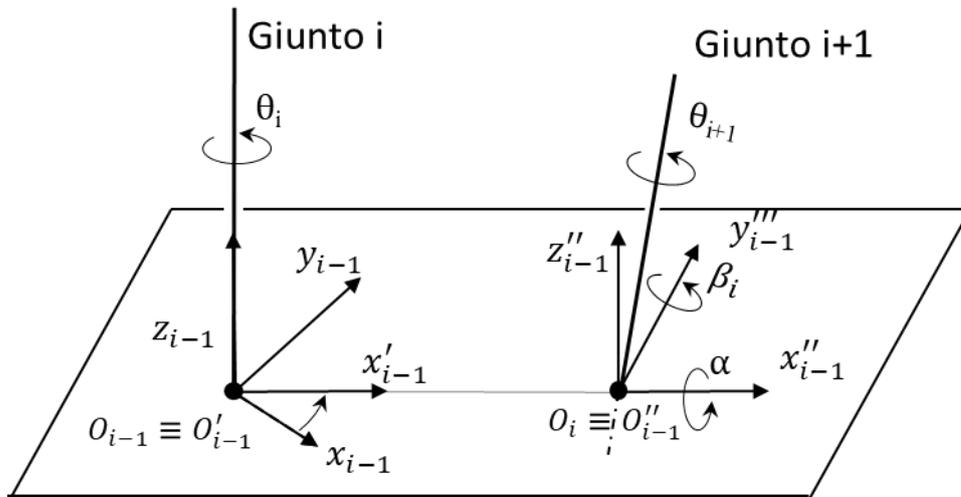


Figura 3.5: Disposizione assi giunto convenzione di Hayati

Rispetto al precedente DH, si elimina la traslazione di offset d_i lungo z_{i-1} dato che è "arbitraria" e si introduce una rotazione di β_i lungo l'asse di rotazione y (precedentemente rototraslato). Le varie trasformazioni omogenee si evolvono in vari sistemi di riferimento intermedi:

- il SR'_i si ottiene attuando una rotazione del SR_i lungo l'asse z_{i-1} dell'angolo θ_i ;
- il SR''_i mediante la traslazione del SR'_i lungo l'asse x'_{i-1} della distanza a_i ;
- il SR'''_i attraverso la dalla rotazione del SR''_i lungo l'asse x''_{i-1} dell'angolo α_i ;
- il SR_{i+1} infine sarà dato dalla rotazione del SR'''_i lungo l'asse y'''_{i-1} dell'angolo β_i .

L'angolo β_i come α_i viene detto angolo di *twist* ed esprime l'angolo da z_{i-1} a z_i misurato rispetto a y'''_i . L'asse y'''_i è un'asse posizionato su di un sistema di

riferimento intermedio e rappresenta l'asse y rototraslato dalle trasformazioni omogenee effettuate prima della rotazione lungo l'asse y.

Questa convenzione ha la proprietà che piccoli errori di posizionamento ed orientamento dei *link* paralleli, possono essere l'identificati attraverso piccole variazioni dei parametri di DH.

Per assegnare i parametri di cinematici secondo questa convenzione non si dovrà più seguire lo schema dato da DH.

Hayati propone altre regole per identificare i nuovi parametri cinematici, composta da quattro step.

- 1) Si traccia il piano ortogonale all'asse z_{i-1} . L'intersezione di questo piano con l'asse z_i rappresenta l'origine O_i (l'origine del frame i-esimo).
- 2) Si ruota il SR i-1 lungo l'asse z_{i-1} per allineare x_{i-1} con la linea che connette le due origini O_{i-1} e O_i .
- 3) Si trasla l'origine del sistema precedentemente ruotato verso O_i .
- 4) Si eseguono due rotazioni del sistema ottenuto al punto tre lungo gli assi x e y fino ad ottenere l'asse z_i

Attraverso il set di quattro trasformazioni omogenee che geometricamente sono rappresentabili dai quattro step di Hayati, è possibile modellare dei *link* ad assi paralleli ed è possibile identificare in fase di calibrazione delle piccole modifiche dei parametri cinematici.

Questo modello va adoperato solo sui *link* che presentano i giunti paralleli, per gli altri *link* rimane valida la classica convenzione di DH.

Questa convenzione ha posto una pietra miliare nella storia della calibrazione; infatti, viene citata nella maggioranza dei lavori raccolti nella Tabella 2.1 e delle revisioni incontrate durante lo sviluppo di questa tesi [15][11].

Anche gli studi più recenti tuttora citano Hayati e fanno uso della rotazione attorno all'asse y per descrivere in modo più efficiente i *link* aventi due assi paralleli [25] [44]. Le trasformazioni di Hayati non sono da definirsi una convenzione obbligatoria o fissa, ogni studio fa un uso differente della rotazione lungo y.

Ad esempio, quando viene utilizzata la convenzione di DH modificata (non approfondita in questa tesi), le trasformazioni omogenee hanno quest'ordine.

$$A_i^{i-1} = R_{x,\alpha_i} T_{x,a_i} R_{y,\beta_i} R_{z,\theta_i} \quad (3.6)$$

Ad un ordine diverso va una interpretazione geometrica leggermente diversa, ma i risultati rimangono gli stessi: per poter descrivere piccole oscillazioni dei parametri cinematici di due giunti ad assi paralleli è necessario introdurre una rotazione lungo l'asse y.

Seppur il valore di d_i è arbitrario nella convenzione di DH, per i *link* paralleli, è molto comune trovare dei valori $d_i = 0$ in letteratura. Ciò non esclude che possano esistere dei manipolatori, con una disposizione dei *link* tale da rendere conveniente la scelta di un $d_i \neq 0$ in questi *link*, per descrivere in modo efficace il sistema fisico reale. La convenzione di Hayati elimina la traslazione di offset d_i perdendo così un parametro che potrebbe essere descrittivo di un eventuale manipolatore preso in analisi.

A questo proposito altre pubblicazioni hanno introdotto un altro modello più completo tale da non perdere eventuali informazioni dal forte significato fisico.

Il modello di DH standard e quello di Hayati possono essere combinati in un'unica forma che è immune al problema numerico associato alla convenzione classica di DH. Questo modello, chiamato modello GDH (*General DH form*) ha il vantaggio di poter descrivere tutti i *link* del manipolatore seriale in modo indistinto[45]. Le trasformazioni omogenee che compongono questo modello sono:

$$A_i^{i-1} = R_{z,\theta_i} T_{z,d_i} T_{x,a_i} R_{x,\alpha_i} R_{y,\beta_i} = \begin{bmatrix} c_{\theta_i} c_{\beta_i} - s_{\theta_i} s_{\alpha_i} s_{\beta_i} & -s_{\theta_i} c_{\alpha_i} & c_{\theta_i} s_{\beta_i} + s_{\theta_i} s_{\alpha_i} c_{\beta_i} & a_i c_{\theta_i} \\ s_{\theta_i} c_{\beta_i} + c_{\theta_i} s_{\alpha_i} s_{\beta_i} & c_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\beta_i} - c_{\theta_i} s_{\alpha_i} c_{\beta_i} & a_i s_{\theta_i} \\ -c_{\alpha_i} s_{\beta_i} & s_{\alpha_i} & c_{\alpha_i} c_{\beta_i} & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.7)$$

In pratica si va ad aggiungere alle quattro trasformazioni di DH standard (3. 2) una rotazione lungo y.

Il modello è stato proposto anche con altri ordinamenti delle trasformazioni omogenee ma quest'ordine è stato selezionato tra tanti perché rimane il più aderente all'originale convenzione di Hayati. L'unica differenza tra la (3. 5) e la (3. 7) si presenta nell'elemento (3,4) delle due matrici omogenee.

Infatti, quando $d_i=0$ nei *link* paralleli, il modello GDH è identico alla convenzione di Hayati, mentre per tutti gli altri *link*, dove $\beta_i=0$ si otterrà esattamente la convenzione di DH standard. Con GDH non c'è alcuna perdita di informazioni, anzi

c'è la possibilità di poter mantenere all'occorrenza $d_i \neq 0$ anche con dei *link* paralleli, informazione che altrimenti andrebbe persa con Hayati.

Questo modello può essere applicato a tutti i *link*, a patto che vengano vincolati in modo opportuno i valori di d_i e di β_i , rispettivamente nei *link* paralleli e non paralleli durante l'identificazione dei parametri cinematici. In fase di calibrazione con quest'ultima convenzione, devono essere identificate solo 4 variazioni dei parametri cinematici per *link*, seppur esistano 5 parametri cinematici a descrivere ciascun *link*. In caso contrario si otterrebbe un sistema mal condizionato e ridondante.

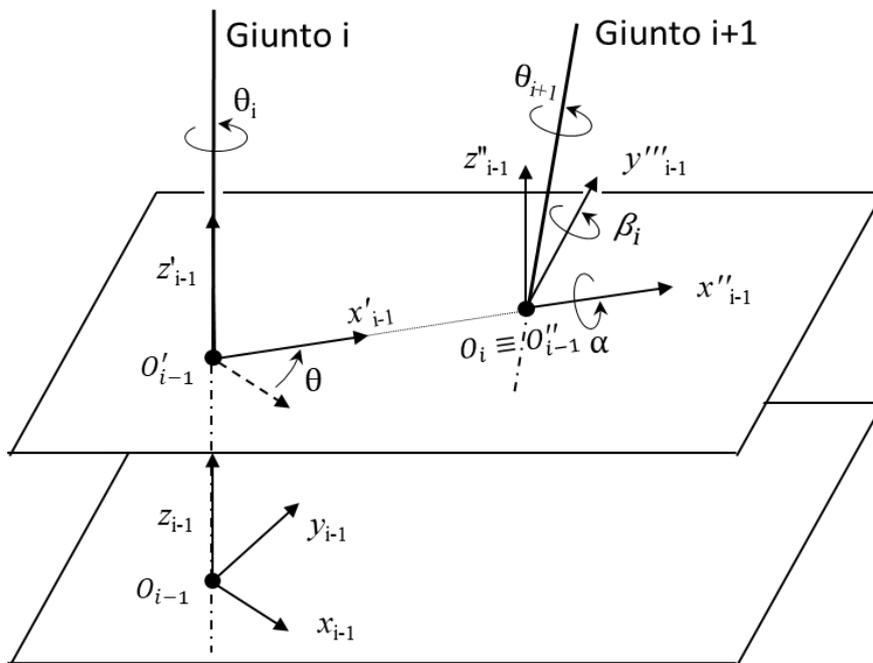


Figura 3.6: Disposizione assi giunto convenzione GDH

I sistemi di riferimento di questo modello saranno:

- il SR'_i si ottiene attuando una rototraslazione del SR_i lungo l'asse z_{i-1} attraverso i parametri θ_i e d_i ; il SR''_i mediante la traslazione del SR'_i lungo l'asse x'_{i-1} della distanza a_i ;
- il SR''_i mediante la traslazione del SR'_i lungo l'asse x'_{i-1} della distanza a_i ;
- il SR'''_i attraverso la dalla rotazione del SR''_i lungo l'asse x''_{i-1} dell'angolo α_i ;
- il SR_{i+1} infine sarà dato dalla rotazione del SR'''_i lungo l'asse y'''_{i-1} dell'angolo β_i .

La modifica da apportare ai quattro step di Hayati per ottenere 5 parametri nei *link* paralleli consisterebbe nell'aggiungere uno step. Questo risulterà la traslazione del SR_{i-1} lungo l'asse z_{i-1} , della distanza d_i fissata, sino a raggiungere l'origine O'_{i-1} . Successivamente si potrà procedere con i 4 punti elencati da Hayati, considerando il SR_{i-1} ma traslato sull'origine O'_{i-1} .

In questo modo si andrà a tracciare un offset del piano che identifica l'origine O_i spostandola nel SR O_{i-1} , ottenendo un punto con maggiore senso fisico e qui si applicheranno i quattro step di Hayati.

Ricapitolando per le analisi di questa tesi verranno utilizzate le seguenti convenzioni:

- 1) DH standard: per tutti i *link* non paralleli (Figura 3.2).
- 2) Hayati: per i *link* paralleli (Figura 3.5).
- 3) GDH: per poter rappresentare allo stesso modo tutti i *link* del manipolatore (Figura 3.6).

Come si può notare da dalla Figura 3.3 gli assi z_1 , z_2 e z_3 sono paralleli; infatti, osservando i parametri cinematici del braccio del robot in Tabella 3.1 si hanno per il *link* 2 e 3, d ed α nulli.

Per una corretta interpretazione fisica delle geometrie di questi due *link* sarà necessario introdurre degli angoli β quando si andrà a considerare i SR_2 e SR_3 .

Dato che i valori di d sono fissati a zero, la convenzione di Hayati e quella GDH saranno perfettamente equivalenti.

3.2.1 Confronto grafico tra DH Standard e GDH/Hayati

Unendo i parametri di DH standard del robot e le modifiche ai parametri di DH identificati dalla UR, si ottengono i parametri del robot reale secondo la convenzione di DH standard fornito dalla calibrazione del costruttore.

	Giunto 1*	Giunto 2*	Giunto 3*	Giunto 4*	Giunto 5*	Giunto 6*
a [m]	0,00011	-0,11000	-0,37513	0,00003	-0,00007	0
α [rad]	0,4995 π	0,000	-0,004	0,4998 π	-0,5001 π	0
d [m]	0,08908	1731,812	-1704,890	-26,812	0,09488	0,08249
θ [rad]	0,0001	0,417 π	-0,511 π	0,094 π	0,0002	-0,0001

Tabella 3.3: Parametri di DH standard del robot UR5 calibrato dalla UR senza senso fisico

L'asterisco nella Tabella 3.3 specifica il fatto che ci si sta riferendo ad i parametri identificati dal costruttore.

Prendendo come esempio il *link* due del robot trattato, rappresentandolo graficamente secondo la convenzione di DH standard (Figura 3.7).

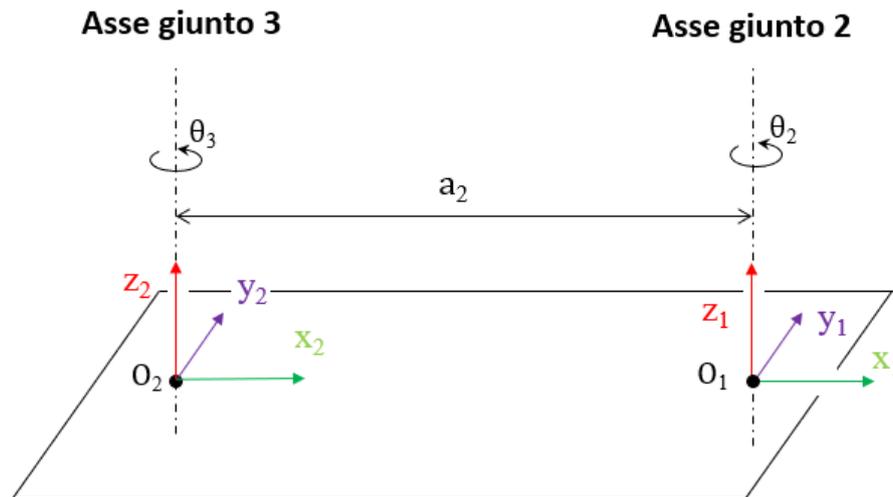


Figura 3.7: Parametri di DH standard teorici del *link* 2 del UR5

Osservando i parametri del robot teorico Tabella 3.1 si ha che gli assi z_1 e z_2 sono tra loro paralleli. Questa situazione si presenta se α_i assume un valore pari a 0 o $\frac{\pi}{2}$ e inoltre $a_i \neq 0$ altrimenti i due assi sarebbero coincidenti.

Dovremmo aspettarci piccole modifiche ai parametri di DH e piccoli spostamenti (errori di montaggio e di produzione) della posizione dei giunti dei *link*.

Se cerchiamo di rappresentare i SR per il *link* 2 con le modifiche ad i parametri di DH ottenute dalla calibrazione del costruttore Tabella 3.3 otterremo una condizione completamente diversa.

La Figura 3.8 ha scopo rappresentativo ed è fuori scala; confrontando i valori dei parametri di DH del giunto 2 risulta evidente la forte differenza tra il *link* 2 del robot non calibrato ed il *link* 2* calibrato dalla UR; infatti, il *link* 2 vede passare la distanza di offset da un valore $d_2 = 0$ ad un valore $d_{2^*} = 1731,81$ m a seguito della calibrazione.

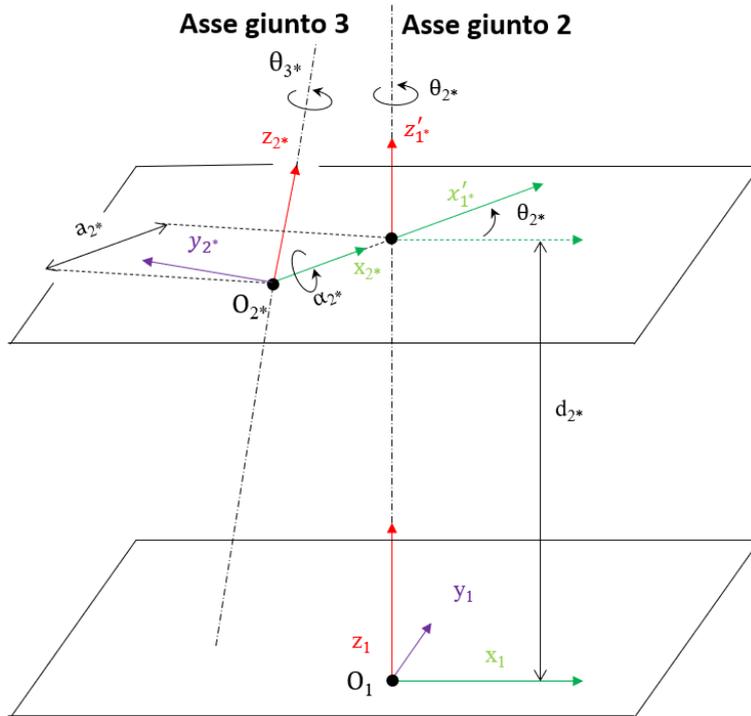


Figura 3.8 Parametri di DH standard identificati dalla UR del *link* 2 del UR5

Per comprendere questa forte differenza tra i parametri di DH tra il robot identificato ed il robot reale è sufficiente analizzare un caso semplificato. Assumendo che esista un errore sul *link* 2 tale da generare una piccola rotazione lungo y_2 , questo genererà un disallineamento dell'asse z_2 . Utilizzando come modello quello fornito da Hayati si otterrebbe una configurazione come in Figura 3.9.

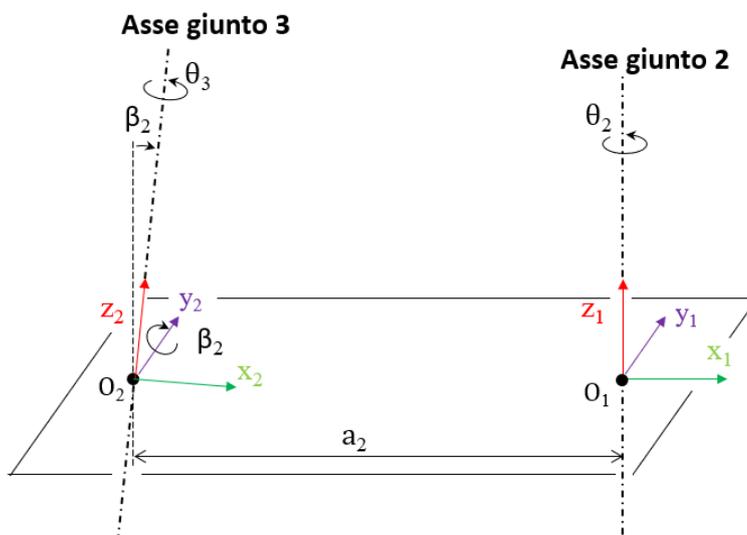


Figura 3.9: *Link* 2 con disallineamento di rotazione lungo l'asse y_2 , convenzione Hayati

Adottando invece la convenzione di DH si otterrà la configurazione rappresentata in Figura 3.10.

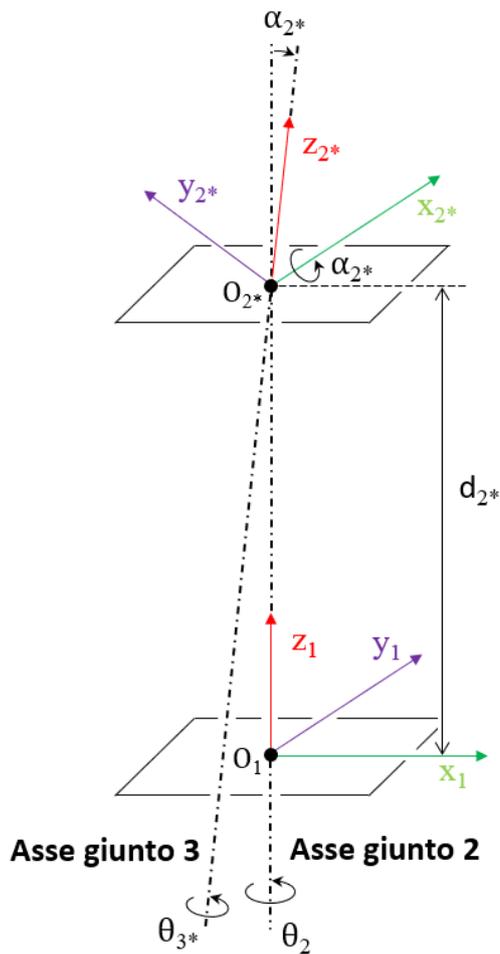


Figura 3.10 : Esempio *Link 2** con disallineamento di rotazione lungo l'asse y_2 , convenzione DH standard

L'origine del *link* O_{2^*} si sposta perdendo di senso fisico, gli assi x_{2^*} e y_{2^*} si orientano in modo tale da permettere di orientare l'asse z_{2^*} in modo corretto, ossia secondo l'asse giunto 3. Si è passa da una configurazione a *link* paralleli, dove esistono infiniti punti aventi distanza minima tra i due assi, ad una configurazione a *link* quasi paralleli. In questa configurazione, l'origine O_{2^*} deve essere quel punto appartenente all'asse giunto 3 situato ad una distanza minima rispetto all'asse giunto 2.

Dato che non è possibile ruotare direttamente lungo l'asse y_2 , bisogna traslare fino al punto di minima distanza tra i due assi, ruotare l'asse x_1 di 90° e successivamente ruotare lungo x dell'angolo α_2 che assumerà la funzione della rotazione β_2 .

Entrambi i metodi permettono di identificare l'asse di rotazione corretto come si può vedere dalla Figura 3.11.

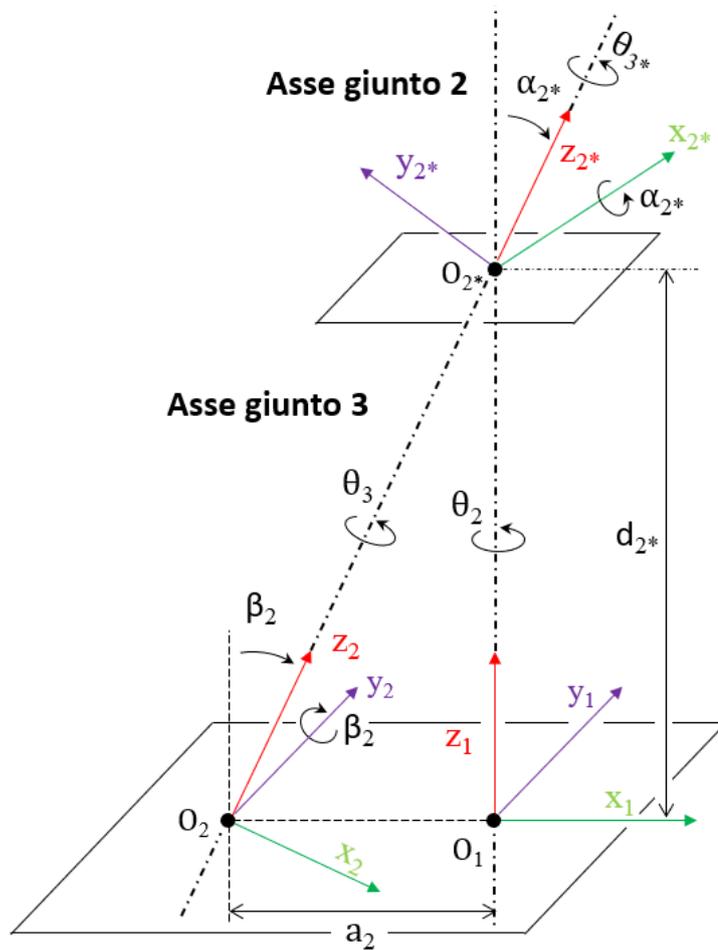


Figura 3.11 : Esempio *Link 2* con disallineamento di rotazione lungo l'asse y_2 , convenzione Hayati e DH standard

Si può osservare graficamente come, per avere piccole modifiche ai parametri di DH, si renda necessario aggiungere/considerare una rotazione attorno all'asse y .

La necessità di considerare la rotazione lungo l'asse y sarà necessaria anche per il *link 3*, dato che anch'esso è parallelo al *link 2*.

Il *link 6*, seppur l'asse z_6 parallelo all'asse z_5 non ha bisogno di questa convenzione, in quanto i due assi sono coincidenti. Una piccola rotazione lungo y non implicherebbe uno spostamento dell'origine del SR_6 in un punto senza senso fisico.

