

Università degli Studi di Padova

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE
Corso di Laurea Magistrale in Ingegneria Aerospaziale

**Studio, analisi e progettazione di un braccio robotico
ridondante per l'esplorazione planetaria**

Candidato:
Erik Bertolutti
Matricola 1084072

Relatore:
Prof. Stefano Debei

Ai miei genitori, che mi hanno permesso tutto questo
Ai miei amici, per il supporto in questi anni
A tutte le persone importanti della mia vita
Semplicemente per esserci

Sommario

In questa tesi viene presentata la progettazione di un braccio robotico economico per una competizione studentesca, con budget di peso e potenza limitati. L'obiettivo di tale competizione è la progettazione di un rover simil-marziano: Si parte dalla definizione dello spazio di lavoro richiesto; vengono poi discusse le analisi dei carichi a cui è sottoposto, e in particolare il modello del terreno utilizzato per stimare le forze a cui il braccio è soggetto. Segue un'analisi delle sezioni possibili per rimanere entro i requisiti e la scelta dei motori utilizzati per attuarlo. In seguito viene proposta una variazione del braccio che possa essere studiata in futuro, aggiungendo 1 G.d.L., che possa prendere il braccio più flessibile. Di tale variazione si è studiata la cinematica, risolvendo la ridondanza con una metodologia proposta da Chow et al., sfruttando la popolarità tra il nucleo dello Jacobiano e una funzione costo, e vengono presentati alcuni risultati scelti per delle configurazioni.

1	Introduzione	1
1.1	Rover: la storia	1
1.1.1	Linea temporale dei Rover impiegati	2
1.2	Challenges Studenteschi	5
1.3	ERC 2015	5
1.3.1	Task della competizione	6
1.3.2	Requisiti del Rover	7
1.3.3	MORPHEUS: Team e Rover	8
2	Cinematica dei manipolatori	11
2.1	Geometria del manipolatore: convenzione di Denavit-Hartenberg	12
2.2	Convenzione del sistema di riferimento per il singolo giunto	13
2.3	Utilizzo dei parametri	14
2.4	Cinematica inversa	15
2.5	Cinematica di velocità	17
2.5.1	Velocità lineare	17
2.5.2	Velocità angolare	18
2.6	Cinematica di accelerazione	22
2.7	Pianificazione della traiettoria	25
2.7.1	Traiettorie parametrizzate	27
2.8	Jacobiano del manipolatore	28
2.8.1	Trasformazione delle forze per il cambio di coordinate	31

3	Dinamica dei manipolatori	35
3.1	Equazioni dinamiche del manipolatore	37
4	Approccio Matriciale Alternativo	39
4.1	Matrici e Tensori di Velocità e Accelerazione	40
4.1.1	Definizioni di base	40
4.1.2	Cambio di sistema di riferimento	41
4.1.3	Cinematica Relativa	42
4.2	Dinamica	42
4.2.1	Matrice delle forze/coppie esterne (di azione)	42
4.2.2	Matrice di quantità di moto/momento angolare	43
4.2.3	Matrice di inerzia	43
4.2.4	Cambio di sistema di riferimento	44
4.2.5	Relazioni tra le matrici di cinematica e dinamica	44
4.2.6	Operatore "antisimmetria" (skew operator)	45
4.3	Matrice L	46
4.4	Metodo risolutivo	46
5	Progettazione del braccio robotico	49
5.1	La storia	49
5.1.1	Mitologia, leggende e automi	49
5.1.2	La robotica moderna	50
5.2	Introduzione	52
5.3	Requisiti	52
5.4	Scelta delle configurazioni	53
5.5	Stima dei carichi e scelta delle sezioni	55
5.5.1	Modello del terreno	55
5.6	Modello dei carichi	59
5.7	Materiali	62
5.8	Scelta dei motori	63
5.9	Driver Motori	68
5.10	Design definitivo	68
6	Cinematica del Braccio	73
6.1	Cinematica diretta	75
6.2	Cinematica inversa	77
6.2.1	Risoluzione "alla Pieper"	77

6.2.2	Ottimizzazione della soluzione per la cinematica inversa	80
6.2.3	Completamento della cinematica inversa	81
6.3	Esempi di risoluzione della cinematica inversa	82
7	Conclusioni e sviluppi futuri	89
A	Codice simbolico e soluzioni analitiche	91
A.1	Componenti della matrice di trasformazione	91
A.2	Risoluzione dell'equazione non lineare (6.9)	92
B	Codice MATLAB	95
B.0.1	Plot analisi del terreno	95
B.0.2	Jacobiano simbolico	98
B.0.3	Analisi simbolica con metodo presentato al capitolo 4	100
B.0.4	Matrice di trasformazione DH	103
B.0.5	Matrice di trasformazione DH simbolica	104
B.0.6	Inversa della matrice di trasformazione	105
B.0.7	Cinematica inversa	106
B.0.8	Path-planning con polinomio	112
B.0.9	Funzione non lineare da minimizzare	122

Elenco delle figure

1.1	Rover	4
1.2	Logo del MORPHEUS Team	8
1.3	Quadro generale del Rover MORPHEUS	9
2.1	Schema di un manipolatore	11
2.2	Convenzione di Denavit-Hattemberg	12
2.3	Esempio di configurazioni multiple	16
2.4	Velocità lineare	17
2.5	Sistemi di riferimento	19
2.6	Giunti $i -esimo$ e $i + 1 -esimo$	20
2.7	Accelerazione del baricentro	25
2.8	Posizione delle forze	32
3.1	Es. per corpo $i -esimo$	36
5.1	Spazio di lavoro	54
5.2	Modello dello scavo	57
5.3	Dimensioni della paletta	58
5.4	Sensibilità della forza all'angolo di contatto col terreno	58
5.5	Sensibilità della forza alla densità del terreno	59
5.6	Modello dei carichi 1	59
5.7	Modello dei carichi 2	60
5.8	Esempio di modello in Patran	61

5.9	Modello Solidworks del carico di punta	62
5.10	Una tipologia di motore passo-passo	63
5.11	Motore DC a magneti permanenti	64
5.12	Motore DC Brushless	65
5.13	Motore PS150-12-196	66
5.14	Attuatore lineare L16-140-150-12-P	66
5.15	Motoriduttore BS138F-12-208	67
5.16	Attuatore lineare L12-30-50-12-I	67
5.17	Driver LAC	68
5.18	Braccio con end effector 1	69
5.19	Particolari del primo end-effector	70
5.20	Braccio con end effector 2	71
5.21	Particolari del secondo end-effector	72
6.1	Render del braccio modificato	74
6.2	Braccio modificato con l'aggiunta di un G.d.L.	74
6.3	Modello del cinematismo a 4 G.d.L.	78
6.4	Configurazione del braccio	84
6.5	Progressione dei θ_i , $\dot{\theta}_i$ e $\ddot{\theta}_i$	86
6.6	Errori in direzione radiale	87
6.7	Errori lungo z	87

Elenco delle tabelle

1.1	Budget rover	9
1.2	Budget rover	10
5.1	Proprietà del suolo marziano	56
5.2	Elenco motori	68
6.1	Parametri di Denavit-Hartenberg	75
6.2	Valori degli angoli	83

CAPITOLO 1

Introduzione

1.1 Rover: la storia

Nel panorama delle ricerche attuali in ambito extra planetario, si è sempre maggiormente diffuso il concetto di rover. Questa parola, che può essere anche tradotta dall'inglese in vagabondo, rappresenta un veicolo in grado di muoversi autonomamente, o guidato da terra, oppure può essere usato nell'accezione di veicolo pilotato da astronauti. I primi due casi citati sono quelli di importanza maggiore, in quanto sfruttando una programmazione del rover, oppure una serie di comandi da terra, si è in grado di esplorare superfici maggiori del pianeta obiettivo. Inoltre i rover possono e sono spesso provvisti di un qualche laboratorio mobile, e questo permette una analisi in loco dei campioni che vengono prelevati dal robot stesso. Com'è facile intuire ciò presenta notevoli vantaggi: non dovendo tornare alla base per effettuare le analisi il raggio di azione del veicolo viene notevolmente incrementato, com'è stato dimostrato da esperimenti precedenti. Questa tipologia di esploratori planetari è stata utilizzata principalmente sulla luna e su Marte, con la presenza di molteplici missioni che ne impiegavano. La prima missione ad utilizzare un rover è stata la Luna 17 del 1970 lanciata dall'Urss. Da allora le missioni ad utilizzare un rover sono state molteplici, fino ad arrivare all'ultima Mars Science Laboratory del 2012, con il rover Curiosity.

1.1.1 Linea temporale dei Rover impiegati

Nella seguente linea temporale vengono inserite unicamente le missioni che hanno avuto successo. Le date proposte sono quelle di attivazione/atterraggio dei rover e non di lancio da terra.

Missioni passate

1970 - Luna 17

Lunokhod 1 è stato il primo robot controllato da remoto che sia atterrato su un qualsiasi corpo celeste. Esso è stato lanciato con la missione Luna 17, pesava circa 756 kg ed era in grado di effettuare analisi del suolo lunare. È stato per molto tempo il rover con il tempo di operatività maggiore: esso infatti ha lavorato per 11 mesi. In totale ha percorso una distanza di 10,5 km.

1971 - Apollo 15

Vedi Apollo 17.

1972 - Apollo 16

Vedi Apollo 17.

1972 - Apollo 17

Nelle ultime tre le missioni Apollo è stato utilizzato il rover lunare sviluppato dalla NASA. Popolarmente conosciuto come Moon Buggy, per la similitudine con la terrestre Dune Buggy, poteva trasportare due astronauti in modo da poter estendere il range di attività degli stessi. In ogni missione il rover è stato utilizzato per tre traversate, una al giorno. Nell'ordine la distanza totale percorsa da questi tre rover è stata di 27,76 km, 26,55 km, e di 35,89 km.

1973 - Luna 21

Lunokhod 2 venne lanciato con la sonda Luna 21 e fu il secondo rover robotico a scendere sulla superficie della luna, rimase attivo per quattro mesi circa e percorse la distanza totale di 37 km. Anche esso aveva a sua disposizione una serie di sensori per lo studio della superficie lunare.

1997 - Mars Pathfinder

Questa è stata la prima missione ad aver portato un rover su un altro corpo celeste diverso dalla luna: Marte. Mars Pathfinder era diviso in un lander, il Mars Pathfinder vero e proprio, ed un rover chiamato Sojourner. Aveva la disposizione

uno spettrometro a raggi X (APXS), ed alcune telecamere. Riuscì a percorrere solamente una distanza di 104 m prima che le comunicazioni furono interrotte.

2004 - Mars Exploration Rover A

Comunemente noto con il nome di *Spirit*, assieme al rover gemello *Opportunity*, è stato uno dei due rover lanciati nel corso della missione Mars Exploration Rover. Progettato per un tempo di missione di 90 sol, equivalenti a circa 92 giorni terrestri, grazie a folate di vento che hanno permesso di pulire pannelli solari, il rover è stato in grado di operare per ben 2269 giorni prima che si perdesse il contatto. A sua disposizione aveva delle telecamere, degli spettrometri, e degli strumenti per l'abrasione della rocce.

2004 - Mars Exploration Rover B

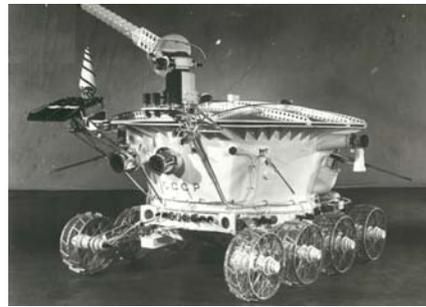
Il MER-B, comunemente noto con il nome di *Opportunity*, a differenza del sopra citato *Spirit*, è tuttora operativo. Il 28 luglio del 2014 la NASA ha annunciato che il rover ha percorso una distanza di oltre 40 km sul pianeta di Marte, divenendo pertanto il rover ad aver percorso più chilometri su un corpo extra terrestre. La strumentazione a sua disposizione è in tutto e per tutto uguale a quella di *Spirit*.

2012 - Mars Science Laboratory

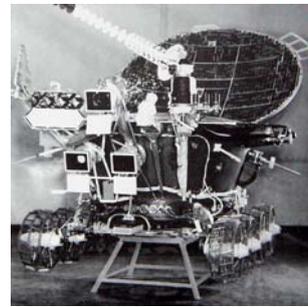
Sulla scia delle due missioni precedenti nel 2011 la NASA ha lanciato da Cape Canaveral la missione Mars Science Laboratory. L'obiettivo di questa missione è investigare il clima e la geologia marziana, e verificare se siano mai esistite delle condizioni favorevoli alla vita. Il rover, *Curiosity*, è pertanto fornito di strumenti per la spettroscopia, telecamere, cromatografia dei gas, detector per le radiazioni e un braccio robotico. Le dimensioni per accogliere tutta questa strumentazione sono quelle di una macchina, per un peso complessivo di circa 900 chili. La missione è tuttora operativa.

2013 - Chang'e 3

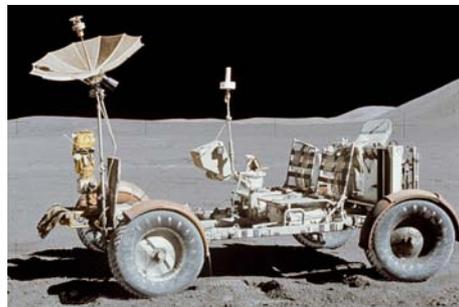
Questa missione fa parte della seconda fase del Chinese Lunar Exploration Program, ed è la prima ad avere un rover (cinese). Yutu, questo è il nome del rover, ha una massa di circa 140 kg e, come strumentazione, presenta un georadar, spettrometri a raggi X e ad infrarossi per analizzare il suolo lunare, e una stereocamera che permette la Visual Odometry. A causa di un qualche non meglio specificato problema, il rover è al momento operativo ma non in grado di muoversi.



(a) Lunokhod 1



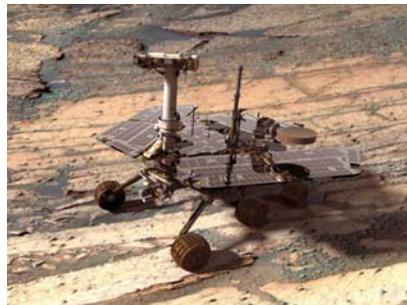
(b) Lunokhod 2



(c) Lunar Roving Vehicle



(d) Sojourner



(e) Spirit/Opportunity



(f) Curiosity



(g) Yutu



(h) I 3 rover NASA assieme

Figura 1.1: Rover

Missioni future

2017 - Chandrayaan-2

Seconda missione di esplorazione lunare dell'India, è partita come una partnership tra l'agenzia spaziale russa Roskosmos e la ISRO, ma per la non possibilità di fornire il lander entro i tempi prestabiliti, la Russia è stata esclusa. La missione prevede un pay-load del rover composto da spettroscopi a laser o a raggi X, e sembra vi sia anche l'intenzione di inserire un sismometro per lo studio dei terremoti lunari.

2018 - ExoMars

La missione ExoMars nasce come collaborazione dell'agenzia spaziale europea (ESA) e dell'agenzia spaziale russa. La missione si articolerà in due lanci: uno nel 2016 e l'altro nel 2018. È proprio con l'ultimo che verrà trasportato l'ExoMars rover, progettato per navigare in modo autonomo sulla superficie marziana. L'obiettivo di questo rover sarà quello di verificare la presenza di biomolecole e biosignature sul o sotto il suolo marziano, pertanto è anche provvisto di un sistema di carotaggio per l'estrazione di campioni.

2020 - Mars 2020

La missione Mars 2020 fa parte del programma di esplorazione di Marte della NASA. Il design di questo rover sarà basato su quello di Curiosity. Gli strumenti scientifici proposti per la missione si basano sugli obiettivi che sono stati proposti: planetologia e uno studio in preparazione a future missioni manned su Marte.

1.2 Challenges Studenteschi

Per approfondire lo studio dei rover e permettere a degli studenti di formarsi sono state indette nel mondo più competizioni. Alcune tra queste sono la NASA Human Exploration Rover Challenge, la University Rover Challenge (URC) e la European Rover Challenge (ERC). La prima sfida ha come obiettivo lo sviluppo e il test di una Moon Buggy simile a quelle impiegate nelle missioni Apollo 15, 16 e 17; le altre sul design di un robot radiocomandato che deve poter superare diverse prove preparate in modo da simulare situazioni che possono accadere o sulla terra o in caso di esplorazione extra terrestre.

1.3 ERC 2015

L'European Rover Challenge nasce come controparte europea della University Rover Challenge, in modo da poter permettere la partecipazione ad un contest studentesco ad

università europee, le quali possono non avere i mezzi per partecipare a quello americano. Infine, questo permette un miglioramento delle capacità degli studenti universitari europei. L'obiettivo principale di questa competizione è di creare un rover che possa affrontare varie prove sul terreno appositamente preparato in modo da sembrare marziano. La competizione infatti è stata pensata con delle prove che possano rispecchiare varie situazioni: siano esse a carattere di ricerca scientifica, di aiuto per un eventuale astronauta, di attraversamento autonomo e evitamento di ostacoli, e persino una prova che si basa su una riparazione radiocomandata, sia essa su un altro pianeta oppure in condizioni pericolose per gli esseri umani.

1.3.1 Task della competizione

Task 1 - Science Task La prima prova è definita come quella a carattere scientifico. In questa prova il rover è incaricato di ottenere tre campioni in tre zone diverse. Nella fattispecie i campioni saranno: un pezzo di roccia (un sasso) del peso di almeno 100 g, per il quale è consigliata un'estrazione afferrandolo in un qualche modo; quello che viene chiamato terreno sfuso, nella quantità di almeno 200 g, viene consigliato di utilizzare una paletta; terreno ad una profondità di almeno 15 cm per almeno 25 g, viene consigliato l'utilizzo di un trapano.

Task 2 - Terrain Traversal Task La seconda prova, ovvero l'attraversamento "cieco", ha come obiettivo il raggiungimento di una posizione secondo un percorso più o meno definito. Questo deve essere effettuato unicamente con l'ausilio del sistema di navigazione del rover, senza nessuna informazione dalle fotocamere. Durante la prima fase della prova il percorso può essere scelto dal team, mentre durante la seconda parte questo è prefissato in un tratto rettilineo e con un ostacolo posto lungo di esso.

Task 3 - Maintenance Task La terza prova, soprannominata il reattore, ha come obiettivo l'utilizzo di strumentazioni poste ad una certa distanza dalla partenza. Nella fattispecie è necessario riattivare degli interruttori e fare delle misure di potenza elettrica. Gli interruttori possono essere posti in un range di altezza abbastanza ampio, e cioè tra i 0,2 e degli 1,5 m. Inoltre deve essere disponibile una metodologia per misurare intensità di corrente e tensione elettrica.

Task 4 Assistance Task La quarta prova, ossia quella che si riferisce ad un caso di riparazioni di emergenza, ha come obiettivo quello di mostrare la capacità del rover di potere trasportare ad un qualsivoglia operatore un possibile pezzo di ricambio. Questo,

a causa della delicatezza del pezzo stesso, deve essere fatto evitando di dare scossoni e con una metodologia indicata: l'oggetto dell'essere raggiunto ed acquisito da rover, e in seguito deve essere trasportato in un'altra posizione e depositato nel giusto verso.

1.3.2 Requisiti del Rover

1. Il rover può costare al massimo 15.000 €, nei quali devono essere compresi i costi di: parti del rover, equipaggiamento e lavorazioni.
2. Il rover durante la singola Task non avere peso superiore ai 50 kg. Si hanno 20 kg ulteriori a disposizione per i pezzi di ricambio.
3. Il rover non deve essere collegato a fili o cavi durante il suo funzionamento.
4. Le Task sono indipendenti, il rover o essere modificato, riparato, e ottimizzato tra le singole prove.
5. Il rover deve essere radiocomandato in tempo reale.
6. Non è necessario che il rover sia completamente automatizzato, anche se questo procura dei bonus. Questi non vengono assegnati in caso di controllo da parte di un operatore.
7. Il rover deve essere fornito di un pulsante di emergenza sia in grado di isolare le batterie dal resto sistema.
8. Le comunicazioni con il rover devono utilizzare frequenze legalmente disponibili e livelli di potenza a norma di legge.
9. Perdita di controllo del rover porta a squalifica
10. Il rover deve essere in grado di partecipare a tutte le prove.

1.3.3 MORPHEUS: Team e Rover



Figura 1.2: Logo del MORPHEUS Team

Il progetto Morpheus inizia alla fine del 2014, nel mese di novembre. Alcuni studenti dell'università di Padova, su iniziativa dei professori Stefano Debei e Marco Pertile, e con l'aiuto di alcuni dottorandi, si trovano per fondare per il primo anno in assoluto un team che possa progettare un rover per l'esplorazione extra planetaria, in modo da poter partecipare a delle competizioni studentesche. Tale progetto, nato come esperienza istruttiva per dei futuri ingegneri, ha come obiettivo secondario il riconoscimento da parte della istituzione della necessità di fornire agli studenti la possibilità di scegliere un canale dedicato alla robotica spaziale. Il gruppo che si viene a formare è composto da studenti di dipartimenti diversi: ingegneria aerospaziale, elettronica, informatica e dell'informazione. La multiforme provenienza di studenti è necessaria per la progettazione di un sistema così complesso e completo.

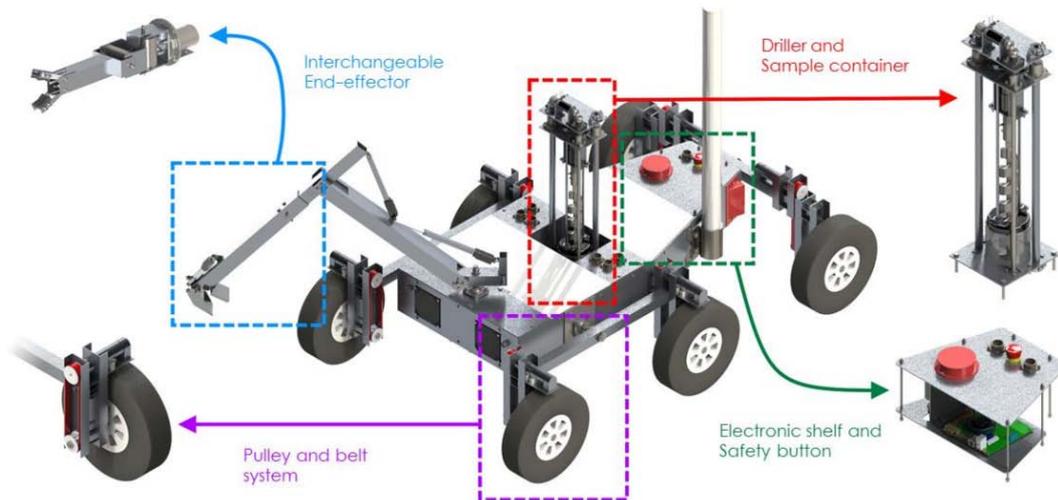


Figura 1.3: Quadro generale del Rover MORPHEUS

In sede di decisione per la divisione del budget adibito al progetto si è deciso di effettuare le seguenti divisioni, dietro computo preliminare di costi, necessità di potenza richiesta e peso dei vari sottosistemi del rover.

Per quanto riguarda quello che è il budget per il rover:

Sottosistema	Massa[kg]	Costo[€]
Bus	38	7800
Arm	3	1000
Drill	3	1000
Electronics	2	2000
Totale	46	11800

Tabella 1.1: Budget Rover

Mentre invece per le disponibilità di attrezzature, ricambi e pezzi sostitutivi del rover si ha:

Componenti addizionali	Massa[kg]	Costo[€]
Batteria Extra	3	400
End-effector 2	2	500
Equipaggiamento	12	1000
Totale	15	1900

Tabella 1.2: Budget pezzi di ricambio ed equipaggiamento

Un manipolatore è un Robot in catena aperta.

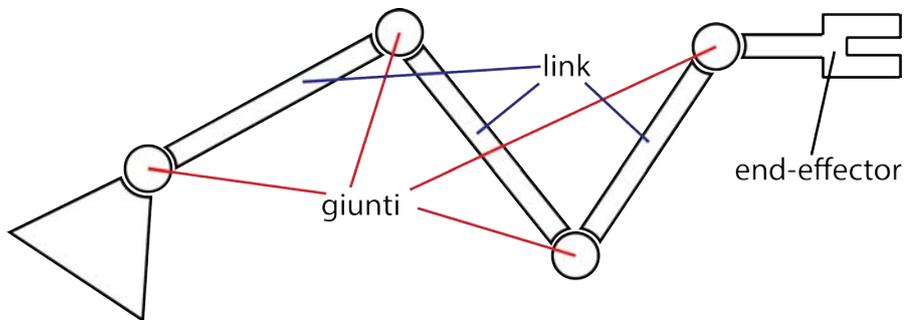


Figura 2.1: Schema di un manipolatore

Esso è composto da bracci (o link) collegati tra loro da giunti, che possono essere di vario tipo a seconda dei gradi di libertà permessi (moto relativo tra i bracci).

- Rotoidali: 1 grado di libertà rotazionale relativo.
- Prismatici: 1 grado di libertà traslazionale relativo.

Giunti più complessi possono essere comunque schematizzati e modellizzati attraverso questi due tipi di giunti (es: giunto sferico schematizzabile come composizione di tre giunti rotoidali).

2.1 Geometria del manipolatore: convenzione di Denavit-Hartenberg

Si definiscono:

- a_{i-1} = distanza i due assi $i-1$ e i .
- α_{i-1} = angolo compreso tra i due assi.
- d_i = distanza tra i segmenti a_{i-1} e a_i .
- θ_i = angolo compreso tra i segmenti a_{i-1} e a_i .

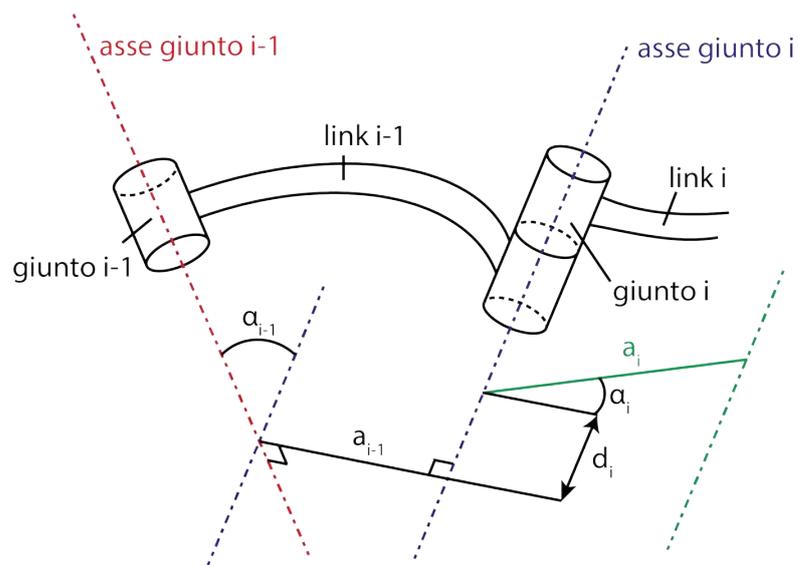


Figura 2.2: Convenzione di Denavit-Hartenberg

Ogni corpo del manipolatore ha il suo sistema di riferimento.

Nel caso di giunto rotoidale:

- $\theta_i = 0$ se il manipolatore è in configurazione di riposo.
- $d_i = 0$
- a_i, α_i, d_i sono costanti mentre θ_i varia nel tempo.

Nel caso di giunto prismatico:

- $d_i = 0$ se il manipolatore è in configurazione di riposo.



- $\theta_i = 0$
- a_i, α_i, θ_i sono costanti mentre d_i varia nel tempo.

2.2 Convenzione del sistema di riferimento per il singolo giunto

- \hat{z}_i coincidente con l'asse del giunto i -esimo.
- Origine del giunto O_i nell'intersezione tra a_i e l'asse i .
- \hat{x}_i lungo a_i dal giunto i -esimo al giunto $i+1$ -esimo.
- \hat{y}_i definito dalla regola della mano destra.
- Il sistema di riferimento "zero" è coincidente con il sistema di riferimento uno per $\theta_1 = d_1 = 0$.
- Il sistema di riferimento n -esimo (end-effector) dipende dal tipo di giunto:
 1. Rotoidale: \hat{x}_n parallelo a \hat{x}_{n-1} per $\theta_n = 0$ e origine tale che $d_n = 0$.
 2. Prismatico: \hat{x}_n tale che $\theta_n = 0$ e origine nell'intersezione tra \hat{x}_{n-1} e asse n per $d_n = 0$.

Per definire i parametri, una volta fissati i sistemi di riferimento, si utilizza la seguente convenzione:

- a_i =distanza tra \hat{z}_i e \hat{z}_{i+1} lungo \hat{x}_i .
- α_i =angolo compreso tra \hat{z}_i e \hat{z}_{i+1} attorno a \hat{x}_i .
- d_i =distanza tra \hat{x}_{i-1} e \hat{x}_i lungo \hat{z}_i .
- θ_i =angolo compreso tra \hat{x}_{i-1} e \hat{x}_i attorno a \hat{z}_i .

Nella costruzione del modello si possono riscontrare alcune ambiguità come:

- Scelta del verso di \hat{z}_i .
- L'intersezione degli assi di due giunti adiacenti comporta un valore di a_i nullo e il verso di \hat{x}_i rimane indefinito.

