



**POLITECNICO DI MILANO**

---

**FACOLTÀ DI INGEGNERIA INDUSTRIALE**  
**Corso di Laurea Specialistica in Ingegneria Meccanica**

**UN MANIPOLATORE PER PROVE DINAMICHE  
IN GALLERIA DEL VENTO: SINTESI  
CINETOSTATICA ED ANALISI DINAMICA.**

**Relatore**

**Ing. Hermes GIBERTI**

**Tesi di laurea di**

**Enrico FIORE**

**Matricola: 799913**

**Anno Accademico 2013/2014**



*A Fabrizia ed alla mia famiglia.*



## RINGRAZIAMENTI

Desidero ringraziare il prof. Hermes Giberti per la disponibilità mostrata e per le opportunità offertemi. I suoi consigli non solo in merito a questo lavoro di tesi sono risultati preziosi. Ringrazio infinitamente Davide Ferrari che mi ha seguito in questi mesi. La sua guida e il suo aiuto sono stati indispensabili. Desidero ringraziarlo per avermi trattato non come un semplice tesista ma come un amico.

Ringrazio i miei genitori Alba e Domenico e le nonne Antonietta e Lucia e tutta la mia famiglia per i sacrifici fatti per mantenermi e per l'immensa fiducia riposta nelle mie capacità.

Ringrazio Fabrizia per avermi sopportato in questi mesi, per essere stata così comprensiva e per tutte le emozioni che è in grado di regalarmi.

Infine desidero ringraziare tutti i miei amici per avermi supportato, per le risate che mi hanno fatto fare, e per avermi tirato sù di morale nelle giornate più buie.

Grazie a tutti di cuore.

*Milano, settembre 2014*



# INDICE

<b>INTRODUZIONE</b>	<b>1</b>
<b>1 STATO DELL'ARTE E SPECIFICHE TECNICHE</b>	<b>3</b>
1.1 I Robot a cinematica parallela . . . . .	3
1.2 Metodologie di progettazione . . . . .	5
1.3 PKM a 6-gdl . . . . .	6
1.4 Specifiche tecniche . . . . .	7
<b>2 CINEMATICA E CINETOSTATICA</b>	<b>11</b>
2.1 Presentazione delle macchine . . . . .	11
2.2 Cinematica inversa . . . . .	11
2.2.1 Piattaforma di Stewart . . . . .	12
2.2.2 Hexaglide ed Hexaslide . . . . .	13
2.3 Analisi di velocità . . . . .	15
2.3.1 Piattaforma di Stewart . . . . .	15
2.3.2 Hexaglide ed Hexaslide . . . . .	15
2.4 Cinetostatica . . . . .	16
2.5 Rigidezza cinematica . . . . .	17
2.6 Singolarità . . . . .	18
2.6.1 Singolarità del 1° tipo . . . . .	19
2.6.2 Singolarità del 2° tipo . . . . .	20
2.6.3 Singolarità del 3° tipo . . . . .	21
2.7 Ellissoidi di manipolabilità ed isotropia . . . . .	21
2.7.1 Ellissoidi di manipolabilità in velocità . . . . .	23
2.7.2 Ellissoidi di manipolabilità in forza . . . . .	24
2.8 Forze di attuazione massime . . . . .	26
<b>3 OTTIMIZZAZIONE CINETOSTATICA</b>	<b>29</b>
3.1 Spazio di lavoro . . . . .	29
3.2 Parametrizzazione delle macchine . . . . .	30
3.2.1 Piattaforma di Gough-Stewart . . . . .	30
3.2.2 Hexaslide . . . . .	30
3.2.3 Hexaglide . . . . .	31
3.3 Algoritmo genetico a singolo obiettivo . . . . .	31