3.3 Proiezione geometrica SR UR5

Come detto nel paragrafo 2.2.3 è possibile identificare le equazioni degli assi giunto del robot, z_i nello spazio in modo geometrico. Partendo dall'ipotesi che la calibrazione fornita dal costruttore è in grado di identificare il corretto orientamento degli assi giunto nello spazio, seppur alcuni centri giunto siano posizionati in posizioni senza alcun senso fisico, si è provato a descrivere il sistema robotico mediante la convenzione GDH usando come base gli assi identificati dal costruttore.

Praticamente si è provato a riscrivere il sistema di riferimento di DH secondo la convenzione GDH.

Da notare che GDH per il nostro manipolatore ingloba le convenzioni di DH e Hayati.

Per i *link* 1,4,5 e 6 la convenzione di DH o di GDH è equivalente; in questi *link* l'ultima trasformazione β_i può essere omessa dato che si assume sia pari a zero.

Per i *link* 2 e 3 per la proiezione dei sistemi di riferimento, si è considerato un passaggio da DH a GDH; dato che i modelli di questa famiglia UR, come nella stragrande maggioranza dei robot incontrati in letteratura, d_i è nullo nei *link* paralleli, Hayati e GDH sono perfettamente sovrapponibili. Basta omettere la trasformazione d_i per questi *link* per poter passare da GDH ad Hayati.

La proiezione si è preferito farla mediante la convenzione GDH in modo da conservare la generalità di rappresentazione di questa convenzione. La routine MATLAB che è stata scritta e che permette il passaggio da un sistema di coordinate all'altro può essere utilizzata anche in altri robot, e varrà anche qualora d_i sia diversa da zero nei *link* paralleli.

Link per link verranno descritte le operazioni per poter calcolare i parametri GDH.

3.3.1 Introduzione degli strumenti utili all'identificazione GDH

È necessario prima fare alcune considerazioni utili all'identificazione dei nuovi parametri cinematici, innanzitutto bisogna approfondire il significato geometrico dei sistemi di riferimento intermedi nelle tre convenzioni precedentemente introdotte.

Le 5 trasformazioni che caratterizzano il SR GDH note dall'eq (3. 7) possono essere trattate come un set di due rototraslazioni (3. 8) e (3. 9) ed una successiva rotazione lungo β_i , seppur le trasformazioni sono in sequenza e definite secondo un preciso ordine, alcune di queste sono indipendenti l'una dall'altra.

Quando si ha una rotazione e, successivamente, una traslazione lungo una stessa direzione, da un punto di vista geometrico, l'ordine delle due trasformazioni non cambia il risultato.

La prima rototraslazione avviene lungo l'asse z e la rotazione e la traslazione lungo lo stesso asse sono tra loro indipendenti.

$$RT_z(\theta_i, d_i) = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 0 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = R_{z,\theta_i} T_{z,d_i} \equiv T_{z,d_i} R_{z,\theta_i} \quad (3. 8)$$

Come si può osservare dalla Figura 3.12, a seguito della prima rototraslazione si identifica il sistema di riferimento intermedio SR'_i che si tratti di convenzione di DH o di GDH. Nel caso di Hayati la rototraslazione RT_z consiste in una semplice rotazione in quanto non esiste la traslazione lungo d_i . Questo sistema di riferimento è rappresentato chiaramente nelle figure Figura 3.2, Figura 3.5 e Figura 3.6.

La seconda rototraslazione avviene lungo l'asse x identifica l'origine del SR_i ed anche in questo caso non è rilevante l'ordine delle trasformazioni ai fini dell'identificazione dell'origine del SR.

$$RT_{x_{i-1}}(a_i, \alpha_i) = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_{x,a_i} R_{x,\alpha_i} \equiv R_{x,\alpha_i} T_{x,a_i} \quad (3. 9)$$

Infine, l'ultima rotazione R_{y,β_i} serve a correggere eventuali disallineamenti in presenza di *link* paralleli.

Il SR''_{i-1} si ottiene traslando il SR'_{i-1} lungo x della quantità a_i ed è comodo ai fini dell'identificazione geometrica (Figura 3.2, Figura 3.5 e Figura 3.6).

In questo modo si sarà identificata l'origine del SR_i e rimarranno solo due rotazioni tali da permettere l'allineamento dell'asse z''_{i-1} verso z_i .

Il passaggio da SR''_{i-1} a SR_i avviene mediante due rotazioni per GDH /Hayati ed una sola rotazione lungo x per la convenzione DH.

La trasformazione (3. 7) può quindi essere riscritta, grazie alle rototraslazioni (3. 8) e (3. 9) senza perdere il significato geometrico e matematico delle trasformazioni che la compongono, si otterrà un diverso ordine di rotazioni e traslazioni.

$$A_i^{i-1} = RT_{z_i}(\theta_i, d_i)RT_{x_i}(a_i, \alpha_i)R_{y,\beta_i} = T_{z,d_i}R_{z,\theta_i}T_{x,a_i}R_{x,\alpha_i}R_{y,\beta_i} \quad (3. 10)$$

E sulla base di questo riordinamento delle trasformazioni, si andranno a collocare i SR intermedi SR' e SR'' schematizzati in Figura 3.12.

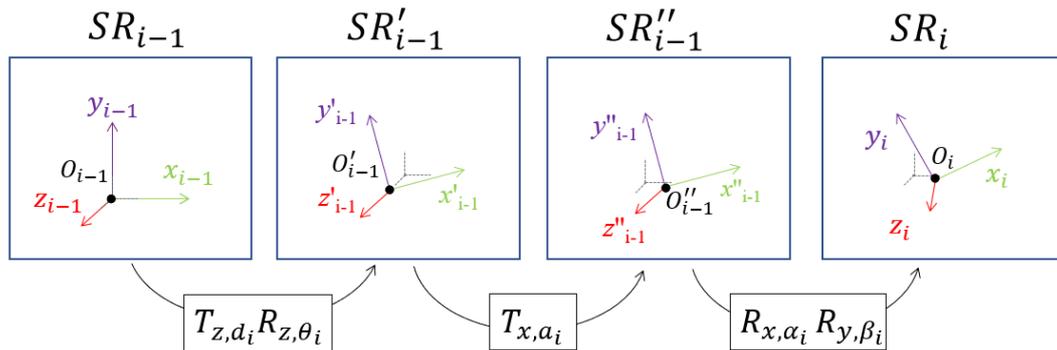


Figura 3.12: Riepilogo dei sistemi di riferimento intermedi e delle trasformazioni omogenee

Il SR'''_{i-1} presente nelle Figura 3.5 e Figura 3.6 è utile per mostrare la rotazione β_i

In presenza di link paralleli ma non è interessante ai fini dell'identificazione geometrica dei parametri cinematici. Come si è potuto già notare dalla Figura 3.8 e i SR dei *link* rappresentati attraverso la convenzione di DH standard che integrano le correzioni dei parametri di DH forniti dal costruttore sono contraddistinti da un asterisco. Anche la Figura 3.11 usa gli asterischi per trattare l'ipotetica configurazione che verrebbe identificata dal costruttore. Il SR_{i^*} contrassegnato dall'asterisco andrà quindi a rappresentare il sistema di riferimento generato dalla convenzione di DH standard che utilizza le modifiche ai parametri DH forniti dal costruttore.

All'interno delle coordinate dei frame dei SR_{i^*} saranno contenute le informazioni che descrivono gli assi giunto identificati dalla Universal Robots.

Imponendo la posa (3. 11) ad i parametri di DH identificati dal costruttore (Tabella 3.3), si otterrà la configurazione dei sistemi di riferimento con il quale verrà effettuata la proiezione e l'identificazione cinematica.

$$q_i = 0 \quad \forall i = 1:6 \quad (3. 11)$$

Per conoscere i sistemi di riferimento dei *link* nello spazio basta calcolare le matrici di trasformazione omogenea, utilizzando l'Eq (3. 2) per ciascun link.

Si ottengono le trasformazioni omogenee dei SR_{i^*} ossia dei *link* ottenuti mediante i parametri aggiornati forniti dalla Universal Robots.

$$A_{1^*}^0, A_{2^*}^1, A_{3^*}^2, A_{4^*}^3, A_{5^*}^4, A_{6^*}^5 \quad (3. 12)$$

Questi possono essere espressi rispetto all'origine O_0 attraverso l'Eq. (3. 4):

$$A_{1^*}^0, A_{2^*}^0, A_{3^*}^0, A_{4^*}^0, A_{5^*}^0, A_{6^*}^0 \quad (3. 13)$$

Attraverso queste matrici è possibile estrarre l'informazione dei vari assi giunto nello spazio.

I $SR_{\underline{i}}$ identificati da un trattino al di sotto del numero del *link*, rappresentano il modello teorico del robot, costruito attraverso le informazioni fornite dal sito della Universal Robots.

Imponendo la stessa posa (3. 11) ai parametri teorici del robot (Tabella 3.1) e calcolando le trasformazioni omogenee utilizzando le equazioni (3. 2) e (3. 4) si ottengono le trasformazioni omogenee del robot senza modifiche:

$$A_{\underline{1}}^0, A_{\underline{2}}^1, A_{\underline{3}}^2, A_{\underline{4}}^3, A_{\underline{5}}^4, A_{\underline{6}}^5 \quad (3. 14)$$

$$A_{\underline{1}}^0, A_{\underline{2}}^0, A_{\underline{3}}^0, A_{\underline{4}}^0, A_{\underline{5}}^0, A_{\underline{6}}^0 \quad (3. 15)$$

I SR_{i^*} e i $SR_{\underline{i}}$ e le loro matrici omogenee serviranno a identificare i SR_i . Questi sono i sistemi di riferimento incogniti dei quali si cerca di identificare i parametri nella convenzione GDH. Da notare che non si usa nessun pedice particolare o asterisco per identificarli; verrà utilizzato solo il numero del link alla quale saranno riferiti.

In Figura 3.13 sono presenti, per la posa selezionata (3. 11), i SR_{i^*} trovati dal costruttore mediante la sua calibrazione e i $SR_{\underline{i}}$ del robot in assenza di modifiche ai parametri di DH. Con le frecce nere è rappresentato il robot teorico ed in rosso il robot calibrato dal costruttore senza senso fisico. In azzurro invece sono mostrati i SR_i incogniti nella convenzione GDH. I SR_2^* e SR_3^* non hanno senso fisico nello spazio e non sono facilmente rappresentabili in una vista 2D. Invece, i *link* 1, 4, 5 e 6 del SR_{i^*} , come verrà analizzato nel paragrafo 3.3.2 dovrebbero essere perfettamente sovrapponibili con i $SR_{\underline{i}}$ in azzurro; inoltre, dato che ci si aspetta delle piccole modifiche ai parametri cinematici dei SR_i rispetto ai $SR_{\underline{i}}$ ci si aspetta variazioni tra i due tipi di sistemi di riferimento molto piccole.

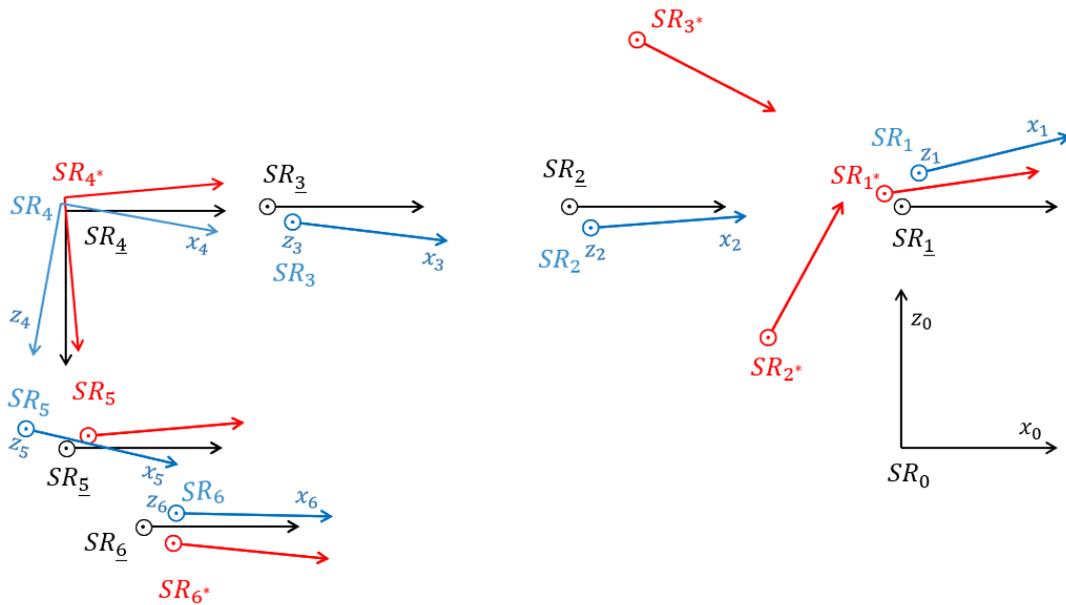


Figura 3.13: Sistemi di riferimento proiezione geometrica

I sistemi di riferimento rappresentati Figura 3.13 sono stati volutamente tracciati non sovrapposti e con delle differenze di orientamento particolarmente enfatizzate e fuori scala per maggiore chiarezza grafica. Queste immagini hanno lo scopo di mostrare e identificare i vari sistemi di riferimento e la loro nomenclatura. I SR_i vengono calcolati in modo sequenziale, ossia si parte dal primo *link* e si sale in direzione prossimale al manipolatore fino al TCP (*link* 6). Immaginando di aver ottenuto il SR_{i-1} nel sistema di coordinate GDH, viene conveniente riferire il SR_i^* successivo dato dal costruttore rispetto al precedente SR_{i-1} . In questo caso si otterranno le seguenti trasformazioni omogenee.

$$A_{1^*}^0, A_{2^*}^1, A_{3^*}^2, A_{4^*}^3, A_{5^*}^4, A_{6^*}^5 \quad (3.16)$$

In quest'ultime matrici omogenee rimane conservata l'informazione degli assi giunto identificati dal costruttore ma l'informazione di questi assi sarà riferita al *link* precedente dei quali sono noti i parametri GDH, precedentemente identificato nello spazio.

All'interno delle matrici omogenee (3.16) sono contenute le equazioni degli assi giunti espressi rispetto al *link* precedente, per poter estrarre le equazioni parametriche degli assi giunto dalle matrici. Ad esempio considerando matrice $A_{i^*}^{i-1}$:

$$A_{i^*}^{i-1} = \begin{bmatrix} \mathbf{x}_{i^*} & \mathbf{y}_{i^*} & \mathbf{z}_{i^*} & \mathbf{p}_{i^*} \\ & \mathbf{0} & & 1 \end{bmatrix} \quad (3.17)$$

I vettori d'interesse sono \mathbf{z}_{i^*} e \mathbf{p}_{i^*}

$$\mathbf{z}_{i^*} = \begin{pmatrix} (z_{i^*})_{x_{i-1}} \\ (z_{i^*})_{y_{i-1}} \\ (z_{i^*})_{z_{i-1}} \end{pmatrix} \quad (3.18)$$

$$\mathbf{p}_{i^*} = \begin{pmatrix} (p_{i^*})_{x_{i-1}} \\ (p_{i^*})_{y_{i-1}} \\ (p_{i^*})_{z_{i-1}} \end{pmatrix} \quad (3.19)$$

l'Eq. parametrica dell'asse z_{i^*} rispetto alle coordinate del *link* i-1 sarà:

$$\begin{cases} x_{i-1} = (p_{i^*})_{x_{i-1}} + t(z_{i^*})_{x_{i-1}} \\ y_{i-1} = (p_{i^*})_{y_{i-1}} + t(z_{i^*})_{y_{i-1}} \\ z_{i-1} = (p_{i^*})_{z_{i-1}} + t(z_{i^*})_{z_{i-1}} \end{cases} \quad (3.20)$$

Dove t rappresenta il parametro dell'Eq. parametrica (3.20).

Infine, le matrici omogenee in coordinate GDH si scriveranno A_i^{i-1} .

Terminate le dovute premesse fatte in questo paragrafo, si può procedere all'identificazione dei parametri cinematici in convenzione GDH.

3.3.2 Identificazione dei parametri cinematici GDH

Di seguito le operazioni per poter calcolare la matrice A_i^{i-1} di ciascun *link*.

3.3.2.1 Link 1: Base-Spalla

A_1^0 coincide con il SR A_1^0 , non c'è la necessità di proiettare i sistemi di riferimento. Le modifiche ai parametri di DH coincidono con quelle del costruttore. In coordinate di GDH bisogna considerare una rotazione lungo β_1 nulla, i quattro parametri di DH standard forniti dal costruttore sono da considerarsi aventi senso fisico.

3.3.2.2 Link 2: Spalla-Gomito

Esprimendo le coordinate del SR_2^* rispetto al SR_1 avremo:

$$A_{2^*}^1 = (A_1^0)^{-1} A_{2^*}^0 \quad (3.21)$$

$$A_{2^*}^1 = \begin{bmatrix} (x_{2^*})_{x_1} & (y_{2^*})_{x_1} & (z_{2^*})_{x_1} & (p_{2^*})_{x_1} \\ (x_{2^*})_{y_1} & (y_{2^*})_{y_1} & (z_{2^*})_{y_1} & (p_{2^*})_{y_1} \\ (x_{2^*})_{z_1} & (y_{2^*})_{z_1} & (z_{2^*})_{z_1} & (p_{2^*})_{z_1} \\ \mathbf{0} & & & 1 \end{bmatrix} \quad (3.22)$$

Estraendo dalla matrice (3. 22) le ultime e due colonne si ottengono i vettori \mathbf{z}_2^* e \mathbf{p}_2^* così come nella (3. 18) e (3. 19) da utilizzare nell'Eq. (3. 20) che porterà alla seguente equazione parametrica:

$$\begin{cases} x_1 = (p_2^*)_{x_1} + t(z_2^*)_{x_1} \\ y_1 = (p_2^*)_{y_1} + t(z_2^*)_{y_1} \\ z_1 = (p_2^*)_{z_1} + t(z_2^*)_{z_1} \end{cases} \quad (3. 23)$$

L'equazione parametrica rappresenta l'asse giunto 3 ed è espressa rispetto al SR_1 ; i valori x_1 , y_1 e z_1 si riferiscono alle coordinate prese rispetto a questo SR.

La variabile t rappresenta il parametro dell'Eq. (3. 23).

Nota la retta dobbiamo calcolare i parametri di GDH che sono dati da una serie di roto traslazioni. Per un riscontro grafico riferirsi alla Figura 3.14.

Dal SR_1 , espresso dalle rette x_1 , y_1 e z_1 , la trasformazione omogenea A_1^1 sarà la matrice identità 4x4

Per passare dal SR_1 al SR_2 il cui asse z_2 è appartenente all'asse giunto 3, bisogna passare dai sistemi di riferimento intermedi $SR_{1'}$ e $SR_{1''}$ seguendo la trasformazione omogenea (3. 10). $R_{z,\theta_i}T_{z,d_i} \equiv R_{z,\theta_i}$

Attraverso le prime tre trasformazioni T_{z,d_2} , R_{z,θ_2} e T_{x,a_2} della (3. 10), si passa attraverso il $SR_{1'}$ dal SR_1 al $SR_{1''}$; l'origine di $SR_{1''}$ coincide con l'origine del SR_2 ed apparterrà all'asse giunto 3 di cui conosciamo le equazioni parametriche.

Rappresentando geometricamente le trasformazioni, si effettuerà prima la traslazione lungo d_2 . Dato che il *link* è parallelo al *link* precedente la distanza di offset viene vincolata al valore nominale (d_2). Si sposta in questo modo l'origine del SR da 1 a 1'.

Da ricordare che il valore di d_2 è pari a 0 nel robot analizzato, non si ha propriamente una traslazione lungo z_1 . In Figura 3.14 è stato rappresentato questo parametro di DH in quanto si è scelto di prenderlo in considerazione in modo da rendere l'approccio adatto all'identificazione delle coordinate secondo la convenzione GDH.

A seguito della traslazione si ruota andando ad orientare l'asse $x_{1'}$ in direzione del centro del $SR_{1''}$; infatti la rotazione θ_2 serve ad orientare la direzione lungo il quale verrà effettuata la successiva traslazione a_2 .

Dal primo SR intermedio $SR_{1'}$ si passa $SR_{1''}$ attraverso la traslazione a_2 .

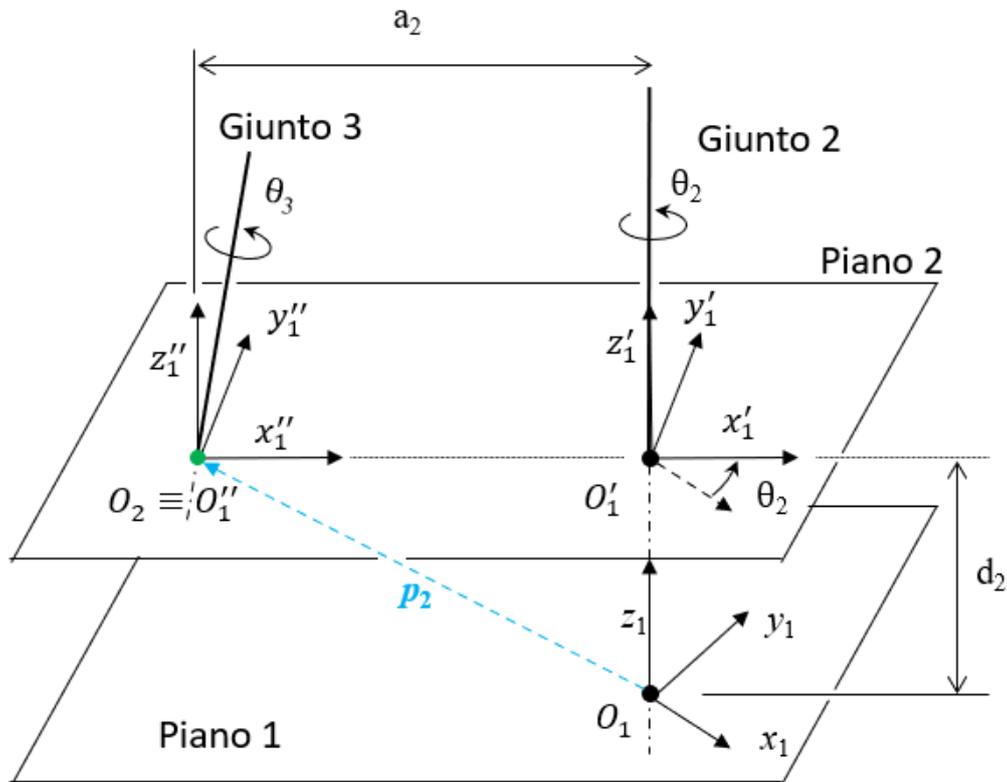


Figura 3.14 Link 2 sistemi di riferimento intermedi

Mostrato il senso geometrico delle prime tre trasformazioni sul *link 2* e nota l'equazione parametrica dell'asse giunto 3, si può procedere all'identificazione dei parametri θ_2 e a_2 (d_2 è già vincolato).

Noto l'asse giunto 3 è possibile identificare subito la posizione dell'origine del SR_2 nello spazio. Bisogna calcolare dalla terza equazione della (3. 23) il valore t^* tale per cui $z_1 = d_2$. Questo punto sarà l'intersezione dell'asse giunto 3 con il Piano 1 generato dagli assi $x_{1'}$ e $y_{1'}$ del $SR_{1'}$ rappresentato in Figura 3.14.

$$t^* = \frac{d_2 - (p_2^*)_{z_1}}{(z_2^*)_{z_1}} \quad (3. 24)$$

Praticamente è come se fosse stato fatto un offset del piano generato dagli assi x_1 e y_1 lungo la direzione z_1 della distanza d_2 dove il punto dell'asse 3 reale, tagliato da questo piano, rappresenta la nuova origine del SR_2 .

Quando nell' equazione parametrica (3. 24) t assumerà il valore t^* , si conoscerà nelle coordinate x_1, y_1 e z_1 del SR_1 l'origine del O_2 rappresentato dal punto verde.

I valori x_1 , y_1 e z_1 rappresentano le tre coordinate del vettore \mathbf{p}_2 , rappresentato in azzurro; questo vettore si andrà a posizionare sulla quarta colonna della matrice omogenea A_2^1 .

$$\mathbf{p}_2 = \begin{pmatrix} (p_2)_{x_1} \\ (p_2)_{y_1} \\ (p_2)_{z_1} \end{pmatrix} \quad (3.25)$$

$$A_2^1 = \begin{bmatrix} \mathbf{x}_2 & \mathbf{y}_2 & \mathbf{z}_2 & \mathbf{p}_2 \\ & \mathbf{0} & & 1 \end{bmatrix} \quad (3.26)$$

I valori di θ_2 e a_2 sono collegati:

a_2 rappresenta la distanza minima tra il centro del punto appena ottenuto

$$\begin{pmatrix} x_1 = (p_2)_{x_1} \\ y_1 = (p_2)_{y_1} \\ z_1 = d_2 \end{pmatrix} \text{ ed il centro del } SR_{1'} \text{ di coordinate } \begin{pmatrix} x_1 = 0 \\ y_1 = 0 \\ z_1 = d_2 \end{pmatrix};$$

θ_2 rappresenta l'angolo che bisogna ruotare per poter orientare x_1 lungo x_1' in direzione delle coordinate della nuova origine del SR2.

Numericamente si avrà:

$$a_2 = \text{sign}((p_2)_{x_1}) \sqrt{(p_2)_{x_1}^2 + (p_2)_{y_1}^2} \quad (3.27)$$

Dato che a_2 è la traslazione lungo x, il suo segno sarà dato dal valore assoluto di x; infatti, ci si aspetta un valore di θ_2 piccolo.

$$\theta_2 = \arctan2\left(\frac{(p_2)_{y_1}}{a_2}, \frac{(p_2)_{x_1}}{a_2}\right) \quad (3.28)$$

Nel calcolo di θ_2 bisogna assolutamente normalizzare i valori di $p_{x_1}^1$ e $p_{y_1}^1$ dividendoli per a_2 in quanto un segno scorretto potrebbe determinare un valore errato dell'angolo θ_2 .