- Se gli assi dei giunti sono paralleli o coincidenti a_i non esiste e non è definita l'origine del sistema di riferimento.
- Altre ambiguità sono dovute alla presenza di coppie prismatiche.

2.3 Utilizzo dei parametri

1. Rotazione del sistema di riferimento $i-1$ di un angolo α_{i-1} attorno a \hat{x}_{i-1} passando al sistema $\{O', x'_i, y'_i, z'_i\}$ con $x'_i \equiv x_{i-1}$.
2. traslazione di a_{i-1} lungo \hat{x}_i passando al sistema $\{O'', x''_i, y''_i, z''_i\}$.
3. Rotazione di un angolo θ_i attorno a \hat{z}''_i .
4. traslazione di d_i lungo \hat{z}'''_i con $\hat{z}'''_i \equiv \hat{z}''_i$.

In questo modo si fa coincidere il sistema di riferimento $i-1$ -esimo con il sistema i -esimo.

La matrice di trasformazione risulta essere allora:

$${}^{i-1}_i[T] = [R(\hat{x}_{i-1}, \alpha_{i-1})] [T_R(\hat{x}_{i-1}, a_{i-1})] [R(\hat{z}''_i, \theta_i)] [T_R(\hat{z}''_i, d_i)] \quad (2.1)$$

dove:

$$[R(\hat{x}_{i-1}, \alpha_{i-1})] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_{i-1}) & -\sin(\alpha_{i-1}) & 0 \\ 0 & \sin(\alpha_{i-1}) & \cos(\alpha_{i-1}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

$$[T_R(\hat{x}_{i-1}, a_{i-1})] = \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

$$[R(\hat{z}''_i, \theta_i)] = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

$$[T_R(\hat{z}_i'', d_i)] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

Si ottiene così:

$${}^{i-1}_i[T] = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & a_{i-1} \\ \sin(\theta_i) \cos(\alpha_{i-1}) & \cos(\theta_i) \cos(\alpha_{i-1}) & -\sin(\alpha_{i-1}) & -d_i \sin(\alpha_{i-1}) \\ \sin(\theta_i) \sin(\alpha_{i-1}) & \cos(\theta_i) \sin(\alpha_{i-1}) & \cos(\alpha_{i-1}) & d_i \cos(\alpha_{i-1}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Per passare dal sistema 0 al sistema N si ha perciò:

$${}^0_N[T] = {}^0_1[T] {}^1_2[T] \dots {}^{N-2}_{N-1}[T] {}^{N-1}_N[T] = f(\theta_1, d_1, \theta_2, d_2, \dots, \theta_{N-1}, d_{N-1}, \theta_N, d_N) \quad (2.7)$$

2.4 Cinematica inversa

L'obiettivo è determinare la matrice

$${}^0_N[T] = {}^0_N[T](q_1, q_2, \dots, q_{N-1}, q_N) \quad (2.8)$$

con $q_i = \theta_i$ nel caso di giunto rotoidale e $q_i = d_i$ per i giunti prismatici. Assegno pertanto una rotazione e una posizione per determinare una matrice numerica (con l'apice).

$${}^0_N[T]' = \begin{bmatrix} [R]' & \vec{t}' \\ 0 & \dots & 1 \end{bmatrix} \quad (2.9)$$

tale che:

$${}^N_0[T](q_1, q_2, \dots, q_{N-1}, q_N) = {}^0_N[T]' \quad (2.10)$$

In generale posso avere al massimo 6 equazioni indipendenti, 3 di rotazione e 3 di traslazione.

Non è sempre possibile risolvere il sistema attraverso la cinematica inversa, ma se la posizione dell'end-effector ha almeno una soluzione si dice che essa appartiene allo spazio di lavoro *del manipolatore/nello spazio cartesiano*. Si definisce pertanto:

- Spazio di lavoro raggiungibile: insieme delle posizioni raggiungibili da almeno una direzione.
- Spazio di lavoro destro (da DEXTEROUS): insieme delle posizioni raggiungibili da qualsiasi direzione.

Lo spazio destro è un sottoinsieme dello spazio raggiungibile. Ci può essere più di una soluzione (configurazione possibile) nello spazio dei giunti per raggiungere un punto dello spazio cartesiano come mostrato in figura.

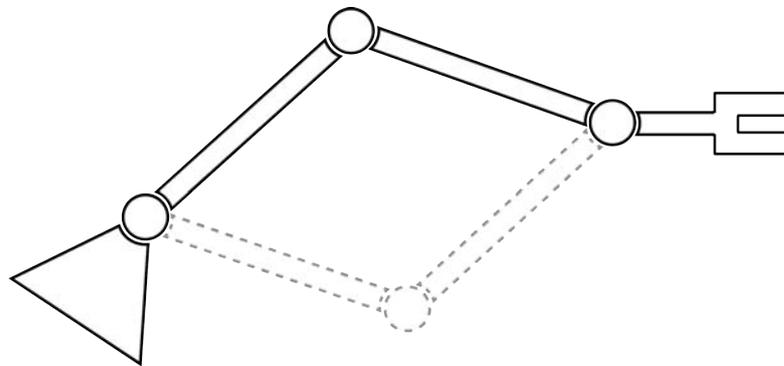


Figura 2.3: Esempio di configurazioni multiple

Il numero di soluzioni o configurazioni possibili aumenta con l'aumentare di:

- Numero di giunti.
- Parametri di giunto diversi da zero.
- Corsa dei giunti.

E' quindi possibile scegliere tra diverse configurazioni; generalmente la scelta ricade sulla soluzione che minimizza il costo complessivo (in termini energetici o temporali). Si tende a movimentare il meno possibile i giunti più vicini al telaio (base) e di far traslare o ruotare maggiormente i giunti prossimi all'end-effector in quanto, in questo modo, si riducono le inerzie movimentate. Nell'analisi si deve comunque tener conto del peso dei bracci e della presenza di ostacoli sul percorso.

Un manipolatore si definisce *Risolubile* se posso trovare tutte le soluzioni possibili.

Per determinare le soluzioni esistono due approci:

- Approccio algebrico: consiste nella risoluzione di equazioni.
- Approccio geometrico: si ragiona in termini di geometria del manipolatore.

Esistono poi anche approcci di tipo numerico che tuttavia risultano essere generalmente più lenti a causa della forte non-linearità.

Si cerca sempre di sviluppare manipolatori risolvibili in forma chiusa; una condizione sufficiente affinché ciò sia verificato è che almeno tre assi consecutivi si intersechino in un solo punto.

2.5 Cinematica di velocità

2.5.1 Velocità lineare

Per la velocità lineare si ha che, derivando la posizione del generico punto P:

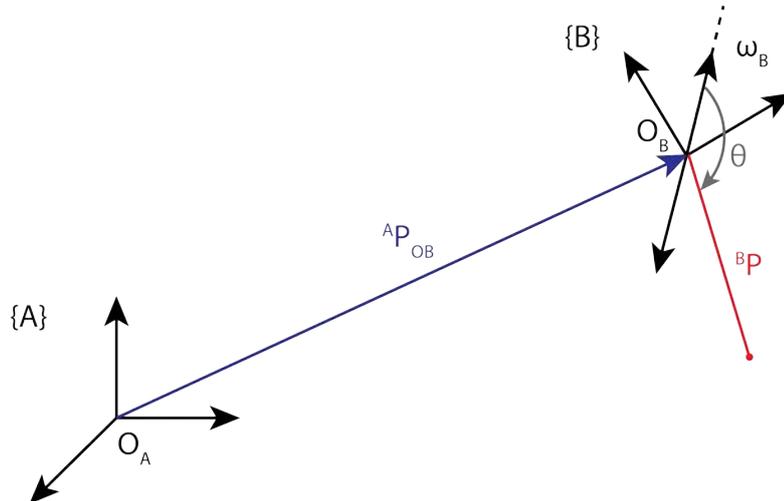


Figura 2.4: Posizionamento dei sistemi di riferimento per il calcolo della velocità lineare

$${}^A \vec{P} = {}^A \vec{P}_{OB} + {}^A [R] {}^B \vec{P} \quad (2.11)$$

$${}^A \vec{v} = {}^A \vec{v}_{OB} + {}^A [R] {}^B \vec{v}_{P,B} + \omega_B \times {}^A [R] {}^B \vec{P} \quad (2.12)$$

I termini che compaiono a secondo membro dell'equazione costituiscono la velocità di trascinamento (primo e terzo termine) e la velocità relativa (secondo); ${}^A [R]$ è la matrice di rotazione.

Poiché:

$${}^A \vec{\omega}_B \times {}^A [R] {}^B \vec{P} = |\omega_B| \left| {}^B \vec{P} \right| \sin(\theta) \quad (2.13)$$

$${}^A_B[\dot{R}]^B \vec{P} = {}^A\omega_B \times {}^A_B[R]^B \vec{P} \quad (2.14)$$

si ha:

$${}^A\vec{v}_P = {}^A\vec{v}_{OB} + {}^A_B[R]^B \vec{v}_{P,B} + {}^A_B[\dot{R}]^B \vec{P} \quad (2.15)$$

Essendo definito il vettore ${}^A\omega_B$ come

$${}^A\omega_B = [{}^A\omega_x, {}^A\omega_y, {}^A\omega_z] \quad (2.16)$$

Si ricorda che per definizione:

$${}^A_B[R] {}^A_B[R]^T = [I] \quad (2.17)$$

per cui, derivando nel tempo si ottiene:

$${}^A_B[\dot{R}] {}^A_B[R]^T + {}^A_B[R] {}^A_B[\dot{R}]^T = 0 \quad (2.18)$$

Si definisce S_ω come:

$$\vec{S}_\omega = {}^A_B[\dot{R}] {}^A_B[R]^T \quad (2.19)$$

Si ha quindi:

$$\vec{S}_\omega + \vec{S}_\omega^T \quad (2.20)$$

$${}^A_B[\dot{R}] = \vec{S}_\omega {}^A_B[R] \quad (2.21)$$

Possiamo pertanto scrivere l'equazione della velocità lineare come:

$${}^A\vec{v}_P = {}^A\vec{v}_{OB} + {}^A_B[R]^B \vec{v}_{P,B} + \vec{S}_\omega {}^A_B[R]^B \vec{P} \quad (2.22)$$

oppure

$${}^A\vec{v}_P = {}^A\vec{v}_{OB} + {}^A_B[R]^B \vec{v}_{P,B} + {}^A\omega_B \times {}^A_B[R]^B \vec{P} \quad (2.23)$$

2.5.2 Velocità angolare

Per la velocità angolare si può scrivere, facendo riferimento all'immagine 2.5:

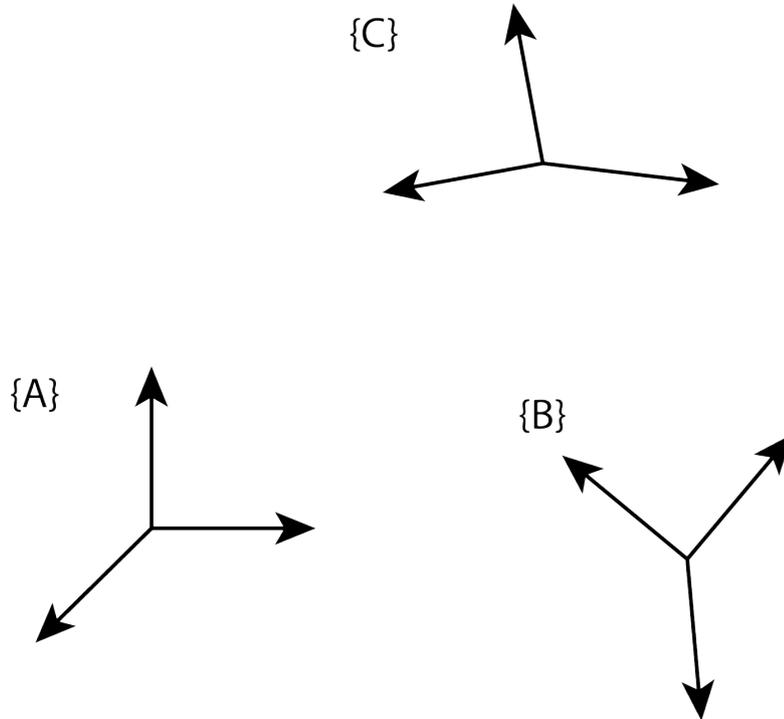


Figura 2.5: Sistemi di riferimento per il calcolo della velocità angolare

$${}^A\vec{\omega}_C = {}^A\vec{\omega}_B + {}^A\vec{\omega}_{C,B} \quad (2.24)$$

da cui

$${}^A\vec{\omega}_C = {}^A\vec{\omega}_B + {}^A_B[R] {}^B\vec{\omega}_{C,B} \quad (2.25)$$

Riscriviamo ora l'equazione della velocità lineare espressa nel sistema di riferimento B:

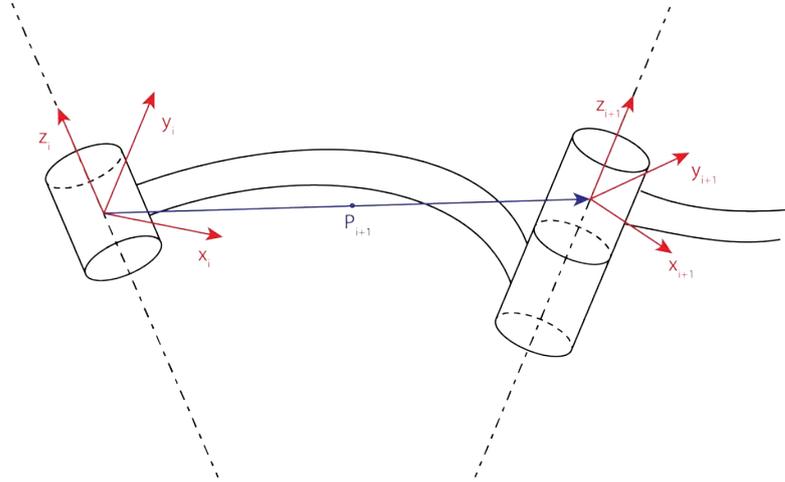
$${}^B\vec{v}_P = {}^B\vec{v}_{OB} + {}^B\vec{v}_{P,B} + {}^B\vec{\omega}_B \times {}^B\vec{P} \quad (2.26)$$

mentre per la velocità angolare si ha:

$${}^B\vec{\omega}_C = {}^B\vec{\omega}_B + {}^B\vec{\omega}_{C,B} \quad (2.27)$$

Applichiamo tali equazioni al manipolatore tra il giunto i -esimo e il successivo $i+1$ -esimo.

Siano:


 Figura 2.6: Sistema di riferimento per i giunti i -esimo e $i+1$ -esimo

- \vec{P} = posizione dell'origine del sistema $i+1$ -esimo espressa ne sistema i -esimo.
- $B = i$
- $C = i + 1$
- $A = 0$

Sostituendo nell'equazione della velocità angolare si ha:

$${}^i\vec{\omega}_{i+1} = {}^i\vec{\omega}_i + {}^i\vec{\omega}_{i+1,i} \quad (2.28)$$

dove:

$${}^{i+1}\vec{\omega}_{i+1} = {}^{i+1}_i[R] {}^i\vec{\omega}_i + {}^{i+1}\vec{\omega}_{i+1,i} \quad (2.29)$$

con:

- ${}^{i+1}\vec{\omega}_{i+1,i} = 0$ per un giunto prismatico
- ${}^{i+1}\vec{\omega}_{i+1,i} = f(\theta_{i+1})$ per un giunto rotoidale

Per un giunto rotoidale l'equazione che esprime la velocità angolare risulta quindi:

$${}^{i+1}\vec{\omega}_{i+1} = {}^{i+1}_i[R] {}^i\vec{\omega}_i + \dot{\theta}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.30)$$

avendo posto:

$${}^{i+1}\vec{\omega}_{i+1,i} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix} = \dot{\theta}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.31)$$

Nel giunto prismatico invece:

$${}^{i+1}\vec{\omega}_{i+1} = {}^{i+1}_i[R] {}^i\vec{\omega}_i \quad (2.32)$$

Riscriviamo ora l'equazione della velocità lineare espressa nel sistema di riferimento i -esimo.

$${}^i\vec{v}_{i+1} = {}^i\vec{v}_i + {}_{i+1}^i[R] {}^{i+1}\vec{v}_{i+1,i} + {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \quad (2.33)$$

- Nel caso di giunto rotoidale si ha che ${}^{i+1}\vec{v}_{i+1,i} = 0$ e quindi l'equazione della velocità lineare diviene:

$${}^i\vec{v}_{i+1} = {}^i\vec{v}_i + {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \quad (2.34)$$

- Per un giunto prismatico invece:

$${}^{i+1}\vec{v}_{i+1,i} = \begin{bmatrix} 0 \\ 0 \\ \dot{d}_{i+1} \end{bmatrix} = \dot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.35)$$

da cui segue:

$${}^i\vec{v}_{i+1} = {}^i\vec{v}_i + {}_{i+1}^i[R] \dot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} + {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \quad (2.36)$$

Riscrivendo le equazioni della velocità lineare per il giunto rotoidale e prismatico nel sistema $i+1$ -esimo si ottiene rispettivamente:

$${}^{i+1}\vec{v}_{i+1} = {}_{i+1}^i[R] \left[{}^i\vec{v}_i + {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \right] \quad (2.37)$$

$${}^{i+1}\vec{v}_{i+1} = {}_{i+1}^i[R] \left[{}^i\vec{v}_i + {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \right] + \dot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.38)$$

E' possibile quindi calcolare la velocità del giunto $i+1$ -esimo a partire dalla velocità del giunto precedente (i -esimo) e, attraverso una serie di iterazioni in avanti, calcolare quindi la velocità dell'end-effector a partire dalla base.

$${}^0\vec{v}_N = {}^0[R]{}^N\vec{v}_N \quad (2.39)$$

$${}^0\vec{\omega}_N = {}^0[R]{}^N\vec{\omega}_N \quad (2.40)$$

Velocità lineari e angolari sono quindi funzioni delle variabili di giunto e delle loro derivate nel tempo.

$${}^0\vec{v}_N = f(q, \dot{q}) \quad (2.41)$$

$${}^0\vec{\omega}_N = f'(q, \dot{q}) \quad (2.42)$$

2.6 Cinematica di accelerazione

A partire dall'equazione della velocità lineare:

$${}^A\vec{v}_P = {}^A\vec{v}_{OB} + {}^A[R]{}^B\vec{v}_{P,B} + {}^A\vec{\omega}_B \times {}^A[R]{}^B\vec{P} \quad (2.43)$$

e derivando nel tempo si ottiene:

$$\begin{aligned} {}^A\dot{\vec{v}}_P &= {}^A\dot{\vec{v}}_{OB} + {}^A[R]{}^B\dot{\vec{v}}_{P,B} + {}^A\dot{\vec{\omega}}_B \times {}^A[R]{}^B\vec{v}_{P,B} + {}^A\dot{\vec{\omega}}_B \times {}^A[R]{}^B\vec{P} + \\ &\quad + {}^A\vec{\omega}_B \times {}^A[R]{}^B\dot{\vec{v}}_{P,B} + {}^A\vec{\omega}_B \times {}^A\dot{\vec{\omega}}_B \times {}^A[R]{}^B\vec{P} \end{aligned} \quad (2.44)$$

$${}^A\dot{\vec{v}}_P = {}^A[R]{}^B\dot{\vec{v}}_{P,B} + 2{}^A\vec{\omega}_B \times {}^A[R]{}^B\vec{v}_{P,B} + {}^A\dot{\vec{v}}_{OB} + {}^A\dot{\vec{\omega}}_B \times {}^A[R]{}^B\vec{P} + {}^A\vec{\omega}_B \times {}^A\dot{\vec{\omega}}_B \times {}^A[R]{}^B\vec{P} \quad (2.45)$$

Dove i termini che compaiono a secondo membro dell'ultima equazione sono rispettivamente l'accelerazione relativa, l'accelerazione di Coriolis e l'accelerazione di trascinamento.

Esprimendo l'accelerazione nel sistema B quindi si ottiene:

$${}^B\dot{\vec{v}}_P = {}^B\dot{\vec{v}}_{P,B} + 2{}^B\vec{\omega}_B \times {}^B\vec{v}_{P,B} + {}^B\dot{\vec{v}}_{OB} + {}^B\dot{\vec{\omega}}_B \times {}^B\vec{P} + {}^B\vec{\omega}_B \times {}^B\dot{\vec{\omega}}_B \times {}^B\vec{P} \quad (2.46)$$

Può essere comodo, per studiare poi il generico manipolatore, riscrivere tale equazione rispetto ad un terzo sistema di riferimento C esterno (i+1).

$${}^C \dot{\vec{v}}_P = {}^B \dot{\vec{v}}_{P,B} + 2 {}^C_B [R] {}^B \vec{\omega}_B \times {}^C \vec{v}_{P,B} + {}^C_B [R] \left[{}^B \dot{\vec{v}}_{0,B} + {}^B \dot{\vec{\omega}}_B \times {}^B \vec{P} + {}^B \vec{\omega}_B \times {}^B \vec{\omega}_B \times {}^B \vec{P} \right] \quad (2.47)$$

Dalla velocità angolare:

$${}^A \vec{\omega}_C = {}^A \vec{\omega}_B + {}^A \vec{\omega}_{C,B} = {}^A \vec{\omega}_B + {}^A_B [R] {}^B \vec{\omega}_{C,B} \quad (2.48)$$

derivando ancora rispetto al tempo si ottiene l'espressione dell'accelerazione angolare,

$${}^A \dot{\vec{\omega}}_C = {}^A \dot{\vec{\omega}}_B + {}^A_B [R] {}^B \dot{\vec{\omega}}_{C,B} + {}^A \vec{\omega}_B \times {}^A_B [R] {}^B \vec{\omega}_{C,B} \quad (2.49)$$

La quale, riscritta rispetto al sistema esterno C diviene:

$${}^C \dot{\vec{\omega}}_C = {}^A_B [R] {}^B \dot{\vec{\omega}}_B + {}^C \dot{\vec{\omega}}_{C,B} + \times {}^C_B [R] {}^B \vec{\omega}_B \times {}^C \vec{\omega}_{C,B} \quad (2.50)$$

Ritorniamo ora al manipolatore generico dove:

- A=0
- B=i
- C=i+1

e riscriviamo le equazione delle due accelerazioni.

Per l'accelerazione angolare si ottiene quindi:

$${}^{i+1} \dot{\vec{\omega}}_{i+1} = {}^{i+1}_i [R] {}^i \dot{\vec{\omega}}_i + {}^{i+1} \dot{\vec{\omega}}_{i+1,i} + {}^{i+1}_i [R] {}^i \vec{\omega}_i \times {}^{i+1} \vec{\omega}_{i+1,i} \quad (2.51)$$

posto

$${}^{i+1} \dot{\vec{\omega}}_{i+1,i} = \ddot{\theta}_{i+1} {}^{i+1} \hat{z}_{i+1} \quad (2.52)$$

$${}^{i+1} \vec{\omega}_{i+1,i} = \dot{\theta}_{i+1} {}^{i+1} \hat{z}_{i+1} \quad (2.53)$$

si giunge all'equazione dell'accelerazione angolare per un generico giunto rotoidale.

$${}^{i+1} \dot{\vec{\omega}}_{i+1} = {}^{i+1}_i [R] {}^i \dot{\vec{\omega}}_i + \ddot{\theta}_{i+1} {}^{i+1} \hat{z}_{i+1} + {}^{i+1}_i [R] {}^i \vec{\omega}_i \times \dot{\theta}_{i+1} {}^{i+1} \hat{z}_{i+1} \quad (2.54)$$

Nel caso di giunto prismatico, invece, si ha che:

$$\dot{\theta}_{i+1} = \ddot{\theta}_{i+1} = 0 \quad (2.55)$$

quindi l'accelerazione angolare risulta essere semplicemente:

$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}_i[R] \dot{\omega}_i \quad (2.56)$$

Per l'accelerazione lineare possiamo scrivere:

$${}^{i+1}\dot{v}_{i+1} = {}^{i+1}\dot{v}_{i+1,i} + 2 {}^{i+1}_i[R] \dot{\omega}_i \times {}^{i+1}\vec{v}_{i+1,i} + {}^{i+1}_i[R] \left[{}^i\dot{v}_i + {}^i\dot{\omega}_i \times {}^i\vec{P}_{i+1} + {}^i\vec{\omega}_i \times {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \right] \quad (2.57)$$

e, ponendo

$${}^{i+1}\dot{v}_{i+1,i} = \ddot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.58)$$

$${}^{i+1}\vec{v}_{i+1,i} = \dot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} \quad (2.59)$$

si ottiene l'equazione dell'accelerazione lineare per un giunto prismatico. Questa risulta essere quindi:

$$\begin{aligned} {}^{i+1}\dot{v}_{i+1} = & \ddot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} + 2 {}^{i+1}_i[R] \dot{\omega}_i \times \dot{d}_{i+1} {}^{i+1}\hat{z}_{i+1} + \\ & + {}^{i+1}_i[R] \left[{}^i\dot{v}_i + {}^i\dot{\omega}_i \times {}^i\vec{P}_{i+1} + {}^i\vec{\omega}_i \times {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \right] \end{aligned} \quad (2.60)$$

Per un giunto rotoidale invece, essendo $\dot{d}_{i+1} = \ddot{d}_{i+1} = 0$, l'espressione precedente diviene:

$${}^{i+1}\dot{v}_{i+1} = {}^{i+1}_i[R] \left[{}^i\dot{v}_i + {}^i\dot{\omega}_i \times {}^i\vec{P}_{i+1} + {}^i\vec{\omega}_i \times {}^i\vec{\omega}_i \times {}^i\vec{P}_{i+1} \right] \quad (2.61)$$

Come per la velocità, in sostanza, attraverso iterazioni successive, siamo in grado di determinare l'accelerazione dell'end-effector a partire dalla base. Possiamo quindi scrivere:

$${}^0\dot{v}_N = {}^0_N[R] {}^N\dot{v}_N \quad (2.62)$$

$${}^0\dot{\vec{\omega}}_N = {}^0_N[R]{}^N\dot{\vec{\omega}}_N \quad (2.63)$$

E' inoltre utile ricavare l'accelerazione del baricentro (C).

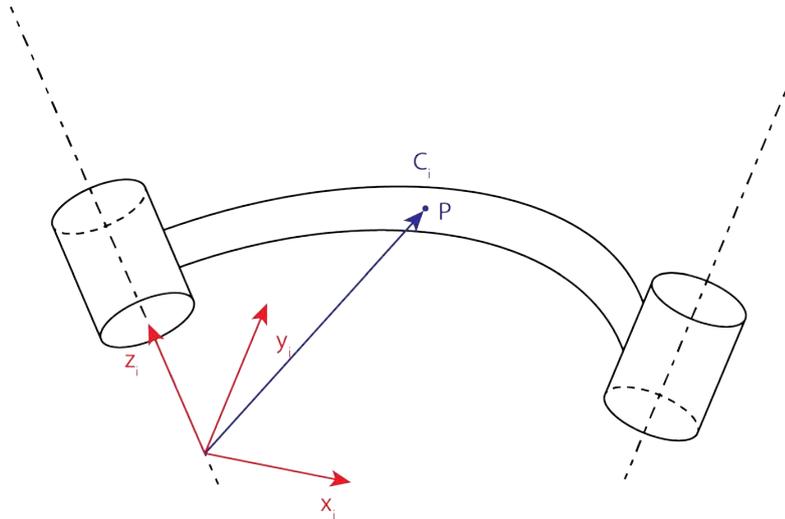


Figura 2.7: Sistema di riferimento per il calcolo dell'accelerazione del baricentro

$${}^i\dot{\vec{v}}_{Ci} = {}^i\dot{\vec{v}}_i + {}^i\dot{\vec{\omega}}_i \times {}^i\dot{\vec{P}}_{Ci} + {}^i\vec{\omega}_i \times {}^i\vec{\omega}_i \times {}^i\dot{\vec{P}}_{Ci} \quad (2.64)$$

A questo punto, per ogni corpo si hanno cinque equazioni:

- velocità del punto i
- velocità del punto $i + 1$
- accelerazione del punto i
- accelerazione del punto $i + 1$
- accelerazione del baricentro

2.7 Pianificazione della traiettoria

Per la pianificazione della traiettoria è necessario passare dallo spazio cartesiano allo spazio dei giunti.

$$\left(\vec{P}_{in}, \vec{P}_f \right) \rightarrow (\theta_{1,in}, \theta_{2,in}, \theta_{3,in}, \theta_{1,f}, \theta_{2,f}, \theta_{3,f}) \quad (2.65)$$

E' quindi necessario trovare una traiettoria per i giunti affinché l'end-effector passi dal punto \vec{P}_i al punto \vec{P}_f , ossia un polinomio funzione del tempo tale che:

$$\theta(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (2.66)$$

$$\dot{\theta}(t) = a_1 + 2a_2t + 3a_3t^2 + 4a_4t^3 + 5a_5t^4 \quad (2.67)$$

$$\ddot{\theta}(t) = 2a_2 + 6a_3t + 12a_4t^2 + 20a_5t^3 \quad (2.68)$$

Si sceglie un polinomio che sia almeno del quinto ordine in modo da poter determinare agevolmente le costanti adatte al fine di soddisfare le condizioni iniziali e finali di posizione, velocità e accelerazione del giunto. in caso di limiti di accelerazionePuò essere comodo anche sfruttare il jerk, ovvero la derivata prima della accelerazione. E persino il jounce, la sua derivata seconda. Un esempio famoso dell'utilizzo di entrambe è stata la progettazione del telescopio Hubble, in modo da limitare la possibilità di avere delle forze distruttive. Imporre un limite al jerk corrisponde ad avere un incremento lineare dell'accelerazione fino al limite della stessa, ed un incremento quadratico della velocità. Imponendo queste condizioni il controllo risulta infatti più delicato e con meno "sobbalzi".