---

3.3.1	Criteri di stop . . . . .	32
3.4	Algoritmo genetico multiobiettivo . . . . .	32
3.5	Funzioni di costo . . . . .	33
3.5.1	Copertura del volume di lavoro . . . . .	33
3.5.2	Copertura area di lavoro . . . . .	33
3.5.3	Ingombro longitudinale . . . . .	34
3.6	Vincoli . . . . .	34
3.7	Risultati dell'ottimizzazione . . . . .	35
3.7.1	Piattaforma di Gough-Stewart . . . . .	36
3.7.2	Hexaslide . . . . .	38
3.7.3	Hexaglide . . . . .	39
<b>4</b>	<b>ANALISI DI VIBRAZIONI</b>	<b>53</b>
4.1	Marker di riferimento e loro parametrizzazione . . . . .	53
4.2	Modellazione piattaforma . . . . .	54
4.3	Modellazione dei link . . . . .	56
4.3.1	Creazione segmenti . . . . .	56
4.3.2	Creazione campi di forze di tipo beam . . . . .	57
4.4	Slider . . . . .	58
4.5	Modellazione delle cinghie . . . . .	58
4.6	Modellazione vite senza fine . . . . .	60
4.7	Vincoli . . . . .	61
4.7.1	Giunti di piattaforma e di base . . . . .	61
4.7.2	Prismatici . . . . .	62
4.7.3	Accoppiamento vite-madrevite . . . . .	62
4.8	Analisi e generazione dei file di testo . . . . .	63
4.9	Risultati . . . . .	64
4.9.1	Hexaglide . . . . .	64
4.9.2	Hexaslide . . . . .	66
<b>5</b>	<b>CLEARANCE</b>	<b>71</b>
5.1	Clearance function . . . . .	71
5.2	Descrizione della piattaforma . . . . .	72
5.3	Descrizione link . . . . .	72
5.4	File generati . . . . .	73
5.5	Risultati . . . . .	74
<b>6</b>	<b>ANALISI DINAMICA E SCELTA DEL GRUPPO MOTORI- DUTTORE</b>	<b>77</b>
6.1	Modello Simmechanics . . . . .	77
6.2	Scelta del gruppo motore-riduttore . . . . .	80
6.3	Coppia resistente . . . . .	82
6.4	Risultati . . . . .	83
	<b>CONCLUSIONI</b>	<b>95</b>



---

<b>A</b>	<b>Matrici di Rotazione e Velocità Angolare</b>	<b>97</b>
A.1	Angoli Nautici XYZ . . . . .	97
A.2	Velocità angolare . . . . .	98
<b>B</b>	<b>Codici ADAMS relativi al calcolo delle frequenze proprie</b>	<b>101</b>
B.1	Creazione del modello e dei marker di riferimento . . . . .	101
B.2	Creazione piattaforma . . . . .	102
B.3	Creazione marker delle viti . . . . .	103
B.4	Creazione link flessibili . . . . .	104
B.5	Creazione slider . . . . .	105
B.6	Creazione Viti . . . . .	106
B.7	Creazione corpi fittizzi . . . . .	107
B.8	Vincoli . . . . .	108
B.9	Analisi . . . . .	111
<b>C</b>	<b>BOSCH CKK25-200</b>	<b>113</b>
<b>D</b>	<b>ROLLON ROBOT SP160</b>	<b>121</b>
<b>E</b>	<b>Dati tecnici motori</b>	<b>125</b>
	<b>BIBLIOGRAFIA</b>	<b>139</b>



## ELENCO DELLE FIGURE

1.1	Robot a cinematica parella industriali: i più diffusi. . . . .	3
1.2	<i>fully parallel manipulators</i> a 6-gdl. . . . .	6
1.3	Sistema di movimentazione dell'Hobby-Eberly Telescope (HET, Texas). . . . .	7
1.4	Toyota-Lexus drive simulator. . . . .	7
1.5	Spaccato della galleria del vento del Politecnico di Milano. . . . .	8
1.6	Attuale set-up sperimentale delle barche a vela e dei generatori eolici nella Galleria delvento del Politecnico di Milano. . . . .	9
2.1	Diagramma dell' $i$ -esima catena cinematica per la piattaforme di Gough-Stewart. . . . .	12
2.2	Diagramma dell' $i$ -esima catena cinematica per le due macchine. . .	14
2.3	Rappresentazione di una singolarità di 1° tipo per un robot scara ed un robot parallelo a 2-gdl. . . . .	20
2.4	Vettore di Plücker. . . . .	21
2.5	Rappresentazione di una singolarità di 2° tipo per un robot planare a 2-gdl. . . . .	22
2.6	Rappresentazione di una singolarità di 3° tipo per un robot planare a 2-gdl. . . . .	22
2.7	Ellissoide di manipolabilità in velocità. . . . .	23
2.8	Poliedro di manipolabilità in velocità. . . . .	24
2.9	Ellissoide di manipolabilità in forza. . . . .	25
2.10	Poliedro di manipolabilità in forza. . . . .	26
3.1	Struttura finale della piattaforma di Gough-Stewart. . . . .	37
3.2	Sezionamento dello spazio di lavoro con piani verticali. . . . .	38
3.3	Volume ad orientazione totale raggiungibile dalla piattaforma di Gough-Stewart. . . . .	40
3.4	Piattaforma di Gough-Stewart: distanza fra i link per vari piani, caso barche. . . . .	41
3.5	Piattaforma di Gough-Stewart: distanza fra i link per vari piani, caso turbine. . . . .	41
3.6	Piattaforma di Gough-Stewart: distanza link-piattaforma, caso barche. . . . .	42