Attraverso la (3.27) e (3.28) si sono ottenuti i primi 3 parametri di GDH.

Applicando le prime tre trasformazioni della (3.10) ci si sposta nel $SR_{1''}$. Per raggiungere il SR_2 mancano le due rotazioni R_{x,α_2} e R_{x,β_2} .

Le due rotazioni andranno ad orientare l'asse z_1'' parallelamente all'asse z_2 coassiale con l'asse giunto3 del quale conosciamo le equazioni parametriche riferite al SR_1 .

Per poter calcolare la rotazione alpha e beta, si riscrive il SR_{2^*} , contenente l'asse reale rispetto al $SR_{2''}$ ricercando la matrice omogenea $A_{2^*}^{1''}$.

$$A_{1''}^0 = A_1^0 T_{z,d_2} R_{z,\theta_2} T_{x,\alpha_2} \quad (3.29)$$

$$A_{2^*}^{1''} = (A_{1''}^0)^{-1} A_2^0 \quad (3.30)$$

$$A_{2^*}^{1''} = \begin{bmatrix} (x_{2^*})_{x_{1''}} & (y_{2^*})_{x_{1''}} & (z_{2^*})_{x_{1''}} & (p_{2^*})_{x_{1''}} \\ (x_{2^*})_{y_{1''}} & (y_{2^*})_{y_{1''}} & (z_{2^*})_{y_{1''}} & (p_{2^*})_{y_{1''}} \\ (x_{2^*})_{z_{1''}} & (y_{2^*})_{z_{1''}} & (z_{2^*})_{z_{1''}} & (p_{2^*})_{z_{1''}} \\ \mathbf{0} & & & 1 \end{bmatrix} \quad (3.31)$$

Si otterranno sempre i vettori come nella matrice (3.22) ma saranno riferiti rispetto al $SR_{1''}$, nelle coordinate $x_{1''}$, $y_{1''}$ e $z_{1''}$.

Del SR_2^* rappresentato in Figura 3.15 i vettori \mathbf{x}_{2^*} , \mathbf{y}_{2^*} e \mathbf{p}_{2^*} non hanno senso fisico; il vettore con significato fisico ai fini della identificazione dei parametri GDH è \mathbf{z}_{2^*} .

$$\mathbf{z}_{2^*} = \begin{pmatrix} (z_{2^*})_{x_{1''}} \\ (z_{2^*})_{y_{1''}} \\ (z_{2^*})_{z_{1''}} \end{pmatrix} \quad (3.32)$$

Il vettore \mathbf{z}_{2^*} coincide con l'orientamento dell'asse giunto 3 riferito rispetto al $SR_{2''}$.

Esprimendo il SR_2 rispetto al $SR_{1''}$ prendendo in considerazione la sottomatrice delle sole rotazioni, si avrà la seguente trasformazione omogenea, data dalle due ultime rotazioni.

$$A_2^{1''} = R_{x,\alpha_2} R_{x,\beta_2} = \begin{bmatrix} c(\beta_2) & 0 & s(\beta_2) \\ s(\alpha_2)s(\beta_2) & c(\alpha_2) & -s(\alpha_2)c(\beta_2) \\ -c(\alpha_2)s(\beta_2) & s(\alpha_2) & c(\alpha_2)c(\beta_2) \end{bmatrix} \quad (3.33)$$

Il terzo vettore colonna di $A_2^{1''}$ rappresenta anch'esso l'orientamento dell'asse giunto 3 riferito rispetto al $SR_{1''}$, e si hanno α_2 e β_2 incogniti.

Vale allora l'uguaglianza:

$$\mathbf{z}_{2^*} = \begin{pmatrix} (z_{2^*})_{x_{1''}} \\ (z_{2^*})_{y_{1''}} \\ (z_{2^*})_{z_{1''}} \end{pmatrix} = \begin{pmatrix} s(\beta_2) \\ -s(\alpha_2)c(\beta_2) \\ c(\alpha_2)c(\beta_2) \end{pmatrix} = \mathbf{z}_2 \quad (3.34)$$

Da cui è possibile ricavare i due parametri di DH in coordinate GDH rimasti:

$$\alpha_2 = \text{atan2} \left(-(z_{2^*})_{y_{1''}}, (z_{2^*})_{z_{1''}} \right) \quad (3.35)$$

$$\beta_2 = \text{atan2} \left((z_{2^*})_{x_{1''}}, \frac{(z_{2^*})_{z_{1''}}}{c(\alpha_2)} \right) \quad (3.36)$$

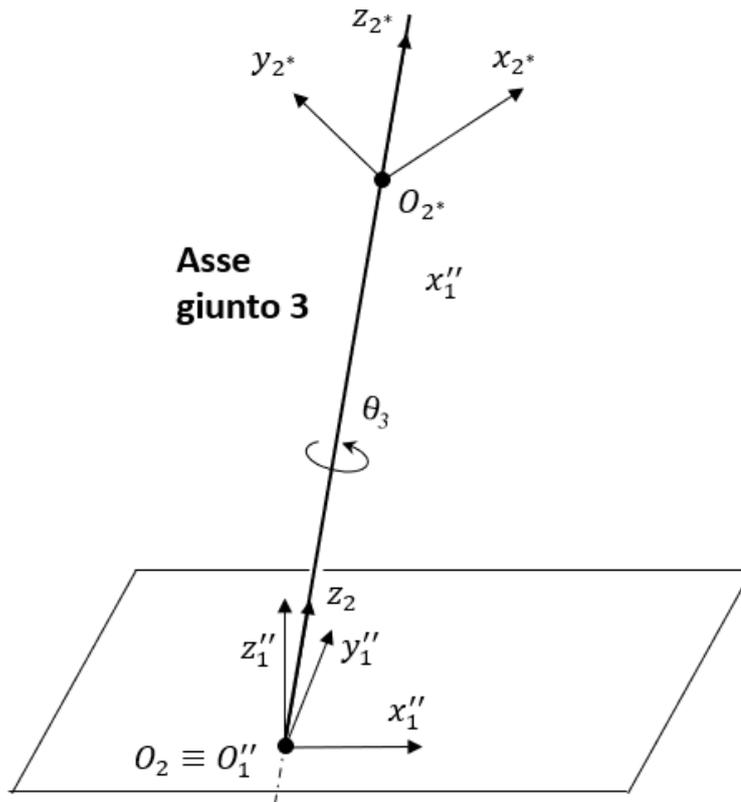


Figura 3.15: Sistemi di riferimento passanti per lo stesso asse giunto

3.3.2.3 Link 3: Gomito-Polso1

Tutte le considerazioni fatte per il *link* 2 sono perfettamente analoghe per il *link* 3, a patto che vengano utilizzati gli indici corretti.

I *link* 1 e 2 diventano rispettivamente i *link* 2 e 3, mentre l'asse giunto alla quale ci si riferisce diventa l'asse giunto 4 e non più 3.

3.3.2.4 Link 4: Polso1-Polso2

Dato che il *link* 3 così come viene espresso dalla UR non ha senso fisico, il *link* 4 ad esso definito è apparentemente senza senso fisico; il *link* 4 essendo riferito al *link* 3, deve compensare l'errato posizionamento del *link* precedente.

Infatti, la matrice $A_{4^*}^{3^*}$ non ha senso fisico leggendone i parametri di DH in Tabella 3.3 in quanto è molto diversa al robot teorico (Tabella 3.1), dal quale ci si aspetta piccole modifiche ai parametri di DH.

Dopo aver studiato il brevetto utilizzato dal costruttore [34] per la calibrazione di questo robot e aver approfondito la letteratura che trattava le varie convenzioni di rappresentazione, si è ipotizzato con successo, che l'orientamento degli assi e la

posizione del SR_{4^*} avessero senso fisico rispetto alla base. Si è imputata la causa dell'assenza di significato fisico dei parametri di DH forniti dal costruttore, esclusivamente alla sua scelta di rappresentare il sistema in coordinate di DH standard, altamente inefficienti in presenza di *link* paralleli. Infatti, in Figura 3.13 il SR_{4^*} è stato rappresentato prossimo al $SR_{\underline{4}}$ sia in posizione che in orientamento. Fatta questa considerazione preliminare si può affermare che la matrice $A_{4^*}^3$ non ha senso fisico, ma la matrice si $A_{4^*}^0$ sì e coincide con A_4^0 .

I parametri di DH per il *link* 4 dovranno essere riscritti correttamente.

Si utilizza la convezione dell'Eq. (3. 7) ossia quella di GDH ma dato che beta è fissato a 0 è equivalente alla (3. 2). La matrice omogenea che esprime il SR_4 sarà data dalla (3. 37)

$$A_4^3 = R_{z,\theta_4} T_{z,d_4} T_{x,a_4} R_{x,\alpha_4} (R_{y,\beta_4=0}) = \begin{bmatrix} c_{\theta_4} & -s_{\theta_4}c_{\alpha_4} & s_{\theta_4}s_{\alpha_4} & a_4c_{\theta_4} \\ s_{\theta_4} & c_{\theta_4}c_{\alpha_4} & -c_{\theta_4}s_{\alpha_4} & a_4s_{\theta_4} \\ 0 & s_{\alpha_4} & c_{\alpha_4} & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3. 37)$$

Dei quali θ_4 , d_4 , a_4 e α_4 sono incognite da ricavare.

Esprimendo le coordinate del SR_{4^*} rispetto al SR_3 si calcola la matrice $A_{4^*}^3$.

$$A_{4^*}^3 = (A_3^0)^{-1} A_{4^*}^0 = \begin{bmatrix} (x_{4^*})_{x_3} & (y_{4^*})_{x_3} & (z_{4^*})_{x_3} & (p_{4^*})_{x_3} \\ (x_{4^*})_{y_3} & (y_{4^*})_{y_3} & (z_{4^*})_{y_3} & (p_{4^*})_{y_3} \\ (x_{4^*})_{z_3} & (y_{4^*})_{z_3} & (z_{4^*})_{z_3} & (p_{4^*})_{z_3} \\ \mathbf{0} & & & 1 \end{bmatrix} \quad (3. 38)$$

Tutte le considerazioni fatte si traducono in una uguaglianza:

$$A_4^3 = A_{4^*}^3 \quad (3. 39)$$

Il che si traduce in forma matriciale come:

$$\begin{bmatrix} c_{\theta_4} & -s_{\theta_4}c_{\alpha_4} & s_{\theta_4}s_{\alpha_4} & a_4c_{\theta_4} \\ s_{\theta_4} & c_{\theta_4}c_{\alpha_4} & -c_{\theta_4}s_{\alpha_4} & a_4s_{\theta_4} \\ 0 & s_{\alpha_4} & c_{\alpha_4} & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} (x_{4^*})_{x_3} & (y_{4^*})_{x_3} & (z_{4^*})_{x_3} & (p_{4^*})_{x_3} \\ (x_{4^*})_{y_3} & (y_{4^*})_{y_3} & (z_{4^*})_{y_3} & (p_{4^*})_{y_3} \\ (x_{4^*})_{z_3} & (y_{4^*})_{z_3} & (z_{4^*})_{z_3} & (p_{4^*})_{z_3} \\ \mathbf{0} & & & 1 \end{bmatrix} \quad (3. 40)$$

Dall'uguaglianza è possibile estrarre i corretti parametri di DH attraverso la cinematica inversa.

$$d_4 = (p_{4^*})_{z_3} \quad (3. 41)$$

$$\theta_4 = \text{atan2}((x_4^*)_{y_3}, (x_4^*)_{x_3}) \quad (3.42)$$

$$a_4 = \frac{(p_4^*)_{y_3}}{(x_4^*)_{y_3}} \quad (3.43)$$

$$\alpha_4 = \text{atan2}((y_4^*)_{z_3}, (z_4^*)_{z_3}) \quad (3.44)$$

3.3.2.5 Link 5 e Link 6: Polso2-Polso3 e Polso3-TCP

Dato che i *link* 5 e 6 non sono paralleli e dato che il *link* precedente era posizionato nello spazio in una posizione con senso fisico, le modifiche ai parametri di DH forniti dal costruttore sono validi. Ricordando sempre che aggiungendo beta pari a 0 come ultima rotazione permetterà di trattare il sistema con la convenzione GDH. A seguito del processo di identificazione dei parametri di GDH, si ottengono le seguenti modifiche ai parametri cinematici.

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,110	-0,156	0,184	0,025	-0,069	0
$\Delta \alpha$ [°]	-0,095	0,004	-0,232	-0,034	-0,013	0
$\Delta \beta$ [°]	0	0,013	0,071	0	0	0
Δd [mm]	-0,075	0	0	1,062	0,229	0,194
$\Delta \theta$ [°]	0,003	-0,019	-0,013	-0,008	0,009	-0,006

Tabella 3.4: Modifiche parametri cinematici da proiezione geometrica (convenzione GDH)

Confrontandolo con la Tabella 3.1 si ha che gli ordini di grandezza delle variazioni rispetto ad i parametri teorici hanno senso fisico, sono sufficienti infatti piccole correzioni dei parametri di DH originari.

Confrontando le modifiche ai parametri cinematici delle Tabella 3.2 e la Tabella 3.4, si nota la sostanziale differenza negli ordini di grandezza delle modifiche dei parametri di DH.

La calibrazione a due robot [34] prevede l'uso di una flangia solidale ai TCP dei due robot utilizzati per calibrazione; i parametri cinematici a_6 e α_6 , che portano dal polso 3 al TCP, vengono attribuiti alla flangia e posti pari a zero sul link finale.

3.3.3 Verifica dei risultati e convalida dei dati

Ottenuti i nuovi parametri cinematici bisogna verificarne l'efficacia, si rende necessario confrontare i nuovi parametri GDH con quelli forniti dalla calibrazione del costruttore.

In MATLAB è stata scritta una routine che prende i nuovi parametri GDH e va a calcolare le differenze che ci sono, in termini di errore di posizionamento ed orientamento del TCP, rispetto ai parametri forniti dal costruttore, del robot reale e teorico. Questa sarà detta fase di convalidazione dei dati.

Si prendono in analisi i punti utilizzati dal costruttore per la calibrazione del robot contenuti dal file 'chainMeasurement.dat', presente all'interno della memoria del robot. In questo file sono memorizzate l'attuazione dei giunti $q_1:q_6$ dei due robot utilizzati dalla Universal Robots per effettuare la sua calibrazione. Si hanno due set di attuazione giunto per ogni punto preso in considerazione dal costruttore, e si hanno a disposizione 346 configurazioni ai giunti. Di queste configurazioni si decide di destinarne il 70% per la fase di identificazione ed il restante 30% per la fase di convalida. Dato che la fase di identificazione è stata conseguita mediante dei metodi geometrici, si procederà con la convalida della proiezione nella convenzione GDH utilizzando l'ultimo 30% dei punti a disposizione.

Questa si effettua calcolando la posizione del TCP del robot attraverso la cinematica diretta per i tre sistemi di riferimento $SR_{\underline{6}}$, SR_{6^*} e SR_6 , rappresentati in Figura 3.13, per tutte le pose della fase di convalida.

Come è stato fatto per la configurazione dei giunti (3. 11), ottenendo nell'Eq. (3. 13) e (3. 15) le matrici $A_{6^*}^0$ e $A_{\underline{6}}^0$, si calcolano per tutte le configurazioni giunti della convalida le rispettive matrici; si avrà quindi a disposizione una nuvola di $SR_{\underline{6}}$ e SR_{6^*} espresse rispetto all'origine comune O_0 .

Sempre per le stesse configurazioni giunti, applicando ad i parametri di GDH identificati le equazioni (3. 4) e (3. 7) si ottengono le matrici A_6^0 che rappresentano il TCP del robot nella convenzione GDH; si avranno dunque i SR_6 per tutte le configurazioni.

Note le coordinate di SR_6 , $SR_{\underline{6}}$ e SR_{6^*} è possibile andare a valutare se la proiezione verso il sistema di riferimento GDH sia andata a buon fine. Si andrà ad analizzare

a quanto ammonta l'errore di posizionamento e di orientamento del TCP in coordinate GDH (dei SR_6 identificati). Per poter fare un confronto significativo tra i sistemi di riferimento, è opportuno esprimere la posa secondo la medesima convenzione di rappresentazione.

Ad esempio, presa in considerazione la matrice A_6^0 , questa rappresenta la trasformazione omogenea dalla base al TCP, la matrice è composta da 4 vettori espressi rispetto alla base.

$$A_6^0 = \begin{bmatrix} (x_6)_{x_0} & (y_6)_{x_0} & (z_6)_{x_0} & (p_6)_{x_0} \\ (x_6)_{y_0} & (y_6)_{y_0} & (z_6)_{y_0} & (p_6)_{y_0} \\ (x_6)_{z_0} & (y_6)_{z_0} & (z_6)_{z_0} & (p_6)_{z_0} \\ \mathbf{0} & & & 1 \end{bmatrix} \quad (3.45)$$

Da questa matrice è possibile estrarre e descrivere in modo più compatto, all'interno di un vettore, la posizione e l'orientamento del SR_6 , ottenendo la posa del SR_6 rispetto alla base. La posizione sarà identificata dal vettore \mathbf{p}_6 , l'orientamento dipenderà dal tipo di convenzione adottata (es. RPY, Eulero); come convenzione si decide di utilizzare quella asse/angolo Figura 3.16.

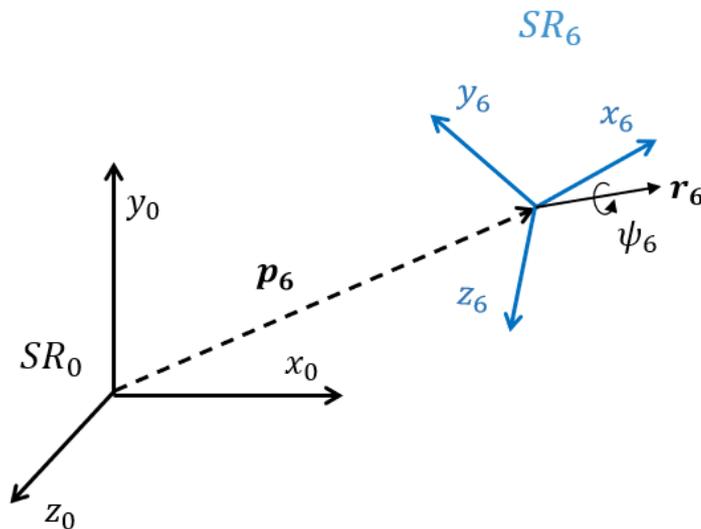


Figura 3.16: Rappresentazione convenzione asse angolo per il TCP

La convenzione asse angolo descrive l'orientamento dato dai vettori \mathbf{x}_6 , \mathbf{y}_6 e \mathbf{z}_6 attraverso un angolo ψ_6 che è scalare ed un asse di rotazione \mathbf{r}_6 che è un vettore.

$$\psi_6 = \arccos\left(\frac{(x_6)_{x_0} + (y_6)_{y_0} + (z_6)_{z_0} - 1}{2}\right) \quad (3.46)$$

$$\mathbf{r}_6 = \frac{1}{2\sin(\psi_6)} \begin{pmatrix} (y_6)_{z_0} - (z_6)_{y_0} \\ (z_6)_{x_0} - (x_6)_{z_0} \\ (x_6)_{y_0} - (y_6)_{x_0} \end{pmatrix} \quad (3.47)$$

Definita la convenzione, si avrà che A_6^0 sarà data dalla rototraslazione composta traslazione tale da spostare in \mathbf{p}_6 e la rotazione lungo \mathbf{r}_6 dell'angolo ψ_6 .

$$A_6^0 = T_{\mathbf{p}_6} R_{\mathbf{r}_6, \psi_6}$$

Si passa dalla matrice A_6^0 al vettore \mathbf{a}_6^0 così definito:

$$\mathbf{a}_6^0 = \begin{pmatrix} (p_6)_{x_0} \\ (p_6)_{y_0} \\ (p_6)_{z_0} \\ \psi_6 \end{pmatrix} = \begin{pmatrix} \mathbf{p}_6 \\ \psi_6 \end{pmatrix} \quad (3.48)$$

Si sceglie di esprimere l'orientamento solo con l'angolo ψ_6 in quanto, ai fini del calcolo della differenza di orientamento, viene comodo trattare con delle grandezze scalari più facilmente confrontabili; inoltre, il costruttore utilizza questa convenzione per la sua calibrazione. Ai fini del calcolo dell'errore di posizione ed orientamento non è particolarmente interessante l'asse di rotazione \mathbf{r}_6 motivo per il quale non viene conservato per descrivere l'orientamento del SR_6 . Dai tre sistemi di riferimento SR_6 , SR_{6^*} e $SR_{\underline{6}}$ si ottengono rispettivamente i tre vettori \mathbf{a}_6^0 , $\mathbf{a}_{6^*}^0$ e $\mathbf{a}_{\underline{6}}^0$ per tutti i punti della convalidazione. L'errore di posizionamento e di orientamento dei SR_6 e $SR_{\underline{6}}$ vengono calcolati per ogni punto rispetto al SR_{6^*} identificato dal costruttore; infatti, esso è l'unico parametro che può essere utilizzato come termine di confronto. I parametri di DH che portano al SR_{6^*} sono gli unici che rappresentano le coordinate del TCP ottenute dalla diretta calibrazione del robot reale. Il SR_{6^*} rappresenta dunque la posizione e l'orientamento del robot reale. Si avranno 4 errori per ogni punto j , per i due sistemi di riferimento SR_6 e $SR_{\underline{6}}$ si avranno:

$$\begin{pmatrix} \delta_{j_x} \\ \delta_{j_y} \\ \delta_{j_z} \\ \gamma_{j_\psi} \end{pmatrix} = \begin{pmatrix} \delta_j \\ \gamma_j \end{pmatrix} = (\mathbf{a}_6^0) - (\mathbf{a}_{6^*}^0) \quad (3.49)$$

$$\begin{pmatrix} \underline{\delta_{j_x}} \\ \underline{\delta_{j_y}} \\ \underline{\delta_{j_z}} \\ \underline{\gamma_{j_\psi}} \end{pmatrix} = \begin{pmatrix} \underline{\delta_j} \\ \underline{\gamma_j} \end{pmatrix} = (\underline{\mathbf{a}}_6^0) - (\underline{\mathbf{a}}_6^{0*}) \quad (3.50)$$

Dove i delta rappresentano gli errori di posizionamento lungo gli assi di riferimento del sistema di riferimento inerziale, mentre gamma rappresenta l'orientamento. Con le grandezze sottolineate si intende il $SR_{\underline{6}}$.

L'errore di posizionamento e di orientamento saranno rappresentati da due valori e_{pos_j} e e_{rot_j} per ogni punto.

$$\begin{pmatrix} e_{pos_j} \\ e_{rot_j} \end{pmatrix} = \begin{pmatrix} \sqrt{\delta_{j_x}^2 + \delta_{j_y}^2 + \delta_{j_z}^2} \\ \gamma_j \end{pmatrix} \quad (3.51)$$

$$\begin{pmatrix} \underline{e_{pos_j}} \\ \underline{e_{rot_j}} \end{pmatrix} = \begin{pmatrix} \sqrt{\underline{\delta_{j_x}^2} + \underline{\delta_{j_y}^2} + \underline{\delta_{j_z}^2}} \\ \underline{\gamma_j} \end{pmatrix} \quad (3.52)$$

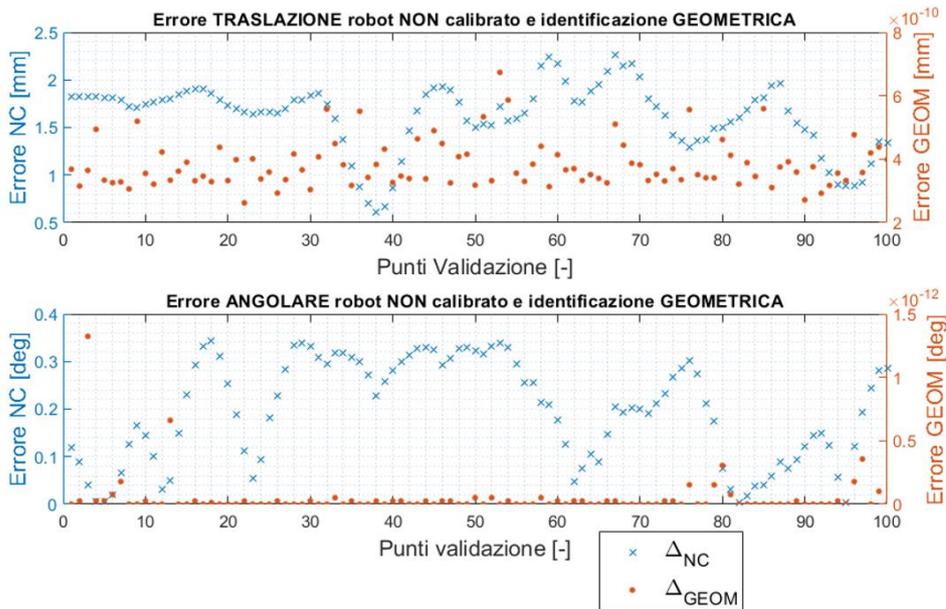


Grafico 3.1: Errori posa punti validazione da proiezione geometrica (convenzione GDH)

Gli errori (3.51) e (3.52) sono calcolati per tutte le pose della fase di convalidazione, sono rappresentati attraverso Δ_{GEOM} e Δ_{NC} che rappresentano

rispettivamente gli errori del robot secondo la convenzione GDH e quelli del robot non calibrato (NC).

Guardando il grafico si nota una sostanziale differenza tra gli errori del robot non calibrato ed il robot della quale sono stati identificati i parametri GDH.