Le condizioni note sono:

$$\theta(0) = \theta_i \quad \dot{\theta}(0) = 0 \quad \ddot{\theta}(0) = 0$$

$$\theta(t_f) = \theta_f \quad \dot{\theta}(t_f) = 0 \quad \ddot{\theta}(t_f) = 0$$

Dalle condizioni iniziali e finali ricavo quindi le costanti.

$$\left\{ \begin{array}{l} \theta(0) = \theta_i \Rightarrow a_0 = \theta_i \\ \dot{\theta}(0) = 0 \Rightarrow a_1 = 0 \\ \ddot{\theta}(0) = 0 \Rightarrow a_2 = 0 \\ \theta(t_f) = \theta_f \Rightarrow \theta_i + a_3t_f^3 + a_4t_f^4 + a_5t_f^5 = \theta_f \\ \dot{\theta}(t_f) = 0 \Rightarrow 3a_3t_f^2 + 4a_4t_f^3 + 5a_5t_f^4 = 0 \\ \ddot{\theta}(t_f) = 0 \Rightarrow 6a_3t_f + 12a_4t_f^2 + 20a_5t_f^3 = 0 \end{array} \right. \quad (2.69)$$

$$a_3 = \frac{10(\theta_f - \theta_i)}{t_f^3} \quad a_4 = \frac{-15(\theta_f - \theta_i)}{t_f^4} \quad a_5 = \frac{6(\theta_f - \theta_i)}{t_f^5} \quad (2.70)$$

2.7.1 Traiettorie parametrizzate

Nel caso in cui si voglia definire delle traiettorie ben definite, come ad esempio linee rette, circonferenze, o qualsivoglia funzione matematica si può riparametrizzare le curve che vogliamo seguire secondo altre funzioni.

Traiettoria rettilinea

In questa sede viene trattata unicamente la traiettoria rettilinea come esempio, in quanto tutte le varie riparametrizzazioni di curve seguono la stessa metodologia, con ovviamente diverse curve di partenza. È noto che la parametrizzazione di un tratto lineare compreso tra gli estremi P_i e P_f , si può scrivere come:

$$P(t) = P_i + \frac{t}{T}(P_f - P_i) \quad (2.71)$$

con t il tempo che trascorre, e T il tempo totale dello spostamento. Si prenda ora una funzione $s(t)$, che noi assumeremo come polinomio del quinto ordine, in quanto si vogliono inserire delle condizioni anche per quanto riguarda la accelerazione, come descritto in precedenza:

$$s(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (2.72)$$

allora l'equazione precedente può essere riscritta ora in funzione del parametro $0 < s(t) < \|P_f - P_i\|$:

$$P(s(t)) = P_i + \frac{s(t)}{\|P_f - P_i\|}(P_f - P_i) \quad (2.73)$$

E, per le stesse considerazioni, tenendo conto che:

$$s(T) = \|P_f - P_i\| \quad (2.74)$$

si ottengono le condizioni

$$a_3 = \frac{10\|P_f - P_i\|}{T^3} \quad a_4 = \frac{-15\|P_f - P_i\|}{T^4} \quad a_5 = \frac{6\|P_f - P_i\|}{T^5} \quad (2.75)$$

e, quindi si ha in conclusione:

$$s(t) = \frac{\|P_f - P_i\|}{T^5} (10T^2t^3 - 15Tt^4 + 6t^5) \quad (2.76)$$

2.8 Jacobiano del manipolatore

Sia $x \in \mathfrak{R}^n$ e $y \in \mathfrak{R}^m$ vettori definiti rispettivamente nello spazio n-dimensionale e m-dimensionale, e sia $f : \mathfrak{R}^n \rightarrow \mathfrak{R}^m$ una funzione tale che

$$\begin{cases} y_1 = f_1(x_1, x_2, \dots, x_n) \\ y_2 = f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ y_m = f_m(x_1, x_2, \dots, x_n) \end{cases} \quad (2.77)$$

allora si può scrivere:

$$\dot{y} = \left[\frac{\partial f_i}{\partial x_j} \right] \dot{x} \quad (2.78)$$

dove

$$\dot{y}_i = \sum_{j=1}^n \left[\frac{\partial f_i}{\partial x_j} \dot{x}_j \right] \quad (2.79)$$

Si definisce con Jacobiano la matrice delle derivate parziali.

$$[J(x)] = \left[\frac{\partial f_i}{\partial x_j} \right] \quad (2.80)$$

In tale contesto valgono le relazioni per le derivate e per gli spostamenti infinitesimi:

$$\dot{y} = [J(x)] \dot{x} \quad (2.81)$$

$$\delta \vec{y} = [J(x)] \delta \vec{x} \quad (2.82)$$

Nello spazio dei manipolatori si ha:

$$\dot{y} = {}^0\vec{V} = \begin{bmatrix} {}^0\dot{v}_N^T \\ {}^0\dot{\omega}_N^T \end{bmatrix} \quad (2.83)$$

e indicando con q le posizioni e con \dot{q} le velocità dei punti visti in precedenza, dalle relazioni:

$$\vec{x} = \vec{q} \quad (2.84)$$

$$\dot{\vec{x}} = \dot{\vec{q}} \quad (2.85)$$

possiamo scrivere:

$${}^0\vec{V} = [J(q)] \dot{\vec{q}} \quad (2.86)$$

dove abbiamo indicato con ${}^0\dot{\vec{V}}$ la velocità dell'end-effector e con $[J(q)]$ lo Jacobiano del manipolatore.

Lo Jacobiano è utile per la determinazione di:

1. velocità e accelerazione dell'end-effector attraverso la cinematica diretta

$${}^0\vec{V} = [J(q)] \dot{\vec{q}} \quad (2.87)$$

$${}^0\dot{\vec{V}} = [J(q)] \ddot{\vec{q}} + [\dot{J}(q)] \dot{\vec{q}} \quad (2.88)$$

2. velocità e accelerazione dei giunti nota la velocità dell'end-effector attraverso la cinematica inversa

$$\dot{\vec{q}} = [J(q)]^{-1} {}^0\vec{V} \quad (2.89)$$

Nel secondo caso si deve tenere presente che l'inversione dello Jacobiano è possibile solo se esso è quadrato e ha determinante non nullo. Il determinante si annulla per le configurazioni singolari, pertanto queste si devono evitare.

Se invece lo Jacobiano non risulta essere quadrato si ricorre alla pseudo-inversione, ossia si definisce una funzione costo $g(\dot{q})$ come segue:

$$g(\dot{q}) = \frac{1}{2} \dot{q}^T [W] \dot{q} \quad (2.90)$$

Dove $[W]$ è la matrice simmetrica dei pesi attribuiti ai diversi giunti; essa si ricava da considerazioni energetiche.

In forma scalare l'equazione precedente diviene:

$$g(\dot{\vec{q}}) = \frac{1}{2} \sum_{i=1}^n \left[\dot{q}_i \sum_{j=1}^n (w_{ij} \dot{q}_j) \right] = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\dot{q}_i w_{ij} \dot{q}_j) \quad (2.91)$$

Introducendo ora il vettore $\vec{\lambda}$ dei moltiplicatori di Lagrange e considerando l'equazione della velocità dell'end-effector dalla cinematica diretta, si ottiene:

$$g'(\dot{\vec{q}}, \lambda) = \frac{1}{2} \dot{\vec{q}}^T W \dot{\vec{q}} + \lambda^T \left({}^0 \dot{\vec{V}} - [J(q)] \dot{\vec{q}} \right) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n [\dot{q}_i w_{ij} \dot{q}_j] + \sum_{i=1}^r \lambda_i \left[v_i - \sum_{j=1}^n J_{ij} \dot{q}_j \right] \quad (2.92)$$

Posto:

$$\frac{\partial g'}{\partial \dot{q}_k} = 0 \quad (2.93)$$

$$\frac{\partial g'}{\partial \lambda} = 0 \quad (2.94)$$

la derivata della funzione costo rispetto alla generica variabile q_k risulta essere:

$$\frac{\partial g'}{\partial \dot{q}_k} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \left[w_{ij} \frac{\partial(\dot{q}_i \dot{q}_j)}{\partial \dot{q}_k} \right] - \sum_{i=1}^r \left[\lambda_i \sum_{j=1}^n J_{ij} \frac{\partial \dot{q}_j}{\partial \dot{q}_k} \right] \quad (2.95)$$

Essendo tuttavia $[W]$ simmetrica e $\frac{\partial \dot{q}_j}{\partial \dot{q}_k} \neq 0 \Leftrightarrow i = j$ posso riscrivere l'equazione precedente come:

$$\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \left[w_{ij} \frac{\partial(\dot{q}_i \dot{q}_j)}{\partial \dot{q}_k} \right] = \sum_{j=1}^n w_{kj} \dot{q}_j \quad (2.96)$$

$$\sum_{i=1}^r \left[\lambda_i \sum_{j=1}^n J_{ij} \frac{\partial \dot{q}_j}{\partial \dot{q}_k} \right] = \sum_{i=1}^r \lambda_i J_{ij} \quad (2.97)$$

e quindi:

$$\frac{\partial g'(\dot{\vec{q}}, \lambda)}{\partial \dot{\vec{q}}} = [W] \dot{\vec{q}} - [J(q)]^T \vec{\lambda} \quad (2.98)$$

Il sistema di equazioni che tiene conto della minimizzazione della funzione costo e dei vincoli è pertanto:

$$\begin{cases} \dot{\vec{q}} = [W]^{-1} [J(q)]^T \vec{\lambda} \\ {}^0\vec{V} = [J(q)] \dot{\vec{q}} \end{cases} \quad (2.99)$$

Si può dimostrare che se lo Jacobiano $[J(q)]$ di dimensioni $r \times n$ ha rango massimo (r) allora la matrice $[J(q)] [W]^{-1} [J(q)]^T$ è invertibile e quadrata, quindi si ha:

$${}^0\vec{V} = [J(q)] [W]^{-1} [J(q)]^T \vec{\lambda} \quad (2.100)$$

da cui:

$$\vec{\lambda} = \left[[J(q)] [W]^{-1} [J(q)]^T \right]^{-1} {}^0\vec{V} \quad (2.101)$$

ed infine:

$$\dot{\vec{q}} = [W]^{-1} [J(q)]^T \left[[J(q)] [W]^{-1} [J(q)]^T \right]^{-1} {}^0\vec{V} \quad (2.102)$$

A partire dall'equazione che lega la velocità dell'end-effector alla velocità dei punti:

$$\vec{V} = [J(q)] \dot{\vec{q}} \quad (2.103)$$

e derivando nel tempo si ottiene la sua accelerazione:

$$\dot{\vec{V}} = [J(q)] \ddot{\vec{q}} + [\dot{J}(q)] \dot{\vec{q}} \quad (2.104)$$

Da cui:

$$\ddot{\vec{q}} = [J(q)]^{-1} \left[\dot{\vec{V}} - [\dot{J}(q)] \dot{\vec{q}} \right] \quad (2.105)$$

dove:

$$\dot{J}_{ij}(q) = \sum_k \frac{\partial J_{ij}(q)}{\partial q_k} \dot{q}_k \quad (2.106)$$

2.8.1 Trasformazione delle forze per il cambio di coordinate

Il sistema può essere considerato in equilibrio istante per istante. È quindi possibile utilizzare il Principio dei lavori Virtuali per un sistema di forze e momenti esterni: il lavoro virtuale compiuto dalle forze esterne è uguale a quello compiuto dalle forze interne.

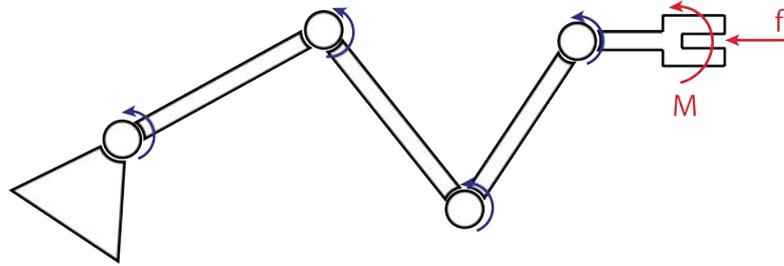


Figura 2.8: Posizione delle forze da considerare per la loro trasformazione, dovuta al cambio di coordinate

$$\delta L_{ext} = \delta L_{in} \quad (2.107)$$

Sia $\{\delta q\}$ il vettore degli spostamenti virtuali. Si ha pertanto che:

$$\delta L_{in} = \{\tau\}^T \{\delta \vec{q}\} \quad (2.108)$$

con

$$\{\tau\} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix} \quad (2.109)$$

vettore delle forze interne. Definito $\{\delta P\}$ lo spostamento infinitesimo virtuale dell'end-effector e $\{\delta \epsilon\}$ la rotazione infinitesima virtuale dell'end-effector, dove $\{\delta \epsilon\} = \{\omega\} \delta t$ si definiscono:

$$\{\delta \vec{X}\} = \begin{bmatrix} \{\delta P\} \\ \{\delta \epsilon\} \end{bmatrix} \quad (2.110)$$

e

$$\{\vec{F}\} = \begin{bmatrix} \{f\} \\ \{M\} \end{bmatrix} \quad (2.111)$$

dove $\{f\}$ è il vettore delle forze esterne¹ agenti sull'end-effector e $\{M\}$ è il vettore

¹Non si considerano le forze esterne che agiscono sui singoli giunti ma solo quelli agenti sull'end-effector.

dei momenti esterni applicati sempre all'end-effector. Per il P.L.V. si ha quindi:

$$\{\tau\}^T \delta \vec{q} = \{\vec{F}\}^T \{\delta \vec{X}\} \quad (2.112)$$

ed essendo:

$$\{\delta \vec{X}\} = [J(q)] \{\delta \vec{q}\} \quad (2.113)$$

possiamo scrivere:

$$\{\tau\}^T \delta \vec{q} = \{\vec{F}\}^T [J(q)] \{\delta \vec{q}\} \quad (2.114)$$

Raccogliendo e semplificando si giunge infine a:

$$\{\tau\}^T \delta \vec{q} = \{\vec{F}\}^T [J(q)] \{\delta \vec{q}\} \quad (2.115)$$

$$\{\vec{F}\}^T [J(q)] = \{\tau\}^T \quad (2.116)$$

$$\{\tau\} = [J(q)]^T \{\vec{F}\} \quad (2.117)$$

CAPITOLO 3

Dinamica dei manipolatori

Definiamo con $f_{i,j}$ e $n_{i,j}$ rispettivamente la forza e il momento (coppia) trasmessi dal corpo i -esimo al j -esimo. Si tenga presente che ogni elemento i -esimo ha il proprio sistema di riferimento ed esso non è per forza centrato nel baricentro dell'elemento stesso.

L'equazione cardinale della dinamica per le forze agenti sul generico elemento diventa:

$${}^i \vec{F}_i = m_i {}^i \dot{v}_{C_i} \quad (3.1)$$

e per il P.L.V. quindi:

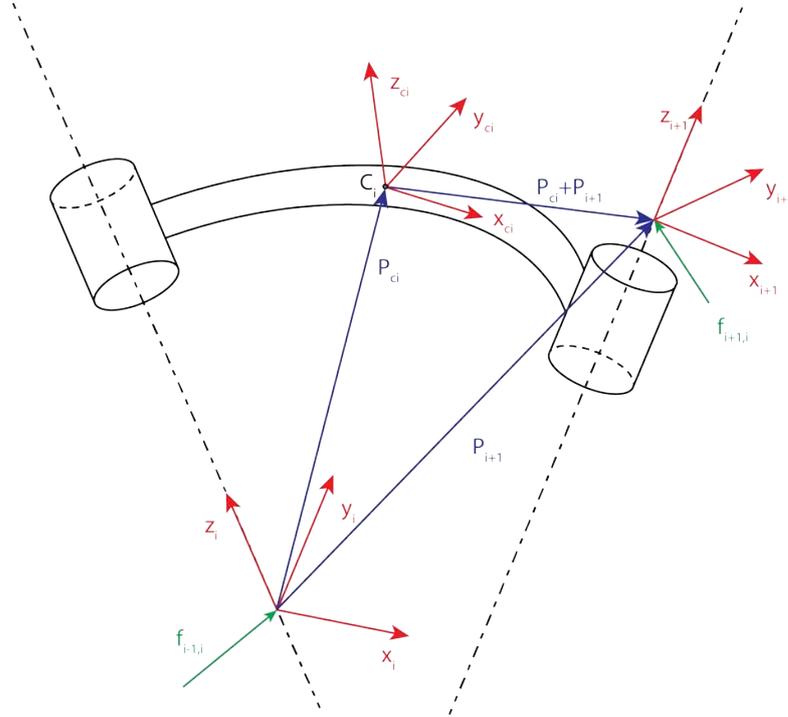
$$\vec{f}_e = {}^i \vec{f}_{i-1,i} + {}^i \vec{f}_{i+1,i} = {}^i \vec{f}_{i-1,i} - {}^i \vec{f}_{i,i+1} \quad (3.2)$$

dalla quale ricaviamo la formula iterativa per calcolare le forze che la base esplica ai corpi, una volta note le forze agenti sull'end-effector:

$${}^i \vec{f}_{i-1,i} = {}^i \vec{F}_i + {}_{i+1}^i [R] {}^i \vec{f}_{i,i+1} \quad (3.3)$$

L'equazione cardinale della dinamica per i momenti agenti sul generico elemento si esprime come:

$$\vec{n}_e = {}^i \vec{N}_i = {}^{C^i} [I]_i {}^i \dot{\omega}_i + {}^i \vec{\omega}_i \times {}^{C^i} [I]_i {}^i \vec{\omega}_i \quad (3.4)$$


 Figura 3.1: Es. corpo i –esimo

dove ${}^{Ci}[I]_i$ è il tensore d'inerzia del corpo con assi paralleli al sistema i -esimo centrato nel baricentro.

Nell'ipotesi che le forze vengano trasmesse all'origine del sistema di riferimento, ossia:

- $f_{i-1,i}$ applicato nell'origine del sistema i -esimo
- $f_{i+1,i}$ applicato nell'origine del sistema $i+1$ -esimo

si può scrivere:

$$\vec{n}_e = {}^i\vec{n}_{i-1,i} + {}^i\vec{n}_{i-1,i} - {}^i\vec{P}_{Ci} \times {}^i\vec{f}_{i-1,i} + ({}^i\vec{P}_{i+1} - {}^i\vec{P}_i) \times {}^i\vec{f}_{i+1,i} \quad (3.5)$$

e sostituendo con:

$${}^i\vec{f}_{i-1,i} = {}^i\vec{F}_i + {}_{i+1}{}^i[R] {}^i\vec{f}_{i,i+1} \quad (3.6)$$

si giunge all'equazione:

$$\vec{n}_e = {}^i\vec{n}_{i-1,i} + {}^i\vec{n}_{i-1,i} - {}^i\vec{P}_{Ci} \times {}^i\vec{F}_i - {}^i\vec{P}_{Ci} \times {}_{i+1}{}^i[R] {}^i\vec{f}_{i,i+1} + {}^i\vec{P}_{i+1} \times {}^i\vec{f}_{i+1,i} - {}^i\vec{P}_i \times {}^i\vec{f}_{i+1,i} \quad (3.7)$$

dove:

$${}^i \vec{f}_{i+1,i} = {}_{i+1}^i [R] {}^i \vec{f}_{i,i+1} \quad (3.8)$$

Sostituendo (3.8) in (3.7) e semplificando si ottiene:

$$\vec{n}_e = {}^i \vec{n}_{i-1,i} + {}^i \vec{n}_{i-1,i} - {}^i \vec{P}_{Ci} \times {}^i \vec{F}_i - {}^i \vec{P}_{i+1} \times {}^i \vec{f}_{i+1,i} \quad (3.9)$$

Le coppie che il corpo i-1-esimo deve trasmettere al corpo i-esimo per l'equilibrio saranno pertanto:

$${}^i \vec{n}_{i-1} = {}^i \vec{N}_i + {}_{i+1}^i [R] {}^{i+1} \vec{n}_{i,i+1} + {}^i \vec{P}_{Ci} \times {}^i \vec{F}_i + {}^i \vec{P}_{i+1} \times {}_{i+1}^i [R] {}^{i+1} \vec{f}_{i,i+1} \quad (3.10)$$

Infine si devono considerare le forze e le coppie che i motori esplicano sul giunto che risultano essere

- per un giunto rotoidale

$$\vec{\tau}_i = {}^i \vec{n}_{i-1} {}^T {}^i \hat{z}_i \quad (3.11)$$

- per un giunto prismatico

$$\vec{\tau}_i = {}^i \vec{f}_{i-1,i} {}^T {}^i \hat{z}_i \quad (3.12)$$

3.1 Equazioni dinamiche del manipolatore

Le coppie e le forze che i motori devono esplicitare sono quindi:

$$\vec{\tau} = \vec{\tau}(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) = [H(q)] \ddot{\vec{q}} + h(\vec{q}, \dot{\vec{q}}) \quad (3.13)$$

dove $[H(q)]$ è la matrice di massa del manipolatore e $h(\vec{q}, \dot{\vec{q}})$ sono i contributi centrifughi, di Coriolis e gravitazionali, i quali possono essere separati come:

$$h(\vec{q}, \dot{\vec{q}}) = c(\vec{q}, \dot{\vec{q}}) \dot{\vec{q}} + G(q) \quad (3.14)$$

L'equazione dell'energia cinetica per il manipolatore invece si presenta come:

$$T = \frac{1}{2} \{\dot{\vec{q}}\}^T [H(q)] \{\dot{\vec{q}}\} \quad (3.15)$$

CAPITOLO 4

Approccio Matriciale Alternativo

Detta $M_{0,1}$ la matrice di rototraslazione 4x4 secondo D-H:

$$M_{0,1} = \left| \begin{array}{ccc|c} R_{0,1} & & & t_{0,1} \\ \hline 0 & 0 & 0 & 1 \end{array} \right| \quad (4.1)$$

Si ricorda che la sua inversa è:

$$M_{0,1}^{-1} = M_{1,0} = \left| \begin{array}{ccc|c} R_{1,0} & & & t_{1,0} \\ \hline 0 & 0 & 0 & 1 \end{array} \right| \quad (4.2)$$

Per la quale, le relazioni matematiche sono:

$$R_{1,0} = R_{0,1}^t = R_{0,1}^{-1} \quad t_{1,0} = -R_{0,1}^t t_{0,1} \quad (4.3)$$

⁰In questo capitolo, per uniformità con gli articoli scientifici a cui si fa riferimento, viene utilizzata la notazione per cui le matrici di trasformazione ${}^{i-1}_i[T] = M_{i,i-1}$

4.1 Matrici e Tensori di Velocità e Accelerazione

4.1.1 Definizioni di base

In modo da estendere la trattazione ad un'analisi cinematica completa, devono essere introdotte 2 nuove matrici: W e H , rispettivamente matrice di velocità e di accelerazione.

$$W = \left| \begin{array}{ccc|c} 0 & -\omega_z & \omega_y & v_x \\ \omega_z & 0 & -\omega_x & v_y \\ -\omega_y & \omega_x & 0 & v_z \\ \hline 0 & 0 & 0 & 1 \end{array} \right| = \left| \begin{array}{ccc|c} \underline{\omega} & & & v_O \\ \hline 0 & 0 & 0 & 1 \end{array} \right| \quad (4.4)$$

Dove si indica con $\underline{\omega}$ la velocità angolare del corpo e con v_O la velocità del punto appartenente al corpo, che nell'istante considerato sta passando per l'origine del sistema di riferimento. La velocità \dot{P} di un punto P si ricorda che può essere ottenuta come

$$\dot{P} = WP = \left| \begin{array}{c|c} \dot{x}_P & x_P \\ \dot{y}_P & y_P \\ \dot{z}_P & z_P \\ \hline 0 & 0 \end{array} \right| = \left| \begin{array}{ccc|c} \underline{\omega} & & & v_O \\ \hline 0 & 0 & 0 & 1 \end{array} \right| \left| \begin{array}{c} x_P \\ y_P \\ z_P \\ 0 \end{array} \right| \quad (4.5)$$

È facile verificare che questa non sia altro che la formulazione matriciale della nota:

$$\vec{v}_P = \vec{v}_O + \vec{\omega} \times (P - O)$$

In modo simile, la accelerazione relativa di un corpo rispetto ad un sistema di riferimento può essere indicata con la matrice di accelerazione H :

$$H = \dot{W} + W^2 = \left| \begin{array}{ccc|c} G & & & a_O \\ \hline 0 & 0 & 0 & 1 \end{array} \right| \quad (4.6)$$

con G matrice 3x3 data da $G = \dot{\underline{\omega}} + \underline{\omega}^2$ e a_O la accelerazione del punto appartenente al

corpo, che nell'istante considerato sta passando per l'origine del sistema di riferimento.

$$\ddot{\mathbf{P}} = \mathbf{HP} = \begin{vmatrix} \ddot{x}_P \\ \ddot{y}_P \\ \ddot{z}_P \\ 0 \end{vmatrix} = \begin{vmatrix} \mathbf{G} & \mathbf{a}_O \\ 0 & 0 & 0 & 0 \end{vmatrix} \begin{vmatrix} x_P \\ y_P \\ z_P \\ 1 \end{vmatrix} \quad (4.7)$$

è facile verificare come questa sia la versione matriciale di:

$$\vec{a}_P = \vec{a}_O + \vec{\omega} \times (P - O) + \dot{\vec{\omega}} \times (P - O)$$

Derivata prima e seconda della matrice di trasformazione $M_{0,1}$ possono essere ottenute ricordando che ogni colonna rappresenta la posizione di un punto. Dalle equazioni 4.5 e 4.7 si possono scrivere le seguenti relazioni:

$$\dot{M}_{0,1} = \mathbf{W}M_{0,1} \quad \mathbf{W} = \dot{M}_{0,1}M_{0,1}^{-1}$$

$$\ddot{M}_{0,1} = \mathbf{H}M_{0,1} \quad \mathbf{H} = \ddot{M}_{0,1}M_{0,1}^{-1}$$

4.1.2 Cambio di sistema di riferimento

il valore assunto dagli elementi della matrice \mathbf{W} dipende dal sistema di riferimento utilizzato per due ragioni: primo, $\underline{\omega}$ e v_O sono rappresentati in funzione del loro sistema di riferimento; secondo, il polo è il punto del corpo passante sull'origine del sistema di riferimento scelto. La matrice \mathbf{H} si comporta allo stesso modo. Detti due sistemi di riferimento r e s si indicano le matrici W_i e H_i in funzione dell' i -esimo sistema. È possibile dimostrare che si comportano come dei tensori e che le loro trasformazioni siano:

$$W_r = M_{r,s}W_sM_{r,s}^{-1} \quad (4.8)$$

$$H_r = M_{r,s}H_sM_{r,s}^{-1} \quad (4.9)$$

chiamando $M_{r,s}$ la matrice di trasformazione del sistema s rappresentato nel sistema r . In altre parole W_r e W_s sono le rappresentazioni cartesiane in r e s di un tensore W . Lo stesso rimane valido per H . Per una dimostrazione si può vedere [5].