3.7	Piattaforma di Gough-Stewart: distanza link-piattaforma, caso turbine. . . . .	42
3.8	Piattaforma di Gough-Stewart: moltiplica delle forze, caso barche. . . . .	43
3.9	Piattaforma di Gough-Stewart: moltiplica delle forze, caso turbine. . . . .	43
3.10	Volume di lavoro non coperto al variare di $\tau$ per Hexaslide e piattaforma di Stewart . . . . .	44
3.11	Struttura finale dell'Hexaslide . . . . .	44
3.12	Volume ad orientazione totale raggiungibile dall'Hexaslide. . . . .	45
3.13	Distanza fra i link per vari piani, caso barche. . . . .	46
3.14	Distanza fra i link per vari piani, caso turbine. . . . .	46
3.15	Distanza fra i link e la piattaforma per vari piani, caso barche. . . . .	47
3.16	Distanza fra i link e la piattaforma per vari piani, caso turbine. . . . .	47
3.17	Moltiplica statica delle forze per vari piani, caso barche. . . . .	48
3.18	Moltiplica statica delle forze per vari piani, caso turbine. . . . .	48
3.19	Azioni interne dei link per macchine ottimizzate con diversi valori di $\tau_{max}$ . . . . .	49
3.20	Frontiera di Pareto per il caso $\tau_{max} = 5$ con $L_c$ . . . . .	49
3.21	Volume ad orientazione totale raggiungibile dall'Hexaslide. . . . .	50
3.22	Distanza minima fra i link. . . . .	51
3.23	Distanza minima fra i link. . . . .	51
3.24	Distanza minima fra i link. . . . .	51
4.1	Disegno della piattaforma dell'Hexaglide. . . . .	55
4.2	Disegno della piattaforma dell'Hexaslide. . . . .	56
4.3	Disegno dei link. . . . .	57
4.4	Disegno delle viti senza fine. . . . .	60
4.5	Vista dall'alto, frontale e laterale in ADAMS dell'Hexaglide. . . . .	65
4.6	Mappa delle frequenze minime dell'Hexaglide sul piano $yz$ nel caso con trasmissione infinitamente rigida. . . . .	65
4.7	Mappa delle frequenze minime dell'Hexaglide sul piano $yz$ nel caso con trasmissione a cinghia. . . . .	66
4.8	Mappa delle frequenze minime dell'Hexaglide sul piano $yz$ nel caso con trasmissione a vite. . . . .	66
4.9	Vista dall'alto, frontale e laterale in ADAMS dell'Hexaslide. . . . .	67
4.10	Mappa delle frequenze minime dell'Hexaslide su vari piani nel caso con trasmissione a vite. . . . .	68
4.11	Mappa delle frequenze minime dell'Hexaslide con trasmissione a cinghia su vari piani. . . . .	69
4.12	Mappa delle frequenze minime dell'Hexaslide con trasmissione a vite su vari piani. . . . .	70
5.1	Disegno costruttivo della piattaforma. . . . .	73
5.2	Mappe di distanza minima fra i link 2 e 5 e la piattaforma. . . . .	74
5.3	Compenetrazione link-piattaforma per una particolare posa. . . . .	74
5.4	Mappe della distanza fra i link con piattaforma allargata. . . . .	75