La UR per poter validare le sue calibrazioni determina delle deviazioni limite al di sopra del quale si ritiene la calibrazione non accettabile.

Prendendo in considerazione i punti di validazione vengono definite la deviazione media e la deviazione standard massimi degli errori di traslazione ed orientamento.

Requisiti di accettabilità forniti dalla UR [9]:

- Errore Medio: 1mm e 2 mrad;
- Deviazione Standard: 0,5mm e 1 mrad.

Provando ad applicare la validazione del robot non calibrato si ottengono nella Tabella 3.5 le caratteristiche degli errori ottenute. Si è scelto di mostrare anche l'errore massimo in quanto rappresenta un parametro molto rilevante per un manipolatore.

Robot	Errore	Non calibrato		Identificazione Analitica GDH	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
UR5 PoliTO	Deviazione Standard	0,354 (70,7%)	1,872 (187,2%)	7,44E-11 (0,0%)	2,70E-12 (0,0%)
	Errore Medio	1,624 (162,4%)	3,418 (170,9%)	3,83E-10 (0,0%)	7,70E-13 (0,0%)
	Errore Massimo	2,263	5,986	6,74E-10	2,31E-11

Tabella 3.5: Confronto deviazione errori TCP usando i parametri cinematici GDH e robot non calibrato

I valori tra parentesi indicano la percentuale ottenuta di errore medio e deviazione standard, rispetto ai limiti di accettabilità imposti dalla UR.

Se la percentuale è maggiore rispetto al 100% l'errore riscontrato non è ammissibile per il costruttore, infatti si osserva che il robot non calibrato non rientra nei limiti imposti dalla UR.

Gli errori riscontrati nella Tabella 3.5, confrontando posa ottenuta dai parametri di GDH identificati e la posa ottenuta dalle modifiche ai parametri di DH del costruttore, sono così piccoli da non avere senso fisico.

Si può affermare che entrambi i modelli portano esattamente alla stessa posa poiché si ottengono degli errori aventi ordine di grandezza pari alla precisione di macchina; infatti, osservando il secondo ed il terzo *link* della Tabella 3.3 si hanno due parametri di DH con ordine di grandezza 10^3 m, l'errore di posizionamento nella Tabella 3.5 ha un ordine di grandezza pari a 10^{-13} m, raggiungendo dunque la precisione di macchina $\left(\frac{10^{-13}}{10^3} = 10^{-16}\right)$.

L'ipotesi che i parametri cinematici identificati dalla Universal Robots avessero senso fisico ma che fossero solo espressi nel sistema di riferimento sbagliato si è rivelata esatta. L'identificazione cinematica mediante la proiezione geometrica è andata a buon fine, è possibile descrivere il sistema attraverso dei parametri cinematici con senso fisico.

Bisogna ricordare che la fase di convalida non ha utilizzato delle misure reali, bensì si è fatto ricorso ad una simulazione. Non sono presenti in questa simulazione incertezze del sistema di misura ed errori di ripetibilità e non geometrici; infatti, si assume che le pose ottenute attraverso l'identificazione della Universal Robots siano quelle di un robot "perfetto" e che queste vengano rilevate da un sistema di misura esente da errori di misurazione.

Questo algoritmo di identificazione può essere utilizzato anche in altre applicazioni simili. Includendo l'UR5 già descritto, sono note le modifiche ai parametri cinematici senza senso fisico di 4 manipolatori: due del Politecnico di Torino e due dell'azienda PST SRL; si è scelto di testare l'algoritmo per indentificare i parametri cinematici con senso fisico di tutti i robot.

Nella Tabella 3.6 sono rappresentati i risultati dell'identificazione dei parametri GDH dell'algoritmo. Questo si è dimostrato funzionante per tale famiglia di robot ed è stato in grado di identificare per ciascun robot dei parametri cinematici con senso fisico. Le attuazioni ai giunti utilizzate per convalidare questi robot sono quelle utilizzate per l'UR5 del Politecnico di Torino.

Robot	Errore	Non calibrato		Proiezione Analitica	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
UR3 PoliT0	Deviazione Standard	0,505 (101,0%)	1,849 (184,9%)	8,48E-14 (0,0%)	2,44E-12 (0,0%)
	Errore Medio	1,183 (118,3%)	2,984 (149,2%)	4,28E-11 (0,0%)	8,13E-13 (0,0%)
	Errore Massimo	2,084	6,185	4,31E-11	2,22E-11
UR5 PoliT0	Deviazione Standard	0,354 (70,7%)	1,872 (187,2%)	7,44E-11 (0,0%)	2,70E-12 (0,0%)
	Errore Medio	1,624 (162,4%)	3,418 (170,9%)	3,83E-10 (0,0%)	7,70E-13 (0,0%)
	Errore Massimo	2,263	5,986	6,74E-10	2,31E-11
UR5 PST Srl	Deviazione Standard	0,483 (96,7%)	2,724 (272,4%)	1,86E-12 (0,0%)	1,52E-12 (0,0%)
	Errore Medio	1,350 (135,0%)	5,172 (258,6%)	5,69E-10 (0,0%)	5,64E-13 (0,0%)
	Errore Massimo	2,093	9,971	5,75E-10	9,77E-15
UR10 PST Srl	Deviazione Standard	1,491 (298,1%)	4,369 (436,9%)	2,30E-13 (0,0%)	1,94E-12 (0,0%)
	Errore Medio	9,371 (937,1%)	8,242 (412,1%)	4,53E-10 (0,0%)	8,28E-13 (0,0%)
	Errore Massimo	11,536	18,184	4,54E-10	1,33E-11

Tabella 3.6: Deviazione errori convenzione GDH e robot non calibrati per 4 robot diversi

Capitolo 4

4 Identificazione dei parametri cinematici

Dopo aver proiettato i sistemi di riferimento dalla convenzione di DH alla convenzione GDH, si è voluto procedere all'identificazione cinematica attraverso dei metodi di identificazione utilizzati in letteratura, descritti nel capitolo 2.2.3.

Proiettare i sistemi di riferimento si è dimostrato efficace, ma è necessaria un'approfondita conoscenza geometrica del problema per ottenere tali risultati; l'algoritmo di proiezione elaborato è valido per robot simili, ossia dei robot antropomorfi aventi la conformazione con gli assi giunti paralleli per gli stessi 2 link.

A seguito della ricerca bibliografica effettuata sulla calibrazione, in questo capitolo si procederà all'identificazione dei parametri cinematici come in un processo di calibrazione, simulando la fase di misurazione. Per simulazione della fase di misurazione si assume che le pose assunte dal TCP con i parametri identificati dal costruttore coincidano con le pose effettivamente raggiunte dal robot fisico; queste verranno trattate nell'identificazione virtualmente come l'output di un sistema di misura esterno. Non saranno presenti quindi l'incertezza del sistema di misura, gli errori di ripetibilità e gli errori non geometrici del manipolatore. Gli unici errori simulati in questo lavoro, usando come riferimento la calibrazione del costruttore, saranno gli errori cinematici.

Dato che la fase di misura non viene fisicamente effettuata sul robot, i risultati ottenuti non potranno essere definiti come il risultato di una calibrazione. Il lavoro consiste in una identificazione cinematica.

Identificando i parametri cinematici attraverso dei metodi numerici attraverso dei metodi di identificazione utilizzati in letteratura, e non è necessario analizzare l'orientamento degli assi giunto di ciascun link nello spazio, come fatto durante l'identificazione mediante proiezione geometrica nel terzo capitolo.

Attraverso questo approccio; questo metodo può essere applicato a qualunque tipologia di robot in quanto si prende in considerazione il posizionamento del TCP come riferimento per l'identificazione.

Si hanno a disposizione le configurazioni contenute nel file 'chainMeasurement.dat' introdotto nel terzo capitolo durante la proiezione dei sistemi di riferimento. Dei 346 punti di presenti, il 70% verrà utilizzato per la fase di identificazione ed il restante 30% per la fase di convalida. Con le stesse configurazioni ai giunti rappresentate nel Grafico 4.1 si effettuerà l'identificazione seguendo diversi tipi di approccio numerico.

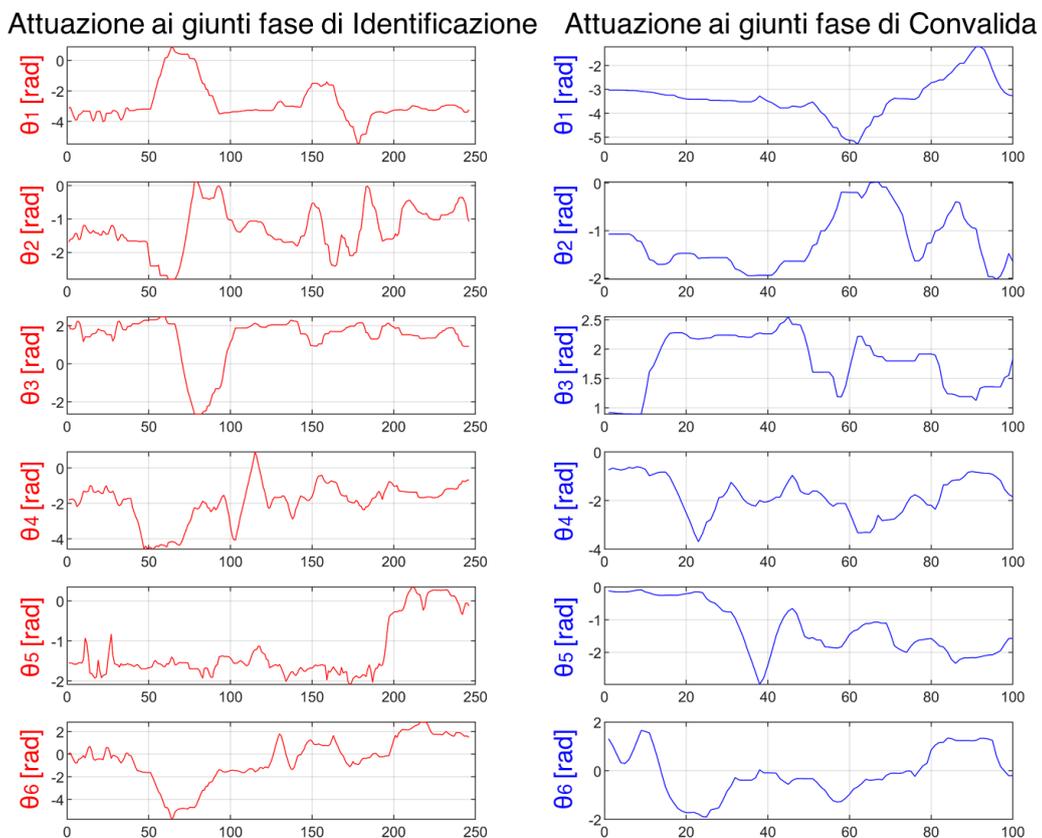


Grafico 4.1: Attuazione ai giunti usati in fase Identificazione e Convalida

Si sceglie di adottare degli approcci sia di tipo analitico che di tipo stocastico.

Gli approcci analitici vengono descritti nel capitolo 4.1 e sono:

- decomposizione ai valori singolari o *Singular Value Decomposition* (SVD);
- metodo di Gauss-Newton o *Least Squares Method* (LSM).

Gli approcci stocastici utilizzati si basano sugli algoritmi genetici, sono descritti nel capitolo 4.2 e sono:

- algoritmo Genetico standard (GA);
- algoritmo Genetico Multiobbiettivo (Multi-GA);
- algoritmo Genetico Multiobbiettivo Ibrido (Multi-GA Ibrido).

4.1 Identificazione SVD e Gauss Newton

Per descrivere la posa raggiunta dai sistemi di riferimento si utilizza la convenzione asse-angolo descritta nel paragrafo 3.3.3.

Utilizzando le attuazioni ai giunti destinati alla fase di identificazione, identificando ciascuna configurazione con la lettera j , attraverso le Eq. (3. 2), (3. 3) e (3. 48) si calcolano le pose $(\mathbf{a}_{6^*}^0)_j$. Queste sono le pose dei sistemi di riferimento $A_{6^*}^0$ utilizzando i parametri cinematici del robot calibrato dal costruttore, che utilizziamo come se fossero le letture di un sistema di misura esterno.

Con le stesse attuazioni ai giunti, si calcolano le pose $(\mathbf{a}_6^0)_j$ che vengono assunte dal sistema di riferimento A_6^0 utilizzando il modello GDH attraverso la (3. 7) mediante i parametri cinematici del robot non calibrato; queste sono le pose che si vorrebbe idealmente raggiungere se il robot non avesse errori geometrici (dato che sono l'unico tipo di errore presente in questo lavoro). Queste pose ideali non vengono raggiunte, attraverso l'Eq. (2. 2) si determinano gli errori di posa \mathbf{e}_j per ciascuna configurazione ai giunti j . Questi errori di posa sono nella convenzione asse-angolo saranno come nell'Eq. (3. 49):

$$\mathbf{e}_j = \begin{pmatrix} \delta_{jx} \\ \delta_{jy} \\ \delta_{jz} \\ \gamma_j \end{pmatrix} = \begin{pmatrix} \delta_j \\ \gamma_j \end{pmatrix} = (\mathbf{a}_6^0)_j - (\mathbf{a}_{6^*}^0)_j \quad (4. 1)$$

Aggregando le pose lette si potrà ottenere l'errore \mathbf{e}_{ag} così come nella (2. 9):

$$\mathbf{e}_{ag} = \begin{pmatrix} \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_{n_{id}} \end{pmatrix} \quad (4. 2)$$

Noto il vettore errore aggregato è necessario calcolare la matrice Jacobiana del robot che si sta calibrando. La Jacobiana di questo problema di identificazione, come si può osservare dall'Eq. (2. 28), è una matrice i cui elementi sono le derivate

parziali della funzione (2. 1) che descrive la posa; è necessario descrivere in modo simbolico la funzione prima di poter calcolare la Jacobiana,

Per farlo, con l'ausilio del *Symbolic Toolbox* di Matlab, bisogna prima calcolare il prodotto delle sei trasformazioni omogenee (3. 3), così da identificare una funzione matriciale $A_6^0(\boldsymbol{\theta}, \boldsymbol{\eta})$ che esprima il SR del TCP.

Ciascuna trasformazione $A_i(\theta_i, \boldsymbol{\eta}_i)$ rappresenta il sistema di riferimento i esimo di un *link* rispetto al precedente. Si sceglie il modello di GDH per ciascuna trasformazione omogenea come nell'Eq (3. 7) e, trattando tutti i parametri cinematici come delle variabili, si ottiene la seguente funzione matriciale:

$$A_6^0(\boldsymbol{\theta}, \boldsymbol{\eta}) = \prod_{i=1}^6 A_i(\theta_i, \boldsymbol{\eta}_i) = \begin{bmatrix} g_{1,1}(\boldsymbol{\theta}, \boldsymbol{\eta}) & \cdots & g_{1,4}(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ \vdots & \ddots & \vdots \\ g_{3,1}(\boldsymbol{\theta}, \boldsymbol{\eta}) & \cdots & g_{3,4}(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ 0 & \cdots & 1 \end{bmatrix} \quad (4. 3)$$

Gli elementi $g_{i,j}(\boldsymbol{\theta}, \boldsymbol{\eta})$ della funzione matriciale $A_6^0(\boldsymbol{\theta}, \boldsymbol{\eta})$ sono delle funzioni scalari. La matrice ed i suoi elementi sono in funzione di trenta variabili. Queste variabili sono contenute nei vettori $\boldsymbol{\theta}$ e $\boldsymbol{\eta}$. Il vettore $\boldsymbol{\eta}$ è composto da quattro vettori, ciascuno dei quali è composto da 6 elementi (uno per *link*):

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_6 \end{bmatrix} \quad (4. 4)$$

$$\boldsymbol{\eta} = [\mathbf{d} \quad \mathbf{a} \quad \boldsymbol{\alpha} \quad \boldsymbol{\beta}] = \begin{bmatrix} d_1 & a_1 & \alpha_1 & \beta_1 \\ \vdots & \vdots & \vdots & \vdots \\ d_6 & a_6 & \alpha_6 & \beta_6 \end{bmatrix} \quad (4. 5)$$

Dalla funzione matriciale, si ricerca la funzione vettoriale che esprime la posa del TCP. Attraverso le Eq.(3. 46) e (3. 48) si ottiene la funzione vettoriale $\mathbf{f}(\boldsymbol{\theta}, \boldsymbol{\eta})$ composta da quattro elementi in trenta variabili.

$$\mathbf{f}(\boldsymbol{\theta}, \boldsymbol{\eta}) = \begin{pmatrix} g_{1,4}(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ g_{2,4}(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ g_{3,4}(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ \arccos\left(\frac{g_{1,1}(\boldsymbol{\theta}, \boldsymbol{\eta}) + g_{2,2}(\boldsymbol{\theta}, \boldsymbol{\eta}) + g_{3,3}(\boldsymbol{\theta}, \boldsymbol{\eta}) - 1}{2}\right) \end{pmatrix} = \begin{bmatrix} f_x(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ f_y(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ f_z(\boldsymbol{\theta}, \boldsymbol{\eta}) \\ f_\psi(\boldsymbol{\theta}, \boldsymbol{\eta}) \end{bmatrix} \quad (4. 6)$$

Seppur il modello permette la presenza di 30 variabili, solo 24 di queste potranno cambiare.

Nota la funzione (4. 6), la Jacobiana viene calcolata attuando la derivata simbolica solo lungo le 24 direzioni che possono variare.

$$J = \left[\frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \boldsymbol{\theta}} \quad \frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \mathbf{d}} \quad \frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \mathbf{a}} \quad \frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \boldsymbol{\alpha}} \quad \frac{\partial f(\boldsymbol{\theta}, \boldsymbol{\eta})}{\partial \boldsymbol{\beta}} \right] \quad (4.7)$$

Le sei componenti da mantenere fisse, coerentemente con quanto osservato nel capitolo 3 sono:

$$\{d_2, d_3, \beta_1, \beta_4, \beta_5, \beta_6\} \quad (4.8)$$

La matrice Jacobiana diventerà della forma:

$$J = \begin{bmatrix} \frac{\partial f_x}{\partial \theta_1} & \dots & \frac{\partial f_x}{\partial d_1} & \frac{\partial f_x}{\partial d_4} & \dots & \frac{\partial f_x}{\partial \beta_3} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_\psi}{\partial \theta_1} & \dots & \frac{\partial f_\psi}{\partial d_1} & \frac{\partial f_\psi}{\partial d_4} & \dots & \frac{\partial f_\psi}{\partial \beta_3} \end{bmatrix} \quad (4.9)$$

Ottenuta la funzione analitica della matrice Jacobiana è possibile calcolarne i valori assunti dai vari elementi al variare delle variabili $\boldsymbol{\theta}$, \mathbf{d} , \mathbf{a} , $\boldsymbol{\alpha}$ e $\boldsymbol{\beta}$.

I valori numerici da inserire sono 30 e sono le variabili espresse nelle Eq. (4. 4) e (4. 5), di questi 24 possono cambiare e sono definibili variabili, considerando costanti quelli presenti nell'Eq.(4. 8); l'espressione simbolica rimane invariata e permette di ottenere la Jacobiana al variare dei valori numerici.

Applicando l'espressione analitica appena ottenuta, per una configurazione giunti $\boldsymbol{\theta}_j$, ed i parametri $\boldsymbol{\eta}$ noti, si ottiene la Jacobiana di una particolare posa $J_j \in \mathbb{R}^{4 \times 24}$ composta da 96 elementi $c_{i,j}$.

$$J_j = \begin{bmatrix} c_{1,1} & \dots & c_{1,24} \\ \vdots & \ddots & \vdots \\ c_{4,1} & \dots & c_{4,24} \end{bmatrix} \quad (4.10)$$

La matrice Jacobiana cambia a seconda della configurazione giunti $\boldsymbol{\theta}_j$. Calcolando ed aggregando le Jacobiane di tutte le pose presenti, si ottiene la matrice Jacobiana aggregata:

$$J_{ag} = \begin{bmatrix} J_1 \\ \vdots \\ J_{n_{id}} \end{bmatrix} \quad (4.11)$$

Noto l'errore aggregato (4. 2) e la matrice Jacobiana aggregata (4. 11), è possibile scegliere un metodo di identificazione e calcolare le modifiche ai parametri di DH del manipolatore.

4.1.1 Identificazione SVD

Il primo metodo utilizzato è la decomposizione ai valori singolari SVD che, a differenza dei minimi quadrati ovvero del metodo di Gauss-Newton, non è iterativo. La SVD non è un metodo di identificazione adottato per le calibrazioni incontrate in letteratura. Si è scelto preliminarmente di utilizzare questo metodo per poter effettuare dei test preliminari ed analizzare il comportamento della Jacobiana ed il suo numero di condizionamento (analisi descritta nel paragrafo 4.1.2).

Le analisi del comportamento della Jacobiana sono presenti nel paragrafo 4.1.2

Per l'analisi SVD, si decompone la matrice J_{ag} :

$$J_{ag} = U \cdot S \cdot V' \quad (4.12)$$

Dove S è una matrice diagonalizzata i quali elementi σ_i rappresentano gli autovalori della matrice J_{ag} .

Le modifiche dei parametri cinematici si calcolano dalle matrici ottenute con la decomposizione (4.12):

$$\Delta k_{ag} = V(S^{-1}(U' * e_{ag})) \quad (4.13)$$

La comodità di questo metodo sta nel fatto che, a differenza della (2.21), non è necessario invertire l'intera matrice Jacobiana, bensì solo la matrice contenente i suoi autovalori; inoltre è necessario effettuare un'unica iterazione per ottenere un risultato.

Le modifiche ai parametri cinematici ottenuti per l'UR5 sono:

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,107	-0,150	0,190	0,027	-0,068	-0,001
$\Delta \alpha$ [°]	-0,094	0,004	-0,231	-0,036	-0,014	0,000
$\Delta \beta$ [°]	0	0,011	0,068	0	0	0
Δd [mm]	-0,072	0	0	1,051	0,229	0,194
$\Delta \theta$ [°]	0,005	-0,019	-0,014	-0,009	0,009	-0,006

Tabella 4.1: Modifiche parametri cinematici identificazione SVD

Ottenute le modifiche dei parametri cinematici, per la fase di convalida, si utilizza il restante 30% delle attuazioni ai giunti disponibili; come è stato già fatto nel paragrafo 3.3.3, si calcolano gli errori di posizione e di orientamento Eq. (3.51) e (3.52) per tutti i punti selezionati.

Ottenendo i seguenti errori:

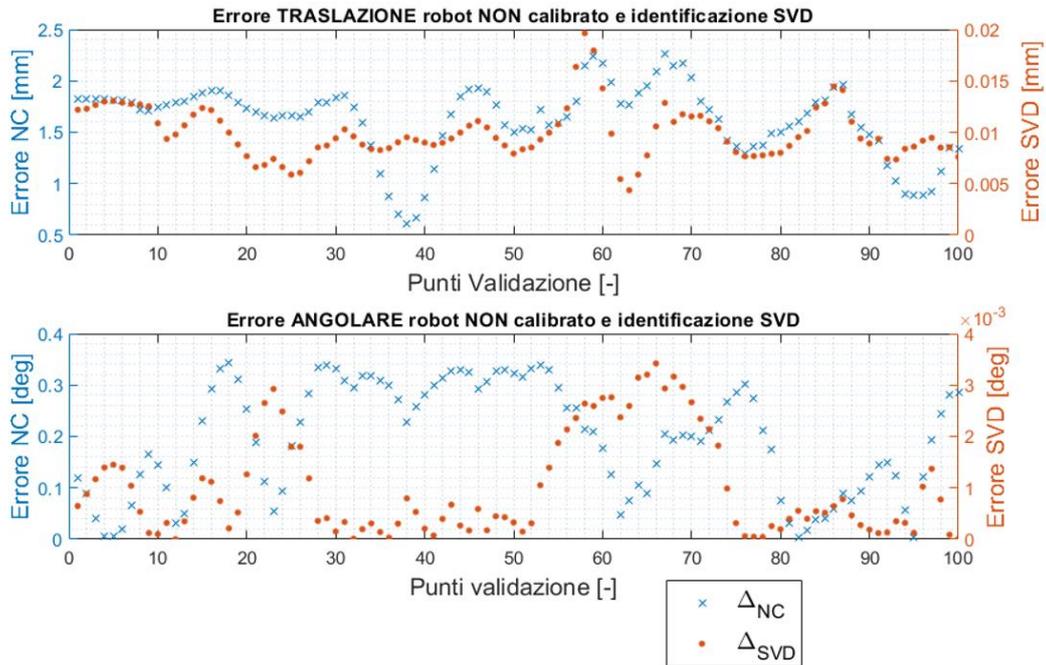


Grafico 4.2: Errori posa punti validazione da identificazione SVD (convenzione GDH)

Nel Grafico 4.1 ed il Grafico 4.2 sono rappresentati gli errori e_i espressi nella Eq.(4.1), questi sono calcolati per tutte le pose della fase di convalidazione e sono stati ripartiti dividendo gli errori di traslazione e di orientamento come nella Eq.(3.51); tutti i grafici degli errori di posa che verranno rappresentati nei prossimi paragrafi, utilizzando altri metodi di identificazione, presenteranno le medesime caratteristiche di quest'ultimi due grafici. Nella legenda Δ_{SVD} e Δ_{NC} rappresentano rispettivamente gli errori del robot a seguito della identificazione SVD e quelli del robot non calibrato (NC). I prossimi grafici porteranno nella legenda gli errori del metodo identificazione utilizzato, attraverso una sigla abbreviata e gli errori non identificati, sempre rappresentati dalla sigla NC.