4.1.3 Cinematica Relativa

il moto relativo di due differenti corpi può essere facilmente studiato utilizzando le matrici presentate. Si considerino tre corpi i, j, k , se il moto relativo tra i corpi i e j , e quello tra j e k , sono noti rispetto ad una terna qualsiasi allora si può scrivere:

$$W_{i,k(r)} = W_{i,j(r)} + W_{j,k(r)} \quad (4.10)$$

$$H_{i,k(r)} = H_{i,j(r)} + H_{j,k(r)} + 2W_{i,j(r)}W_{j,k(r)} \quad (4.11)$$

Per una dimostrazione della 4.10 si veda l'appendice A di [5].

e poiché le matrici W e H sono rappresentazioni cartesiane di tensori, le azioni precedenti possono essere riscritte senza pedici:

$$W_{i,k} = W_{i,j} + W_{j,k} \quad (4.12)$$

$$H_{i,k} = H_{i,j} + H_{j,k} + 2W_{i,j}W_{j,k} \quad (4.13)$$

4.2 Dinamica

4.2.1 Matrice delle forze/coppie esterne (di azione)

Un sistema di forze esterne può essere rappresentato dalla matrice antisimmetrica Φ_k :

$$\Phi_k = \left| \begin{array}{c|c} \underline{c} & \underline{f} \\ \hline -\underline{f}^t & 0 \end{array} \right| = \left| \begin{array}{ccc|c} 0 & -c_z & c_y & f_x \\ c_z & 0 & -c_x & f_y \\ -c_y & c_x & 0 & f_z \\ \hline -f_x & -f_y & -f_z & 0 \end{array} \right| \quad (4.14)$$

con f risultante delle forze, e \underline{c} coppie calcolate rispetto all'origine del sistema di riferimento. Se si è in presenza di una forza infinitesima $f = [f_x f_y f_z 0]^t$ applicata ad un volume infinitesimo dv posizionato in P , la matrice ϕ è definita come l'integrale su tutto il corpo:

$$\Phi = \int (fP^t - Pf^t)dv \quad (4.15)$$

4.2.2 Matrice di quantità di moto/momento angolare

In modo simile si presenta anche la matrice antisimmetrica che si riferisce alla quantità di moto e al momento angolare Γ di un corpo k rispetto al riferimento:

$$\Gamma_k = \left| \begin{array}{c|c} \underline{\gamma} & \underline{\rho} \\ \hline -\underline{\rho}^t & 0 \end{array} \right| = \left| \begin{array}{ccc|c} 0 & -\gamma_z & \gamma_y & \rho_x \\ \gamma_z & 0 & -\gamma_x & \rho_y \\ -\gamma_y & \gamma_x & 0 & \rho_z \\ \hline -\rho_x & -\rho_y & -\rho_z & 0 \end{array} \right| \quad (4.16)$$

con ρ vettore quantità di moto, e γ momento angolare calcolato rispetto all'origine del sistema di riferimento.

$$\Gamma = \int (\dot{P}P^t - P\dot{P}^t)dm \quad (4.17)$$

4.2.3 Matrice di inerzia

La distribuzione di massa del corpo k –esimo può essere rappresentata dalla matrice antisimmetrica J , chiamata anche pseudo matrice di inerzia:

$$J_k = \left| \begin{array}{c|c} \underline{J} & \underline{q} \\ \hline \underline{q}^t & m \end{array} \right| = \left| \begin{array}{ccc|c} I_{xx} & I_{xy} & I_{xz} & q_x \\ I_{xy} & I_{yy} & I_{yz} & q_y \\ I_{xz} & I_{yz} & I_{zz} & q_z \\ \hline q_x & q_y & q_z & m \end{array} \right| \quad (4.18)$$

dove m è la massa e il vettore $q = m |x_g y_g z_g|^t$ è il prodotto della massa per la posizione del centro di massa del corpo. Si ricorda che gli elementi della sottomatrice J , diversa dall'usuale matrice di inerzia, sono definiti come:

$$\begin{aligned} I_{xx} &= \int x^2 dm & I_{yy} &= \int y^2 dm & I_{zz} &= \int z^2 dm \\ I_{xy} &= \int xy dm & I_{xz} &= \int xz dm & I_{yz} &= \int yz dm \end{aligned}$$

Data la definizione degli usuali momenti di inerzia J_{ii} si ha:

$$J_{xx} = \int (y^2 + z^2)dm = I_{yy} + I_{zz} \quad (4.19)$$

da cui:

$$I_{xx} = \frac{-J_{xx} + J_{yy} + J_{zz}}{2} \quad (4.20)$$

In notazione breve quindi:

$$J = \int PP^t dm$$

4.2.4 Cambio di sistema di riferimento

si può dimostrare che le matrici relative alla dinamica Φ , Γ e J sono anch'esse rappresentazioni cartesiane di tensori e che, dati due sistemi di riferimento r e s , valgono le relazioni:

$$\Phi_{k(r)} = M_{r,s} \Phi_{k(s)} M_{r,s}^t$$

$$\Gamma_{k(r)} = M_{r,s} \Gamma_{k(s)} M_{r,s}^t$$

$$J_{k(r)} = M_{r,s} J_{k(s)} M_{r,s}^t$$

dove $M_{r,s}$ è la posizione, in forma matriciale, del sistema s rispetto quello r .

4.2.5 Relazioni tra le matrici di cinematica e dinamica

Con le matrici precedenti si possono scrivere le relazioni classiche della meccanica. Si un sistema di riferimento inerziale (0) , allora la legge di Newton si può scrivere come:

$$\Phi_{k(0)} = H_{0,k} J_{k(0)} - J_{k(0)} H_{0,k}^t \quad (4.21)$$

e si può dimostrare che espandendo la precedente si ottiene:

$$\vec{f} = m\vec{a}_G \quad \vec{c} = I\vec{\dot{\omega}} + \vec{\omega} \times \vec{\gamma} + (G - O) \times \vec{f} \quad (4.22)$$

Dove I è il tensore di inerzia, G la posizione del centro di massa rispetto a (0) , $\vec{\gamma}$ il momento angolare, \vec{f} e \vec{c} sono la forza e la coppia che producono le accelerazioni \vec{a}_G e $\vec{\dot{\omega}}$.

La forza peso che agisce sul corpo può essere calcolata inserendo l'accelerazione di gravità nella matrice H_G e sfruttando l'equazione 4.21:

$$\Phi_{k(0)} = H_{g(0)} J_{k(0)} - J_{k(0)} H_{g(0)}^t \quad H_{g(0)} = \begin{array}{ccc|c} 0 & 0 & 0 & g_x \\ 0 & 0 & 0 & g_y \\ 0 & 0 & 0 & g_z \\ \hline 0 & 0 & 0 & 0 \end{array} \quad (4.23)$$

La quantità di moto e momento angolare di un corpo k la cui velocità è W_k e di inerzia J_k sono dati dalla formula:

$$\Gamma_k = W_k J_k - J_k W_k^t \quad (4.24)$$

e si può nuovamente dimostrare che espandendo la precedente si ottiene:

$$\vec{\rho} = m v \vec{G} \quad \vec{\gamma} = I \vec{\omega} + (G - O) \times \vec{\rho} \quad (4.25)$$

L'energia cinetica del corpo k è espressa alla relazione¹:

$$t_k = \frac{1}{2} \text{tr}(W_{0,k} J_{k(0)} W_{0,k}^t) \quad (4.26)$$

L'energia potenziale dovuta, invece, all'accelerazione di gravità si può esprimere come:

$$p_k = -\text{tr}(H_{g(0)} J_{k(0)}) \quad (4.27)$$

Una volta definite le energie cinetica e potenziale di un corpo, risulta facile estrarre le equazioni differenziali del moto utilizzando le equazioni di Lagrange.

4.2.6 Operatore "antisimmetria" (skew operator)

Per rendere più compatta la notazione, si definisce un operatore, in inglese "skew", tale che per ogni matrice quadrata \mathbf{X} o tensore X si ha:

$$\text{skew}[\mathbf{X}] = \mathbf{X} - \mathbf{X}^t$$

E, poiché anche Φ e Γ sono antisimmetriche, le equazioni 4.21 e 4.24 possono essere riscritte come:

$$\Phi_{k(0)} = \text{skew}[H_{0,k} J_{k(0)}] \quad (4.28)$$

$$\Gamma_{k(0)} = \text{skew}[W_{0,k} J_{k(0)}] \quad (4.29)$$

Si tenga inoltre del fatto che gli elementi indipendenti delle matrici Φ e Γ sono sei per ognuna.

¹si ricorda che con $\text{tr}(A)$ si intende la traccia della matrice, ovvero la somma dei termini sulla diagonale

4.3 Matrice L

Si definisce la matrice che rappresenta l'asse istantaneo di rotazione per una coppia cinematica elicoidale (ISA=istantaneous screw axis). Questa può essere ottenuta dalla matrice W dividendola per il modulo della sua velocità angolare²:

$$L = \frac{W}{|\omega|} = \left| \begin{array}{ccc|c} 0 & -u_z & u_y & b_x \\ u_z & 0 & -u_x & b_y \\ -u_y & u_x & 0 & b_z \\ \hline 0 & 0 & 0 & 0 \end{array} \right| = \left| \begin{array}{c|c} \underline{u} & \vec{b} \\ \hline 0 & 0 & 0 & 0 \end{array} \right|$$

Con $|\omega| = \sqrt{w_x^2 + w_y^2 + w_z^2}$. Se $|\omega| = 0$, la matrice L è definita:

$$L = \frac{W}{|\vec{b}|}$$

Dalla definizione di W si può vedere che \underline{u} è il versore dell'asse ISA, e \vec{b} può essere espresso come:

$$\vec{b} = -\underline{u}\vec{p}_{ax} + p\vec{u}$$

con p passo della vite e \vec{p}_{ax} punto dell'asse.

4.4 Metodo risolutivo

Si riassume ora il metodo utilizzabile per la risoluzione:

1. Si definiscono le matrici $M_{0,i}$
2. Si ricavano le matrici $W_{0,i}$
3. Si ricavano le matrici $H_{0,i}$
4. Si trova l'energia cinetica t_h e quella potenziale v_{ph} del sistema

Si risolve ora sfruttando la metodologia sviluppata da Lagrange, secondo la ben nota:

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = Q_i \quad (4.30)$$

²Si presti attenzione alla definizione della matrice L : essa si può considerare una sorta di *matrice rapporto di velocità generalizzata*

e, da [6] si può vedere che:

$$t_h = \frac{1}{2} \text{Traccia}(W_{h(0)} J_{h(0)} W_{h(0)}^t) \quad (4.31)$$

$$v_{ph} = \text{Traccia}(-H_{g(0)H_{h(0)}}) \quad (4.32)$$

In questo modo si possono agevolmente trovare le equazioni differenziali del moto del braccio, secondo la classica:

$$[H(q)] \ddot{\vec{q}} + C(\vec{q}, \dot{\vec{q}}) \dot{\vec{q}} + G(q) = \vec{\tau} = \vec{\tau}(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) \quad (4.33)$$

dove $[H(q)]$ è la matrice di massa del manipolatore e $h(\vec{q}, \dot{\vec{q}})$ sono i contributi centrifughi e di Coriolis mentre $G(q)$ include quelli gravitazionali.

5.1 La storia

5.1.1 Mitologia, leggende e automi

La parola robot viene coniata nel 1922 dallo scrittore di fantascienza ceco Karel Čapek nella sua opera Rossum's Universal Robots (RUR), e al tempo aveva l'accezione di corvée. All'inizio degli anni '40 del 20° secolo Isaac Asimov coniò la parola robotica per descrivere la scienza associata. Questi però non furono gli inizi della robotica, infatti sin dall'inizio della nostra storia, la storia dell'umanità, abbiamo cercato modi per compiere meno lavoro. La stessa ricerca del golem nella mitologia ebraica dimostra quanto un servo che possa compiere azioni al posto nostro sia un'immagine assolutamente affascinante. Partiamo però dall'inizio. Una leggenda cinese vuole che nel 10° secolo a.C. Yan Shi ha costruito un automa con fattezze umane (questo si può trovare nel Liezi, un testo del Taoismo). Nella mitologia greca, invece, sono gli dei che creano dei robot: Efesto, infatti, creò tre tavoli a tre gambe e un uomo di bronzo, Talos, che difese Creta. L'indiano Lokapannati, del 11° o 12° secolo racconta la storia di Re Ajatashatru di Magadha che raccolse e nasconde le reliquie di Buddha, queste erano protette da dei robot meccanici. Leggende cristiane, invece, affermano che il dottore della Chiesa Alberto Magno abbia creato un uomo di ottone nel 13° secolo era in grado di rispondere a domande complicate ed era impiegato come un domestico. Le leggende non finiscono, ma è ora di passare ai fatti. Nel quarto secolo d.C. il matematico greco Archita di Taranto apostolò uccello

meccanico chiamato il colombo propulso a vapore. Il motore cosmico, a una torre con l'orologio di 10 m costruita a Kaifeng in Cina da Su Song nel 1088 presentava manichini in grado di muoversi. Durante la dinastia Artuqid, l'inventore musulmano Al-Jazari (1136-1206) costruì innumerevoli macchine automatiche, e comprendevano strumenti per la cucina, automi musicali mossi ad acqua e il primo vero e proprio robot programmabile con fattezze umane: si trattava di un robot in grado di suonare dei tamburi, è in grado di cambiare le melodie grazie allo spostamento delle gamme che facevano muovere delle leve. Uno dei primi progetti trovati di un robot con fattezze umane risale al 1495 circa. L'autore di questo progetto era Leonardo da Vinci, e si trattava di un cavaliere meccanico in armatura generali sedersi, agitare le braccia, muovere la sua testa e la sua mandibola.

5.1.2 La robotica moderna

Con la definizione di Asimov negli anni '40 e con la stesura delle tre leggi fondamentali della robotica si apre una nuova epoca della robotica. Le tre leggi fondamentali sono:

1. Un robot non può arrecare danno ad un essere umano o, per merito di una mancata azione, per mettere che un essere umano venga danneggiato.
2. Un robot deve obbedire agli ordini datigli da esseri umani, con l'eccezione di quando questi siano in conflitto con la prima legge.
3. Un robot deve proteggere la propria esistenza, a patto che questo non sia in conflitto con la prima o con la seconda legge.

Passi fondamentali successivi sono stati [9, 14, 16]:

1947 Viene sviluppato il primo teleoperatore elettrico con servo

1948 Viene sviluppato il primo teleoperatore con feedback di forza

1949 Inizia la ricerca per sviluppare una fresa a controllo numerico

1954 George C. Devol fa richiesta di brevetto per un manipolatore programmabile (riceverà il brevetto nel 1961)

1956 George C. Devol e Joseph Engelberger si incontrano e fondano la Unimation Inc.

1961 Il primo robot della Unimation viene installato in una fabbrica della General Motors

1961 Viene sviluppato il primo robot con feedback di forza

- 1963** Viene sviluppato il primo sistema di visione robotica
- 1971** Lo Stanford Arm viene sviluppato all'università di Stanford
- 1973** Viene sviluppato il primo linguaggio di programmazione per robot (WAVE) a Stanford
- 1974** La compagnia Cincinnati Milacron introduce il T^3 , robot con controllo al computer
- 1974** I primi motoriduttori per applicazioni di attuazione robotica diventano disponibili. Nello stesso anno diventano commercialmente disponibili i primi robot con microprocessore
- 1975** La Unimation Inc. registra i primi profitti
- 1976** Bracci robotici vengono utilizzati su Viking I e Viking II e atterrano su Marte
- 1978** Unimation introduce il robot PUMA, basato su studi della General Motors
- 1979** Viene introdotto il modello SCARA in Giappone
- 1981** Viene sviluppato il primo robot direct-drive (ovvero con motori non collegati con organi di trasmissione ai link) all'Università di Caregie-Mellon
- 1983** Viene fondata la Adept Technology che immette sul mercato il robot precedente
- 1986** Il robot sottomarino Jason dell'Istituto oceanografico Wood Hole esplora il relitto del di Titanic
- 1993** Il robot sperimentale ROTEX dell'agenzia spaziale tedesca DLR viene fatto volare all'interno dello Space Shuttle Columbia per dei test
- 1996** La Honda presenta il suo robot con fattezze umane
- 1997** Il rover Sojourner viaggia fino a Marte
- 2001** La Sony produce il primo robot domestico, Aibo
- 2001** Viene lanciato a bordo dello Space Shuttle Endeavor la SSRMS (Space Station Remote Manipulation System)
- 2001** Viene effettuata la prima operazione a distanza
- 2001** Vengono utilizzati dei robot per cercare vittime dopo la tragedia delle torri gemelle
attualmente con robot si definisce un

5.2 Introduzione

Il braccio robotico è stato progettato per rispondere ai requisiti che sono stati delineati nel paragrafo 1.3.2. Data la sostanziale differenza delle tipologie di obiettivi che vengono ricercati, il braccio è stato progettato per essere innanzitutto facilmente rimovibile dalla rover. Questo è motivato dal fatto che durante la task numero due, la missione di attraversamento e raggiungimento di un obiettivo senza l'aiuto di fotocamere, la dinamica del robot risulta semplificata. Inoltre la riduzione dei pesi totali avrebbe portato ad un bonus di punti guadagnati. Ulteriormente vista la differenza di obiettivi da raggiungere, il passo è stato progettato con la possibilità di sostituire l'end-effector. Per minimizzare le spese sono stati progettati unicamente due terminali che, con appropriate modifiche durante le varie prove, sono in grado di rispondere a tutti i requisiti richiesti. Il primo terminale porta il braccio ad avere un totale di 4 gradi di libertà, e si presenta sostanzialmente come la benna di uno scavatore. Il secondo, invece, è un polso a 3 gradi di libertà, terminante con una pinza, ed un maschio per presa Schuko. Esso, a causa della possibilità di pioggia durante le prove è stato posizionato all'interno del polso stesso.

5.3 Requisiti

I requisiti che il braccio robotico deve soddisfare quindi sono:

1. raggiungere il terreno per effettuare scavi o raccogliere oggetti
2. deve poter depositare i campioni raccolti in contenitori
3. avere uno spazio di lavoro compreso tra 0,2 e 1,5 metri di altezza
4. deve essere molto leggero
5. deve essere economico
6. deve essere in grado di misurare tensioni
7. non deve consumare troppa potenza

Come si può intuire, se un unico braccio dovesse rispondere a tutti i requisiti richiesti, sarebbe estremamente sovradimensionato; se si dovesse utilizzare lo spazio di lavoro che si ha nella task 3, e pensare di avere come requisito di carico sollevabile quello della task 1 o 4, i motori/attuatori che si trovano alla base risulterebbero troppo ingombranti, pesanti, e assetati di potenza. Sfruttando però la possibilità di modificare il braccio

tra le varie prove, si è pensato di progettare due end-effector che possano soddisfare i requisiti discordanti delle varie prove. A causa del basso carico previsto nella task 3 e dell'ampio workspace richiesto, per un end-effector si è prediletta la destrezza del braccio e l'ampiezza dello spazio di lavoro a scapito della forza erogabile. Per le prove uno e quattro, invece, i requisiti di spazio di lavoro non sono così stringenti, mentre è necessario erogare delle coppie maggiori rispetto a quelle previste nel caso 3. Questo ha portato all'ideazione di un braccio con lunghezza dei link inferiore, ma con coppia erogabile dall'end-effector in linea con i requisiti richiesti. Il trade-off tra spazio di lavoro e potenza richiesta/coppia erogabile viene così scansato in favore di soluzioni multiple.

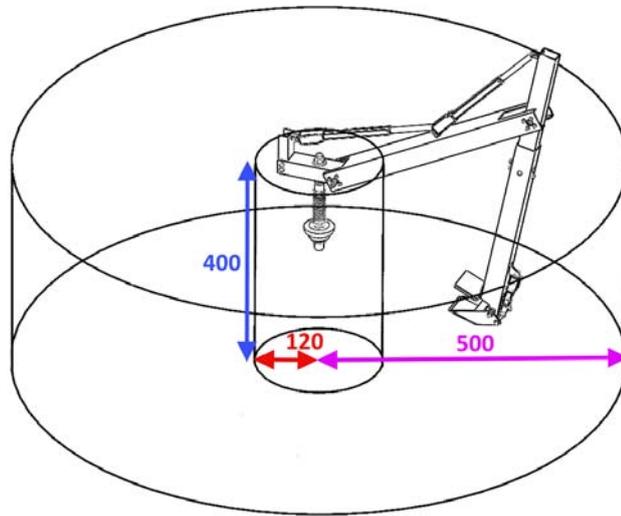
5.4 Scelta delle configurazioni

Lo spazio di lavoro è stato definito in prima battuta in funzione dei requisiti e delle task della competizione. Come è facilmente intuibile vi sono dei requisiti che contrastano tra loro: in due prove il braccio ha la necessità di sollevare un peso non indifferente, ma posto a distanza relativamente breve dal rover; mentre in una lo spazio di lavoro viene ad essere espanso, e le forze richieste da mettere in gioco sono notevolmente ridotte. Queste considerazioni hanno portato alla scelta di un braccio robotico con due end-effector intercambiabili da sostituire fra le prove della competizione. A causa delle differenti tipologie di prova, anche il numero di gradi di libertà delle configurazioni sarà diverso: nelle task dove non è necessaria una grande mobilità i G.d.L. sono 3 più quello dell'attuatore finale; nella task tre invece è necessaria una grande destrezza, pertanto si è optato per un braccio a 6 G.d.L., sempre attuatore finale escluso. Dalla definizione di uno spazio di lavoro preliminare si è scelta la lunghezza delle connessioni:

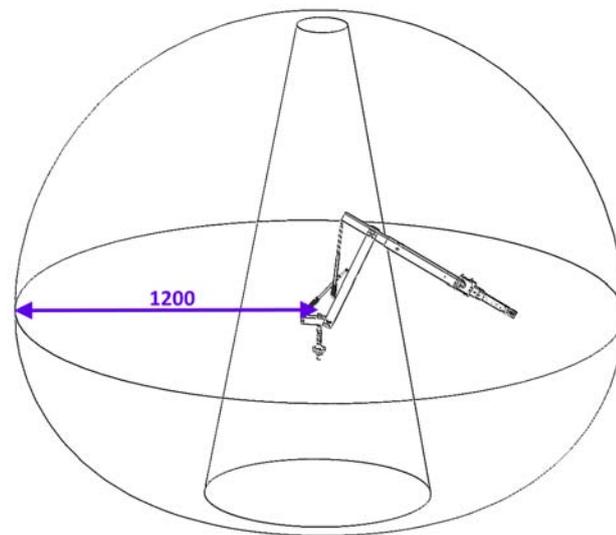
- Task 1/4: $l_1 = 0.5 \text{ m}$ $l_2 = 0.6 \text{ m}$
- Task 3: $l_1 = 0.5 \text{ m}$ $l_2 = 0.65 \text{ m}$ $l_3 \approx l_4 \approx 0.05 \text{ m}$ $l_5 = 0.12 \text{ m}$

Dalla figura 5.1 si può verificare come lo spazio di lavoro che viene ad essere generato risponda ai requisiti richiesti¹.

¹Si tenga nota del fatto che il rover stesso, sul quale verrà montato il braccio, ha un'altezza di 0,32 m.



(a) Task 1 e 4



(b) Task 3

Figura 5.1: Spazio di lavoro

5.5 Stima dei carichi e scelta delle sezioni

Parte integrante del progetto è stata l'analisi dei carichi a cui si prevede essere sottoposto il braccio robotico. Poiché in principio non sono stati forniti dati dettagliati né sulla tipologia di terreno, né sul peso della parte di ricambio da sollevare nella task 4, il lavoro di progettazione è stato molto rallentato. Ulteriori dati mancanti all'inizio erano la tipologia di interruttori e anche la tipologia di presa elettrica impiegati nella task 3. Successivamente sono stati forniti dei dati di massima unicamente sugli interruttori e sul peso della "spare part". In base alla documentazione presentata (e comunicazioni seguenti) dall'organizzazione che presiede l'ERC, i requisiti di carico espressamente formulati sono stati i seguenti:

- sollevamento di un masso del peso di almeno 100 g.
- sollevamento di "loose soil" per un totale di 200 g, non è necessario che siano sollevati in una volta sola.
- sollevamento di un pezzo di ricambio di peso non superiore ai 200 g.

In base alle informazioni contenute nei vari documenti, non è dato sapere, però, le caratteristiche del terreno. In base ai consigli formulati dall'organizzazione per la task 1, il metodo di raccolta del terreno "soffice" più consono sembra essere una paletta. Seguendo il consiglio formulato risulta intuitiva la necessità di quantificare le forze che vengono trasmesse al braccio dal terreno durante le operazioni di scavo.

5.5.1 Modello del terreno

Quando si effettua uno scavo nel terreno, le forze che si generano nel sistema sono per la maggior parte di attrito. Il comportamento attritivo del terreno, per essere modellato, ha bisogno di alcuni dati quali:

- densità media del terreno.
- diametro medio dei grani del terreno.
- coesione del terreno.
- angolo di resistenza al taglio.
- coefficiente di attrito tra metallo e terreno.

Per definire alcuni dati richiesti dal modello di scavo, e tenendo conto del fatto che la competizione vuole simulare un ambiente marziano, si sono assunti dei valori relativi al suolo di Marte. Per completezza si riportano i valori trovati in missioni passate:

Physical Properties of Mars Soils and Analogs.		
	ϕ	Cohesion (Pa)
Phoenix	29 – 47°	200-1200
MER Opportunity	20°	5000-8000
MER Spirit	20 – 25°	500-15000
Pathfinder	15 – 41°	10-600
Viking Landers		
VL-1 drift material	18.0±2.4°	1600±1200
VL-1 blocky material	30.8±2.4°	5100±2700
VL-2 crusty material	34.5±4.7°	1100±800
JSC Mars-1	47°	210
Quartz sand	41.4°	24
MMS up to 5 wt% H2O	32.5°	718

Tabella 5.1: Proprietà del suolo marziano
[2]

Per questa trattazione la tensione di taglio che porta a *failure* il terreno è data dalla relazione di Mohr-Coulomb:

$$\tau = \sigma \tan(\phi) + c \quad (5.1)$$

dove τ [Pa] è la tensione di taglio allo snervamento, σ [Pa] è la tensione normale, ϕ è l'angolo di resistenza al taglio e c è la coesione. Ulteriormente si è cercato un modello che potesse essere rappresentativo del fenomeno che stiamo cercando di riprodurre.

Il modello trovato è stato sviluppato da *Luengo et al.* [13] basandosi sul lavoro di *Reece* [15], e la sua *Fundamental Earth-moving Equation (FEE)*:

$$F = (\gamma g d^2 N_\gamma + c d N_c + q d N_q) w \quad (5.2)$$

con F la resistenza che sente la lama, γ densità del terreno, g accelerazione di gravità, d profondità dello strumento sotto al suolo, c coesione del terreno, q pressione verticale agente sulla superficie, w larghezza dello strumento e N_γ , N_c e N_q sono dei coefficienti che dipendono dalla componente attrittiva del suolo, dalla geometria dello strumento e dall'interazione suolo-strumento.

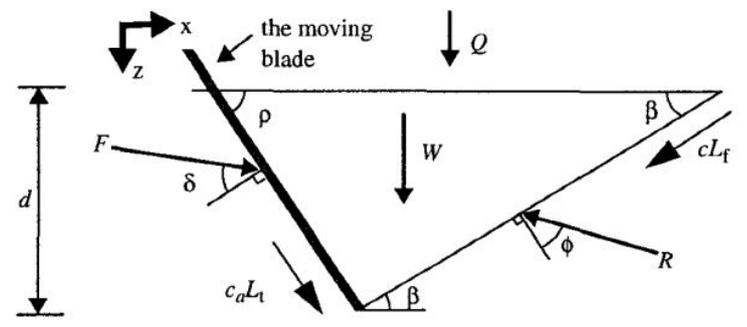


Figura 5.2: Modello dello scavo. Da [15]

Le componenti della forza resistente possono essere scritte come:

$$\begin{aligned} F_x &= F \sin(\rho + \delta) + c_a L_t \cos \rho - R \sin(\beta + \phi) - c L_f \cos \beta = 0 \\ F_z &= -F \cos(\rho + \delta) + c_a L_r \sin \rho + c L_f \sin \beta - R \cos(\beta + \phi) + W + Q = 0 \end{aligned} \quad (5.3)$$

e ricavando F si ottiene:

$$F = \frac{W + Q + cd[1 + \cot \beta \cot(\beta + \phi)] + c_a d[1 - \cot \rho \cot(\beta + \phi)]}{\cos(\rho + \phi) + \sin(\rho + \phi) \cot(\beta + \phi)} \quad (5.4)$$

quest'equazione può essere riscritta nella forma di 5.2, e assumendo che il materiale (terreno) spostato in precedenza sia uniformemente distribuito si ottiene:

$$\begin{aligned} N_\gamma &= \frac{\cot \rho + \cot \beta}{2[\cos(\rho + \delta) + \sin(\rho + \delta) \cot(\beta + \phi)]} \\ N_c &= \frac{1 + \cot \beta \cot(\beta + \phi)}{\cos(\rho + \delta) + \sin(\rho + \delta) \cot(\beta + \phi)} \\ N_q &= \frac{\cot \rho + \cot \beta}{\cos(\rho + \delta) + \sin(\rho + \delta) \cot(\beta + \phi)} \end{aligned} \quad (5.5)$$

La paletta è stata studiata per rispondere al requisito di raccolta di sabbia in tre raccolte. Pertanto le dimensioni (vedere fig.5.3) sono state calcolate in funzione del volume di terreno previsto.

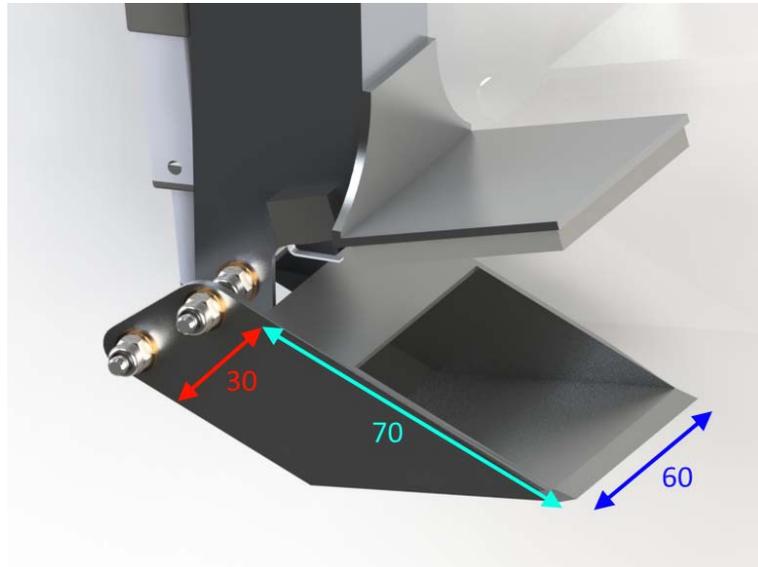


Figura 5.3: Dimensioni della paletta

Sfruttando il modello sopra proposto si sono fatti variare i parametri per effettuare una stima sulle forze che vengono esplicate sulla paletta. I risultati mostrano come non vi sia una forte variazione delle forze agenti sul *tool* in caso di profondità di scavo modeste. In ogni caso le forze agenti sul end-effector, rimanendo inferiori ad una plausibile profondità di 10 mm, non superano i 30 Newton.