---

5.5	Mappe della distanza fra i link e la piattaforma con piattaforma allargata. . . . .	76
6.1	Schema turbina. . . . .	78
6.2	Schema barca. . . . .	78
6.3	Confronto fra i vari valori di azione assiale lungo i link per varie movimentazioni. . . . .	85
6.4	Forze massime in direzione assiale lungo i link. . . . .	85
6.5	Forze massime lungo gli assi delle guide. . . . .	86
6.6	Confronto fra i vari tipi di movimentazione in termini di massima coppia resistente e massimo fattore di carico. . . . .	87
6.7	Massimo fattore di carico per entrambi i tipi di trasmissione. . . . .	90
6.8	$\alpha$ vs. $\beta$ nel caso con trasmissione a vite. . . . .	91
6.9	$\alpha$ vs. $\beta$ nel caso con trasmissione a cinghia. . . . .	92
6.10	Rapporto di trasmissione per vari motori nel caso con trasmissione a vite. . . . .	93
6.11	Rapporto di trasmissione per vari motori nel caso con trasmissione a cinghia. . . . .	94



## ELENCO DELLE TAVOLE

1.1	Specifiche tecniche iniziali. . . . .	9
1.2	Nuove specifiche tecniche. . . . .	10
2.1	Grandezze utilizzate nel problema cineto-statico. . . . .	12
2.2	Componenti dei versori delle guide dell'Hexaslide espresse nel s.d.r. globale. . . . .	13
2.3	Tre tipologie di singolarità. . . . .	19
3.1	Parametri utilizzati per l'ottimizzazione dell'Hexaglide. . . . .	31
3.2	Range di variazione dei parametri della piattaforma di Gough-Stewart. . . . .	36
3.3	Range di variazione dei parametri dell'Hexaslide. . . . .	38
4.1	Dati relativi al materiale dei link. . . . .	58
6.1	Posizioni di partenza attorno alle quali si sviluppano i moti sinusoidali. . . . .	79
6.2	Ampiezza delle sinusoidi. . . . .	79
6.3	Dati delle inerzie dei due tipi di trasmissione. . . . .	83
6.4	Selezione unità di trasmissione: prestazioni fornite vs. richieste. . . . .	84
6.5	Contributo inerziale dato dall'unità di trasmissione, ridotto alla slitta. $F_{1g}$ è la forza per accelerare $m_{eq}$ di 1g. $\max F_{A_i,x} = 1388\text{N}$ è la forza di traslazione massima richiesta nel caso di "barca a vela" per un'accelerazione $\ddot{q}_i \approx 1\text{g}$ . . . . .	86
6.6	Valori di fattore di carico, coppia resistente ed accelerazione angolare per le due trasmissioni. . . . .	88
6.7	Verifica coppia motrice massima: richiesta vs. disponibile, nel caso di movimento lungo $z$ . . . . .	88
6.8	Verifica coppia motrice massima: richiesta vs. disponibile, nei casi di: rotazione di beccheggio, movimento lungo $z$ . . . . .	89





## SOMMARIO

Negli ultimi anni lo sviluppo delle simulazioni CFD ha incrementato la conoscenza dei problemi di interazione fluido-struttura. Questo trend è stato particolarmente importante per settori come le turbine eoliche off-shore galleggianti e le barche a vela, in cui sono coinvolti due fluidi. Ciononostante l'affidabilità dei software CFD richiede ulteriori validazioni sperimentali. A tal fine come approccio complementare a quello delle vasche navali, c'è l'esigenza di effettuare test aeroelastici di tipo dinamico in galleria del vento.

In questo lavoro di tesi è stato affrontato il progetto di una macchina a cinematica parallela che emulasse l'azione del mare: l'architettura della macchina è stata scelta in base alle specifiche fornite dalla galleria del vento; l'ottimizzazione cinetostatica è stata effettuata con degli algoritmi di tipo genetico in ambiente MATLAB; per l'analisi dinamica in piccolo ed il calcolo della distanza minima fra i componenti della macchina è stato costruito un modello multibody in ADAMS; il dimensionamento del gruppo motore-riduttore è stato condotto a valle di una simulazione dinamica del moto in grande in Simulink, considerando le condizioni operative più gravose.