Questo metodo è meno efficace rispetto all'identificazione geometrica fatta nel terzo capitolo e l'identificazione LSM che sarà fatta nel paragrafo 4.1.3. ma è stato molto utile come analisi preliminare.

Nella Tabella 4.2 è presente il confronto tra gli errori di posa riscontrati in questo modello semplificato ed i criteri di accettabilità definiti dal costruttore.

Robot		Non calibrato		SVD	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
UR5 PoliTO	Deviazione Standard	0,354 (70,7%)	1,872 (187,2%)	2,52E-03 (0,5%)	1,72E-02 (1,7%)
	Errore Medio	1,624 (162,4%)	3,418 (170,9%)	9,91E-03 (1,0%)	1,77E-02 (0,9%)
	Errore Massimo	2,263	5,986	2,52E-03	1,72E-02

Tabella 4.2: Confronto errori convalida dell'identificazione SVD e robot non calibrato

In Tabella 4.2, come fatto per la Tabella 3.5, dagli errori rappresentati nel Grafico 4.1 e Grafico 4.2 vengono calcolati la loro Deviazione Standard, l'errore medio e quello massimo. Nella stessa tabella vengono rappresentate le percentuali di questi errori confrontandoli con i limiti di accettabilità imposti dal costruttore, descritti nel paragrafo 3.3.3. Da come si può notare, gli errori medi di posizionamento e di orientamento vengono ridotti rispettivamente del 99,0% e del 99,1% rispetto al limite massimo imposto dal costruttore. Si considerano i parametri cinematici ottenuti accettabili; infatti, confrontando la Tabella 3.4 e la Tabella 4.1 la differenza tra i parametri delle due identificazioni raggiungono i lavori massimi di 6 μm e $0,001^\circ$.

Tutte le future tabelle presenti nei prossimi capitoli, i quali contengono la deviazione standard, l'errore medio e l'errore massimo, saranno ottenute rispettando gli stessi criteri della Tabella 4.2; per evitare troppe ripetizioni future in presenza di errori percentuali, questi saranno sempre da intendersi riferiti ai limiti imposti dalla Universal Robots.

Seppur sono presenti solo le attuazioni ai giunti utilizzati per la calibrazione dell'UR5 presenti nel laboratorio del Politecnico, gli stessi punti sono stati usati per gli altri tre manipolatori introdotti nella Tabella 3.6; si è cercato di identificare i parametri cinematici seguendo la stessa procedura. A differenza dell'identificazione geometrica fatta nel capitolo 3, che è indipendente dal tipo di attuazione ai giunti, in questo caso i punti utilizzati sono specifici dell'UR5 presente nei laboratori del Politecnico; si tratta dunque di un tentativo di identificazione utilizzando delle attuazioni ai giunti non specifiche per i robot presi in questione. I risultati della validazione a seguito dell'identificazione dei parametri cinematici di questi robot sono riassunti nella tabella successiva.

Robot	entità	SVD	
		posizione [mm]	rotazione [mrad]
UR3 PoliTO	Deviazione Standard	4,10E-04 (0,1%)	2,57E-03 (0,3%)
	Errore Medio	4,39E-03 (0,4%)	6,79E-03 (0,3%)
	Errore Massimo	5,18E-03	1,10E-02
UR5 PoliTO	Deviazione Standard	2,52E-03 (0,5%)	1,72E-02 (1,7%)
	Errore Medio	9,91E-03 (1,0%)	1,77E-02 (0,9%)
	Errore Massimo	1,96E-02	5,97E-02
UR5 PST Srl	Deviazione Standard	1,80E-02 (3,6%)	0,189 (18,9%)
	Errore Medio	5,12E-02 (5,1%)	0,214 (10,7%)
	Errore Massimo	9,75E-02	0,885
UR10 PST Srl	Deviazione Standard	0,171 (34,2%)	0,543 (54,3%)
	Errore Medio	0,337 (33,7%)	0,608 (30,4%)
	Errore Massimo	0,885	2,596

Tabella 4.3: Errori convalida identificazione SVD

4.1.2 Analisi SVD della matrice Jacobiana

Oltre alla praticità d'uso, la decomposizione ai valori singolari è uno strumento molto utile per analizzare il numero di condizionamento della Jacobiana e la sua capacità di descrivere il sistema che si vuole linearizzare. Una Jacobiana avente un numero di condizionamento troppo elevato genera un sistema sovradeterminato; in presenza di tale sistema non è possibile convergere ad una soluzione[8].

Il numero di condizionamento della Jacobiana rappresenta un forte ostacolo nel processo di identificazione dei parametri cinematici. Generalmente, nei processi di calibrazione, vengono effettuate delle misure ridondanti al fine di minimizzare il rumore di misura associato ai diversi sensori montati sul manipolatore [46]. L'elevato uso di misure ridondanti può aumentare il numero di condizionamento della matrice Jacobiana. Il numero di osservazioni necessarie può essere ridotto se le misure vengono eseguite in configurazioni nei quali gli errori di posa sono più

sensibili alle variazioni dei parametri cinematici del modello, cioè dove sono meno reattivi alle fonti di errore non modellate.[11]

In questa simulazione non sono presenti fonti di errore non modellate, il rumore di misura risulta completamente assente, ma, nonostante ciò, il numero di condizionamento ha un impatto sulla convergenza dei metodi numerici adottati per l'identificazione dei parametri. Infatti, in presenza di un modello dove più parametri cinematici risultano linearmente dipendenti, non è possibile ottenere una soluzione mediante un metodo numerico deterministico in quanto avremmo un sistema sovradeterminato; la scelta dei parametri cinematici deve garantire che questi siano tra loro linearmente indipendenti, non devono esserci ridondanze di rappresentazione nel modello adottato[16]. Le pose misurate per l'identificazione cinematica devono essere scelte in modo tale da permettere che ciascuna variabile (parametro cinematico) risulti indipendente dalle altre. Per evitare eventuali problemi con il numero di condizionamento della matrice Jacobiana, sono stati utilizzate le attuazioni ai giunti utilizzate dal costruttore per la calibrazione dell'UR5. A dimostrare la buona distribuzione delle pose adottate dal costruttore è sufficiente analizzare gli autovalori σ_i ottenuti dalla decomposizione SVD della matrice Jacobiana (4. 11) effettuata nell'Eq (4. 12).

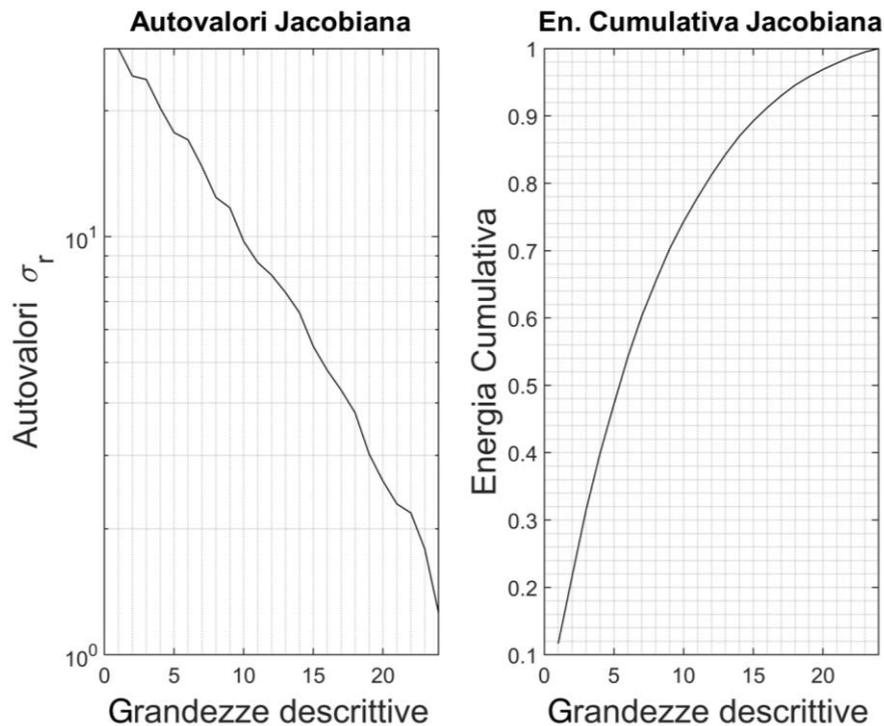


Grafico 4.3: Autovalori ed energia Cumulativa Jacobiana variabili linearmente indipendenti

Nel Grafico 4.3 sono rappresentati gli autovalori e l'energia cumulativa della Jacobiana in convenzione GDH all'aumentare del numero delle grandezze descrittive. Le grandezze descrittive sono le componenti dei parametri cinematici θ e η delle Eq. (4. 4) e (4. 5) escludendo le 6 componenti dell'Eq. (4. 8) della quale non vengono considerate le variazioni.

Il numero di condizionamento si può calcolare attraverso il rapporto degli autovalori massimo e minimo; per il set di attuazione ai giunti utilizzato il numero di condizionamento della matrice Jacobiana è:

$$ncond = \frac{\sigma_{max}}{\sigma_{min}} = 22,407 \quad (4. 14)$$

Un numero di condizionamento così basso permette di convergere ad una soluzione. L'energia cumulativa può variare entro i valori zero ed uno e consiste, all'aumentare delle grandezze descrittive j del sistema, nella relazione:

$$en. cum. = \frac{\sum_{i=1}^j \sigma_i}{\sum_{i=1}^N \sigma_i} \quad (4. 15)$$

Dove N è il numero di grandezze descrittive, in questo caso 24.

L'energia cumulativa è un coefficiente che rappresenta il contributo che dà l'aggiunta di ciascuna variabile nel rappresentare la matrice Jacobiana decomposta. La matrice Jacobiana analizzata nel Grafico 4.3 raggiunge il valore della sua energia cumulativa attraverso 24 grandezze descrittive ed il numero di condizionamento calcolato nell'Eq. (4. 14) è sufficientemente basso per poter permettere l'identificazione cinematica.

Calcolando invece la regressione della matrice Jacobiana (4. 11) derivando solo lungo le variabili espresse dalla convenzione di DH si ottiene il Grafico 4.4. L'energia cumulativa raggiunge un valore pari a 1 utilizzando 22 variabili; ciò vuol dire che 22 variabili sono sufficienti a descrivere il 100% della matrice Jacobiana decomposta. Le restanti 2 variabili non contribuiscono alla caratterizzazione della matrice Jacobiana, bensì porteranno il sistema ad essere sovradeterminato. Infatti, osservando la curva degli autovalori, si osserva una riduzione di ben 17 ordini di grandezza nel passaggio da σ_{22} a σ_{24} , portando il numero di condizionamento da 22,24 con 22 variabili a $3,04e+18$ per 24 variabili. Le due variabili ridondanti che rendono il sistema sovradeterminato sono state identificate da Hayati [16] e sono i parametri d_i che collegano i giunti ad assi paralleli.

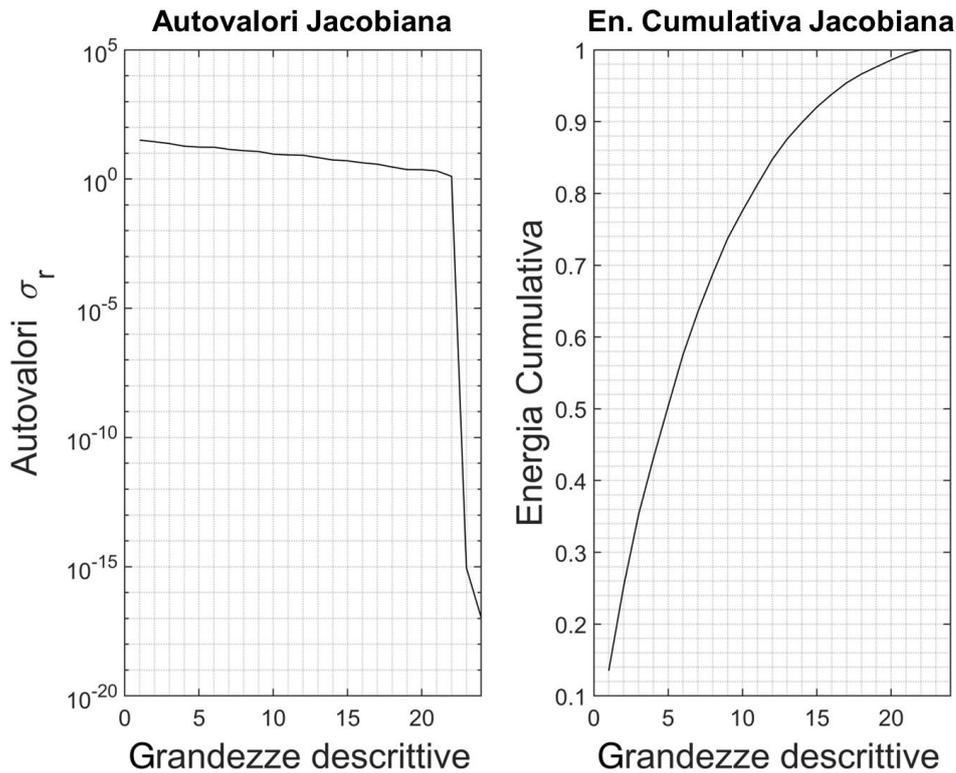


Grafico 4.4: Autovalori ed En. Cumulativa Jacobiana variabili DH

Il numero di condizionamento è molto impattante, valori troppo elevati di questo numero devono essere previsti ed evitati. Bisogna scegliere accuratamente le pose e selezionare come variabili i parametri cinematici corretti e strettamente necessari, per poter convergere ad un risultato.

Bisogna tenere in considerazione che ottenere l'energia cumulativa pari ad uno, non vuol dire che si riesce a descrivere l'intero sistema; vuol dire che è possibile descrivere in modo completo la matrice Jacobiana la cui regressione ha generato le informazioni nel Grafico 4.4, con 22 variabili. È ragionevole calcolare l'energia cumulativa della matrice Jacobiana nella convenzione di DH rappresentata nel Grafico 4.4 rispetto a quella utilizzata Grafico 4.3, ottenendo così il Grafico 4.5. Questo è possibile sostituendo al denominatore dell'Eq.(4. 15) della regressione della matrice in convenzione di DH, la sommatoria degli autovalori della matrice con la quale la si sta confrontando. Seppur il Grafico 4.4 converge dopo 22 variabili, questo non è in grado di descrivere tutta l'energia cumulativa del problema preso in considerazione.

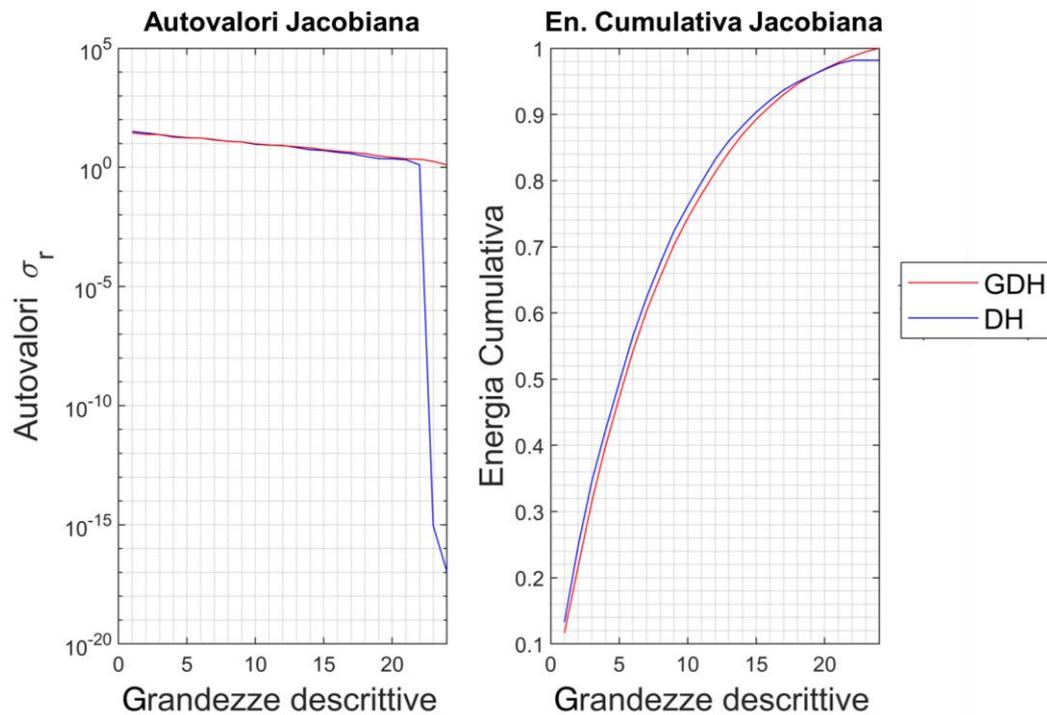


Grafico 4.5: Confronto autovalori e Energia Cumulativa modello GDH e modello DH

La differenza dell'energia cumulativa nella convenzione di DH rispetto alla convenzione GDH adottata in questa tesi, utilizzando 24 variabili, è dello 1,8%; questa leggera differenza giustifica la scelta di alcuni autori di troncare i parametri cinematici ridondanti in fase di calibrazione cinematica [24][28][25], senza ricercare un modello rappresentativo più completo.

4.1.3 Identificazione LSM o Gauss-Newton

Il secondo metodo utilizzato è quello di Gauss-Newton ed è quello più riscontrato nei lavori di calibrazione confrontati in Tabella 2.1.

L'identificazione in questo caso è di tipo iterativo, per risolvere il problema si usano le equazioni (2. 33) e (2. 34). Sostituendo un'equazione dentro l'altra si otterrà:

$$\Delta \mathbf{k}_{ag}^{(i)} = \Delta \mathbf{k}_{ag}^{(i-1)} + \left(\mathbf{J}^{(i)T} \mathbf{J}^{(i)} \right)^{-1} \mathbf{J}^{(i)T} \mathbf{e}_{ag}^{(i)} \quad (4. 16)$$

Utilizzando sempre gli stessi punti di convalida e quindi le variabili ai giunti utilizzati nell'identificazione del paragrafo precedente, si procede con il calcolo iterativo. Si inizializza il vettore $\Delta \mathbf{k}_{ag}$:

$$\Delta \mathbf{k}_{ag}^{(0)} = 0 \quad (4. 17)$$

Come criterio di convergenza si è scelto la radice quadrata della funzione obiettivo (2. 17) [45] , si procede con il calcolo in loop finché:

$$\sqrt{L} > 10^{-11} \quad (4. 18)$$

Nel Grafico 4.6 si ha l'andamento del parametro di convergenza selezionato al crescere delle iterazioni.

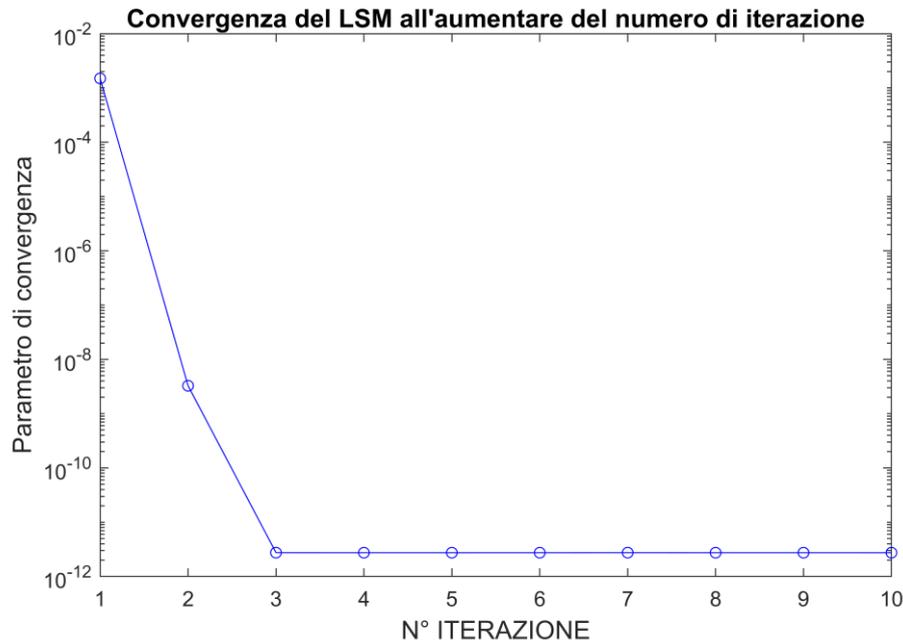


Grafico 4.6: Riduzione parametro di convergenza nel metodo LSM

L'algorithmo converge molto velocemente ad un minimo, sono necessarie sole tre iterazioni. Le modifiche ai parametri cinematici ottenute sono:

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,110	-0,156	0,184	0,025	-0,069	0,000
$\Delta \alpha$ [°]	-0,095	0,004	-0,232	-0,034	-0,013	0,000
$\Delta \beta$ [°]	0	0,013	0,071	0	0	0
Δd [mm]	-0,075	0	0	1,062	0,229	0,194
$\Delta \theta$ [°]	0,003	-0,019	-0,013	-0,008	0,009	-0,006

Tabella 4.4 Modifiche ai parametri cinematici identificata mediante LSM

Nel Grafico 4.7 è rappresentato l'andamento degli errori per i punti di convalida n_{cv} ed in Tabella 4.5 gli errori dati dalla proiezione geometrica e quelli ottenuti mediante il LSM, confrontati con i limiti di accettabilità della Universal Robots.

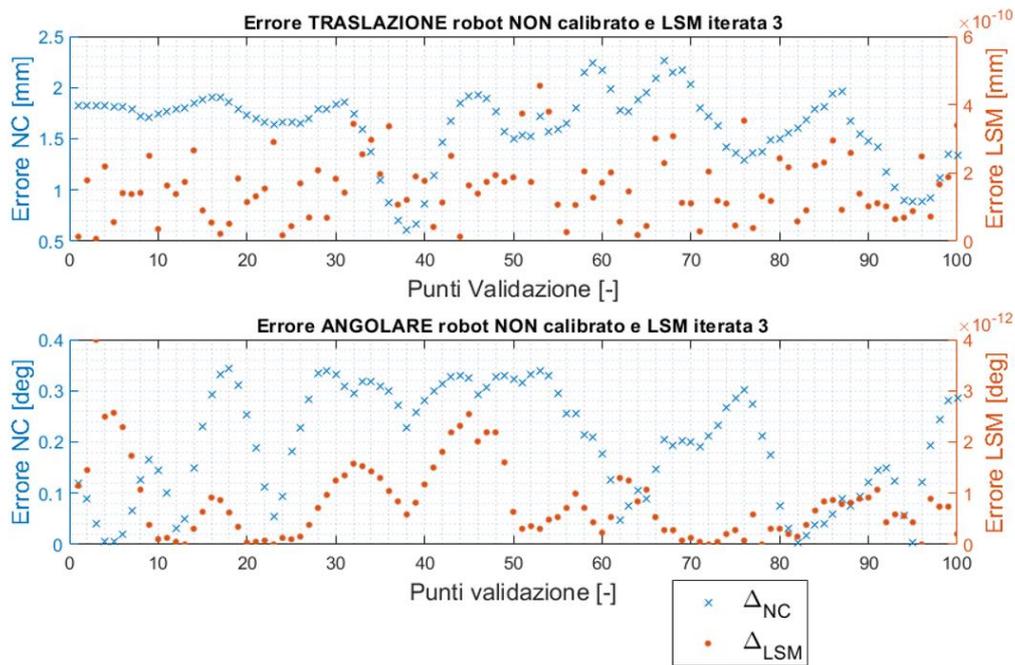


Grafico 4.7: Errori posa punti validazione da identificazione LSM (convenzione GDH)

Robot	Errore	LSM		Proiezione Analitica	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
UR5 PoliTO	Deviazione Standard	9,57E-11 (0,0%)	1,29E-11 (0,0%)	7,44E-11 (0,0%)	2,70E-12 (0,0%)
	Errore Medio	1,56E-10 (0,0%)	1,41E-11 (0,0%)	3,83E-10 (0,0%)	7,70E-13 (0,0%)
	Errore Massimo	4,55E-10	6,97E-11	6,74E-10	2,31E-11

Tabella 4.5: Confronto errori convalida dell'identificazione LSM e proiezione geometrica

I parametri identificati attraverso il LSM sono approssimabili a quelli ottenuti mediante la proiezione geometrica; infatti, le differenze tra i due sono così piccole da limitarsi ad essere delle semplici differenze numeriche trascurabili. Queste sono date dal fatto che vengono considerate tutte le cifre significative delle modifiche dei parametri cinematici ottenute mediante i due metodi di identificazione. Troncando i parametri cinematici al terzo ordine di grandezza si ottengono le stesse modifiche dei parametri cinematici come si può osservare nelle Tabella 4.4 e Tabella 3.4.

In Tabella 4.6 i risultati per il LSM per gli altri manipolatori considerati.

Robot	Errore	LSM	
		posizione [mm]	rotazione [mrad]
UR3 PoliTO	Deviazione Standard	3,00E-13 (0,0%)	2,36E-12 (0,0%)
	Errore Medio	5,96E-13 (0,0%)	3,36E-12 (0,0%)
	Errore Massimo	1,39E-12	1,15E-11
UR5 PoliTO	Deviazione Standard	9,57E-11 (0,0%)	1,29E-11 (0,0%)
	Errore Medio	1,56E-10 (0,0%)	1,41E-11 (0,0%)
	Errore Massimo	4,55E-10	6,97E-11
UR5 PST Srl	Deviazione Standard	5,26E-12 (0,0%)	6,79E-12 (0,0%)
	Errore Medio	1,20E-11 (0,0%)	1,11E-11 (0,0%)
	Errore Massimo	3,00E-11	2,69E-11
UR10 PST Srl	Deviazione Standard	7,89E-13 (0,0%)	3,85E-12 (0,0%)
	Errore Medio	1,24E-12 (0,0%)	4,67E-12 (0,0%)
	Errore Massimo	3,98E-12	1,73E-11

Tabella 4.6: Errori identificazione LSM per 4 robot diversi

4.2 Identificazione con algoritmi genetici

Si è voluto utilizzare gli algoritmi genetici per poter identificare i parametri cinematici, questi algoritmi permettono la soluzione di problemi molto complessi, senza dover necessariamente conoscere eventuali metodi risolutivi specifici al fine di minimizzare una certa funzione obiettivo.