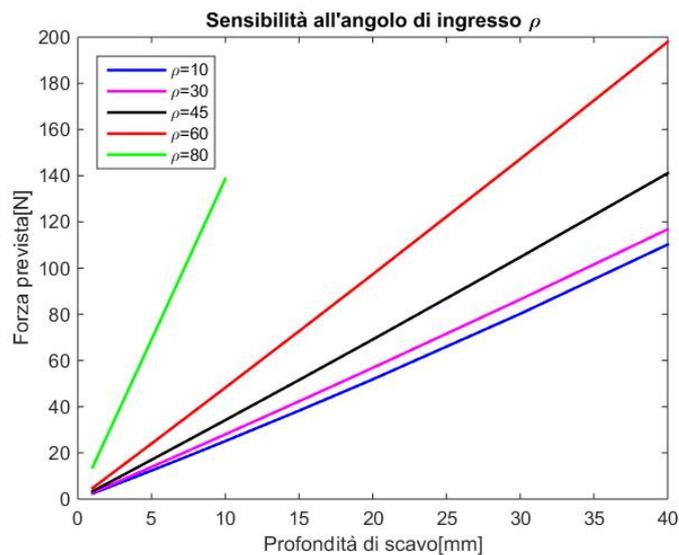


Figura 5.4: Sensibilità della forza all'angolo di contatto col terreno

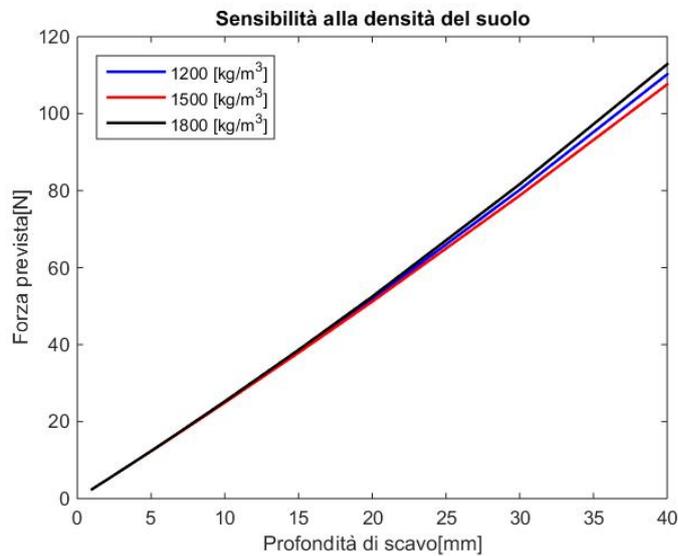


Figura 5.5: Sensibilità della forza alla densità del terreno

5.6 Modello dei carichi

Per poter dimensionare accuratamente il braccio robotico, è stato necessario creare dei modelli, anche se semplificati, dei carichi a cui sarebbe stato sottoposto.

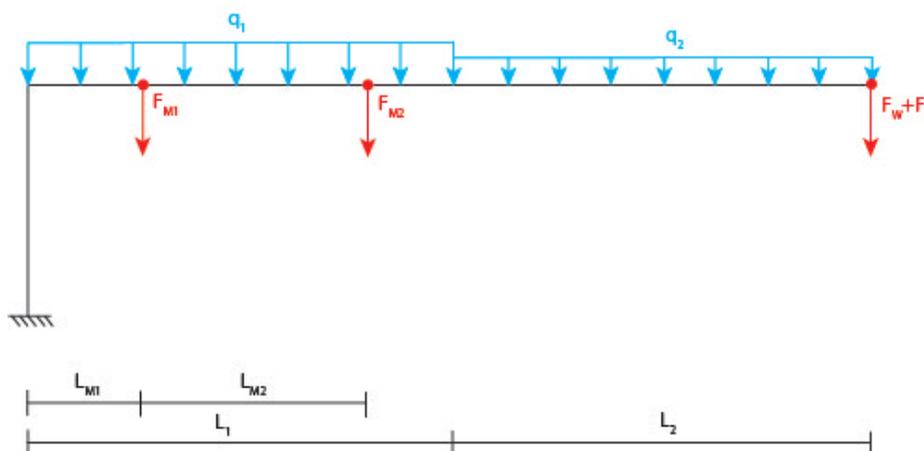


Figura 5.6: Modello dei carichi 1

Per definire il momento d'inerzia d'area della sezione, in base ai due modelli indicati in figura, si è impostato il problema andando a definire una freccia massima che i vari link possano avere.

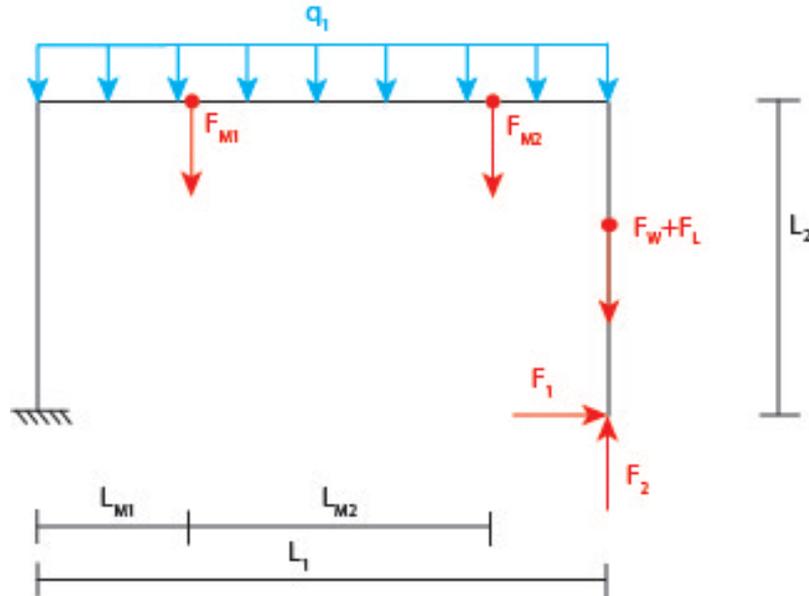


Figura 5.7: Modello dei carichi 2

In base all'equazione classica di scienza delle costruzioni della freccia in funzione del momento applicato:

$$v''_{xx} = -\frac{M_x(x)}{EI_{xx}} \quad (5.6)$$

Si è utilizzato il modello lineare, in quanto l'ipotesi di piccoli spostamenti viene rispettata. Pertanto dividendo il problema nelle sue componenti e tramite il principio di sovrapposizione degli effetti si è giunti a delle equazioni differenziali, risolvibili con le condizioni al contorno. Tramite un'integrazione si può quindi arrivare ad una soluzione.

Impostando così i parametri richiesti si è giunti al minimo momento d'inerzia d'area che devono avere le varie sezioni per soddisfare i requisiti imposti. Da catalogo, si è cercato poi delle soluzioni commerciali che avessero I_{xx} e I_{yy} superiori a quelli calcolati. Una volta dimensionate le varie sezioni, si sono fatti dei semplici test con dei modelli agli elementi finiti in Nastran per verificare che il comportamento sia quello atteso. Gli elementi utilizzati a causa della scarsità di tempo sono sempre stati quelli monodimensionali, e non si è avuto la possibilità di studiare meglio il braccio in progettazione.

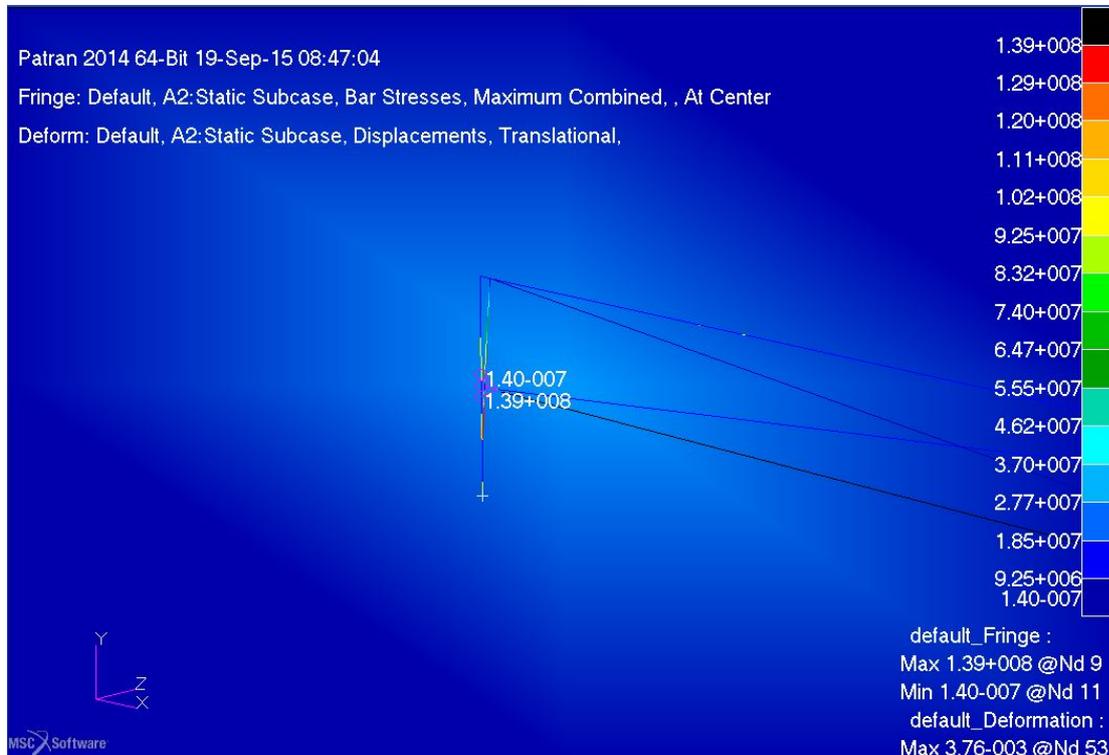


Figura 5.8: Esempio di modello in Patran

Nella letteratura si può trovare che in caso di accelerazioni del braccio inferiori ai $0.5 g$, le forze dinamiche che vengono ad essere esercitate sono dell'ordine del 10-20% di quelle gravitazionali. Pertanto si è scelto di non effettuare una analisi dinamica del braccio, ma di utilizzare dei coefficienti di sicurezza importanti in modo da supplire a questa mancanza. questo ha permesso di ridurre i tempi di progettazione, che si sono rivelati estremamente stringenti.

Il processo è stato poi integrato con la scelta dei motori, partendo dall'estremità più lontana alla base e risalendo. Iterando e di volta in volta scegliendo delle soluzioni commerciali si è arrivati ad una soluzione.

Lo stesso processo è stato effettuato anche per verificare il carico che porterebbe a buckling alcune sezioni. Dopo aver constatato che con i carichi in gioco dal punto di vista teorico si stava molto lontani dal carico critico, si sono verificati i risultati anche con FEM. Si porta ora un esempio di soluzione con Solidworks:

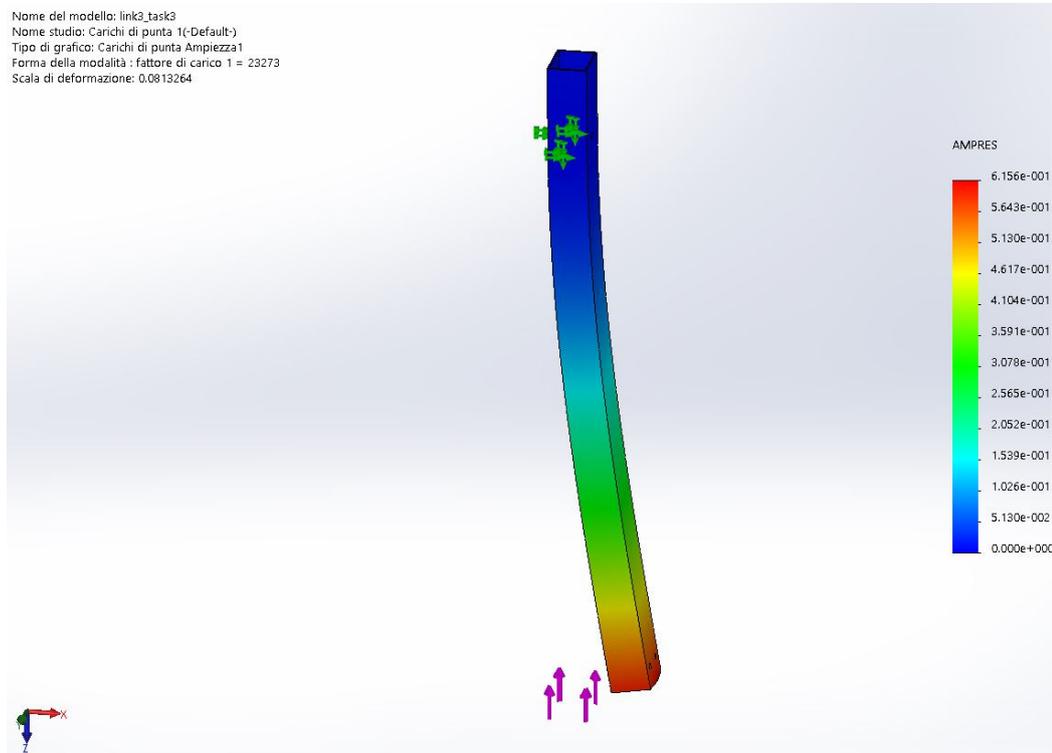


Figura 5.9: Modello Solidworks del carico di punta

5.7 Materiali

I requisiti particolarmente stringenti in fatto di peso hanno subito fatto concentrare la scelta su materiali a peso specifico molto ridotto. In un primo momento si era pensato di utilizzare una struttura in composito (resina epossidica e fibre di carbonio), la quale però presentava costi elevati, tempi di attesa per la produzione lunghi e problematiche relative agli inserti da utilizzare nelle connessioni. Il connubio con il requisito di costo molto contenuto ha fatto propendere quindi per una struttura in alluminio, per la sua economicità e l'alto rapporto resistenza/peso. In un primo momento si è pensato di utilizzare dei profilati di alluminio tipo Bosh Rexroth che avrebbero semplificato notevolmente la costruzione della struttura del braccio robotico. Tale idea è stata successivamente scartata e si è deciso di utilizzare dei semplici profilati a sezione quadrata o rettangolare, che permettono una maggiore facilità di progettazione per le connessioni tra i giunti rispetto a quelli a sezione circolare. Tale scelta è stata effettuata sulla base di un risparmio notevole di peso, a parità di rigidità.

5.8 Scelta dei motori

La scelta dei motori è stata un'azione abbastanza ardua. È risultato necessario, come prima cosa, definire la tipologia di attuatore: se questo sia da scegliere lineare o rotativo. Successivamente un passo importante è stata la decisione di che tipologia di motore sia provvisto questo attuatore, nella fattispecie se venga utilizzato un motore passo-passo, DC oppure DC brushless. Una volta effettuata questa decisione si può passare alla ricerca di una soluzione commerciale che possa rispondere sia da un lato ai requisiti di alimentazione, sia dei requisiti di potenza erogabile. Per capire le scelte possibili entriamo ora in dettaglio:

Motori Passo-Passo I motori passo-passo sono molto utilizzati su robot piccoli e semplici, in quanto non troppo costosi e molto semplici da comandare. Questi motori sono in grado di suddividere il giro in molteplici step, e pertanto sono in grado di funzionare in open-loop. Questo però viene a mancare nel caso in cui la coppia resistente sia troppo alta, e allora il motore "salta il passo". I difetti di tali motori sono principalmente due: generalmente hanno tempo di assestamento abbastanza a lungo, e in più hanno un rapporto peso potenza inferiore rispetto alle altre tipologie di motore elettrico. Se utilizzati in configurazione closed-loop essi si comportano in modo simile ai DC.

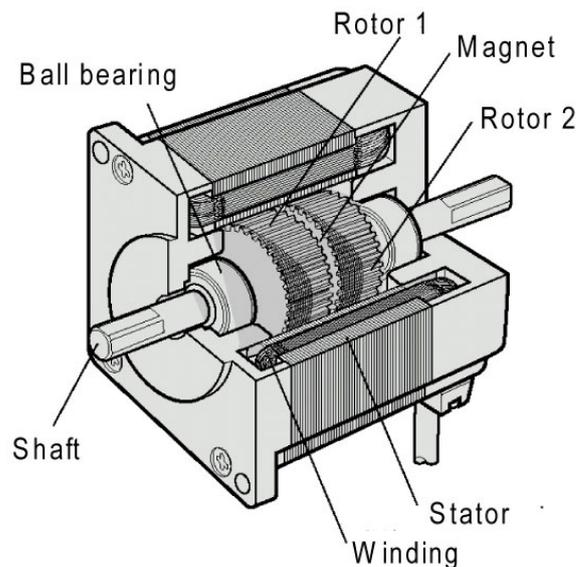


Figura 5.10: Una tipologia di motore passo-passo

Motori DC a magneti permanenti I motori a corrente continua a magneti permanenti sono ampiamente utilizzati. Possono avere magneti in ferrite, i più economici, oppure in neodimio. Gli ultimi generalmente sono la tipologia di motore elettrico con un miglior rapporto potenza peso. Tipicamente gli avvolgimenti di questi motori sono in rame. Vantaggi di questi motori sono: bassa induttanza, basso attrito e nessuna coppia di *cogging*. Svantaggi importanti dei motori DC sono: bassa capacità termica a causa della loro bassa massa e poche vie per asportare calore, attrito tra spazzole e commutatore, scintille, spazzole che si usurano.

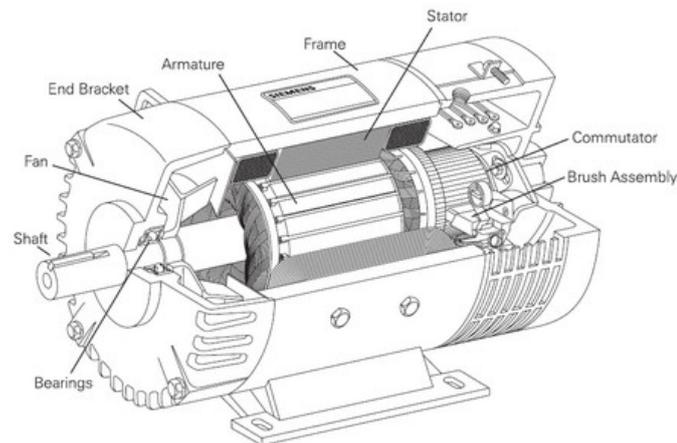


Figura 5.11: Motore DC a magneti permanenti

DC Brushless Questi motori nascono per rispondere alle problematiche di cui sopra: non vi sono contatti, spazzole o commutatore. Data la maggiore semplicità dal punto di vista costruttivo, questi motori hanno delle buone performances ad un costo ridotto. Questo, però, si paga con un maggiore complessità del controllo e del driver. I DC brushless, infatti, si comportano come dei motori AC e pertanto hanno bisogno di circuiti di commutazione.

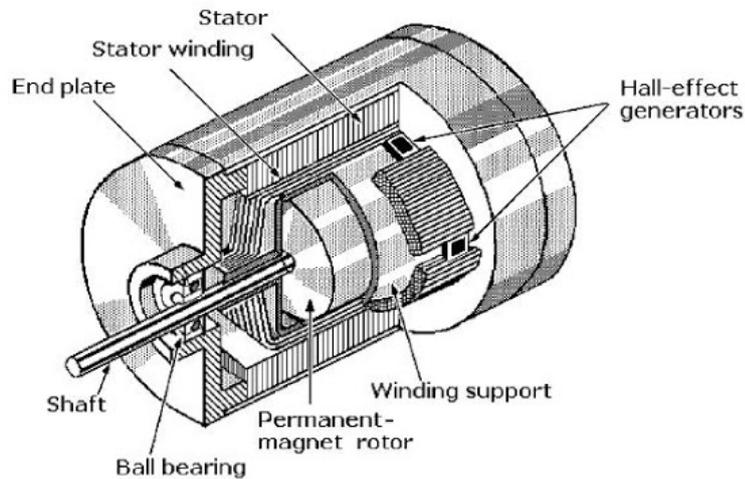


Figura 5.12: Motore DC Brushless

I parametri fondamentali da tenere in conto quando si sceglie un motore sono i seguenti:

- Alto rapporto potenza/peso
- Massa contenuta
- Bassa inerzia
- Rendimento
- Precisione
- Sensori incorporati
- Riduttori efficienti
- coppia erogabile dal riduttore senza che si rompa.

Il passo successivo è stato scegliere la tipologia di attuatore: se lineare o rotativo. Risulta palese come questa scelta influenzi pesantemente il design di tutto il braccio, e come sia di importanza fondamentale. Si ha ora una descrizione dei motori scelti partendo dalla base fino all'end-effector.

Il primo motore, in quanto non deve fornire nessuna coppia che contrasti l'azione della forza peso, ma debba unicamente contrastare le coppie in generate dalla componente inerziale, è stato scelto rotativo. In vista di successivi cambiamenti ed evoluzioni del

braccio robotico nei prossimi anni, e della necessità di autorità notevole per evitare oscillazioni, si è deciso di sovradimensionare questo attuatore; la scelta è ricaduta sul modello PS150 a 12V (fig. 5.13). Questo motore può erogare una coppia nominale di 1,4 Nm, ad una velocità di 19 rpm in condizioni di carico. Il fabbisogno di corrente in condizioni di carico nominale è di 950 mA, che si riducono a meno di 200 a vuoto. A questo motoriduttore è stato ulteriormente applicato un sistema di riduzione del numero di giri con una coppia conica con rapporto di trasmissione 4:1. Dato il basso rapporto di riduzione l'efficienza di quest'ultima è molto elevata (0,98-0,99).



Figura 5.13: Motore PS150-12-196

Tenendo in considerazione le coppie erogabili da motoriduttori rotativi commerciali ed i loro rispettivi prezzi, si è giunti alla conclusione che gli attuatori lineari fossero la scelta migliore per i motori 2 e 3 alla base. Il vantaggio principale di questa tipologia di attuatori è la possibilità di bloccare la posizione senza che sia necessaria una alimentazione; il meccanismo di trasmissione del moto per gli attuatori scelti risulta non reversibile. Dopo attente riflessioni, e avendo notato che risulta esserci un vuoto produttivo per un range di forze prodotte da attuatori lineari, si sono scelti per entrambi i giunti degli attuatori della Firgelli, e nello specifico il modello L16-140-150-12-P (fig. 5.14).



Figura 5.14: Attuatore lineare L16-140-150-12-P

Per quanto riguarda, invece, i motori scelti per il polso la scelta è ricaduta nuovamente su dei motori prodotti dalla Micro Motors s.r.l. Questa azienda offre ad un prezzo competitivo dei piccoli motori completi di riduttore e resolver che riescono a generare delle copie considerevoli con dimensioni contenute e pesi ridotti.



Figura 5.15: Motoriduttore BS138F-12-208

come attuatore finale per la movimentazione delle pinze si è scelto di utilizzare lo stesso attuatore usato per l'end-effector della task 1/4. La scelta è stata effettuata sulla base di considerazioni sulla possibilità di intercambiare i due attuatori in caso di failure di uno dei due. Questa soluzione, assieme alle precedenti di utilizzare per più gradi di libertà lo stesso modello di motore, ha permesso di abbattere i costi per le parti di ricambio, talvolta nemmeno necessarie, come in questo ultimo caso. L'attuatore finale è stato trovato nel Firgelli L12-30-50-12-I:



Figura 5.16: Attuatore lineare L12-30-50-12-I

Riassumendo tutti i motori acquistati, compresi di costo, peso e coppia/forza erogata possiamo scrivere:

Modello	Prezzo[€]	Peso[g]	Forza[N] or Coppia[Nm]	Task	Funzione
PS150-12-196-2S	130	200	1,4 Nm	1-3-4	Motore alla base
L16-140-150-12-P	100	84	200 N	1-3-4	Attuatori 2 e 3
L12-30-50-12-I	100	34	50 N	1-3-4	Attuatore end-effector
BS138F-12-208-2S	50	90	0,2 Nm	3	Wrist Motors(x3)

Tabella 5.2: Elenco motori

5.9 Driver Motori

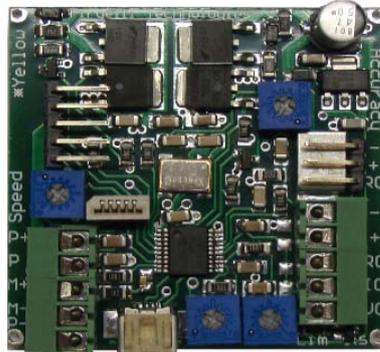


Figura 5.17: Driver LAC

I driver per il controllo dei motori dipendono dalla tipologia di attuatore utilizzato. Nel caso degli attuatori della Firgelli, è disponibile un controllore dedicato di costo non eccessivo se venduto assieme ad essi. Esso si può vedere in figura 5.17.

Per quanto riguarda il controllo degli altri motori, questo è stato progettato ad hoc da membri del team, poiché driver e controllori per motori commerciali sarebbero risultati troppo ingombranti e avrebbero pesato troppo sul budget.

5.10 Design definitivo

Il design definitivo del braccio è stato quindi pensato in relazione a tutti i paragrafi di cui sopra e si è giunti a queste conclusioni:

1. braccio con end-effector uno: cinematismo molto semplice, lunghezze dei giunti minimizzate, robustezza per resistere durante prove che possono danneggiare lo strumento.



Figura 5.18: Braccio con end effector 1

2. braccio con end-effector due: cinematismo più complicato, lunghezze dei giunti in modo da raggiungere lo spazio di lavoro richiesto, flessibilità e protezione dalle intemperie delle parti che andranno a fare misure di tensione.

L'end-effector che stia nella configurazione per la task 1-4, assieme alla base del braccio, assomiglia a quella di uno scavatore. In questo modo, durante le operazioni di scavo, si è aiutati dalla forza di gravità per fornire più carico; inoltre il fatto di scavare verso il rover evita problematiche come il sollevarsi delle ruote minimizza la possibilità di incastrarsi e rischiare così di rompere il braccio. Si presentano dei particolari del primo end-effector in figura 5.19 :

Per quanto riguarda il secondo end-effector, a causa delle ristrette risorse economiche, si è verificata la disponibilità di soluzioni commerciali e si è deciso di scartare la possibilità di inserire un polso a 3 gradi di libertà immediatamente con la prima iterazione del braccio. Questo è dovuto alla difficoltà di trovare dei giunti asso-radiali di dimensioni, peso e costi contenuti. Si è deciso quindi di fornire i gradi di libertà in maniera sequenziale, mettendo al primo posto, partendo dalla base del braccio, il grado di libertà con asse di rotazione parallelo alla congiungente tra il link precedente e quello successivo. In questo modo si è portato più vicino alla base il peso del giunto, diminuendo il braccio della forza, e quindi così le necessità di coppie alla base. L'attuatore scelto per muovere le pinze è lo stesso dell'end-effector precedente: in questo modo si è potuto risparmiare su pezzi di ricambio e uniformare le richieste dal punto di vista elettronico e facilitare il lavoro al team. La connessione del sensore di tensione viene inserita all'interno della



(a) Vista laterale



(b) Vista frontale a tre-quarti



(c) Vista posteriore a tre-quarti

Figura 5.19: Particolari del primo end-effector



Figura 5.20: Braccio con end effector 2

parte finale in quanto la competizione si sarebbe potuta svolgere anche in caso di pioggia lieve. Le conseguenze di una mancata protezione in questo caso sono ovvie.