Dato che non possono esistere problemi di mal condizionamento in quanto non si considera la Jacobiana, si può pensare di poter cercare delle modifiche ai parametri di DH anche nella convenzione standard che soddisfino la corretta posizione del TCP ed al contempo abbiano senso fisico.

Dei punti a disposizione, così come è stato fatto per le precedenti identificazioni, si usano il 70% per la prima fase detta di *training* che consiste nell'identificazione cinematica. Dopodiché si procede con la fase di validazione, con il restante 30%.

Durante la fase di *training* si cercano, come anticipato, le modifiche ai parametri di DH secondo due convenzioni: la convenzione di DH standard e la convenzione di GDH.

Grazie al “*Global Optimization Toolbox*” di Matlab è possibile utilizzare il metodo GA senza dover scrivere l’algoritmo manualmente.

Per poter utilizzare l’algoritmo è necessario specificare quali sono le variabili che si sta ricercando ed il range entro il quale queste si desidera che varino.

Le variabili scelte sono 24 per entrambi i SR e si ricercano delle soluzioni (modifiche ai parametri cinematici) in un range di $\pm 5\text{mm}$ e $\pm 2^\circ$. I risultati che si aspetta di ottenere dal robot sono delle correzioni dei parametri cinematici confrontabili con le tolleranze di produzione ed assemblaggio del robot, ossia dell’ordine del decimo di millimetro e del decimo di grado. Siccome si stanno ricercando delle piccole modifiche ai parametri cinematici, sarebbe interessante scegliere dei range più piccoli rispetto a quelli adottati per le variabili; tuttavia, l’algoritmo non risponde molto bene in presenza di un range troppo limitati (es $\pm 0.4\text{mm}$ e $\pm 0.2^\circ$), motivo per il quale si è proceduto lasciando dei range piuttosto ampi rispetto a quelli attesi.

Bisogna inoltre definire altri parametri che andranno a condizionare il funzionamento dell’algoritmo quali: popolazione dell’algoritmo, numero di generazioni massime, generazioni di stallo e tolleranza della soluzione. La tolleranza della soluzione viene fissata a 10^{-6} m, ordine di grandezza al di sotto del quale non avrebbe senso considerare valori più piccoli in quanto poco sensati da un punto di vista fisico. La popolazione scelta per ciascuna iterazione dell’algoritmo è di mille individui, vengono fissate inoltre il numero di iterazioni massimo posto pari a cento generazioni ed il numero massimo di generazioni di stallo, pari a 50 generazioni. Le altre caratteristiche dell’algoritmo quali le percentuali di crossover e mutazioni, vengono lasciate ad i valori di default del software di calcolo.

L’algoritmo genetico, a seconda della funzione di fitness utilizzata e dei criteri di convergenza e della tolleranza richiesta dalla soluzione, ricerca una condizione di minimo globale e le correzioni ai parametri di DH.

Sono state ricercate le modifiche dei parametri di DH usando tre metodi aumentando il livello di complessità, i metodi usati sono: l’algoritmo genetico

semplice, l'algoritmo genetico multiobiettivo ed infine l'algoritmo genetico multiobiettivo ibrido.

4.2.1 Algoritmo genetico semplice (GA)

Si è adottata come funzione di fitness ($F_{fitness}$) la somma degli errori di posizione e rotazionali fedelmente a quanto illustrato nel brevetto che descrive la calibrazione eseguita dalla UR [34]

$$F_{fitness} = E_{pos}^2 + E_{rot}^2 \quad (4.19)$$

Dove E_{pos} e E_{rot} sono la sommatoria della norma degli errori di posizione ed orientamento per ogni punto.

Prendendo gli errori di posizione ed orientamento descritti nella (3.51)

$$E_{pos}^2 = \sum_{j=1}^N \delta_{j_x}^2 + \delta_{j_y}^2 + \delta_{j_z}^2 = \sum_{j=1}^N e_{pos_j}^2 \quad (4.20)$$

$$E_{rot}^2 = \sum_{j=1}^N \gamma_j^2 = \sum_{j=1}^N e_{rot_j}^2 \quad (4.21)$$

Quindi la funzione che si vuole minimizzare con l'algoritmo genetico sarà:

$$F_{fitness} = \sum_{i=1}^N \delta_{i_x}^2 + \delta_{i_y}^2 + \delta_{i_z}^2 + \gamma_{i_\theta}^2 \quad (4.22)$$

Tutti gli errori vengono sommati per tutti i punti e si cerca di minimizzarne la somma quadratica. La funzione di fitness equivale alla funzione obiettivo definita nella (2.17) e (2.22).

Avviando l'algoritmo, durante la fase di training, questo converge dopo cento generazioni per entrambe le convenzioni (DH e GDH).

In Tabella 4.7 i valori delle funzioni di fitness ottenute ed i valori delle sue due componenti principali.

Parametri Cinematici UR5	E_{pos} [mm]	E_{rot} [mrad]	Ffitness
Non Calibrato	17,13	39,10	1822,6
GA GDH	31,75	27,87	1785,4
GA STD	30,70	57,27	4223,5

Tabella 4.7: Parametri funzione fitness algoritmo genetico

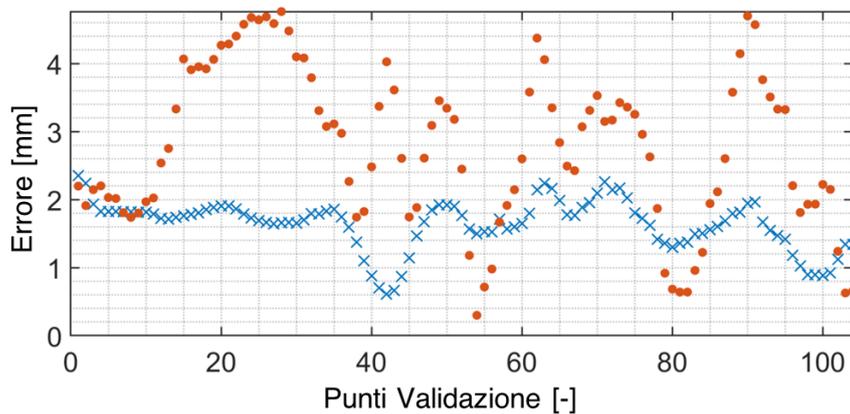
	Giunto					
	Giunto 1	Giunto 2	3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	-2,422	-0,866	-0,998	3,739	1,802	-0,581
$\Delta \alpha$ [°]	0,054	-0,028	0,202	0,431	-0,909	0,520
Δd [mm]	-0,848	1,725	0,342	0,464	0,741	-1,628
$\Delta \theta$ [°]	0,031	-0,550	0,169	-0,126	0,243	-0,542

Tabella 4.8: Modifiche ai parametri cinematici GA (Convenzione DH)

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	1,067	0,400	2,253	-0,240	-3,087	0,629
$\Delta \alpha$ [°]	-0,465	0,093	-0,540	0,405	0,368	0,108
$\Delta \beta$ [°]	0	0,305	0,573	0	0	0
Δd [mm]	0,027	0	0	3,005	1,047	0,484
$\Delta \theta$ [°]	-0,148	0,331	-0,288	0,387	-0,594	-0,473

Tabella 4.9: Modifiche ai parametri cinematici GA (Convenzione GDH)

Errore di traslazione Robot NON calibrato e Identificazione GA



Errore di rotazione Robot NON calibrato e Identificazione GA

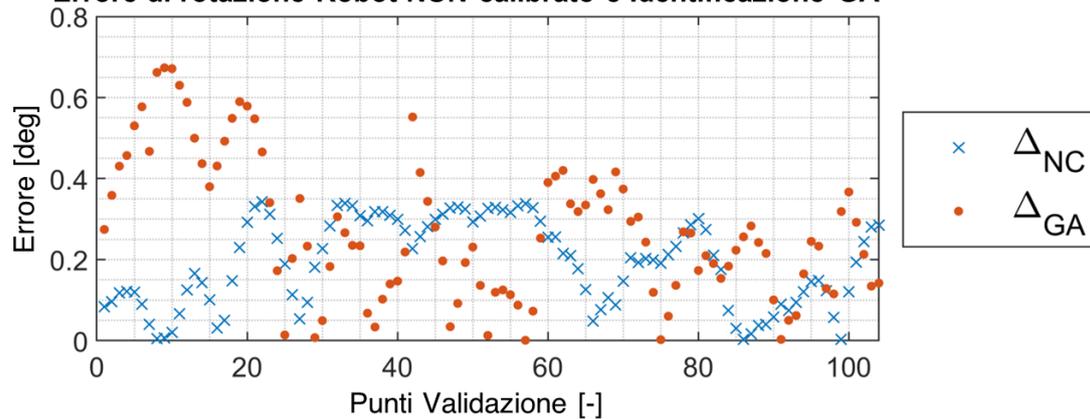


Grafico 4.8: Errori posa punti validazione da identificazione GA (convenzione DH)

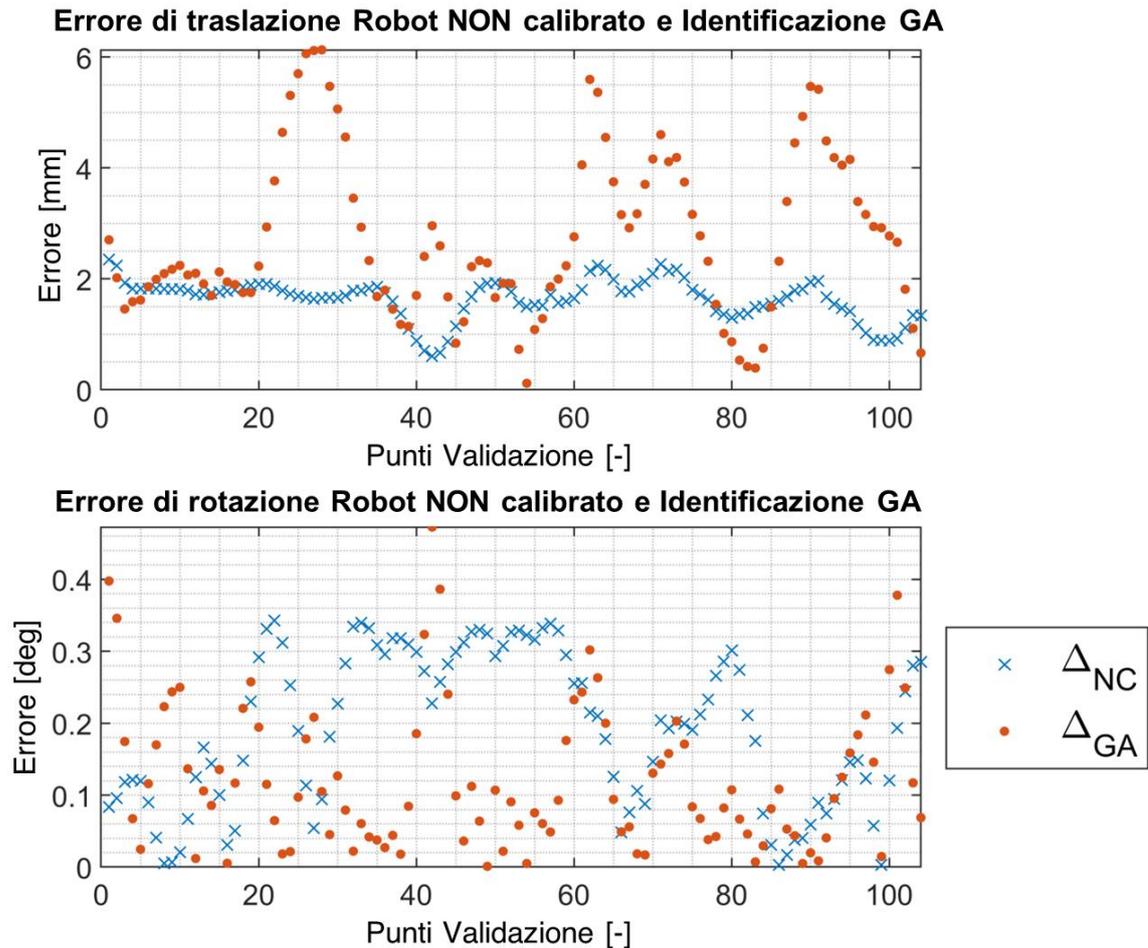


Grafico 4.9: Errori posa punti validazione da identificazione GA (convenzione GDH)

Come si può già notare dalle funzioni fitness in Tabella 4.7, l'algoritmo non è riuscito a diminuire di molto la funzione di fitness, non è attesa quindi una buona riduzione degli errori in fase di validazione.

Le modifiche ai parametri cinematici ottenute nella convenzione di DH e di GDH rispettivamente sono presenti nella Tabella 4.8 e Tabella 4.9.

Confrontando i risultati in Tabella 4.9 con quelli in Tabella 3.4 non si trova alcuna correlazione. Nel Grafico 4.8 e Grafico 4.9 gli errori per i punti di convalida nella convenzione DH e GDH. Analizzando le funzioni di fitness e le sue componenti presenti in Tabella 4.7, non si sono ottenuti miglioramenti; infatti, l'errore di posizione è aumentato in entrambe le convenzioni, mentre c'è un margine di miglioramento sull'errore di orientamento nella convenzione GDH.

In Tabella 4.10 le caratteristiche degli errori ottenuti confrontate con i la soglia di accettabilità imposta dalla Universal Robots.

Identificaz.	Errore	DH		GDH	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
GA	Deviazione Standard	1,151 (230,2%)	3,004 (300,4%)	1,478 (295,6%)	1,749 (174,9%)
	Errore Medio	2,785 (278,5%)	4,754 (237,7%)	2,744 (274,4%)	2,108 (105,4%)
	Errore Massimo	4,764	11,754	6,131	8,249

Tabella 4.10: Caratteristiche errori identificazione GA

Questi risultati portano con l'esigenza di aumentare il livello di complessità degli algoritmi genetici da utilizzare. Infatti, risultati così poco soddisfacenti potrebbero essere dovuti al fatto che, nella funzione obiettivo, si sommano traslazioni con rotazioni, quando queste dovrebbero essere almeno normalizzate.

4.2.2 Algoritmo Genetico Multiobiettivo (Multi-GA)

Date le considerazioni precedentemente effettuate sull'algoritmo genetico standard, si è deciso di modificare la funzione di fitness per cercare di ottenere un risultato accettabile. Gli errori E_{rot} e E_{pos} da come si può osservare nella Tabella 4.7 non possiedono la stessa unità di misura.

Si decide di dividere la funzione di fitness precedente in due e cercare una soluzione che minimizzi due funzioni obiettivo: E_{rot} ed E_{pos} . Gli algoritmi genetici con più di una funzione obiettivo vengono detti Algoritmi Genetici Multiobiettivo.

Per minimizzare i due obiettivi è necessario definire su Matlab la Ffitness come una funzione in due variabili da minimizzare:

$$Ffitness = \begin{pmatrix} E_{pos} \\ E_{rot} \end{pmatrix} \quad (4.23)$$

Le due funzioni di costo non è detto che presentino contemporaneamente un minimo assoluto per la stessa soluzione. Infatti, attraverso il GA multi-obiettivo si ottiene come output un insieme di soluzioni, rappresentabili sul Diagramma di Pareto nel Grafico 4.10, ottenuto con 50 popolazioni.

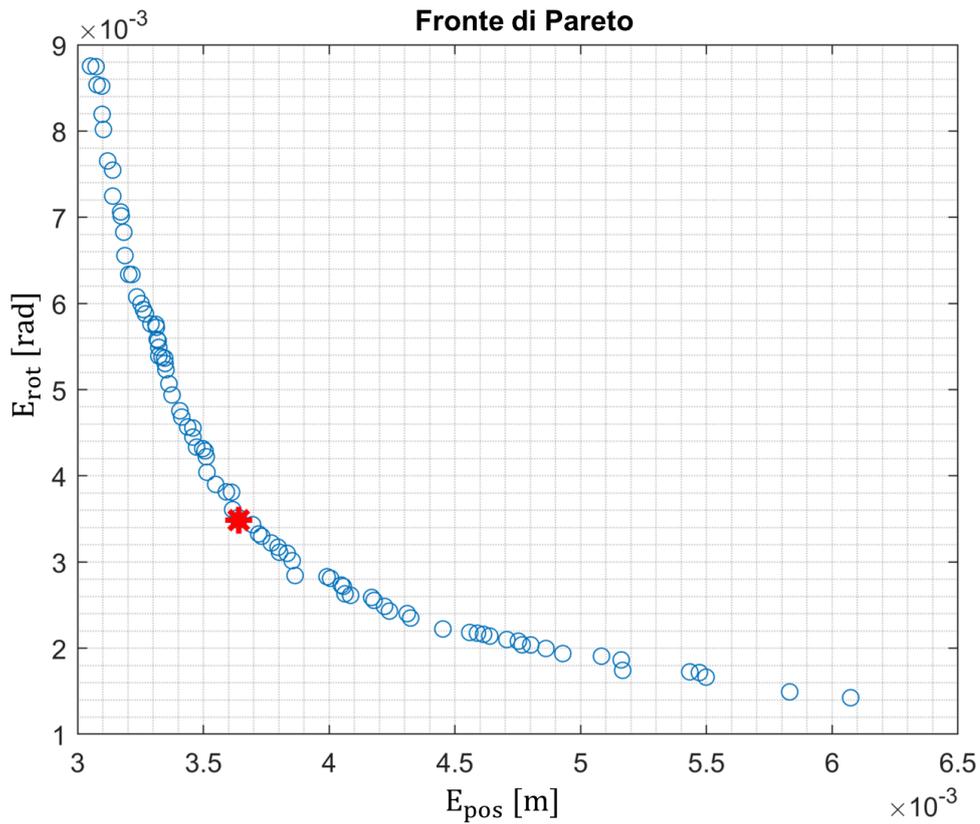


Grafico 4.10: Diagramma di Pareto Multi-GA (convenzione GDH)

Ciascuna soluzione porta una riduzione di entrambe le funzioni costo. Si può osservare dal diagramma che, favorendo la minimizzazione di una delle funzioni di fitness, non si ottiene una riduzione ottimale dell'altra.

Tra le possibilità si sceglie la soluzione che può essere considerata come ottimo, presente sul ginocchio del Diagramma di Pareto identificata dal segno rosso.

Per la soluzione di ottimo si ottengono le seguenti funzioni di fitness:

Parametri Cinematici UR5	E_{pos} [mm]	E_{rot} [mrad]
Non Calibrato	17,13	39,10
Multi-GA GDH	3.486	3.640
Multi-GA DH	2,940	5,606

Tabella 4.11 Parametri funzione fitness Multi-GA

Confrontando le funzioni di fitness ottenute con i risultati ottenuti mediante gli algoritmi genetici, si ha per entrambe le convenzioni una riduzione di E_{pos} e E_{rot} significativa. Si ottengono delle modifiche dei parametri cinematici anche nella convenzione di DH, e da quanto si può notare dalle funzioni obiettivo in Tabella

4.11 presentano una riduzione delle funzioni obiettivo. In Tabella 4.12 e Tabella 4.13 le modifiche ai parametri cinematici ottenute nelle due convenzioni.

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,019	0,071	0,314	0,061	-0,183	-0,052
$\Delta \alpha$ [°]	-0,092	-0,017	-0,189	-0,049	0,000	0,028
Δd [mm]	0,028	0,267	0,144	0,318	0,189	0,208
$\Delta \theta$ [°]	0,017	-0,018	-0,051	0,056	0,041	0,007

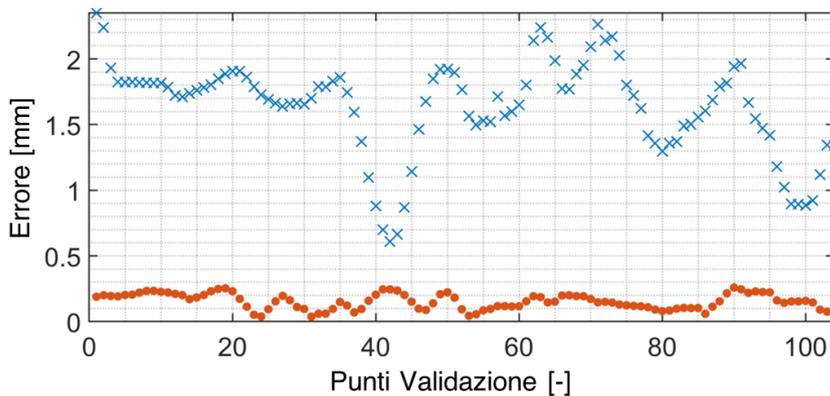
Tabella 4.12: Modifiche ai parametri cinematici Multi-GA (Convenzione DH)

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	-0,118	0,046	0,271	0,041	-0,344	0,026
$\Delta \alpha$ [°]	-0,098	0,008	-0,204	-0,011	-0,016	0,043
$\Delta \beta$ [°]	0	0,000	0,060	0	0	0
Δd [mm]	-0,099	0	0	0,642	0,165	0,203
$\Delta \theta$ [°]	0,047	-0,029	-0,034	0,039	-0,006	-0,076

Tabella 4.13: Modifiche ai parametri cinematici Multi-GA (Convenzione GDH)

Nel Grafico 4.11 e Grafico 4.12, l'andamento degli errori per le due convenzioni.

Errore di traslazione Robot NON calibrato e Identificazione Multi-GA



Errore di rotazione Robot NON calibrato e Identificazione Multi-GA

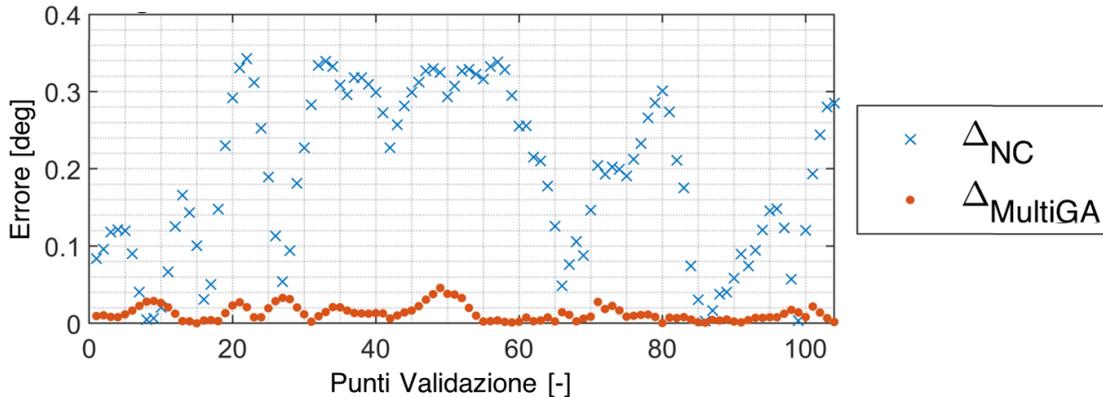
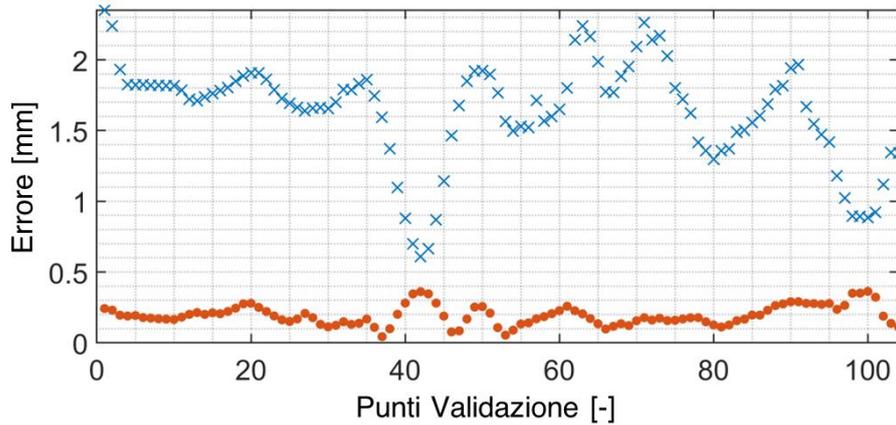


Grafico 4.11: Errori posa punti validazione da identificazione Multi-GA (convenzione DH)

Confrontando la Tabella 4.13 con quelli in Tabella 3.4, si raggiunge un risultato prossimo a quello identificato analiticamente mediante la proiezione geometrica; gli errori si sono ridotti e la media della differenza tra i parametri cinematici delle due tabelle raggiungono i 50 μm e gli 0.02° , con una differenza massima tra i parametri cinematici pari a 0.42 mm e $0,07^\circ$.

Errore di traslazione Robot NON calibrato e Identificazione Multi-GA



Errore di rotazione Robot NON calibrato e Identificazione Multi-GA

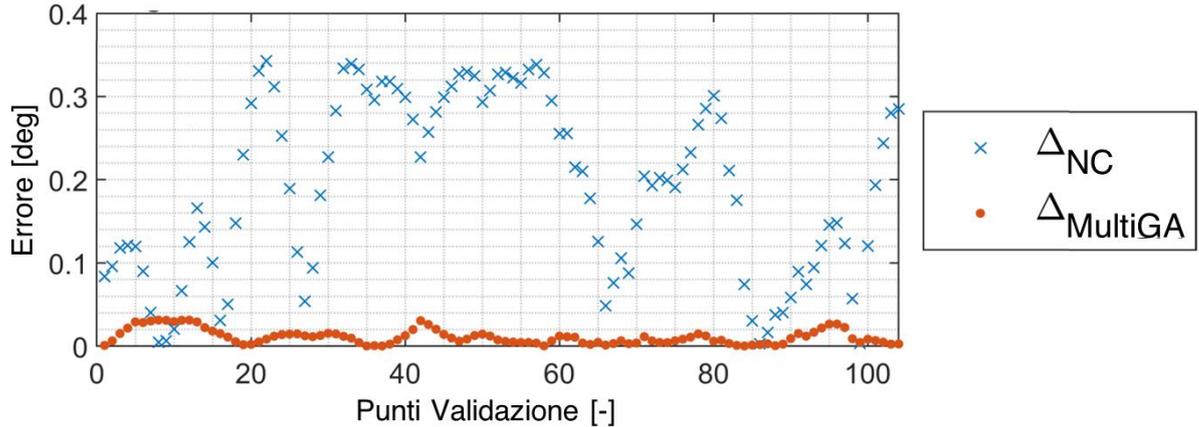


Grafico 4.12: Errori posa punti validazione da identificazione Multi-GA (convenzione GDH)

Le caratteristiche degli errori ottenute rispetto ai limiti di accettabilità sono:

Algoritmo	Errore	DH		GDH	
		posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
Multi-GA	Deviazione Standard	5,87E-02 (11,7%)	0,222 (17,5%)	6,94E-02 (13,9%)	0,193 (15,5%)
	Errore Medio	0,153 (15,3%)	0,222 (11,1%)	0,194 (19,4%)	0,193 (9,7%)
	Errore Massimo	0,259	0,799	0,363	0,548

Tabella 4.14: Caratteristiche errori identificazione Multi-GA

Si hanno netti miglioramenti per entrambe le convenzioni, questi errori rientrano ampiamente all'interno dei criteri di accettabilità della Universal Robots.

Seppur si ottengono dei miglioramenti delle funzioni obiettivo ricercate, date le considerazioni fatte nel terzo capitolo, ci si aspetta degli errori minori nella convenzione GDH. Bisogna tenere inoltre in considerazione che questa è una simulazione e non sono presenti errori di ripetibilità, non geometrici e del sistema di misura. Considerando che mediante la proiezione geometrica si sono raggiunti errori con ordine di grandezza confrontabile a quella della precisione di macchina, si sceglie di complicare ulteriormente il metodo di identificazione stocastico, al fine di ricercare delle modifiche ai parametri cinematici con senso fisico con un errore maggiormente ridotto.

4.2.3 Algoritmo Genetico Multiobiettivo ibrido (Multi-GA Ibrido)

Attraverso gli algoritmi geometrici multi-obiettivo si è riscontrato un miglioramento dei risultati, ma bisogna ricordare che si sta lavorando con dei sistemi contenenti un elevato numero di variabili. Gli algoritmi genetici non garantiscono la convergenza dato che la ricerca delle soluzioni è casuale. Considerando che viene generato un insieme di soluzioni appartenente al diagramma di Pareto, sarebbe necessario un elevato numero di popolazioni e quindi di tempo di calcolo per poter cercare di avvicinarsi maggiormente ad un minimo tale da poter generare dei risultati soddisfacenti. Inoltre, data la natura stocastica dell'algoritmo, non potrebbe mai esserci la garanzia che il risultato raggiunto sia di fatto un minimo. A tal proposito si decide di far uso degli Algoritmi Genetici Ibridi Multiobiettivo, che seppur continuano a non garantire la certezza di convergere ad un minimo assoluto, sfruttano metodi ibridi per cercare di minimizzare di più e con meno iterazioni le funzioni di fitness selezionate.

Questo algoritmo combina la capacità degli algoritmi genetici multi-obiettivo di avvicinarsi ad una serie di presunti minimi globali, e la capacità di convergere ad un minimo locale dei metodi a gradiente.

L'algoritmo genetico multi-obiettivo ibrido consiste in due ottimizzazioni successive:

- 1) Ottimizzazione Globale mediante Algoritmo Genetico Multiobiettivo (creazione di un fronte di Pareto provvisorio).
- 2) Ottimizzazione locale mediante il metodo di quasi-Newton di ciascuno dei minimi globali ottenuti precedentemente (si minimizzano le soluzioni del fronte di Pareto precedente e se ne crea uno nuovo).

I minimi locali dei minimi globali rappresenteranno i possibili minimi globali del problema ed andranno a formare un nuovo fronte di Pareto.

Aggiungendo il secondo step di ottimizzazione locale, a seguito della prima ottimizzazione globale, si otterranno dei minimi per i vari intorno delle soluzioni globali. Il primo step di ottimizzazione non può garantire da solo il raggiungimento di un minimo in quanto si tratta di un metodo stocastico.

Lo step d'ottimizzazione aggiuntivo viene implementato attraverso Matlab specificando che si desidera l'uso di un Algoritmo Genetico Multiobiettivo ibrido. È necessario richiamare l'oggetto delle impostazioni dell'algoritmo genetico e specificare nel campo HybridFcn che si vuole usare la funzione 'fggoalattain'. Attraverso questa funzione Matlab applicherà la funzione 'fmincon' a tutte le soluzioni del diagramma di Pareto.

Come risultato, specificando il tipo di funzione ibrida che si vorrà usare, si otterrà come output il fronte di Pareto finale, ossia quello ottenuto a seguito di entrambi i passi di ottimizzazione.

La funzione 'fmincon' consiste in un metodo di identificazione a gradiente, si usa la metodologia quasi-Newton, che è una modifica del metodo di Newton descritto nel capitolo 3. La differenza sta nel fatto che la matrice Hessiana viene approssimata, essa viene calcolata dal rapporto incrementale finito dei gradienti di due passi d'iterazione consecutivi.

Nel Grafico 4.13 il diagramma di Pareto ottenuto per la convenzione GDH; il fronte di Pareto che si ottiene presenta meno punti rispetto al Pareto senza l'ottimizzazione locale (Grafico 4.10). I valori delle funzioni di fitness risultano più 'schiate' sui due assi, segnale del fatto che ne sono stati ridotti ulteriormente i valori.

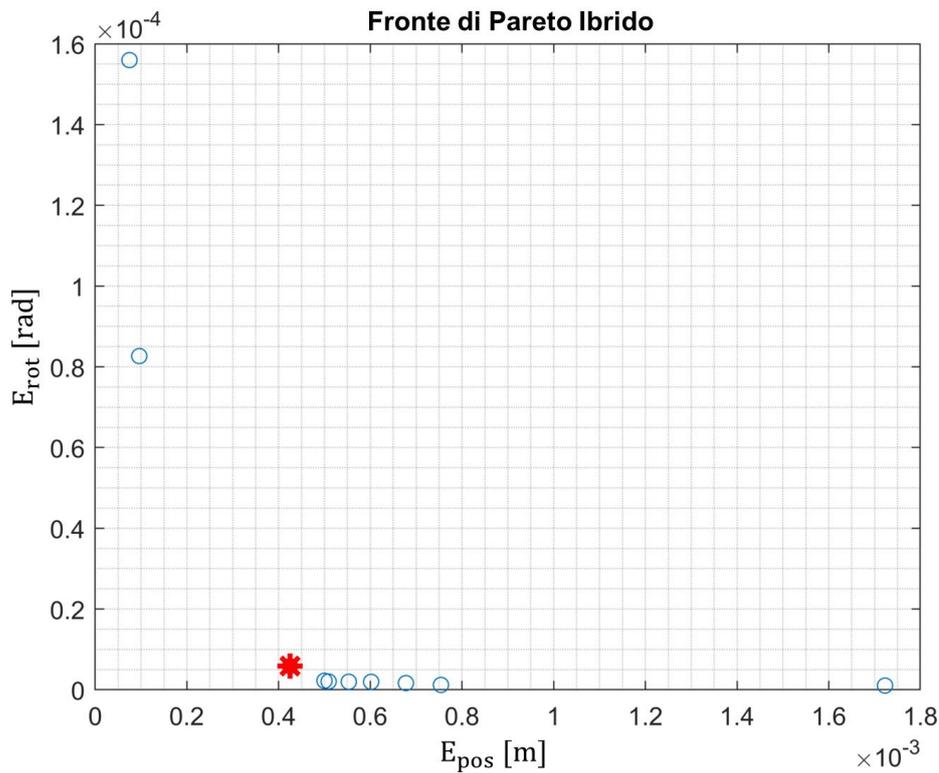


Grafico 4.13 Diagramma di Pareto Multi-GA Ibrido (convenzione GDH)

Le soluzioni di ottimo selezionate presentano le seguenti funzioni obbiettivo:

Parametri Cinematici UR5	E_{pos} [mm]	E_{rot} [mrad]
Non Calibrato	162,4	341,8
GA Multiobbiettivo Ibrido GDH	0,425	5.85E-03
GA Multiobbiettivo Ibrido DH	1,746	5,59E-03

Tabella 4.15: Parametri funzione fitness Multi-GA Ibrido

Come ci si aspettava, in Tabella 4.15, si ottiene un minimo globale con la convenzione GDH avente delle funzioni obbiettivo più piccoli di quelle nella convenzione DH standard.

Le modifiche ai parametri cinematici ottenute nelle due convenzioni sono:

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,083	-0,119	0,330	0,042	-0,066	0,018
$\Delta \alpha$ [°]	-0,097	-0,011	-0,195	-0,049	-0,018	0,022
Δd [mm]	-0,074	0,428	0,788	-0,253	0,236	0,174
$\Delta \theta$ [°]	0,012	-0,016	-0,029	0,010	0,033	0,018

Tabella 4.16: Modifiche ai parametri cinematici Multi-GA Ibrido (Convenzione DH)

Come si può notare dalle modifiche dei parametri in convenzione DH non sono comparabili con quelli ottenuti mediante l'identificazione geometrica effettuata nel capitolo 3.

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
Δa [mm]	0,098	-0,158	0,171	0,024	-0,035	-0,011
$\Delta \alpha$ [°]	-0,095	0,003	-0,232	-0,034	-0,013	0,000
$\Delta \beta$ [°]	0	0,013	0,070	0	0	0
Δd [mm]	-0,093	0	0	1,116	0,229	0,194
$\Delta \theta$ [°]	-0,001	-0,024	-0,011	-0,005	0,009	-0,002

Tabella 4.17: Modifiche ai parametri cinematici Multi-GA Ibrido (Convenzione GDH)

Invece le modifiche dei parametri nella convenzione GDH raggiungono un risultato prossimo a quello identificato analiticamente mediante la proiezione geometrica; infatti, confrontando le modifiche dei parametri ottenute in Tabella 3.4 e i risultati appena ottenuti Tabella 4.17, si ha una differenza media tra i parametri cinematici di $3\mu\text{m}$ e 10^{-5}° , ed un valore massimo della differenza pari a $18\mu\text{m}$ e $0,005^\circ$.

Gli errori di posizione e di orientamento riscontrati nelle due convenzioni sono:

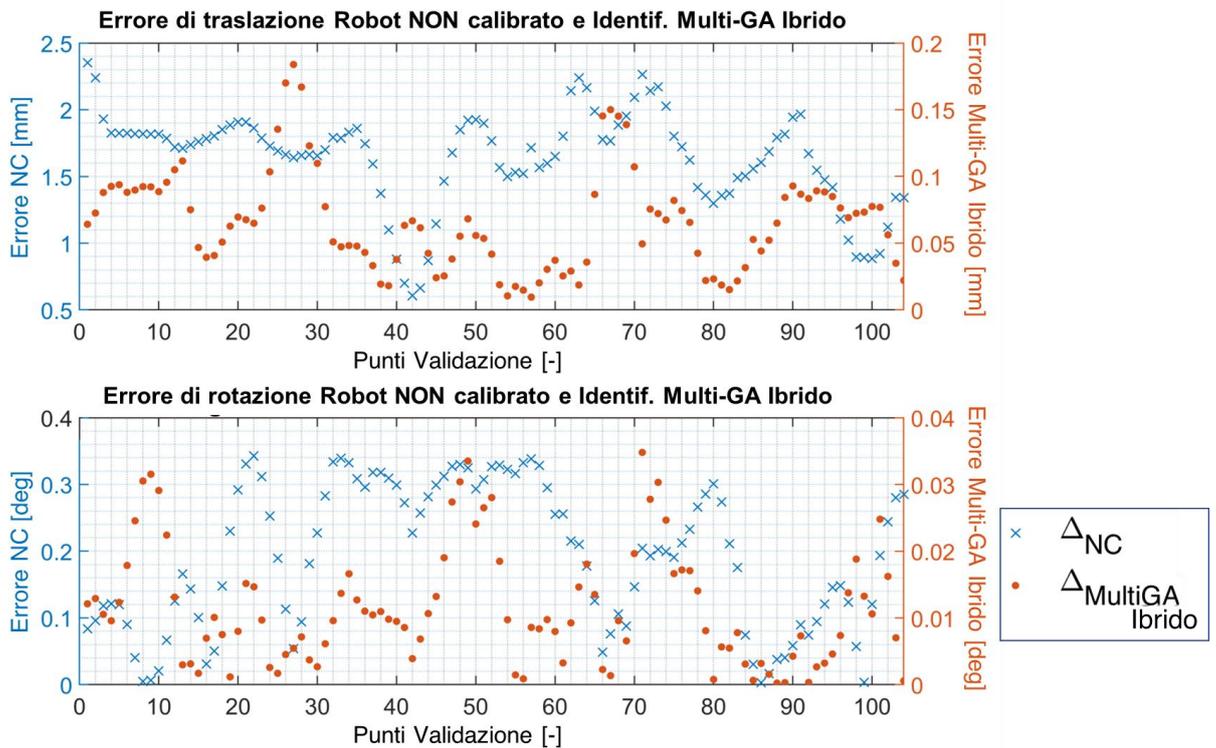


Grafico 4.14: Errori posa punti validazione da identificazione Multi-GA Ibrido (convenzione DH)

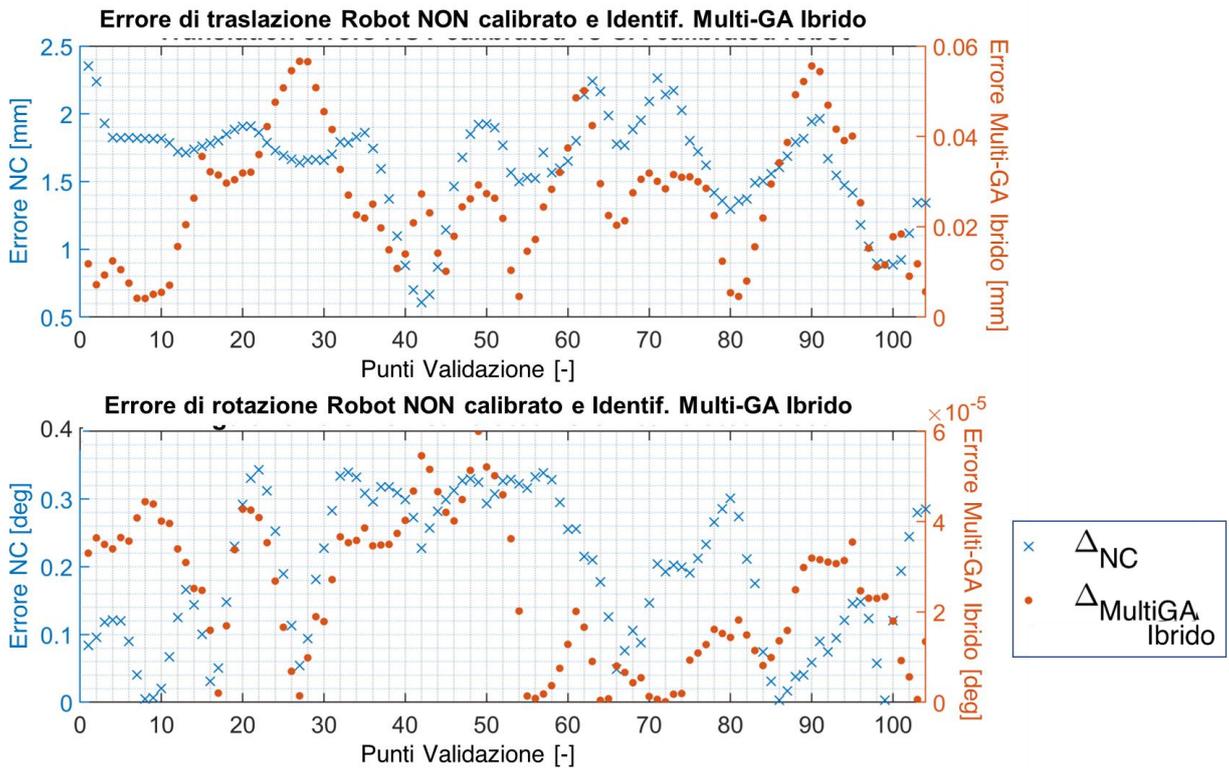


Grafico 4.15: Errori posa punti validazione da identificazione Multi GA Ibrido (convenzione GDH)

Confrontando gli errori ottenuti con le soglie di accettabilità indicate dalla Universal Robot si ottiene:

		DH		GDH	
Algoritmo	Errore	posizione [mm]	rotazione [mrad]	posizione [mm]	rotazione [mrad]
Multi-GA Ibrido	Deviazione Standard	3,729E-02 (7,5%)	0,1538 (15,4%)	1,42E-02 (2,8%)	2,76E-04 (0,03%)
	Errore Medio	6,601E-02 (6,6%)	0,2005 (10,0%)	0,026 (2,6%)	4,24E-04 (0,02%)
	Errore Massimo	0,1838	0,6073	0,057	1,05E-03

Tabella 4.18: Caratteristiche errori identificazione GA Multiobbiettivo Ibrido

Passando dalla convenzione di DH a quella GDH, mediante questo metodo di identificazione, si ha una riduzione degli errori di posizione e di orientamento presenti in Tabella 4.18.

Nella convenzione GDH diminuiscono gli errori medi di posizione e di orientamento, rispetto ai limiti imposti dalla UR, rispettivamente del 97.4% e del 99,8%; si può affermare quindi che anche attraverso un metodo stocastico e senza calcolare la Jacobiana è possibile identificare dei parametri cinematici con senso fisico per l'UR5.

Da questi risultati emerge il limite della convenzione di DH che non può raggiungere, per piccole modifiche dei parametri cinematici, l'accuratezza ottenibile mediante la convenzione GDH, in quanto il sistema è sovradeterminato. Le modifiche ai valori cinematici ottenuti nella convenzione di DH sono una combinazione di parametri tali per cui si riesce a ridurre il più possibile le funzioni di fitness scelte. Queste modifiche però, date tutte le considerazioni fatte nel terzo capitolo, non sarebbero realmente rappresentative del sistema preso in oggetto; infatti, si è mostrato che in presenza di giunti paralleli modellati nella convenzione di DH standard, è impossibile ottenere delle piccole perturbazioni dei parametri cinematici. L'algoritmo, a causa di questi limiti, ha trovato dei minimi prossimi, ma non potrebbe mai descrivere in modo completo il sistema cinematico reale, poiché il modello, in presenza di piccole perturbazioni dei parametri cinematici, è insufficiente. Seppur teoricamente inadatto, il modello di DH fornisce dei risultati accettabili, questo è stato possibile grazie all'uso di un metodo numerico stocastico.

Capitolo 5

5 Errori geometrici

Un manipolatore può accusare nel tempo un deterioramento della sua capacità di posizionarsi in modo accurato nello spazio, questo può portare a una riduzione della qualità della produzione e dell'efficienza produttiva. È importante comprendere come e quanto degrada la “salute posizionale dei sistemi robotici” in modo da poter ottimizzare le strategie di manutenzione e controllo a livello di sistema.

In letteratura è stato sviluppato un modello *high fidelity* capace di descrivere tutti gli errori presenti, sia di natura geometrica che di natura non geometrica [1]; in questo modello vengono introdotte delle matrici di errore che contengono tutte le fonti di errore. I parametri cinematici identificati nel terzo e nel quarto capitolo verranno utilizzati per implementare questo modello dal forte significato fisico.

Per ciascuno dei sei assi giunto del manipolatore vengono considerati sei errori spaziali: tre spostamenti nello spazio (direzioni x, y e z) e tre errori di rotazione degli assi (*roll, pitch yaw*). In Figura 5.1 si mostra un asse di rotazione i-esimo e gli errori anzidetti .

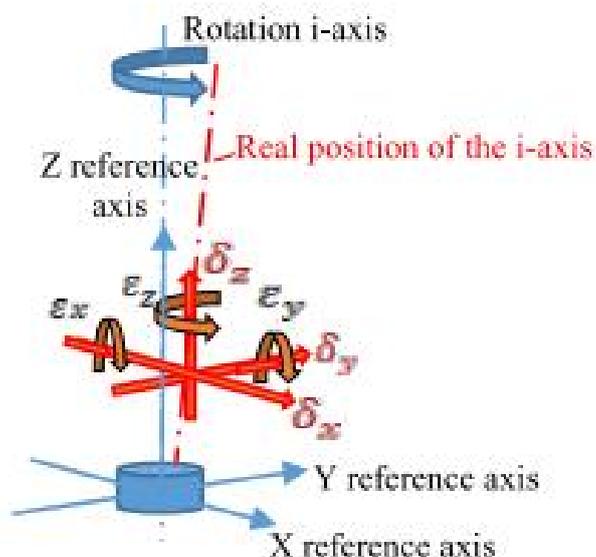


Figura 5.1: Errori geometrici asse i-esimo [1]

Gli errori degli assi dei giunti (lineari o rotativi) descritti come errori geometrici possono essere funzione della posizione del giunto in quanto possono inglobare errori non geometrici.

In Figura 5.1, l'asse reale è stato deviato dalla posizione di progetto, la ragione della deviazione possono essere data da una combinazione di errori geometrici, errori non geometrici o anche da gli effetti della degradazione del sistema.

I sei errori di questo asse sono così definiti :

δ_x - errore di spostamento in direzione X;

δ_y - errore di spostamento in direzione Y;

δ_z - errore di spostamento in direzione Z;

ε_x - errore di rotazione attorno all'asse X;

ε_y - errore di rotazione attorno all'asse Y;

ε_z - errore di posizionamento angolare (anche detto errore di scala dell'asse di rotazione).

Il modello di errore del giunto è descritto come:

$$E_{(i-1)}^i(\theta) = \begin{bmatrix} 1 & -\varepsilon_z(\theta) & \varepsilon_y(\theta) & \delta_x(\theta) \\ \varepsilon_z(\theta) & 1 & -\varepsilon_x(\theta) & \delta_y(\theta) \\ -\varepsilon_y(\theta) & \varepsilon_x(\theta) & 1 & \delta_z(\theta) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5. 1)$$

Dove $E_{(i-1)}^i$ sarà la trasformazione che descrive questi errori, questa matrice si interpone tra le matrici di trasformazione e precede la matrice $A_{(i-1)}^i(\theta)$ che porta dall'asse i-1 al quello i-esimo. Come si nota dalla (5. 1) questi errori sono definiti al variare di θ e non sono necessariamente da considerarsi costanti; questa caratteristica permette di modellare non solo gli errori geometrici, i quali sono indipendenti dalla posizionamento del robot, ma bensì, anche gli errori non geometrici e le particolari degradazioni dei giunti i quali possono non essere costanti al variare della variabile del giunto. Per questa applicazione andremo a utilizzare le modifiche dei parametri cinematici GDH ottenute mediante la proiezione geometrica, nel terzo capitolo. Si trasformeranno i parametri cinematici in matrici di errori geometrici. Gli errori geometrici che andranno a comporre la matrice di errore andranno ad assumere tutti un valore costante, in quanto le modifiche ai parametri cinematici sono costanti.

Seppur non si sfrutta a pieno la potenzialità di questo modello, dato che si identificano degli errori geometrici costanti, gli effetti sull'accuratezza resteranno quelli presenti a seguito dell'identificazione cinematica; come quanto detto nel secondo capitolo, il 90% percento dell'inaccuratezza migliora a seguito della calibrazione cinematica [15], il solo uso dei parametri cinematici è molto significativo ai fini dell'accuratezza del manipolatore. Si massimizzano i risultati minimizzando gli sforzi computazionali dato che non si dovranno identificare delle funzioni per ciascun errore. La matrice errore costante sarà:

$$E_{(i-1)}^i = \begin{bmatrix} 1 & -\varepsilon_z & \varepsilon_y & \delta_x \\ \varepsilon_z & 1 & -\varepsilon_x & \delta_y \\ -\varepsilon_y & \varepsilon_x & 1 & \delta_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.2)$$

Dato che questi errori hanno uno spiccato senso fisico, per sfruttare al meglio questo modello, anche le modifiche dei parametri di DH devono avercelo. In caso contrario gli errori che verrebbero ottenuti nella matrice d'errore non avrebbero alcun senso fisico o, ancora peggio, porterebbero a delle equazioni indeterminate.

La necessità di un senso fisico va a giustificare la necessità dell'identificazione dei parametri cinematici effettuata in questo lavoro di tesi, perché con le identificazioni dei parametri fatta dal costruttore, non sarebbe possibile costruire un modello high-fidelity della macchina.

Il modello di cinematica diretta per esprimere la posa del TCP è:

$$H_0^6 = A_0^1 E_0^1 A_1^2 E_1^2 A_2^3 E_2^3 A_3^4 E_3^4 A_4^5 E_4^5 A_5^6 E_5^6 \quad (5.3)$$

Dove A_{i-1}^i sono le matrici di trasformazione omogenea degli assi giunti espressi espressi nella convenzione di DH standard; queste matrici utilizzano i parametri di DH della casa costruttrice da modello.