Come per il caso precedente si presentano ora dei particolari dell'end-effector in figura 5.21:



(a)

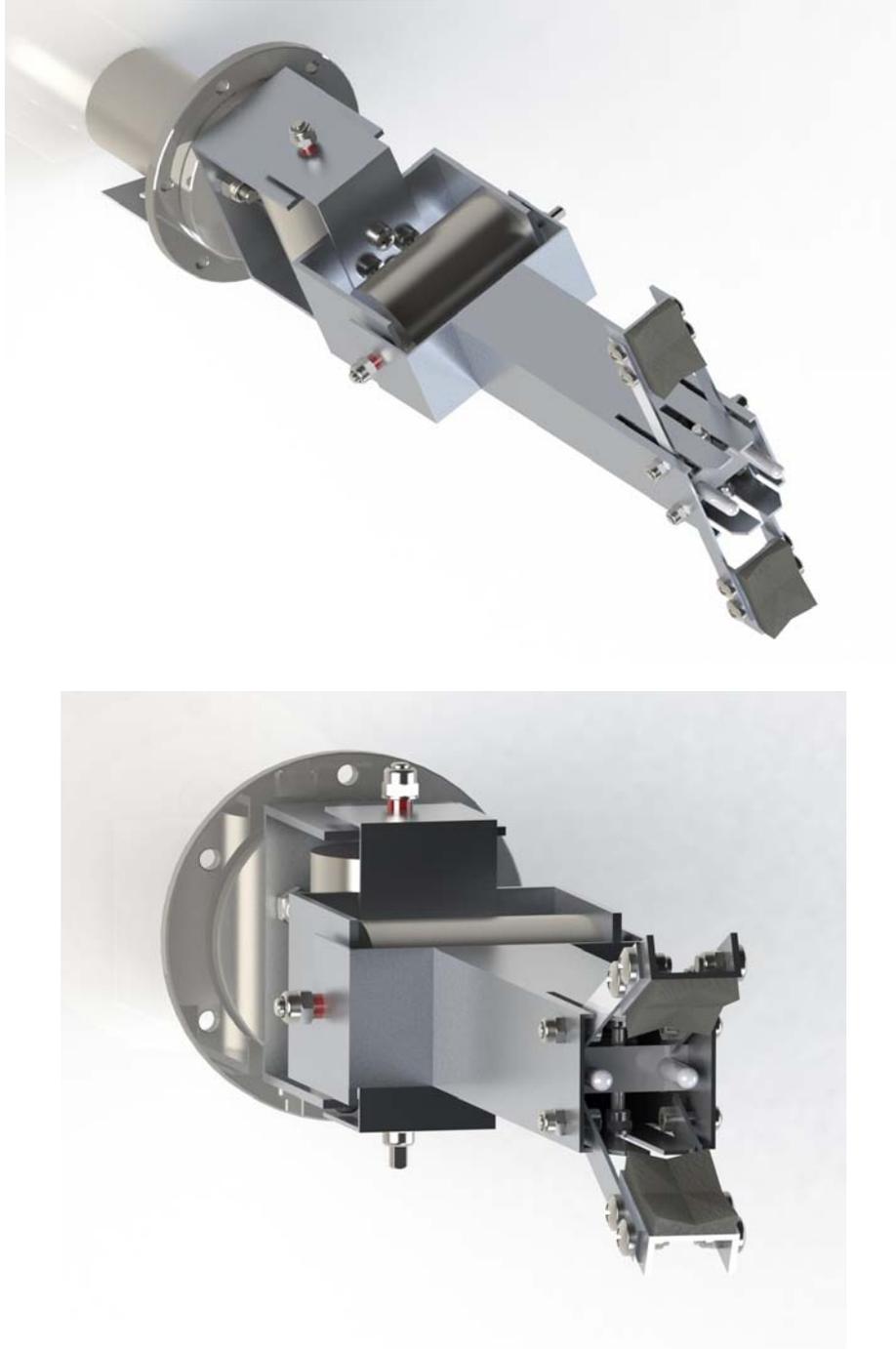


Figura 5.21: Particolari del secondo end-effector

CAPITOLO 6

Cinematica del Braccio

Poiché l'analisi del controllo del braccio robotico progettato in precedenza è stata effettuata da altri membri del gruppo MORPHEUS si è pensato di aggiungere un ulteriore grado di libertà al braccio. questa scelta è stata effettuata in modo da poter sviluppare un'evoluzione del braccio attuale. La aggiunta di una ulteriore coppia cinematica rende il braccio ridondante, e questo oltre a permettere configurazioni che possano evitare gli ostacoli, può essere corredato con un controllo che vada a minimizzare la potenza richiesta, ottimizzare i consumi. D'altro canto il fatto che il braccio sia ridondante, complica maggiormente il problema della cinematica inversa. La soluzione del problema non è più univoca (o perlomeno in numero molto limitato), e per alcune posizioni obiettivo le configurazioni possono essere potenzialmente infinite. Il braccio robotico viene modificato come da modello seguente.



Figura 6.1: Render del braccio modificato

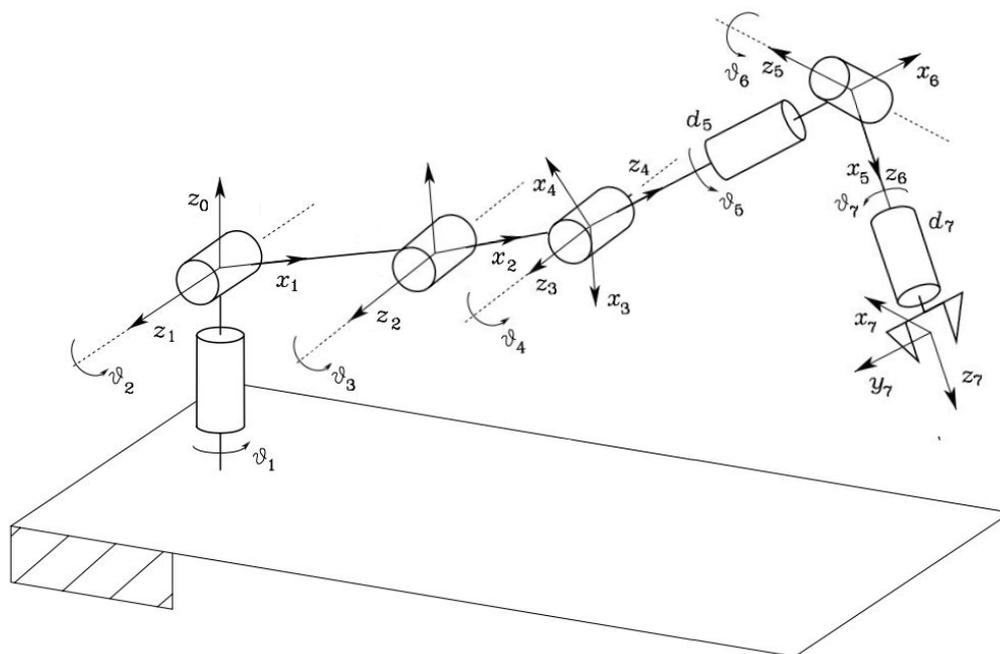


Figura 6.2: Modello del braccio modificato con l'aggiunta di un G.d.L.

i valori che sono stati definiti per le lunghezze dei giunti sono:

$$\begin{aligned}
 l_1 &= 0.45 \text{ m} & l_2 &= 0.463 \text{ m} & l_3 &= 0.4 \text{ m} \\
 d_5 &= 0.293 \text{ m} & d_7 &= 0.1 \text{ m}
 \end{aligned}$$

6.1 Cinematica diretta

Sfruttando la convenzione di Denavit-Hartenberg, definita nel capitolo 2.1 si può ottenere la seguente tabella con i coefficienti DH:

Giunto	a_{i-1}	α_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	l_1	$\frac{\pi}{2}$	0	θ_2
3	l_2	0	0	θ_3
4	l_3	0	0	θ_4
5	0	$\frac{\pi}{2}$	d_5	θ_5
6	0	$-\frac{\pi}{2}$	0	θ_6
7	0	$\frac{\pi}{2}$	d_7	θ_7

Tabella 6.1: Parametri di Denavit-Hartenberg

Le singole matrici di trasformazione tra un sistema di riferimento e il successivo sono

quindi:

$$\begin{aligned}
 M_{0,1} &= \begin{pmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{1,2} &= \begin{pmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & l_1 \\ 0 & 0 & -1 & 0 \\ \sin(\theta_2) & \cos(\theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{2,3} &= \begin{pmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & l_2 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{3,4} &= \begin{pmatrix} \cos(\theta_4 + \frac{\pi}{2}) & -\sin(\theta_4 + \frac{\pi}{2}) & 0 & l_3 \\ \sin(\theta_4 + \frac{\pi}{2}) & \cos(\theta_4 + \frac{\pi}{2}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{4,5} &= \begin{pmatrix} \cos(\theta_5) & -\sin(\theta_5) & 0 & 0 \\ 0 & 0 & -1 & d_5 \\ \sin(\theta_5) & \cos(\theta_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{5,6} &= \begin{pmatrix} \cos(\theta_6) & -\sin(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_6) & -\cos(\theta_6) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 M_{6,7} &= \begin{pmatrix} \cos(\theta_7) & -\sin(\theta_7) & 0 & 0 \\ 0 & 0 & -1 & d_7 \\ \sin(\theta_7) & \cos(\theta_7) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
 \end{aligned}$$

Facendo la produttoria tra tutte le matrici in serie si può ottenere la matrice di trasferimento totale del sistema¹:

¹Le componenti della matrice $M_{0,7}$ si possono trovare in appendice A.1.

$$M_{0,7} = \prod_{k=1}^{k=7} M_{k-1,k} = \begin{pmatrix} M_{1,1} & M_{1,2} & M_{1,3} & M_{1,4} \\ M_{2,1} & M_{2,2} & M_{2,3} & M_{2,4} \\ M_{3,1} & M_{3,2} & M_{3,3} & M_{3,4} \\ 0 & 0 & 0 & 2 \end{pmatrix} \quad (6.1)$$

6.2 Cinematica inversa

Come si può vedere dalla figura 6.2, il braccio in questa configurazione è provvisto di un polso a 3 gradi di libertà. si può notare come gli assi delle ultime coppie cinematiche, secondo la convenzione di Denavit-Hartenberg, si incontrino tutti in uno stesso punto. Questa condizione è stata molto studiata in passato e permette di sfruttare un algoritmo per la soluzione ideato da Pieper. Questo algoritmo che viene utilizzato generalmente per bracci a 6 G.d.L., è stato modificato in modo da permettere di trovare tutte le configurazioni possibili per il braccio in esame. La particolare disposizione delle coppie cinematiche, in questo caso tutte rotoidali, e nello specifico con le prime tre ad asse di rotazione parallelo, permette di dividere il problema in due più piccoli.

6.2.1 Risoluzione "alla Pieper"

Pieper ha sviluppato una soluzione per risolvere con un polinomio del quarto grado un braccio a 6 G.d.L. data la disposizione delle terne del polso, la soluzione viene ad essere disgiunta in quanto la direzione scelta per l'avvicinamento all'obiettivo viene assolta unicamente dal polso. Chiamato W il centro del polso, dove si incontrano i tre assi, e definita la posizione da raggiungere con l'end-effector \vec{p}_e e il suo orientamento $R_e = [\vec{n}_e, \vec{s}_e, \vec{a}_e]$, la posizione di W può essere trovata come:

$$\vec{p}_w = \vec{p}_e - d_7 \vec{a}_e \quad (6.2)$$

In questo caso, ora rimane solo da risolvere un braccio a 4 G.d.L., che per la particolare scelta della disposizione delle coppie cinematiche, è piano. Per risolvere la cinematica inversa secondo Pieper si procede poi come segue:

1. si trova la posizione del polso come in (6.2).
2. si risolve la cinematica inversa per (q_1, q_2, q_3) .
3. si calcola la matrice di rotazione $R_3^0(q_1, q_2, q_3)$.
4. si ricava la matrice $R_6^3(\theta_4, \theta_5, \theta_6) = R_3^{0T} R$, con R matrice di rotazione totale.

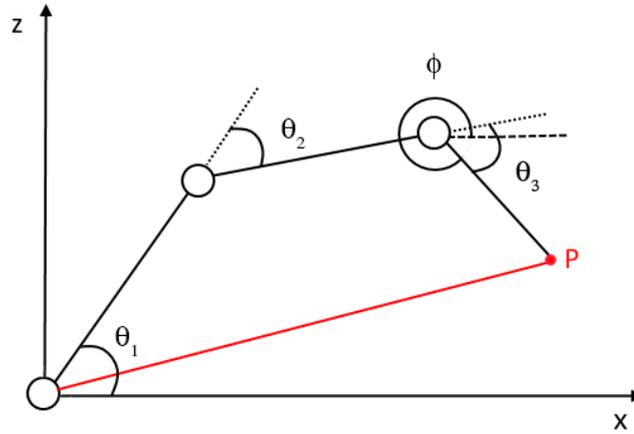


Figura 6.3: Modello del cinematismo a 4 G.d.L.

5. si risolve la cinematica inversa per l'orientamento $(\theta_4, \theta_5, \theta_6)$.

A differenza di quello che viene effettuato nella lista qui sopra indicata, si dovrà risolvere una cinematica inversa per un braccio ridondante a 4 G.d.L. La semplicità nella disposizione delle coppie cinematiche ci è di aiuto: si ottiene infatti un braccio piano 3R, già studiato molteplici volte[11].

Le equazioni della cinematica diretta per il braccio piano in figura 6.3 sono, mantenendo la notazione del braccio totale:

$$p_x = l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3) + l_4 \cos(\theta_2 + \theta_3 + \theta_4) \quad (6.3)$$

$$p_z = l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3) + l_4 \sin(\theta_2 + \theta_3 + \theta_4) \quad (6.4)$$

$$\phi = \theta_2 + \theta_3 + \theta_4 \quad (6.5)$$

Si voglia ora invertire il sistema descritto. Sostituendo l'equazione (6.5) all'interno delle altre due, e raggruppando i termini noti si ottiene:

$$p_x - l_4 \cos(\phi) = l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3) \quad (6.6)$$

$$p_z - l_4 \sin(\phi) = l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3) \quad (6.7)$$

Si rinomina per facilità di lettura $p'_x = p_x - l_4 \cos(\phi)$ e $p'_z = p_z - l_4 \sin(\phi)$.

Elevando al quadrato entrambe le equazioni nel modo indicato e sommandole:

$$\begin{aligned} [p'_x - l_2 \cos(\theta_2)]^2 &= [l_3 \cos(\theta_2 + \theta_3)]^2 \\ [p'_z - l_2 \sin(\theta_2)]^2 &= [l_3 \sin(\theta_2 + \theta_3)]^2 \end{aligned}$$

si riesce a giungere ad un'unica equazione non lineare in funzione di θ_2 .

$$(-2l_2 p'_x) \cos(\theta_2) + (-2l_2 p'_z) \sin(\theta_2) + (p_x'^2 + p_z'^2 + l_2^2 - l_3^2) = 0 \quad (6.8)$$

Si può notare che l'equazione assume la forma

$$P \cos \alpha + Q \sin \alpha + R = 0 \quad (6.9)$$

La soluzione è facilmente trovabile per sostituzione come mostrato in appendice A.2. Come risultato si ottiene:

$$\theta_2 = \gamma \pm \arccos \left(\frac{-(p_x'^2 + p_z'^2 + l_2^2 - l_3^2)}{2l_2 \sqrt{p_x'^2 + p_z'^2}} \right) \quad (6.10)$$

con γ :

$$\gamma = \text{atan2} \left(\frac{-p'_z}{\sqrt{p_x'^2 + p_z'^2}}, \frac{-p'_x}{\sqrt{p_x'^2 + p_z'^2}} \right) \quad (6.11)$$

si tenga nota del fatto che ci sono due soluzioni per θ_1 . sostituendo una di queste soluzioni nelle equazioni (6.6) e (6.7) si ottiene:

$$\begin{aligned} \cos(\theta_2 + \theta_3) &= \frac{p'_x - l_2 \cos(\theta_2)}{l_3} \\ \sin(\theta_2 + \theta_3) &= \frac{p'_z - l_2 \sin(\theta_2)}{l_3} \end{aligned}$$

E utilizzando nuovamente la atan2 otteniamo tutto in funzione di θ_2

$$\theta_3 = \text{atan2} \left(\frac{p'_z - l_2 \sin(\theta_2)}{l_3}, \frac{p'_x - l_2 \cos(\theta_2)}{l_3} \right) - \theta_2 \quad (6.12)$$

per cui per ogni soluzione di θ_2 , si ha una ed una sola soluzione per θ_3 (fermo restando il fatto di definire la variabile ϕ). Infine θ_4 si può ricavare dalla terza equazione iniziale:

$$\theta_4 = \phi - (\theta_2 + \theta_3) \quad (6.13)$$

in conclusione si può affermare che questi tre giunti analizzati possono essere fatti dipendere da una unica variabile ϕ . Questo semplifica di molto il problema, in quanto permette ottenere immediatamente gli altri parametri in funzione di uno solo.

Con questa metodologia si può verificare facilmente quali siano l'angolo minimo e massimo del primo e del terzo giunto affinché si possa definire un range. Le possibilità studiate per la soluzione di questa tipologia di problema sono molteplici: alcune semplicemente fissano uno dei gradi di libertà, come la trattazione precedente, ai quali fanno seguire gli altri; altre si basano sulla minimizzazione di una funzione costo del tipo:

$$g(\dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{W} \dot{\mathbf{q}} \quad (6.14)$$

che viene risolta con il metodo dei moltiplicatori di Lagrange. W è una matrice di pesi per i vari giunti. La strada che invece si va a seguire ora in questa trattazione per ottimizzare i risultati è quella dei link virtuali.

6.2.2 Ottimizzazione della soluzione per la cinematica inversa

Come abbiamo visto qua sopra, poiché la catena cinematica in esame è ridondante si possono ottenere senza considerazioni aggiuntive, unicamente due equazioni, ma in tre incognite. Pertanto si cerca una metodologia per risolvere il problema. la scelta è ricaduta sulla trattazione sviluppata da *Chung et al.* [17] per motivi di semplicità, ovvero sull'utilizzo della teoria dei link virtuali. Il metodo di ottimizzazione utilizzato si basa sulla trattazione con lo Jacobiano esteso, che sfrutta la ortogonalità tra un vettore gradiente ∇C di una funzione $C(\theta)$ che analizza le prestazioni, e il vettore Kernel (nucleo) \mathbf{n} dello Jacobiano del manipolatore (in questo caso si tratta solo del meccanismo piano). Per l'ortogonalità si può scrivere:

$$\mathbf{n} \cdot \nabla C = 0 \quad (6.15)$$

lo Jacobiano può essere facilmente ottenuto, e il nucleo di \mathbf{J} può essere ottenuto come segue:

$$\mathbf{n} = \begin{bmatrix} l_3 l_4 \sin(\theta_4) \\ -(l_3 l_4 \sin(\theta_4) + l_2 l_4 \sin(\theta_3 + \theta_4)) \\ l_2 l_3 \sin(\theta_3) + l_2 l_4 \sin(\theta_3 + \theta_4) \end{bmatrix} \quad (6.16)$$

sfruttando l'indice di prestazioni definito in [17], si può ottenere la equazione aggiuntiva come:

$$-h_2(l_3l_4 \sin(\theta_4) + l_2l_4 \sin(\theta_3 + \theta_4)) + h_3(l_2l_3 \sin(\theta_3) + l_2l_4 \sin(\theta_3 + \theta_4)) \quad (6.17)$$

con:

$$\begin{aligned} h_2 &= -l_3l_4[l_2l_4 \cos(\theta_3 + \theta_4)(l_2l_3 \sin(\theta_3) + l_2l_4 \sin(\theta_3 + \theta_4)) \\ &\quad + (l_3l_4 \sin(\theta_4) + l_2l_4 \sin(\theta_3 + \theta_4))(l_2l_3 \cos(\theta_3) + l_2l_4 \cos(\theta_3 + \theta_4))] \\ h_3 &= -l_3l_4 \cos(\theta_4)(l_3l_4 \sin(\theta_4) + l_2l_4 \sin(\theta_3 + \theta_4))(l_2l_3 \sin(\theta_3) + l_2l_4 \sin(\theta_3 + \theta_4)) \\ &\quad - l_3l_4 \sin(\theta_4)(l_3l_4 \sin(\theta_4) + l_2l_4 \cos(\theta_3 + \theta_4))(l_2l_3 \sin(\theta_3) + l_2l_4 \sin(\theta_3 + \theta_4)) \\ &\quad - l_3l_4(l_3l_4 \sin(\theta_4) + l_2l_4 \sin(\theta_3 + \theta_4))l_2l_4 \cos(\theta_3 + \cos(\theta_4)) \end{aligned}$$

Mettendo ora a sistema le equazioni (6.3), (6.4) e (6.17) si può risolvere ora il problema con un metodo di risoluzione non lineare.

6.2.3 Completamento della cinematica inversa

Per quanto riguarda θ_1 la soluzione è molto semplice. Essendo l'unico giunto che permette un moto fuori piano dalla base al polso, si può facilmente trovare:

$$\theta_1 = \text{atan2}(p_y, p_x) \quad (6.18)$$

Come si è visto in precedenza si può ricavare la matrice di rotazione $R_7^4(\theta_5, \theta_6, \theta_7)$ in funzione della rotazione totale R e della matrice di rotazione che abbiamo ottenuto con la soluzione del problema precedente.

$$R_7^4(\theta_5, \theta_6, \theta_7) = R_4^0 R \quad (6.19)$$

Il comportamento del polso, dal punto di vista delle rotazioni è paragonabile a una rotazione con angoli di Eulero 3-2-3 rispetto alla terna di riferimento 4. Sia quindi la matrice di rotazione:

$$R_7^4 = \begin{bmatrix} n_x^4 & s_x^4 & a_x^4 \\ n_y^4 & s_y^4 & a_y^4 \\ n_z^4 & s_z^4 & a_z^4 \end{bmatrix}$$

si ricorda che la matrice di rotazione di Eulero per una rotazione ZYZ è:

$$R_7^4 = \begin{bmatrix} c\theta_5 c\theta_6 c\theta_7 - s\theta_5 s\theta_7 & -c\theta_5 c\theta_6 s\theta_7 - s\theta_5 c\theta_7 & c\theta_5 s\theta_6 & c\theta_5 s\theta_6 d_7 \\ s\theta_5 c\theta_6 c\theta_7 + c\theta_5 s\theta_7 & -s\theta_5 c\theta_6 s\theta_7 + c\theta_5 c\theta_7 & s\theta_5 s\theta_6 & s\theta_5 s\theta_6 d_7 \\ -s\theta_6 c\theta_7 & s\theta_6 s\theta_7 & c\theta_6 & c\theta_6 d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.20)$$

si hanno ora due casi: per $\theta_6 \in (0, \pi)$

$$\begin{aligned} \theta_5 &= \text{atan2}(a_y^4, a_x^4) \\ \theta_6 &= \text{atan2}\left(\sqrt{(a_x^4)^2 + (a_y^4)^2}, a_z^4\right) \\ \theta_7 &= \text{atan2}(s_z^4, -n_z^4) \end{aligned} \quad (6.21)$$

e per $\theta_6 \in (-\pi, 0)$

$$\begin{aligned} \theta_5 &= \text{atan2}(-a_y^4, -a_x^4) \\ \theta_6 &= \text{atan2}\left(-\sqrt{(a_x^4)^2 + (a_y^4)^2}, a_z^4\right) \\ \theta_7 &= \text{atan2}(-s_z^4, n_z^4) \end{aligned} \quad (6.22)$$

Si sono definiti quindi ora tutti gli angoli che corrispondono alla posizione iniziale e finale. Rimane da considerare la pianificazione di traiettoria che può essere effettuata, ad esempio, come espresso nel paragrafo 2.7.

6.3 Esempi di risoluzione della cinematica inversa

Si presentano ora dei risultati ottenuti con MATLAB. Il programma è stato scritto nell'ottica di verificare che il braccio sia in grado di raggiungere dal punto di vista della programmazione tutte le configurazioni possibili all'interno dello spazio destro (dall'inglese *dexterous*, ovvero agile). Questo è stato ottenuto con un programma che, in funzione delle posizioni iniziale e finale, dia come output gli angoli dei giunti, e l'errore relativo sia al posizionamento che all'orientamento dell'end-effector, e successivamente fornisca anche i valori di θ_i , $\dot{\theta}_i$ e $\ddot{\theta}_i$ in funzione del tempo. Questi sono stati trovati mediante l'utilizzo del polinomio del quinto ordine, come descritto in precedenza.

La posizione finale è sempre presa in modo casuale, e la posizione iniziale è sempre presa con tutti gli angoli nulli. Il caso preso in esame ora ha come dati obiettivo in ingresso:

- $P_x = 0.9100$
- $P_y = -0.7110$
- $P_x = -0.0521$

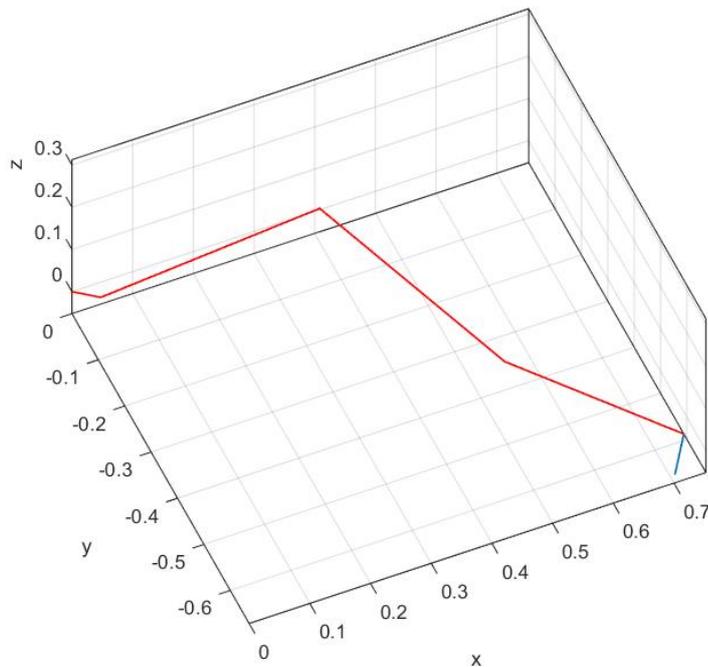
La configurazione finale del braccio è rappresentata in figura 6.4.

Gli angoli ottenuti con la cinematica inversa, sono racchiusi in tabella:

$\theta_{j(i)}$	Valore	$\theta_{j(f)}$	Valore
$\theta_{1(i)}$	0	$\theta_{1(f)}$	-0.6632
$\theta_{2(i)}$	0	$\theta_{2(f)}$	0.7409
$\theta_{3(i)}$	0	$\theta_{3(f)}$	-1.5046
$\theta_{4(i)}$	0	$\theta_{4(f)}$	0.4588
$\theta_{5(i)}$	0	$\theta_{5(f)}$	-1.3311
$\theta_{6(i)}$	0	$\theta_{6(f)}$	1.5246
$\theta_{7(i)}$	0	$\theta_{7(f)}$	2.5703

Tabella 6.2: Valori degli angoli

una volta trovati gli angoli iniziali e finali, con la pianificazione di traiettoria si è ottenuto i seguenti risultati:



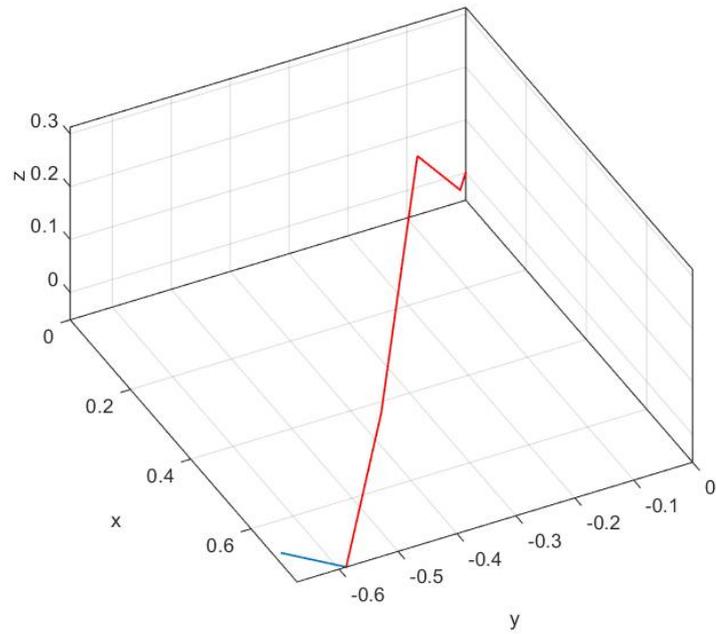
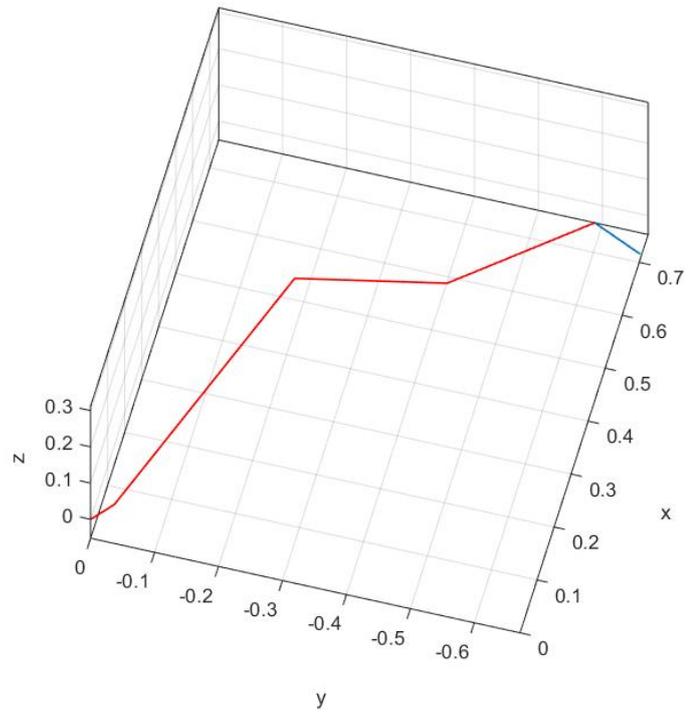
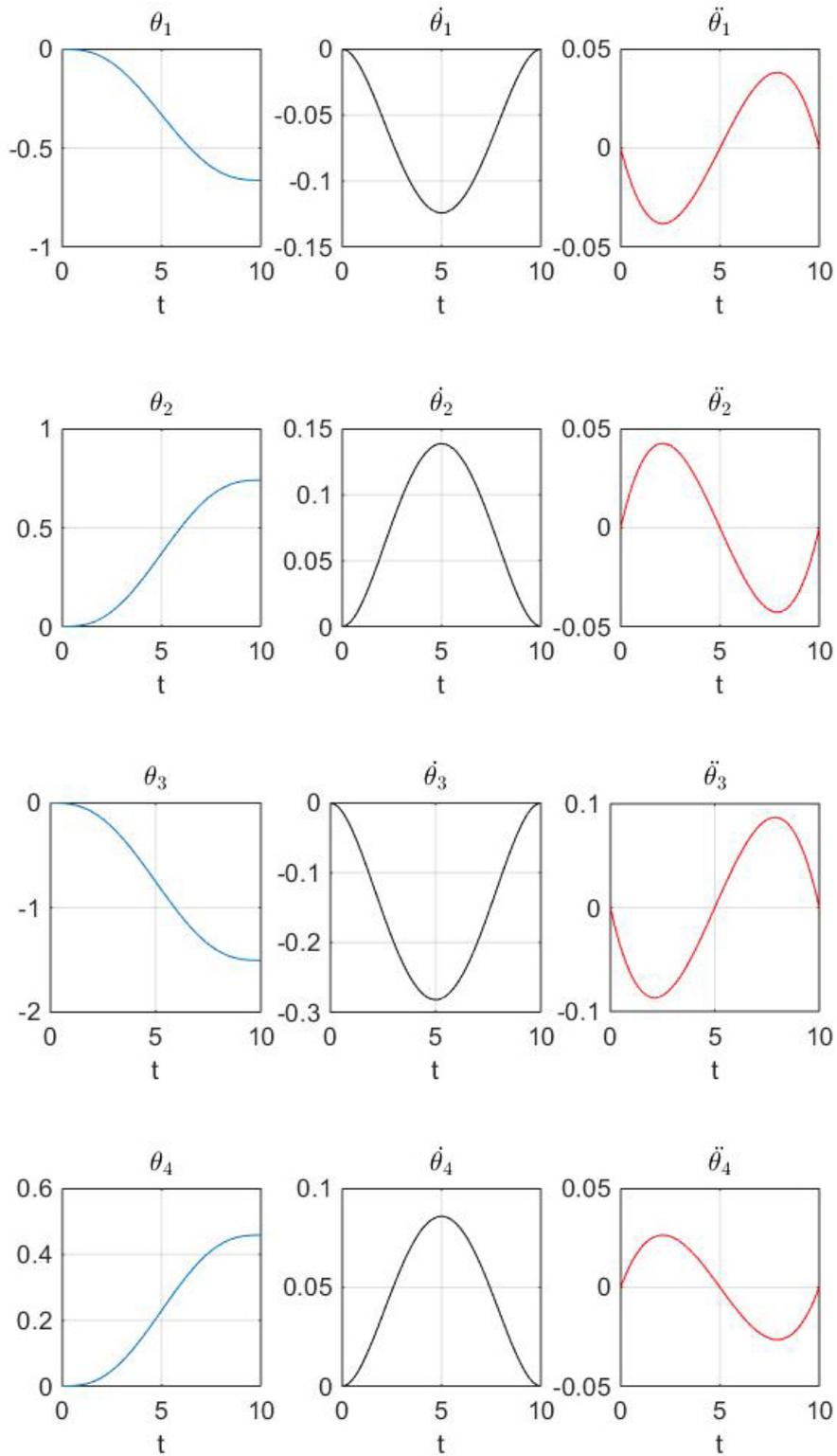
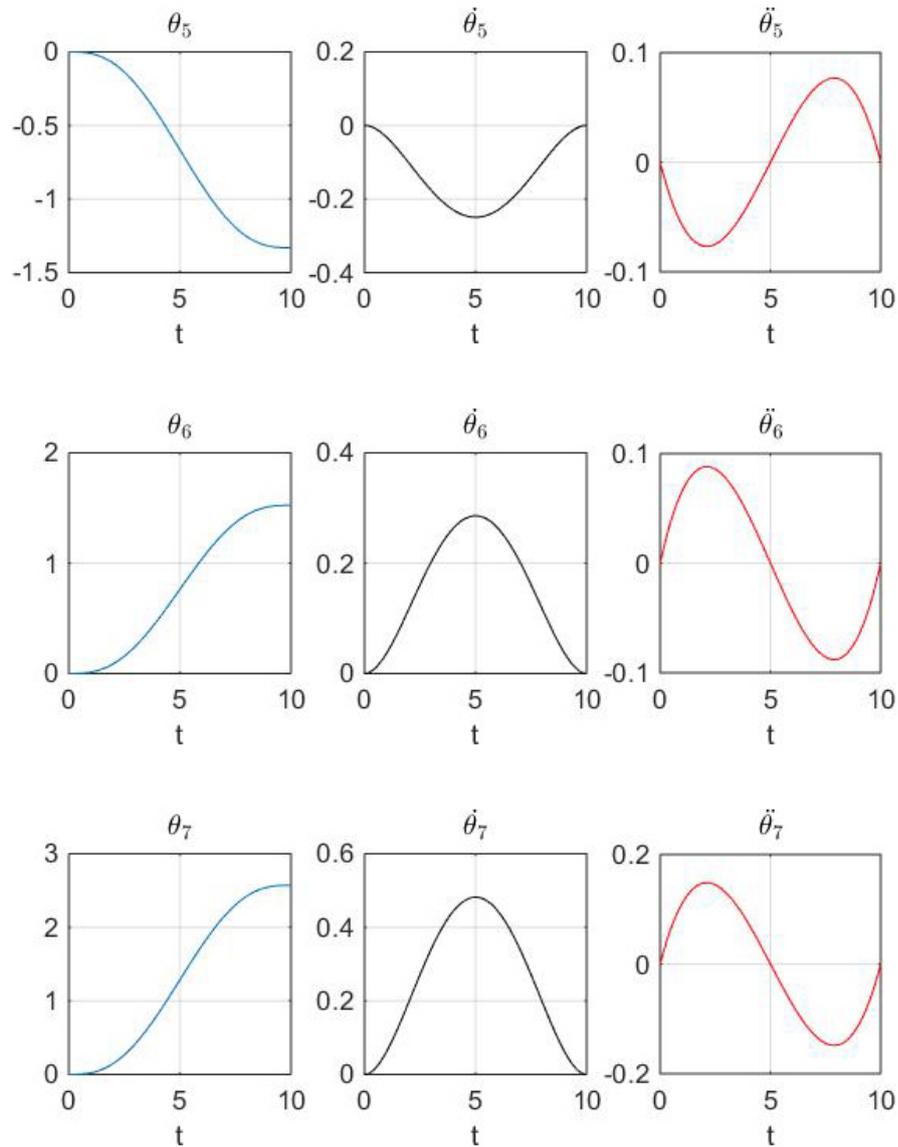


Figura 6.4: Configurazione del braccio




 Figura 6.5: Progressione dei θ_i , $\dot{\theta}_i$ e $\ddot{\theta}_i$

Inserendo un ciclo che vada a calcolare più volte i risultati per posizioni diverse di arrivo, mantenendo però come dati iniziali i valori nulli precedenti si ottiene²:

²Si tenga conto del fatto che questo è riferito unicamente al posizionamento del centro del polso.

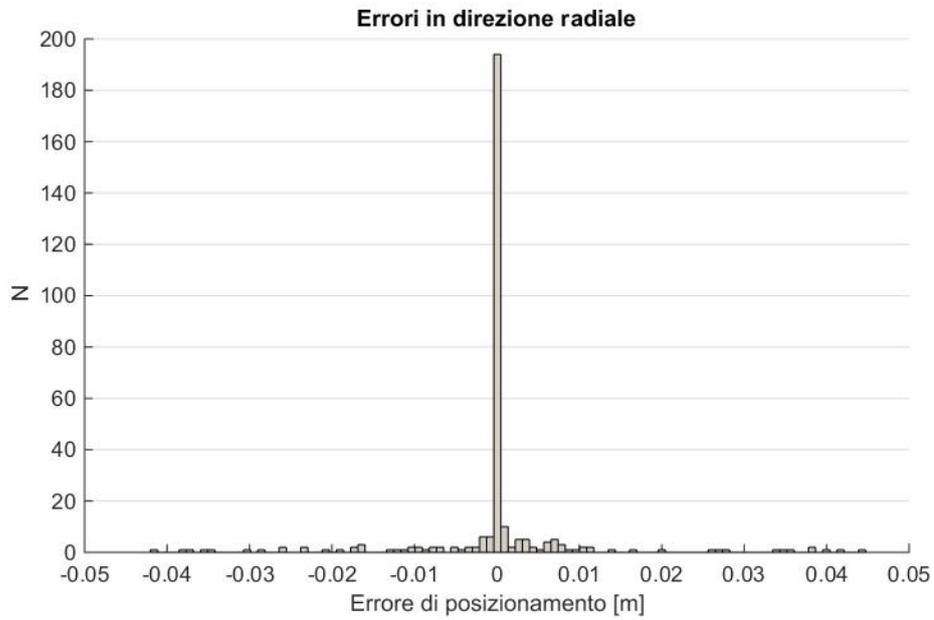


Figura 6.6: Errori in direzione radiale

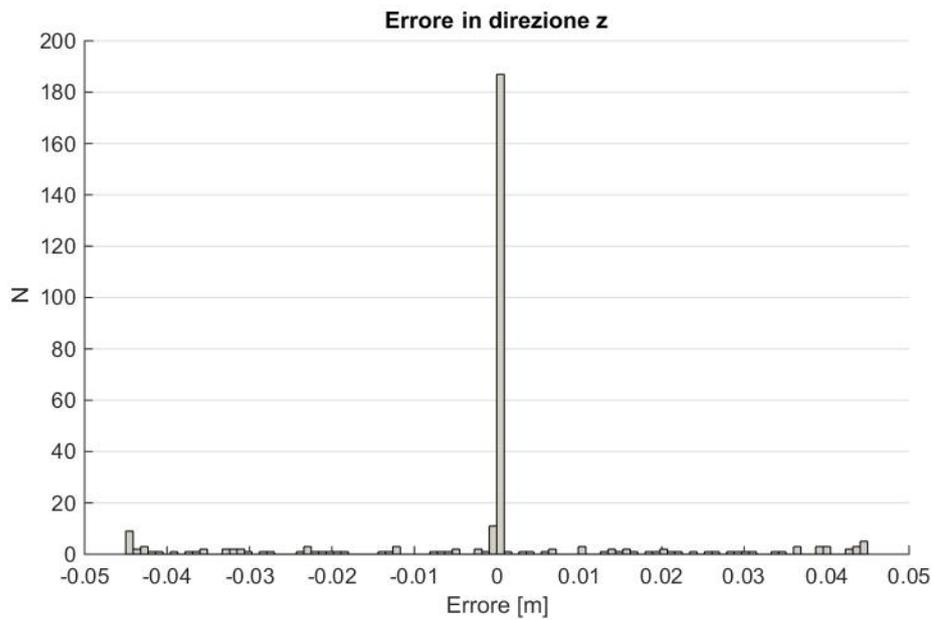


Figura 6.7: Errori lungo z

Come si può notare dalle figure precedenti, gli errori di posizionamento non sono particolarmente elevati nella gran parte dei casi, e risultano tali generalmente per errori di carattere numerico dovuti alle funzioni trigonometriche. Inoltre non ci sono stati problemi per singolarità, e si è giunti a soluzione, seppur con degli errori, in tutte le iterazioni del programma. La metodologia descritta, infine, è sufficientemente rapida per essere utilizzata in *real-time*, in quanto il problema è ben formulato e generalmente la soluzione è stata trovata in poche iterazioni.

CAPITOLO 7

Conclusioni e sviluppi futuri

Il rover MORPHEUS è, alla data odierna, ancora allo stadio di costruzione dei componenti. Nel prossimo futuro esso sarà completato e potrà fornire le basi per una continuazione del progetto, in vista dell'ERC 2016. Sarà utilizzato a scopi dimostrativi in scuole, in modo da pubblicizzare l'Università degli Studi di Padova, il Dipartimento di Ingegneria Industriale, Elettronica, e Informatica. La parte relativa al controllo è stata sviluppata da Enrico Mion ed Elena Toffanin, anche se un riscontro sul lavoro effettuato, sia esso dal punto di vista della progettazione meccanica che elettronica o informatica, si avrà unicamente durante la campagna di test, anche a causa della scarsa documentazione relativa agli attuatori.

Per quanto riguarda il lavoro relativo al controllo del braccio robotico sviluppato in questo testo sicuramente è possibile sviluppare un approccio migliore per il path-planning. In questa sede, infatti, non si è tenuto conto di possibili impedimenti lungo il percorso, del jerk e del tempo di percorrenza. un ottimo approccio per il futuro, sembra quello di utilizzare lo spazio delle fasi: questa metodologia infatti permette di sfruttare al massimo i limiti di coppia, velocità, posizione del braccio ottimizzando il tempo di percorrenza tra due posizioni. Questo caso, però, richiede delle teorie e degli algoritmi sicuramente più complicati di quelli che sono l'obiettivo di questo testo, anche se andranno studiate meglio in futuro per le prossime iterazioni del braccio.

Lavori successivi comprendono anche lo sviluppo della dinamica del braccio a 7 gradi di libertà, e lo sfruttamento, ad esempio, di un controllo nel cosiddetto spazio operazio-

nale (*operational space control*), che consente, tramite le leggi della dinamica e funzioni costo, che le velocità si mantengano finite e non tendano a salire troppo. Esistono poi altre metodologie di controllo, sicuramente più efficienti, ma anche più complesse, che permetterebbero al braccio di avere delle performances migliori.

A.1 Componenti della matrice di trasformazione

$$M_{1,1} = s(\theta_7)(c(\theta_5)s(\theta_1) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)s(\theta_5)) \\ + c(\theta_7)(c(\theta_6)s(\theta_1)s(\theta_5) - s(\theta_2 + \theta_3 + \theta_4)c(\theta_1)s(\theta_6) + c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_5)c(\theta_6))$$

$$M_{1,2} = c(\theta_7)(c(\theta_5)s(\theta_1) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)s(\theta_5)) \\ - s(\theta_7)(c(\theta_6)s(\theta_1)s(\theta_5) - s(\theta_2 + \theta_3 + \theta_4)c(\theta_1)s(\theta_6) + c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_5)c(\theta_6))$$

$$M_{1,3} = s(\theta_6)(s(\theta_1)s(\theta_5) + c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_5)) + s(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_6)$$

$$M_{1,4} = a_1c(\theta_1) + a_2c(\theta_1)c(\theta_2) + d_5s(\theta_2 + \theta_3 + \theta_4)c(\theta_1) + a_3c(\theta_1)c(\theta_2)c(\theta_3) \\ - a_3c(\theta_1)s(\theta_2)s(\theta_3) + d_7s(\theta_1)s(\theta_5)s(\theta_6) + d_7s(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_6) \\ + d_7c(\theta_2 + \theta_3 + \theta_4)c(\theta_1)c(\theta_5)s(\theta_6)$$

$$M_{2,1} = -s(\theta_7)(c(\theta_1)c(\theta_5) + c(\theta_2 + \theta_3 + \theta_4)s(\theta_1)s(\theta_5)) - c(\theta_7)(s(\theta_2 + \theta_3 + \theta_4)s(\theta_1)s(\theta_6) + c(\theta_1)c(\theta_6)s(\theta_5) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_5)c(\theta_6)s(\theta_1))$$

$$M_{2,2} = s(\theta_7)(s(\theta_2 + \theta_3 + \theta_4)s(\theta_1)s(\theta_6) + c(\theta_1)c(\theta_6)s(\theta_5) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_5)c(\theta_6)s(\theta_1)) - c(\theta_7)(c(\theta_1)c(\theta_5) + c(\theta_2 + \theta_3 + \theta_4)s(\theta_1)s(\theta_5))$$

$$M_{2,3} = s(\theta_2 + \theta_3 + \theta_4)c(\theta_6)s(\theta_1) - s(\theta_6)(c(\theta_1)s(\theta_5) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_5)s(\theta_1))$$

$$M_{2,4} = a_1s(\theta_1) + a_2c(\theta_2)s(\theta_1) + d_5s(\theta_2 + \theta_3 + \theta_4)s(\theta_1) + a_3c(\theta_2)c(\theta_3)s(\theta_1) - d_7c(\theta_1)s(\theta_5)s(\theta_6) - a_3s(\theta_1)s(\theta_2)s(\theta_3) + d_7s(\theta_2 + \theta_3 + \theta_4)c(\theta_6)s(\theta_1) + d_7c(\theta_2 + \theta_3 + \theta_4)c(\theta_5)s(\theta_1)s(\theta_6)$$

$$M_{3,1} = c(\theta_7)(c(\theta_2 + \theta_3 + \theta_4)s(\theta_6) + s(\theta_2 + \theta_3 + \theta_4)c(\theta_5)c(\theta_6)) - s(\theta_2 + \theta_3 + \theta_4)s(\theta_5)s(\theta_7)$$

$$M_{3,2} = -s(\theta_7)(c(\theta_2 + \theta_3 + \theta_4)s(\theta_6) + s(\theta_2 + \theta_3 + \theta_4)c(\theta_5)c(\theta_6)) - s(\theta_2 + \theta_3 + \theta_4)c(\theta_7)s(\theta_5)$$

$$M_{3,3} = s(\theta_2 + \theta_3 + \theta_4)c(\theta_5)s(\theta_6) - c(\theta_2 + \theta_3 + \theta_4)c(\theta_6)$$

$$M_{3,4} = a_3s(\theta_2 + \theta_3) + a_2s(\theta_2) - d_5c(\theta_2 + \theta_3 + \theta_4) - d_7c(\theta_2 + \theta_3)c(\theta_4)c(\theta_6) + d_7s(\theta_2 + \theta_3)c(\theta_6)s(\theta_4) + d_7c(\theta_2 + \theta_3)c(\theta_5)s(\theta_4)s(\theta_6) + d_7s(\theta_2 + \theta_3)c(\theta_4)c(\theta_5)s(\theta_6)$$

A.2 Risoluzione dell'equazione non lineare (6.9)

Si vuole trovare ora le soluzioni dell'equazione:

$$P \cos \alpha * Q \sin \alpha + R = 0 \tag{A.1}$$

Si definisca γ tale che:

$$\cos \gamma = \frac{P}{\sqrt{P^2 + Q^2}} \quad \sin \gamma = \frac{Q}{\sqrt{P^2 + Q^2}}$$

Questo è sempre possibile. γ può essere determinato utilizzando la funzione atan2 :

$$\gamma = \text{atan2}\left(\frac{P}{\sqrt{P^2 + Q^2}}, \frac{Q}{\sqrt{P^2 + Q^2}}\right)$$

Si può riscrivere ora l'equazione (6.9) come:

$$\cos(\alpha - \gamma) = -\frac{R}{\sqrt{P^2 + Q^2}}$$

dalla quale si possono ottenere le due soluzioni per α in funzione dell'angolo γ :

$$\alpha = \gamma \pm \arccos\left(-\frac{R}{\sqrt{P^2 + Q^2}}\right)$$

APPENDICE B

Codice MATLAB

B.0.1 Plot analisi del terreno

```
1 %% I dati per gli scavi sono stati calcolati in un foglio excel
2
3 x=[1;2;3;4;5;6;7;8;9;10;15;20;30;40];
4 y1
   =[2.436;4.889;7.358;9.844;12.345;14.864;17.398;19.949;22.516;25.100;...
5     38.264;51.836;80.209;110.217];
6 y2
   =[2.765;5.538;8.319;11.108;13.905;16.710;19.523;22.344;25.172;28.009;...
7     42.310;56.810;86.404;116.791];
8 y3
   =[3.385;6.778;10.178;13.585;17.000;20.422;23.851;27.287;30.731;34.182;...
9     51.546;69.092;104.729;141.094];
10 y4
    =[4.787;9.582;14.385;19.197;24.017;28.845;33.682;38.527;43.380;48.241;...
11    72.674;97.314;147.216;197.950];
```

```
12 y5
    =[13.780;27.579;41.396;55.232;69.086;82.959;96.850;110.760;124.688;...
13     138.635;208.646;279.119;421.453;565.637];
14 y1l
    =[2.435;4.883;7.343;9.817;12.305;14.805;17.318;19.844;22.384;24.936;...
15     37.895;51.182;78.736;107.599];
16 y1p
    =[2.438;4.896;7.373;9.870;12.386;14.923;17.478;20.054;22.649;25.264;...
17     38.632;52.491;81.681;112.835];
18
19 plot(x,y1,'b','linewidth',1.5)
20 %set(gca,'FontSize',12,'LineWidth',1.25)
21 hold on
22 plot(x,y2,'m','linewidth',1.5)
23 plot(x,y3,'k','linewidth',1.5)
24 plot(x,y4,'r','linewidth',1.5)
25 plot(x(1:10),y5(1:10),'g','linewidth',1.5)
26 xlabel('Profondita'' di scavo [mm]')
27 ylabel('Forza prevista [N]')
28 title('Sensibilita'' all''angolo di ingresso \rho')
29 legend('\rho=10','\rho=30','\rho=45','\rho=60','\rho=80','
30     Location',...
31     'northwest')
31 figure
32 plot(x,y1,'b','linewidth',1.5)
33 hold on
34 plot(x,y1l,'r','linewidth',1.5)
35 plot(x,y1p,'k','linewidth',1.5)
36 xlabel('Profondita'' di scavo [mm]')
37 ylabel('Forza prevista [N]')
38 title('Sensibilita'' alla densita'' del suolo')
39 legend('1200 [kg/m^3] ','1500 [kg/m^3] ','1800 [kg/m^3] ','
40     Location',...
```



'northwest')

B.0.2 Jacobiano simbolico

```
1 syms l1 l2 l3 d5 t1 t2 t3 t4 t5 t6 t7
2
3 global A1 A2 A3 A4 A5 A6 A7
4
5 R1=A1(1:3,1:3);
6 R2=A2(1:3,1:3);
7 R3=A3(1:3,1:3);
8 R4=A4(1:3,1:3);
9 R5=A5(1:3,1:3);
10 R6=A6(1:3,1:3);
11 R7=A7(1:3,1:3);
12
13 Z0=[0 0 1]';
14 Z1=R1*Z0;
15 Z2=R1*R2*Z0;
16 Z3=R1*R2*R3*Z0;
17 Z4=R1*R2*R3*R4*Z0;
18 Z5=R1*R2*R3*R4*R5*Z0;
19 Z6=R1*R2*R3*R4*R5*R6*Z0;
20 Z7=R1*R2*R3*R4*R5*R6*R7*Z0;
21
22 P0=[0 0 0 1]';
23 PE=A1*A2*A3*A4*A5*A6*A7*P0;
24 P1=A1*P0;
25 P2=A1*A2*P0;
26 P3=A1*A2*A3*P0;
27 P4=A1*A2*A3*A4*P0;
28 P5=A1*A2*A3*A4*A5*P0;
29 P6=A1*A2*A3*A4*A5*A6*P0;
30 P7=A1*A2*A3*A4*A5*A6*A7*P0;
31
32 P0=P0(1:3);
33 P1=P1(1:3);
34 P2=P2(1:3);
```



35 P3=P3(1:3);

36 P4=P4(1:3);

37 P5=P5(1:3);

38 P6=P6(1:3);

39 P7=P7(1:3);

40 PE=PE(1:3);

41

42 JW=[cross(Z1,PE-P0) cross(Z2,PE-P1) cross(Z3,PE-P2) cross(Z4,PE-
P3) cross(Z5,PE-P4) cross(Z6,PE-P5) cross(Z7,PE-P6)];

43 JZ=[Z0 Z1 Z2 Z3 Z4 Z5 Z6];

44 J=[JW;JZ];

B.0.3 Analisi simbolica con metodo presentato al capitolo 4

```
1 %% definizione costanti
2
3 a2=45/1000;
4 a3=463/1000;
5 a4=400/1000;
6 d5=293/1000;
7 d7=0.1;
8
9 %% definizione variabili
10 syms t T;
11 t1=sym('t1(t)');
12 t2=sym('t2(t)');
13 t3=sym('t3(t)');
14 t4=sym('t4(t)');
15 t5=sym('t5(t)');
16 t6=sym('t6(t)');
17 t7=sym('t7(t)');
18 q=[t1;t2;t3;t4;t5;t6;t7];
19 dotq=diff(q);
20 ddotq=diff(dotq);
21
22 %% matrici di dh
23 M1=[ cos(t1), 0, sin(t1), a2*cos(t1); sin(t1), 0, -cos(t1), a2*
      sin(t1); 0, 1, 0, 0; 0, 0,
      0, 1];
24 M2=[ cos(t2), -sin(t2), 0, a3*cos(t2); sin(t2), cos(t2), 0, a3*
      sin(t2); 0, 0, 1, 0; 0,
      0, 1];
25 M3=[ cos(t3), -sin(t3), 0, a4*cos(t3); sin(t3), cos(t3), 0, a4*
      sin(t3); 0, 0, 1, 0; 0,
      0, 1];
26 M4=[ cos(t4), 0, sin(t4), 0; sin(t4), 0, -cos(t4), 0; 0,
      1, 0, 0; 0, 0, 0, 1];
27 M5=[ cos(t5), 0, -sin(t5), 0; sin(t5), 0, cos(t5), 0;
```



```

    0, -1,      0, d5;      0, 0,      0, 1];
28 M6=[ cos(t6), 0,  sin(t6), 0; sin(t6), 0, -cos(t6), 0;      0,
    1,      0, 0;      0, 0,      0, 1];
29 M7=[ cos(t7), -sin(t7), 0, 0; sin(t7),  cos(t7), 0, 0;
    0,      0, 1, d7;      0,      0, 0, 1];
30
31 M1I=iinv(M1);
32 M2I=iinv(M2);
33 M3I=iinv(M3);
34 M4I=iinv(M4);
35 M5I=iinv(M5);
36 M6I=iinv(M6);
37 M7I=iinv(M7);
38 T=sym(zeros(4,4,7));
39 W=sym(zeros(4,4,7));
40 TI=sym(zeros(4,4,7));
41 H=sym(zeros(4,4,7));
42 W0=sym(zeros(4,4,7));
43 H0=sym(zeros(4,4,7));
44 W0tot=sym(zeros(4));
45 H0tot=sym(zeros(4));
46 T(:, :, 1)=M1;
47 T(:, :, 2)=M1*M2;
48 T(:, :, 3)=M1*M2*M3;
49 T(:, :, 4)=M1*M2*M3*M4;
50 T(:, :, 5)=M1*M2*M3*M4*M5;
51 T(:, :, 6)=M1*M2*M3*M4*M5*M6;
52 T(:, :, 7)=M1*M2*M3*M4*M5*M6*M7;
53 for i=1:7
54 TI(:, :, i)=iinv(T(:, :, i));
55 W(1,2,i)=-dotq(i);
56 W(2,1,i)=dotq(i);
57 W(4,1:4,i)=0;
58 W(1:2,4,i)=0;
59 end
60 for i=1:7
```

```
61     H(:, :, i) = diff(W(:, :, i), t) + (W(:, :, i) * W(:, :, i));
62
63 end
64
65 for i = 1:7
66     W0(:, :, i) = T(:, :, i) * W(:, :, i) * TI(:, :, i);
67     H0(:, :, i) = T(:, :, i) * H(:, :, i) * TI(:, :, i);
68 end
69     W0tot = W0(:, :, 1);
70     H0tot = H0(:, :, 1) + 2 * W0(:, :, 1);
71 for i = 2:7
72     W0tot = W0tot + W0(:, :, i);
73     H0tot = H0H0tot + H0(:, :, i) + 2 * W0tot * W0(:, :, i);
74 end
```

B.0.4 Matrice di trasformazione DH

```
1 function M=dh(theta , alpha , a , d)
2
3 %M=[cos(theta),-sin(theta)*cos(alpha),sin(theta)*sin(alpha),a*
      cos(theta);sin(theta),cos(theta)*cos(alpha),-cos(theta)*sin(
      alpha),a*sin(theta);0,sin(alpha),cos(alpha),d;0,0,0,1];
4 M=[cos(theta),-sin(theta),0,a;sin(theta)*cos(alpha),cos(theta)*
      cos(alpha),-sin(alpha),d*sin(alpha);sin(theta)*sin(alpha),
      cos(theta)*sin(alpha),cos(alpha),d*cos(alpha);0,0,0,1];
```

B.0.5 Matrice di trasformazione DH simbolica

```
1 syms th1 th2 th3 th4 th5 th6 th7 l1 l2 l3 d5 d7
2 alpha=[0 pi/2 0 0 pi/2 -pi/2 pi/2];
3 theta=[th1 th2 th3 th4+pi/2 th5 th6 th7];
4 d=[0 0 0 0 d5 0 d7];
5 a=[0 l1 l2 l3 0 0 0];
6
7
8
9 M=sym(zeros(4,4,7));
10 for i =1:7
11 M(:, :, i)=[cos(theta(i)), -sin(theta(i)), 0, a(i); sin(theta(i))*cos(
    alpha(i)), ...
12     cos(theta(i))*cos(alpha(i)), -sin(alpha(i)), d(i)*sin(alpha(i)
    ); ...
13     sin(theta(i))*sin(alpha(i)), cos(theta(i))*sin(alpha(i)), cos(
    alpha(i)), ...
14     d(i)*cos(alpha(i)); 0, 0, 0, 1];
15 end
```



B.0.6 Inversa della matrice di trasformazione

```
1 function MI=iinv(M)
2
3 R=M(1:3,1:3);
4 t=M(1:3,4);
5 RI=R';
6 tI=-RI*t;
7 MI(1:3,1:3)=RI;
8 MI(1:3,4)=tI;
9 MI(4,:)=[0,0,0,1];
```

B.0.7 Cinematica inversa

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %%%
3 %%% Erik Bertolutti
4 %%% Soluzione della cinematica inversa del braccio robotico
5 %%% In assolvimento parziale alla Tesi di Laurea Magistrale
6 %%%
7
8 % si definiscono i parametri di DH del braccio robotico in esame
9 global px py pz thetai thetaf
10
11 a=zeros(7,1);
12 d=zeros(7,1);
13
14 a(2)=45/1000;
15 a(3)=463/1000;
16 a(4)=400/1000;
17 d(5)=293/1000;
18 d(7)=0.1;
19 alpha=[0; pi/2; 0; 0; pi/2; -pi/2; pi/2];
20 valori=zeros(3,3000);
21
22 %% definizione della posizione iniziale del braccio
23 % for j=1:300;
24 thetai=[0;0;0;0;0;0;0];
25
26 x0 = [thetai(2);thetai(3);thetai(4)]; % valori di partenza per
    la
27 % soluzione del problema non lineare
28 % e'' utile scegliere i valori in funzione della posizione di
    partenza del
29 % braccio
30
31
32 %% generazione di una posizione finale casuale
```