L'obbiettivo è quello di calcolare le 6 matrici di errore ed i 6 errori presenti presenti in ciascuna matrice $E_{(i-1)}^i$. Si procede dunque identificando la soluzione analitica di ciascun errore a partire dalle modifiche cinematiche ottenute nella convenzione GDH.

Si scrive il seguente sistema di equazioni per ciascun *link*:

$$A_{i-1}^i E_{i-1}^i = \mathcal{A}_{i-1}^i \quad (5.4)$$

Dove \mathcal{A}_{i-1}^i rappresenta la matrice di trasformazione omogenea di due assi giunti consecutivi di un *link*, nella convezione di GDH con i parametri di DH identificati

con senso fisico. Dato che si sta cercando una soluzione analitica, per ridurre il numero di variabili dove possibile e semplificare le equazioni, i *link* 1, 4, 5 e 6 sono espressi nella convezione di DH standard, mentre i *link* 2 e 3 vengono espressi nella convezione GDH.

Otterremo così due set di 12 equazioni, rispettivamente validi per le due convenzioni prese in considerazione.

I 2 set di equazioni saranno organizzati in forma matriciale in una matrice 4x4 l'ultima riga della matrice che racchiude le equazioni sarà composta da soli 0 ed 1 e risulta poco interessante, saranno presenti dodici equazioni interessanti così disposte:

$$equazioni = \begin{bmatrix} (1) & (4) & (7) & (10) \\ (2) & (5) & (8) & (11) \\ (3) & (6) & (9) & (12) \\ - & - & - & - \end{bmatrix} \quad (5.5)$$

Da queste 12 equazioni si estrapolano delle equazioni utili per ottenere gli errori, non tutte verranno utilizzate per ottenere gli errori ricercati.

Soluzioni analitiche *Link* 1, 4, 5, 6

Le prime tre equazioni contenenti i tre errori di spostamento saranno prese dalle equazioni 10, 11 e 12 e saranno:

$$\begin{cases} a_n c(\theta_n) + \delta_x c(\theta_n) - \delta_y c(\alpha_n) s(\theta_n) + \delta_z c(\alpha_n) s(\theta_n) = a_c c(\theta_c) \\ a_n s(\theta_n) + \delta_x s(\theta_n) - \delta_y c(\alpha_n) c(\theta_n) + \delta_z s(\alpha_n) c(\theta_n) = a_c s(\theta_c) \\ d_n + \delta_z c(\alpha_n) + \delta_y s(\alpha_n) = d_c \end{cases} \quad (5.6)$$

Dove i pedici n e c rappresentano rispettivamente i parametri cinematici nominali e quelli identificati con senso fisico.

Le successive tre equazioni contenenti i tre errori di orientamento saranno prese dalle equazioni 2, 3, 6 e 9.

le equazioni 2 e 3 dipendono dai valori di α_n e α_c .

$$\begin{cases} s(\alpha_n) + \varepsilon_x c(\alpha_n) = s(\alpha_c) & se \alpha_n \neq \pm \frac{\pi}{2} \\ c(\alpha_n) - \varepsilon_x s(\alpha_n) = c(\alpha_c) & se \alpha_n = \pm \frac{\pi}{2} \end{cases} \quad (5.7)$$

Dato che α_n assume valori paria a 0 e $\pm \frac{\pi}{2}$, si usano due distinte equazioni al variare dei valori del parametro, per evitare un sistema indeterminato.

Fatte le considerazioni su α_n il sistema di equazioni per determinare l'orientamento sarà:

$$\begin{cases} s(\alpha_n) + \varepsilon_x c(\alpha_n) = s(\alpha_c) & \text{se } \alpha_n \neq \pm \frac{\pi}{2} \\ c(\alpha_n) - \varepsilon_x s(\alpha_n) = c(\alpha_c) & \text{se } \alpha_n = \pm \frac{\pi}{2} \\ s(\theta_n) + \varepsilon_z c(\alpha_n) c(\theta_n) + \varepsilon_y s(\alpha_n) c(\theta_n) = s(\theta_c) \\ \varepsilon_z s(\alpha_n) - \varepsilon_y c(\alpha_n) = 0 \end{cases} \quad (5.8)$$

Attraverso il calcolo simbolico di MATLAB e raggruppando alcune formule trigonometriche è possibile ottenere la soluzione analitica dei tre sistemi di equazioni, definendo l'espressione analitica dei 6 errori geometrici.

A seguire le equazioni che definiscono gli errori geometrici.

$$\begin{cases} \delta_x = a_c c(\theta_n - \theta_c) - a_n \\ \delta_y = (d_c - d_n) s(\alpha_n) + a_c c(\alpha_n) s(\theta_c - \theta_n) \\ \delta_z = (d_c - d_n) c(\alpha_n) + a_c s(\alpha_n) s(\theta_n - \theta_c) \end{cases} \quad (5.9)$$

$$\begin{cases} \varepsilon_x = \frac{s(\alpha_c)}{c(\alpha_n)} - s(\alpha_n) & \text{se } \alpha_n \neq \pm \frac{\pi}{2} \\ \varepsilon_x = c(\alpha_n) - \frac{c(\alpha_c)}{s(\alpha_n)} & \text{se } \alpha_n = \pm \frac{\pi}{2} \\ \varepsilon_y = \frac{s(\theta_c)}{c(\theta_n)} - s(\alpha_n) s(\theta_n) \\ \varepsilon_z = \frac{s(\theta_c)}{c(\theta_n)} - c(\alpha_n) s(\theta_n) \end{cases} \quad (5.10)$$

Soluzioni analitiche *Link 2 e 3*

Considerando i *link* ad assi paralleli si prendono le 12 equazioni come nella (5.5) e si procede allo stesso modo dei link 1, 4, 5 e 6.

Le tre equazioni contenenti i tre errori di spostamento come per i *link* precedenti saranno prese dalle equazioni 10, 11 e 12 del set di equazioni matriciali ottenuto.

Il sistema di equazioni che descrive la posizione è:

$$\begin{cases} a_n c(\theta_n) + \delta_x c(\theta_n) - \delta_y c(\alpha_n) s(\theta_n) + \delta_z c(\alpha_n) s(\theta_n) = a_c c(\theta_c) \\ a_n s(\theta_n) + \delta_x s(\theta_n) - \delta_y c(\alpha_n) c(\theta_n) + \delta_z s(\alpha_n) c(\theta_n) = a_c s(\theta_c) \\ \delta_z c(\alpha_n) + \delta_y s(\alpha_n) = 0 \end{cases} \quad (5.11)$$

Il sistema di equazioni sarà lo stesso descritto nell'Eq. (5.9) con la differenza che nella terza equazione, in quanto il parametro d_c non può variare e viene vincolato al valore nominale d_n , i valori d_n e d_c si semplificano.

Le successive tre equazioni contenenti i tre errori di orientamento saranno prese dalle equazioni 2, 3, 6 e 9.

Come nella eq (5. 7) le equazioni 2 e 3 dipendono dai valori di α_n e α_c .

$$\begin{cases} s(\alpha_n) + \varepsilon_x c(\alpha_n) = s(\alpha_c) & \text{se } \alpha_n \neq \pm \frac{\pi}{2} \\ c(\alpha_n) - \varepsilon_x s(\alpha_n) = c(\alpha_c) c(\beta_c) & \text{se } \alpha_n = \pm \frac{\pi}{2} \end{cases} \quad (5. 12)$$

equazione 5:

$$s(\theta_n) + \varepsilon_z c(\alpha_n) c(\theta_n) + \varepsilon_y s(\alpha_n) c(\theta_n) = c(\beta_c) s(\theta_c) + s(\alpha_c) s(\beta_c) c(\theta_c) \quad (5. 13)$$

Il sistema di equazioni che descrive le rotazioni sarà:

$$\begin{cases} s(\alpha_n) + \varepsilon_x c(\alpha_n) = s(\alpha_c) & \text{se } \alpha_n \neq \pm \frac{\pi}{2} \\ c(\alpha_n) - \varepsilon_x s(\alpha_n) = c(\alpha_c) c(\beta_c) & \text{se } \alpha_n = \pm \frac{\pi}{2} \\ s(\theta_n) + \varepsilon_z c(\alpha_n) c(\theta_n) + \varepsilon_y s(\alpha_n) c(\theta_n) = c(\beta_c) s(\theta_c) + s(\alpha_c) s(\beta_c) c(\theta_c) \\ \varepsilon_z s(\alpha_n) - \varepsilon_y c(\alpha_n) = -c(\alpha_c) s(\beta_c) \end{cases} \quad (5. 14)$$

Le soluzioni analitiche degli errori per i *link* 2 e 3 saranno:

$$\begin{cases} \delta_x = a_c c(\theta_n - \theta_c) - a_n \\ \delta_y = a_c c(\alpha_n) s(\theta_c - \theta_n) \\ \delta_z = a_c s(\alpha_n) s(\theta_n - \theta_c) \end{cases} \quad (5. 15)$$

$$\begin{cases} \varepsilon_x = \frac{s(\alpha_c)}{c(\alpha_n)} - s(\alpha_n) & \text{se } \alpha_n \neq \pm \frac{\pi}{2} \\ \varepsilon_x = c(\alpha_n) - \frac{c(\alpha_c) c(\beta_c)}{s(\alpha_n)} & \text{se } \alpha_n = \pm \frac{\pi}{2} \\ \varepsilon_y = s(\alpha_n) \left(s(\theta_c) c(\beta_c) - s(\theta_n) + \frac{s(\alpha_c) s(\beta_c) c(\theta_c)}{c(\theta_n)} \right) + c(\alpha_n) (c(\alpha_c) s(\beta_c) c(\theta_c)) \\ \varepsilon_z = c(\alpha_n) \left(c(\theta_c) c(\beta_c) - s(\theta_n) + \frac{s(\alpha_c) s(\beta_c) c(\theta_c)}{c(\theta_n)} \right) - s(\alpha_n) (c(\alpha_c) s(\beta_c) c(\theta_c)) \end{cases} \quad (5. 16)$$

Ottenute tutte le equazioni analitiche si può procedere al calcolo degli errori geometrici:

	Giunto 1	Giunto 2	Giunto 3	Giunto 4	Giunto 5	Giunto 6
δ_x [mm]	0,1099	-0,1557	0,1841	0,0253	-0,0689	0
δ_y [mm]	-0,0749	0,1411	0,0896	1,0623	-0,2293	0
δ_z [mm]	-6,60E-06	0	0	3,38E-06	-1,14E-05	0,1937
ε_x [°]	-0,0946	0,0035	-0,2321	-0,0342	0	0

ε_y [°]	0,0034	0,0131	0,0705	-0,0077	-0,0095	0
ε_z [°]	0	-0,019	-0,0134	0	0	-0,0063

Tabella 5.1: Errori geometrici con senso fisico robot UR5

Come si può osservare dalla Tabella 5.1 gli errori massimi degli assi giunto sono dell'ordine del decimo di millimetro e del decimo di grado. Questi ordini di grandezza hanno senso fisico e permettono di descrivere geometricamente le deviazioni degli assi giunto, questi risultati possono contribuire all'implementazione di un modello high fidelity a scopo prognostico.

Capitolo 6

6 Conclusioni e sviluppi futuri

Dopo aver descritto la prognostica e l'utilità della calibrazione come strumento di identificazione dei parametri fisici di un robot reale, è stata approfondita la letteratura che descrive la calibrazione e la sua evoluzione nel tempo.

Sono state descritte le quattro fasi della calibrazione: modellazione, misurazione, identificazione e implementazione, dando una panoramica dello stato dell'arte della calibrazione industriale odierna.

Le prime tre fasi sono state trattate cercando di raggiungere un buon livello di profondità in quanto guidano il processo di calibrazione fino all'identificazione dei parametri dei robot.

Per la fase di modellazione, è stata dedicata una cospicua parte del terzo capitolo, attraverso questa è stato possibile comprendere i limiti e le applicazioni dei sistemi di riferimento di DH, Hayati, GDH. L'approfondimento sulla modellazione ha permesso di comprendere perché i parametri cinematici del costruttore erano senza senso fisico e che, senza l'ausilio di un modello corretto, non sarebbe mai stato possibile descrivere i giunti ad assi paralleli del manipolatore reale. Le tecnologie ed i metodi della fase di misurazione, elencati nel secondo capitolo, hanno permesso di comprendere come avviene la misura delle pose in un processo di calibrazione, facendo particolare attenzione a descrivere le tecniche di calibrazione adottate dalla Universal Robots per i suoi manipolatori. Nella fase di identificazione, descritta nel secondo capitolo, si sono descritti alcuni dei metodi più adottati in letteratura e si sono approfonditi quelli più interessanti ai fini del lavoro di identificazione svolto in questa tesi.

L'identificazione di tipo geometrico e quella LSM hanno portato ai risultati migliori, entrambi hanno raggiunto lo stesso livello di accuratezza dei parametri cinematici forniti dal costruttore.

Il metodo SVD ha permesso di identificare i parametri cinematici senza dover effettuare iterazioni, diminuendo gli errori medi di posizione ed orientamento del 99,0% e del 99,1% rispetto alla soglia di accettabilità del costruttore. La regressione lineare è stata molto utile anche nelle fasi preliminari, prima dell'identificazione di tipo deterministico, in quanto ha permesso di verificare il numero di condizionamento della matrice Jacobiana e le variabili (parametri cinematici) da considerare.

Il metodo GA si è dimostrato insufficiente e si è optato per un metodo GA Multiobiettivo. Attraverso questo si sono ottenuti buoni risultati sia per la convenzione DH che per la convenzione GDH. Si sono ottenute delle riduzioni degli errori di traslazione e rotazione rispetto ai limiti imposti dal costruttore pari a 84.7%, 88.9%, 80.6% e 90.3%, rispettivamente i primi due per la convenzione di DH e gli altri per la convenzione GDH. Non essendoci marcate differenze tra le due convenzioni, si è cercato di usare gli Algoritmi Genetici Multiobiettivo Ibridi. Il risultato migliore tra le due convenzioni va alla convenzione GDH, con degli errori di posizione ed orientamento ridotti del 97.4% e del 99.98%. Nella convenzione DH, seppur si raggiungono riduzioni degli errori inferiori rispetto a GDH, con un errore medio di posizione ed orientamento pari a 93,4% e 90,0%, rappresentano un risultato interessante, in quanto non risentono degli effetti del numero di condizionamento; questi risultati possono considerarsi accettabili, ma sono da preferire quelli identificati mediante la convenzione GDH. Tutti i risultati ottenuti mediante i vari metodi di identificazione adottati, che rientrano nelle soglie di accettabilità imposte dalla Universal Robots hanno senso fisico; questi risultati sono frutto di una simulazione, non tengono conto di errori di ripetibilità, errori non geometrici ed incertezza del sistema di misura. Infine, utilizzando i parametri cinematici ottenuti mediante la proiezione geometrica, sono stati calcolati gli errori geometrici utili per caratterizzare un modello *high fidelity*. In eventuali lavori futuri, noti i parametri geometrici, bisognerà identificare anche gli errori non geometrici, attraverso i quali implementare un modello high fidelity ancora più conforme alla realtà. Sarebbe inoltre interessante utilizzare le informazioni raccolte in questa tesi per poter realizzare un processo di calibrazione vero e proprio, attraverso un sistema di misura esterno. Un altro tipo di calibrazione che potrebbe essere approfondito è

la calibrazione a due robot effettuata dal costruttore; dato che gli strumenti di misura esterni adottati per la calibrazione esterna sono molto costosi, si rivelerebbe un metodo di calibrazione più accessibile, ma al contempo utile all'identificazione dei parametri cinematici.

BIBLIOGRAFIA

- [1] G. Qiao, C. Schlenoff, and B. A. Weiss, “Quick positional health assessment for industrial robot prognostics and health management (PHM),” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2017, pp. 1815–1820, doi: 10.1109/ICRA.2017.7989214.
- [2] P. W. Kalgren, C. S. Byington, M. J. Roemer, and M. J. Watson, “Defining PHM, a lexical evolution of maintenance and logistics,” in *2006 IEEE autotestcon*, 2006, pp. 353–358.
- [3] A. Carvalho Bittencourt, “Modeling and diagnosis of friction and wear in industrial robots.” Linköping University Electronic Press, 2014.
- [4] B. Abichou, A. Voisin, and B. Iung, “Bottom-up capacities inference for health indicator fusion within multi-level industrial systems,” in *2012 IEEE Conference on Prognostics and Health Management*, 2012, pp. 1–7.
- [5] M. Sorli, S. Mauro, A. De Martin, A. Raviola, and R. Guida, “Sviluppo di un modello high-fidelity del robot UR5 a fini diagnostici e prognostici,” 2020.
- [6] M. Sorli, G. Porcelli, S. Mauro, A. Raviola, and S. Pastorelli, “Dynamic parameters identification of a UR5 robot manipulator,” 2020.
- [7] H. N. Nguyen, J. Zhou, and H. J. Kang, “A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network,” *Neurocomputing*, vol. 151, no. P3, pp. 996–1005, 2015, doi: 10.1016/j.neucom.2014.03.085.
- [8] B. W. Mooring, Z. S. Roth, and M. R. Driels, *Fundamentals of manipulator calibration*. Wiley-interscience, 1991.
- [9] Universal Robots, *Kinematic Calibration Manual for CB3*. 2009.
- [10] K. Schröer, “Theory of Kinematic Modeling and Numerical Procedures for Robot Calibration,” in *Robot Calibration. Bernhardt and Albright. UK: Chapman & Hall, Springer Science & Business Media*, 1993, pp. 157–193.
- [11] B. Karan and M. Vukobratović, “Calibration and accuracy of manipulation robot models-An overview,” *Mech. Mach. Theory*, vol. 29, no. 3, pp. 479–500, 1994, doi: 10.1016/0094-114X(94)90130-9.
- [12] D. E. Whitney, C. A. Lozinski, and J. M. Rourke, “Industrial robot forward calibration method and results,” *J. Dyn. Sys., Meas., Control.*, 1986.

BIBLIOGRAFIA

- [13] R. P. Judd and A. B. Knasinski, “A Technique to Calibrate Industrial Robots with Experimental Verification,” *IEEE Transactions on Robotics and Automation*, vol. 6, no. 1. pp. 20–30, 1990, doi: 10.1109/70.88114.
- [14] H. W. Stone, *Kinematic modeling, identification, and control of robotic manipulators*, vol. 29, no. 2. 1987.
- [15] Z. S. Roth, B. W. Mooring, and B. Ravani, “An Overview of Robot Calibration,” *IEEE J. Robot. Autom.*, vol. 3, no. 5, pp. 377–385, 1987, doi: 10.1109/JRA.1987.1087124.
- [16] L. S. Ginani and J. M. S. T. Motta, “Theoretical and practical aspects of robot calibration with experimental verification,” *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 33, no. 1, pp. 15–21, 2011, doi: 10.1590/S1678-58782011000100003.
- [17] M. Lind, “Automatic Robot Joint Offset Calibration,” no. June 2012, 2014.
- [18] Y. Wu, A. Klimchik, S. Caro, B. Furet, and A. Pashkevich, “Geometric calibration of industrial robots using enhanced partial pose measurements and design of experiments,” *Robot. Comput. Integr. Manuf.*, vol. 35, pp. 151–168, 2015, doi: 10.1016/j.rcim.2015.03.007.
- [19] G. Duelen and K. Schröer, “Robot calibration-Method and results,” *Robot. Comput. Integr. Manuf.*, vol. 8, no. 4, pp. 223–231, 1991, doi: 10.1016/0736-5845(91)90034-P.
- [20] J. Zhou and H. J. Kang, “A hybrid least-squares genetic algorithm-based algorithm for simultaneous identification of geometric and compliance errors in industrial robots,” *Adv. Mech. Eng.*, vol. 7, no. 6, pp. 1–12, 2015, doi: 10.1177/1687814015590289.
- [21] W. K. Veitschegger and C. H. Wu, “Robot Calibration and Compensation,” *IEEE J. Robot. Autom.*, vol. 4, no. 6, pp. 643–656, 1988, doi: 10.1109/56.9302.
- [22] H. N. Nguyen, J. Zhou, and H. J. Kang, “A new full pose measurement method for robot calibration,” *Sensors (Basel)*, vol. 13, no. 7, pp. 9132–9147, 2013, doi: 10.3390/s130709132.
- [23] A. Nubiola and I. A. Bonev, “Absolute calibration of an ABB IRB 1600 robot using a laser tracker,” *Robot. Comput. Integr. Manuf.*, vol. 29, no. 1, pp. 236–245, 2013, doi: 10.1016/j.rcim.2012.06.004.
- [24] A. Nubiola and I. A. Bonev, “Absolute robot calibration with a single telescoping ballbar,” *Precis. Eng.*, vol. 38, no. 3, pp. 472–480, 2014, doi: 10.1016/j.precisioneng.2014.01.001.
- [25] T. Messay, R. Ordóñez, and E. Marcil, “Computationally efficient and robust kinematic calibration methodologies and their application to industrial robots,” *Robot. Comput. Integr. Manuf.*, vol. 37, pp. 33–48, 2016, doi:

BIBLIOGRAFIA

- 10.1016/j.rcim.2015.06.003.
- [26] M. Barati, A. R. Khoogar, and M. Nasirian, “Estimation and calibration of robot link parameters with intelligent techniques,” *Iran. J. Electr. Electron. Eng.*, vol. 7, no. 4, pp. 225–234, 2011.
- [27] J. Santolaria, J. J. Aguilar, J. A. Yagüe, and J. Pastor, “Kinematic parameter estimation technique for calibration and repeatability improvement of articulated arm coordinate measuring machines,” *Precis. Eng.*, vol. 32, no. 4, pp. 251–268, 2008, doi: 10.1016/j.precisioneng.2007.09.002.
- [28] A. Joubair and I. A. Bonev, “Non-kinematic calibration of a six-axis serial robot using planar constraints,” *Precis. Eng.*, vol. 40, pp. 325–333, 2015, doi: 10.1016/j.precisioneng.2014.12.002.
- [29] X. Chen, Q. Zhang, and Y. Sun, “Non-kinematic calibration of industrial robots using a rigid–flexible coupling error model and a full pose measurement method,” *Robot. Comput. Integr. Manuf.*, vol. 57, no. November 2017, pp. 46–58, 2019, doi: 10.1016/j.rcim.2018.07.002.
- [30] J. Ziegert and P. Datsoris, “Basic considerations for robot calibration,” in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, 1988, pp. 932–938 vol.2, doi: 10.1109/ROBOT.1988.12179.
- [31] S. Ahmad, “Analysis of robot drive train errors, their static effects, and their compensations,” *IEEE J. Robot. Autom.*, vol. 4, no. 2, pp. 117–128, 1988, doi: 10.1109/56.2075.
- [32] C. Gong, J. Yuan, and J. Ni, “Nongeometric error identification and compensation for robotic system by inverse calibration,” *Int. J. Mach. Tools Manuf.*, vol. 40, no. 14, pp. 2119–2137, 2000, doi: 10.1016/S0890-6955(00)00023-7.
- [33] Y. Liu, B. Liang, W. Qiang, and J. Yan-shu, “Improvement on Robots Positioning Accuracy Based on Genetic Algorithm,” pp. 387–392, 2007, doi: 10.1109/cesa.2006.4281683.
- [34] S.-K. Rune, E. H. Østergaard, and H. G. Petersen, “Calibration and Programming of Robots.” Google Patents, 2016.
- [35] G. Du, P. Zhang, and D. Li, “Online robot calibration based on hybrid sensors using Kalman Filters,” *Robot. Comput. Integr. Manuf.*, vol. 31, pp. 91–100, 2015, doi: 10.1016/j.rcim.2014.08.002.
- [36] H. G. BAŞ, “The Accuracy of Using Theodolite in Close Range Engineering Measurements,” vol. XXXIII, pp. 38–44, 2000.
- [37] C. Mathworks, “Optimization Toolbox™ User’s Guide R 2014 b,” 2014.
- [38] L. S. Ginani, “Theoretical and practical aspects of robot calibration with experimental verification,” *J. Brazilian Soc. Mech. Sci. Eng.*, pp. 1–13, 2011.

BIBLIOGRAFIA

- [39] S. Hayati and M. Mirmirani, “Improving the absolute positioning accuracy of robot manipulators,” *J. Robot. Syst.*, vol. 2, no. 4, pp. 397–413, 1985, doi: 10.1002/rob.4620020406.
- [40] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.
- [41] K. Wang, “Application of genetic algorithms to robot kinematics calibration,” *Int. J. Syst. Sci.*, vol. 40, no. 2, pp. 147–153, 2009, doi: 10.1080/00207720802630644.
- [42] F. Fei, H. Hongjie, and G. Zhongtong, “Application of genetic algorithm PSO in parameter identification of SCARA robot,” *Proc. - 2017 Chinese Autom. Congr. CAC 2017*, vol. 2017-Janua, pp. 923–927, 2017, doi: 10.1109/CAC.2017.8242898.
- [43] Universal Robots, “DH PARAMETERS FOR CALCULATIONS OF KINEMATICS AND DYNAMICS.” <https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/> (accessed Sep. 01, 2022).
- [44] A. Marwan, M. Simic, and F. Imad, “Calibration method for articulated industrial robots,” *Procedia Comput. Sci.*, vol. 112, pp. 1601–1610, 2017, doi: 10.1016/j.procs.2017.08.246.
- [45] M. J. D. Hayes and P. L. O. Leary, “Kinematic Calibration Procedure for Serial Robots with Six Revolute Axes,” pp. 1–27, 2001.
- [46] Y. Sun and J. M. Hollerbach, “Observability index selection for robot calibration,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 831–836, 2008, doi: 10.1109/ROBOT.2008.4543308.