```
33
34 % la posizione e'' funzione della distanza dal centro
35 ltot=sum(a)+sum(d)-a(2)-d(7);
36 ele=rand*pi-pi/2;
37 dec=rand*2*pi;
38 pz=ltot*sin(ele);
39 px=ltot*cos(ele)*cos(dec);
40 py=ltot*cos(ele)*sin(dec);
41
42 % la disposizione della matrice finale viene generata
    casualmente
43 P=orth(rand(3));
44
45 %% richiamo della posizione finale in funzione dei parametri
46
47 % si definisca ora l'orientamento della terna finale secondo il
    sistema
48 % riferimento O.
49 % P=[0, 1, 0;1, 0, 0;0, 0, 1];
50
51 % vado a definire la posizione del centro del polso per comodita
    ,
52 % in caso venga data la posizione pe dell' end effector si de-
    commenti la
53 % linea successiva e si commenti px, py, pz.
54
55 % p=pe-d(7).*(P(:,3));
56 % px=p(1);
57 % py=p(2);
58 % pz=p(3);
59 %
60 % px=0.3;
61 % pz=0.3;
62 % py=-0.7;
63
64 % valori(:,j)=[px;py;pz]
```

```

65 %% definita la posizione del centro del polso, e'' possibile
    risolvere la
66 % ridondanza
67
68 options = optimoptions('fsolve','Display','iter');
69 options.MaxFunEvals=1000;
70 options.TolX = 1e-10;
71 [x,fval] = fsolve(@link3,x0,options);
72 thetaf(1)=atan2(py,px);
73 % px=px-a(2)*cos(thetaf(1));
74 thetaf(2)=x(1);
75 thetaf(3)=x(2);
76 thetaf(4)=x(3);
77
78 %% le matrici di trasformazione di DH sono state trovate in
    forma simbolica su Maple, e poi trasferite
79 % M1=[ cos(thetaf(1)), 0, sin(thetaf(1)), a(1)*cos(thetaf(1));
    sin(thetaf(1)), 0, -cos(thetaf(1)), a(1)*sin(thetaf(1));
    0, 1, 0, 0; 0, 0, 0,
    1];
80 % M2=[ cos(thetaf(2)), -sin(thetaf(2)), 0, a(2)*cos(thetaf(2));
    sin(thetaf(2)), cos(thetaf(2)), 0, a(2)*sin(thetaf(2));
    0, 0, 1, 0; 0, 0, 0,
    1];
81 % M3=[ cos(thetaf(3)), -sin(thetaf(3)), 0, a(3)*cos(thetaf(3));
    sin(thetaf(3)), cos(thetaf(3)), 0, a(3)*sin(thetaf(3));
    0, 0, 1, 0; 0, 0, 0,
    1];
82 % M4=[ cos(thetaf(4)), 0, sin(thetaf(4)), 0; sin(thetaf(4)), 0,
    -cos(thetaf(4)), 0; 0, 1, 0, 0; 0, 0,
    0, 1];
83
84 for i=1:4
85     if i==4
86         M(:, :, i)=dh(thetaf(i)-pi/2,alpha(i),a(i),d(i));
87     else

```

```

88 M(:, :, i)=dh(thetaf(i), alpha(i), a(i), d(i));
89     end
90 end
91 Ma=M(:, :, 1)*M(:, :, 2)*M(:, :, 3)*M(:, :, 4);
92 MI=iinv(Ma);
93 R=MI(1:3, 1:3)*P;
94
95 % thet(6)=atan2(sqrt(R(1,3)^2+R(2,3)^2), R(3,3));
96 disc=acos((Ma(1:3,3) '*R(:, 3)));
97 if disc >= 0 && disc <= pi
98 % if thet(6) >= 0 && thet(6) <= pi
99 thet(5)=atan2(R(2,3), R(1,3));
100 thet(6)=atan2(sqrt(R(1,3)^2+R(2,3)^2), R(3,3));
101 thet(7)=atan2(R(3,2), -R(3,1));
102 else
103 thet(5)=atan2(-R(2,3), -R(1,3));
104 thet(6)=atan2(-sqrt(R(1,3)^2+R(2,3)^2), R(3,3));
105 thet(7)=atan2(-R(3,2), R(3,1));
106 end
107
108
109 for i=5:7
110 M(:, :, i)=dh(thetaf(i), alpha(i), a(i), d(i));
111 end
112 M0=[1, 0, 0, 0; 0, 1, 0, 0; 0, 0, 1, 0; 0, 0, 0, 1];
113 T1=M0*M(:, :, 1);
114 T2=M0*M(:, :, 1)*M(:, :, 2);
115 T3=M0*M(:, :, 1)*M(:, :, 2)*M(:, :, 3);
116 T4=M0*M(:, :, 1)*M(:, :, 2)*M(:, :, 3)*M(:, :, 4);
117 T5=M0*M(:, :, 1)*M(:, :, 2)*M(:, :, 3)*M(:, :, 4)*M(:, :, 5);
118 T6=M0*M(:, :, 1)*M(:, :, 2)*M(:, :, 3)*M(:, :, 4)*M(:, :, 5)*M(:, :, 6);
119 T7=M0*M(:, :, 1)*M(:, :, 2)*M(:, :, 3)*M(:, :, 4)*M(:, :, 5)*M(:, :, 6)*M
    (:, :, 7);
120 p=zeros(3, 8);
121 p(1:3, 1)=M0(1:3, 4);
122 p(1:3, 2)=T1(1:3, 4);

```

```
123 p(1:3,3)=T2(1:3,4);
124 p(1:3,4)=T3(1:3,4);
125 p(1:3,5)=T4(1:3,4);
126 p(1:3,6)=T5(1:3,4);
127 p(1:3,7)=T6(1:3,4);
128 p(1:3,8)=T7(1:3,4);
129
130 for i=1:8
131 as(1:3,i)=sum(p(:,1:i),2);
132 end
133
134 %% soluzione con metodo geometrico fino alla posizione del polso
135 q=zeros(3,8);
136 q(1:3,2)=[a(2)*cos(thetaf(1)),a(2)*sin(thetaf(1)),0]';
137 q(1:3,3)=[(a(2)+a(3)*cos(thetaf(2)))*cos(thetaf(1)),(a(2)+a(3)*
    cos(thetaf(2)))*sin(thetaf(1)),a(3)*sin(thetaf(2))]';
138 q(1:3,4)=[(a(2)+a(3)*cos(thetaf(2))+a(4)*cos(thetaf(2)+thetaf(3)
    ))*cos(thetaf(1)),(a(2)+a(3)*cos(thetaf(2))+a(4)*cos(thetaf
    (2)+thetaf(3)))*sin(thetaf(1)),a(3)*sin(thetaf(2))+a(4)*sin(
    thetaf(2)+thetaf(3))]';
139 q(1:3,5)=[(a(2)+a(3)*cos(thetaf(2))+a(4)*cos(thetaf(2)+thetaf(3)
    )+d(5)*cos(thetaf(2)+thetaf(3)+thetaf(4)))*cos(thetaf(1)),(a
    (2)+a(3)*cos(thetaf(2))+a(4)*cos(thetaf(2)+thetaf(3))+d(5)*
    cos(thetaf(2)+thetaf(3)+thetaf(4)))*sin(thetaf(1)),a(3)*sin(
    thetaf(2))+a(4)*sin(thetaf(2)+thetaf(3))+d(5)*sin(thetaf(2)+
    thetaf(3)+thetaf(4))]';
140 rot=zeros(3,3,5);
141
142
143 % q(1:3,6)=M5(1:3,4);
144 % q(1:3,7)=M6(1:3,4);
145 % q(1:3,8)=M7(1:3,4);
146
147
148
149 theta=x;
```



```
150 ex=a(3)*cos(theta(1))+a(4)*cos(theta(1)+theta(2))+d(5)*cos(theta
      (1)+theta(2)+theta(3))-px;
151 ez=a(3)*sin(theta(1))+a(4)*sin(theta(1)+theta(2))+d(5)*sin(theta
      (1)+theta(2)+theta(3))-pz;
152 % end
153 close all
154 plot3(p(1,:),p(2,:),p(3,:))
155 hold on
156 grid on
157 plot3(q(1,1:5),q(2,1:5),q(3,1:5),'r')
158 axis equal
159 xlabel('x')
160 ylabel('y')
161 zlabel('z')
162 axis equal
163 xlabel('x')
164 ylabel('y')
165 zlabel('z')
```

B.0.8 Path-planning con polinomio

```
1 global thetaf thetai
2 cases=input('Digitare 1 per traiettoria casuale angolare ,
3           digitare 2 per traiettoria rettilinea\n');
4 t=10;
5 timestep=1e-1;
6 steps=t/timestep+1;
7 options = optimoptions('fsolve','Display','iter');
8 options.MaxFunEvals=1000;
9 options.TolX = 1e-10;
10
11 switch cases
12     case 1
13
14     for i=1:7
15         a(i)=(thetaf(i)-thetai(i))./t.^3*10;
16         b(i)=-(thetaf(i)-thetai(i))./t.^4*15;
17         c(i)=(thetaf(i)-thetai(i))./t.^5*6;
18     end
19
20     for i=1:steps
21         A1(i,1)=timestep*(i-1);
22         A1(i,2)=a(1)*(timestep*(i-1))^3+b(1)*(timestep*(i-1))^4+c(1)
23             *(timestep*(i-1))^5+thetai(1);
24         A1(i,3)=3*a(1)*(timestep*(i-1))^2+4*b(1)*(timestep*(i-1))
25             ^3+5*c(1)*(timestep*(i-1))^4;
26         A1(i,4)=6*a(1)*(timestep*(i-1))+12*b(1)*(timestep*(i-1))
27             ^2+20*c(1)*(timestep*(i-1))^3;
28         B1(i,1)=(timestep*(i-1));
29         B1(i,2)=a(2)*(timestep*(i-1))^3+b(2)*(timestep*(i-1))^4+c(2)
30             *(timestep*(i-1))^5+thetai(2);
31         B1(i,3)=3*a(2)*(timestep*(i-1))^2+4*b(2)*(timestep*(i-1))
32             ^3+5*c(2)*(timestep*(i-1))^4;
33         B1(i,4)=6*a(2)*(timestep*(i-1))+12*b(2)*(timestep*(i-1))
```

```

      ^2+20*c(2)*(timestep*(i-1))^3;
29 C1(i,1)=(timestep*(i-1));
30 C1(i,2)=a(3)*(timestep*(i-1))^3+b(3)*(timestep*(i-1))^4+c(3)
      *(timestep*(i-1))^5+thetai(3);
31 C1(i,3)=3*a(3)*(timestep*(i-1))^2+4*b(3)*(timestep*(i-1))
      ^3+5*c(3)*(timestep*(i-1))^4;
32 C1(i,4)=6*a(3)*(timestep*(i-1))+12*b(3)*(timestep*(i-1))
      ^2+20*c(3)*(timestep*(i-1))^3;
33 E1(i,1)=timestep*(i-1);
34 E1(i,2)=a(4)*(timestep*(i-1))^3+b(4)*(timestep*(i-1))^4+c(4)
      *(timestep*(i-1))^5+thetai(4);
35 E1(i,3)=3*a(4)*(timestep*(i-1))^2+4*b(4)*(timestep*(i-1))
      ^3+5*c(4)*(timestep*(i-1))^4;
36 E1(i,4)=6*a(4)*(timestep*(i-1))+12*b(4)*(timestep*(i-1))
      ^2+20*c(4)*(timestep*(i-1))^3;
37 F1(i,1)=(timestep*(i-1));
38 F1(i,2)=a(5)*(timestep*(i-1))^3+b(5)*(timestep*(i-1))^4+c(5)
      *(timestep*(i-1))^5+thetai(5);
39 F1(i,3)=3*a(5)*(timestep*(i-1))^2+4*b(5)*(timestep*(i-1))
      ^3+5*c(5)*(timestep*(i-1))^4;
40 F1(i,4)=6*a(5)*(timestep*(i-1))+12*b(5)*(timestep*(i-1))
      ^2+20*c(5)*(timestep*(i-1))^3;
41 G1(i,1)=(timestep*(i-1));
42 G1(i,2)=a(6)*(timestep*(i-1))^3+b(6)*(timestep*(i-1))^4+c(6)
      *(timestep*(i-1))^5+thetai(6);
43 G1(i,3)=3*a(6)*(timestep*(i-1))^2+4*b(6)*(timestep*(i-1))
      ^3+5*c(6)*(timestep*(i-1))^4;
44 G1(i,4)=6*a(6)*(timestep*(i-1))+12*b(6)*(timestep*(i-1))
      ^2+20*c(6)*(timestep*(i-1))^3;
45 H1(i,1)=timestep*(i-1);
46 H1(i,2)=a(7)*(timestep*(i-1))^3+b(7)*(timestep*(i-1))^4+c(7)
      *(timestep*(i-1))^5+thetai(7);
47 H1(i,3)=3*a(7)*(timestep*(i-1))^2+4*b(7)*(timestep*(i-1))
      ^3+5*c(7)*(timestep*(i-1))^4;
48 H1(i,4)=6*a(7)*(timestep*(i-1))+12*b(7)*(timestep*(i-1))
      ^2+20*c(7)*(timestep*(i-1))^3;

```

```
49 end
50
51 close all
52 figure
53 subplot(1,3,1)
54 plot(A1(:,1),A1(:,2))
55 title('$\theta_1$', 'interpreter', 'latex')
56 grid on
57 axis square
58 xlabel('t')
59 subplot(1,3,2)
60 plot(A1(:,1),A1(:,3), 'k')
61 title('$\dot{\theta}_1$', 'interpreter', 'latex')
62 grid on
63 axis square
64 xlabel('t')
65 subplot(1,3,3)
66 plot(A1(:,1),A1(:,4), 'r')
67 title('$\ddot{\theta}_1$', 'interpreter', 'latex')
68 xlabel('t')
69 grid on
70 axis square
71 figure
72 subplot(1,3,1)
73 plot(A1(:,1),B1(:,2))
74 title('$\theta_2$', 'interpreter', 'latex')
75 xlabel('t')
76 grid on
77 axis square
78 subplot(1,3,2)
79 plot(A1(:,1),B1(:,3), 'k')
80 title('$\dot{\theta}_2$', 'interpreter', 'latex')
81 xlabel('t')
82 grid on
83 axis square
84 subplot(1,3,3)
```

```
85 plot(A1(:,1),B1(:,4),'r')
86 title('$\ddot{\theta}_2$', 'interpreter', 'latex')
87 xlabel('t')
88 grid on
89 axis square
90 figure
91 subplot(1,3,1)
92 plot(A1(:,1),C1(:,2))
93 title('$\theta_3$', 'interpreter', 'latex')
94 xlabel('t')
95 grid on
96 axis square
97 subplot(1,3,2)
98 plot(A1(:,1),C1(:,3),'k')
99 title('$\dot{\theta}_3$', 'interpreter', 'latex')
100 xlabel('t')
101 grid on
102 axis square
103 subplot(1,3,3)
104 plot(A1(:,1),C1(:,4),'r')
105 title('$\ddot{\theta}_3$', 'interpreter', 'latex')
106 xlabel('t')
107 grid on
108 axis square
109 figure
110 subplot(1,3,1)
111 plot(A1(:,1),E1(:,2))
112 title('$\theta_4$', 'interpreter', 'latex')
113 xlabel('t')
114 grid on
115 axis square
116 subplot(1,3,2)
117 plot(A1(:,1),E1(:,3),'k')
118 title('$\dot{\theta}_4$', 'interpreter', 'latex')
119 xlabel('t')
120 grid on
```

```
121 axis square
122 subplot(1,3,3)
123 plot(A1(:,1),E1(:,4),'r')
124 title('$\ddot{\theta}_4$', 'interpreter', 'latex')
125 xlabel('t')
126 grid on
127 axis square
128 figure
129 subplot(1,3,1)
130 plot(A1(:,1),F1(:,2))
131 title('$\theta_5$', 'interpreter', 'latex')
132 xlabel('t')
133 grid on
134 axis square
135 subplot(1,3,2)
136 plot(A1(:,1),F1(:,3),'k')
137 title('$\dot{\theta}_5$', 'interpreter', 'latex')
138 xlabel('t')
139 grid on
140 axis square
141 subplot(1,3,3)
142 plot(A1(:,1),F1(:,4),'r')
143 title('$\ddot{\theta}_5$', 'interpreter', 'latex')
144 xlabel('t')
145 grid on
146 axis square
147 figure
148 subplot(1,3,1)
149 plot(A1(:,1),G1(:,2))
150 title('$\theta_6$', 'interpreter', 'latex')
151 xlabel('t')
152 grid on
153 axis square
154 subplot(1,3,2)
155 plot(A1(:,1),G1(:,3),'k')
156 title('$\dot{\theta}_6$', 'interpreter', 'latex')
```

```
157 xlabel('t')
158 grid on
159 axis square
160 subplot(1,3,3)
161 plot(A1(:,1),G1(:,4),'r')
162 title('$\ddot{\theta}_6$', 'interpreter', 'latex')
163 xlabel('t')
164 grid on
165 axis square
166 figure
167 subplot(1,3,1)
168 plot(A1(:,1),H1(:,2))
169 title('$\theta_7$', 'interpreter', 'latex')
170 xlabel('t')
171 grid on
172 axis square
173 subplot(1,3,2)
174 plot(A1(:,1),H1(:,3),'k')
175 title('$\dot{\theta}_7$', 'interpreter', 'latex')
176 xlabel('t')
177 grid on
178 axis square
179 subplot(1,3,3)
180 plot(A1(:,1),H1(:,4),'r')
181 title('$\ddot{\theta}_7$', 'interpreter', 'latex')
182 xlabel('t')
183 grid on
184 axis square
185
186 %
```

%%

```
187     case 2
188 %traiettoria rettilinea
189 pi=[1.2;0;0];
190 pf=rand(3,1);
```

```
191 dist=pf-pi;
192 moddist=sqrt(dist'*dist);
193 x0=[0;0;0]
194 for i=2:steps
195     P=pi+(10*t^2*((timestep)*(i-1))^3-15*t*((timestep)*(i-1))
        ^4+(6*((timestep)*(i-1))^5))*dist;
196
197     [x,fval]=fsolve(@link3,x0,options);
198     thetaf(1)=atan2(py,px);
199     thetaf(2)=x(1);
200     thetaf(3)=x(2);
201     thetaf(4)=x(3);
202     x0=[thetaf(2);thetaf(3);thetaf(4)];
203     for i=1:4
204         if i==4
205             M(:,:,i)=dh(thetaf(i)+pi/2,alpha(i),a(i),d(i));
206         else
207             M(:,:,i)=dh(thetaf(i),alpha(i),a(i),d(i));
208         end
209     end
210     Ma=M(:,:,1)*M(:,:,2)*M(:,:,3)*M(:,:,4);
211     MI=iinv(Ma);
212     R=MI(1:3,1:3)*P;
213
214     thetaf(6)=atan2(sqrt(R(1,3)^2+R(2,3)^2),R(3,3));
215
216     if thetaf(6)>= 0 && thetaf(6)<= pi
217         thetaf(5)=atan2(R(2,3),R(1,3));
218         thetaf(6)=atan2(sqrt(R(1,3)^2+R(2,3)^2),R(3,3));
219         thetaf(7)=atan2(R(3,2),-R(3,1));
220     else
221         thetaf(5)=atan2(-R(2,3),-R(1,3));
222         thetaf(6)=atan2(-sqrt(R(1,3)^2+R(2,3)^2),R(3,3));
223         thetaf(7)=atan2(-R(3,2),R(3,1));
224     end
225
```

```

226 A1(i,1)=timestep*(i-1);
227 A1(i,2)=a(1)*(timestep*(i-1))^3+b(1)*(timestep*(i-1))^4+c(1)
      *(timestep*(i-1))^5+thetai(1);
228 A1(i,3)=3*a(1)*(timestep*(i-1))^2+4*b(1)*(timestep*(i-1))
      ^3+5*c(1)*(timestep*(i-1))^4;
229 A1(i,4)=6*a(1)*(timestep*(i-1))+12*b(1)*(timestep*(i-1))
      ^2+20*c(1)*(timestep*(i-1))^3;
230 B1(i,1)=(timestep*(i-1));
231 B1(i,2)=a(2)*(timestep*(i-1))^3+b(2)*(timestep*(i-1))^4+c(2)
      *(timestep*(i-1))^5+thetai(2);
232 B1(i,3)=3*a(2)*(timestep*(i-1))^2+4*b(2)*(timestep*(i-1))
      ^3+5*c(2)*(timestep*(i-1))^4;
233 B1(i,4)=6*a(2)*(timestep*(i-1))+12*b(2)*(timestep*(i-1))
      ^2+20*c(2)*(timestep*(i-1))^3;
234 C1(i,1)=(timestep*(i-1));
235 C1(i,2)=a(3)*(timestep*(i-1))^3+b(3)*(timestep*(i-1))^4+c(3)
      *(timestep*(i-1))^5+thetai(3);
236 C1(i,3)=3*a(3)*(timestep*(i-1))^2+4*b(3)*(timestep*(i-1))
      ^3+5*c(3)*(timestep*(i-1))^4;
237 C1(i,4)=6*a(3)*(timestep*(i-1))+12*b(3)*(timestep*(i-1))
      ^2+20*c(3)*(timestep*(i-1))^3;
238 E1(i,1)=timestep*(i-1);
239 E1(i,2)=a(4)*(timestep*(i-1))^3+b(4)*(timestep*(i-1))^4+c(4)
      *(timestep*(i-1))^5+thetai(4);
240 E1(i,3)=3*a(4)*(timestep*(i-1))^2+4*b(4)*(timestep*(i-1))
      ^3+5*c(4)*(timestep*(i-1))^4;
241 E1(i,4)=6*a(4)*(timestep*(i-1))+12*b(4)*(timestep*(i-1))
      ^2+20*c(4)*(timestep*(i-1))^3;
242 F1(i,1)=(timestep*(i-1));
243 F1(i,2)=a(5)*(timestep*(i-1))^3+b(5)*(timestep*(i-1))^4+c(5)
      *(timestep*(i-1))^5+thetai(5);
244 F1(i,3)=3*a(5)*(timestep*(i-1))^2+4*b(5)*(timestep*(i-1))
      ^3+5*c(5)*(timestep*(i-1))^4;
245 F1(i,4)=6*a(5)*(timestep*(i-1))+12*b(5)*(timestep*(i-1))
      ^2+20*c(5)*(timestep*(i-1))^3;
246 G1(i,1)=(timestep*(i-1));

```

```
247     G1(i,2)=a(6)*(timestep*(i-1))^3+b(6)*(timestep*(i-1))^4+c(6)
        *(timestep*(i-1))^5+thetai(6);
248     G1(i,3)=3*a(6)*(timestep*(i-1))^2+4*b(6)*(timestep*(i-1))
        ^3+5*c(6)*(timestep*(i-1))^4;
249     G1(i,4)=6*a(6)*(timestep*(i-1))+12*b(6)*(timestep*(i-1))
        ^2+20*c(6)*(timestep*(i-1))^3;
250     H1(i,1)=timestep*(i-1);
251     H1(i,2)=a(7)*(timestep*(i-1))^3+b(7)*(timestep*(i-1))^4+c(7)
        *(timestep*(i-1))^5+thetai(7);
252     H1(i,3)=3*a(7)*(timestep*(i-1))^2+4*b(7)*(timestep*(i-1))
        ^3+5*c(7)*(timestep*(i-1))^4;
253     H1(i,4)=6*a(7)*(timestep*(i-1))+12*b(7)*(timestep*(i-1))
        ^2+20*c(7)*(timestep*(i-1))^3;
254 end
255
256 close all
257 figure
258 plot(A1(:,1),A1(:,2))
259 figure
260 plot(A1(:,1),A1(:,3))
261 figure
262 plot(A1(:,1),A1(:,4))
263 figure
264 plot(A1(:,1),B1(:,2))
265 figure
266 plot(A1(:,1),B1(:,3))
267 figure
268 plot(A1(:,1),B1(:,4))
269 figure
270 plot(A1(:,1),C1(:,2))
271 figure
272 plot(A1(:,1),C1(:,3))
273 figure
274 plot(A1(:,1),C1(:,4))
275 figure
276 plot(A1(:,1),F1(:,2))
```



```
277 figure
278 plot(A1(:,1),F1(:,3))
279 figure
280 plot(A1(:,1),F1(:,4))
281 figure
282 plot(A1(:,1),F1(:,2))
283 figure
284 plot(A1(:,1),F1(:,3))
285 figure
286 plot(A1(:,1),F1(:,4))
287 figure
288 plot(A1(:,1),G1(:,2))
289 figure
290 plot(A1(:,1),G1(:,3))
291 figure
292 plot(A1(:,1),G1(:,4))
293 figure
294 plot(A1(:,1),H1(:,2))
295 figure
296 plot(A1(:,1),H1(:,3))
297 figure
298 plot(A1(:,1),H1(:,4))
299
300
301
302 end
```

B.0.9 Funzione non lineare da minimizzare

```
1 function F=link3(theta)
2 l1=0.463;
3 l2=0.4;
4 l3=0.293;
5 % px=-0.9414+0.0412;
6 % pz=0.5276;
7 global px py pz
8
9 F=[l1*cos(theta(1))+l2*cos(theta(1)+theta(2))+l3*cos(theta(1)+
    theta(2)+theta(3))-px;
10 l1*sin(theta(1))+l2*sin(theta(1)+theta(2))+l3*sin(theta(1)+
    theta(2)+theta(3))-pz;
11 l2*l3*(l1*l3*cos(theta(2)+theta(3))*(l1*l2*sin(theta(2))+l1*
    l3*sin(theta(2)+theta(3)))+(l2*l3*sin(theta(3))+l1*l3*sin
    (theta(2)+theta(3)))*(l1*l2*cos(theta(2))+l1*l3*cos(theta
    (2)+theta(3)))*(l2*l3*sin(theta(3))+l1*l3*sin(theta(2)+
    theta(3)))+ ...
12 (-l2*l3*cos(theta(3))*(l2*l3*sin(theta(3))+l1*l3*sin(theta(2)
    +theta(3)))*(l1*l2*sin(theta(2))+l1*l3*sin(theta(2)+theta
    (3)))-l2*l3*sin(theta(3))*(l2*l3*sin(theta(3))+l1*l3*cos
    (theta(2)+theta(3)))*(l1*l2*sin(theta(2))+l1*l3*sin(theta
    (2)+theta(3)))-l2*l3*(l2*l3*sin(theta(3))+l1*l3*sin(theta
    (2)+theta(3))*l1*l3*cos(theta(2)+cos(theta(3)))*(l1*l2*
    sin(theta(2))+l1*l3*sin(theta(2)+theta(3)))];
```

Testi consultati

- [1] *Appunti dal corso di Robotica Spaziale*. 2014.
- [3] John J. Craig. *Introduction to Robotics: Mechanics and Control*. 2004.
- [4] B. Siciliano , O. Khatib (editors). *Springer Handbook of Robotics*. 2008.
- [7] R. D. Klafter et al. *Robotic Engineering: An Integrated Approach*. A cura di Prentice Hall Int. Ed. 1989.
- [9] *Handbook of Industrial Robotics*. 1999.
- [11] V. Kumar. *Appunti delle lezioni del corso di robotica*.
- [12] Giovanni Marro. *Controlli Automatici*. A cura di Zanichelli. 2004.
- [14] J. Pires. *Industrial Robots Programming*. 2007.
- [16] M. Spong. *Robot dynamics and control*. 1989.

Articoli scientifici consultati

- [2] J. Hanley, M. T. Mellonv, R. E. Arvidson. «Mechanical Strength of Martian Analog Soils». In: *45th Lunar and Planetary Science Conference* (2014).
- [5] G. Legnani et al. «A Homogeneous Matrix Approach to 3D Kinematics and Dynamics, Part 1:theory». In: *Machanisms and Machine Theory* (1996).
- [6] G. Legnani et al. «A Homogeneous Matrix Approach to 3D Kinematics and Dynamics, Part 2: applications to chains of rigid bodies and serial manipulators». In: *Machanisms and Machine Theory* (1996).

- [8] Oleg Ivlev, Axel Gräser. «An Analytical Method for the Inverse Kinematics of Redundant Robots». In: *Intelligent Automation and Active Systems* (1997).
- [10] Wildan Lalo, Thorsten Brandt, Dieter Schramm, Manfred Hiller. «A Linear Optimization Approach To Inverse Kinematics Of Redundant Robots With Respect To Manipulability». In: *ISARC* (2008).
- [13] H. Cannon O. Luengo, S. Singh. «Modeling and Identification of Soil-tool Interaction in Automated Excavation». In: *Proceedings of the 1998 IEEURSJ Intl. Conference on Intelligent Robots and Systems*, ().
- [15] A.R. Reece. «The Fundamental Equation of Earthmoving Mechanics». In: *Proceedings of Institution of Mechanical Engineers* (1964).
- [17] W. J. Chung, W. K. Chug, Y. Youmt. «Inverse Kinematics of Planar Redundant Manipulators using Virtual Link and Displacement Distribution Schemes». In: *Proceedings of the 1991 IEEE International Conference on Robotics and Automation* (1991).
- [18] Jingguo Wang, Yangmin Li, Xinhua Zhao. «Inverse Kinematics and Control of a 7-DOF Redundant Manipulator Based on the Closed-Loop Algorithm». In: *Advanced Robotic Systems International* (2009).