## INTRODUZIONE

Al giorno d'oggi si dà sempre maggiore importanza agli studi sull'interazione fluido-struttura, al fine di ottenere prestazioni sempre più elevate. Per prevedere quale sarà il comportamento in condizioni operative di una struttura, vengono effettuati degli esperimenti in galleria del vento su modelli in scala. I test in galleria vengono utilizzati per validare ed implementare i modelli matematici presenti all'interno dei software per analisi di tipo CFD, e in alcuni casi per ottenere risultati in tempi decisamente inferiori. Questa tesi nasce dall'esigenza di simulare nella camera di prova di tipo civile della galleria del vento del Politecnico di Milano il comportamento di barche a vela e turbine eoliche off-shore aggiungendo all'interazione vento-struttura il contributo idrodinamico fornito dalle onde marine. A questo scopo si è resa necessaria la progettazione di una macchina che emulasse l'azione del mare a fronte di una modellazione matematica del moto ondoso. In questo lavoro vengono presentate le metodologie utilizzate per la sintesi cinetostatica e per l'analisi dinamica di alcune soluzioni costruttive a cinematica parallela.

La tesi è strutturata nei seguenti sei capitoli.

Nel Cap. 1 viene fornita una panoramica generale sui robot a cinematica parallela, indicando quali siano i pregi e i difetti di queste macchine rispetto ai più diffusi robot a cinematica seriale. Ci si focalizza poi sulla categoria di macchine in cui i gradi di libertà dell'elemento terminale sono pari al numero di attuatori. Infine vengono definite le specifiche tecniche fornite dai ricercatori della galleria del vento, a fronte delle quali sono stati scelti determinati tipi di architettura.

Si passa poi al Cap. 2 dove vengono illustrati gli strumenti matematici usati per calcolare la cinematica inversa e per effettuare l'analisi di velocità. Viene poi descritto come le forze agenti sull'elemento terminale del robot si trasmettano lungo i vari elementi che compongono la macchina. Dallo studio della matrice Jacobiana poi vengono definite le pose del robot per cui questo si viene a trovare in una condizione di singolarità, ed in che modo si trasformino le velocità e le forze di attuazione in corrispondenza dell'elemento terminale.

Nel Cap. 3 si affronta l'argomento relativo all'ottimizzazione cinetostatica delle macchine in esame. Dopo una descrizione dei parametri usati per definire le tre macchine scelte, si passa alla descrizione degli algoritmi utilizzati, delle funzioni di costo usate per quantificare la bontà di una soluzione rispetto ad un'altra, e dei vincoli imposti, illustrando infine i risultati ottenuti.

È necessario poi verificare che le azioni a cui sono soggette le macchine non in-

neschino fenomeni vibratori che andrebbero ad influire sul corretto funzionamento. Per cui nel Cap. 4 viene descritto in che modo è stato creato in ambiente ADAMS il modello multibody usato per l'analisi dinamica in piccolo. Viene mostrato in che modo sono stati creati i corpi, sia quelli rigidi che quelli flessibili, come sono stati modellati i vincoli ed i sistemi di trasmissione. Vengono poi messi a confronto i risultati delle analisi delle varie macchine in modo tale da avere un ulteriore elemento di scelta.

Ai fini di evitare collisioni, una volta definito il disegno finale di tutti i componenti della macchina sarà necessario verificare che la distanza fra i vari elementi rispetti determinati limiti di sicurezza. In particolare si vuole verificare che i link non vadano a collidere con il corpo centrale della piattaforma. A tale scopo nel Cap. 5 viene descritto il modello multibody rigido della macchina e le funzioni di ADAMS usate per calcolare la distanza minima fra i corpi.

L'argomento trattato nell'ultimo capitolo (Cap. 6) riguarda la scelta del gruppo motore-riduttore. Una volta calcolati i requisiti in termini di coppia ed accelerazioni richiesti agli attuatori, si descrive la metodologia usata per la scelta da catalogo dei motori e del rapporto di trasmissione della macchina finale.

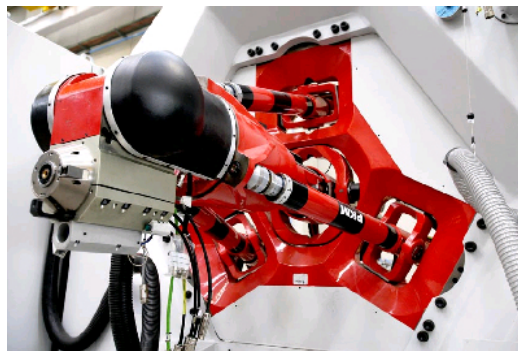
## STATO DELL'ARTE E SPECIFICHE TECNICHE

### 1.1 I Robot a cinematica parallela

I robot a cinematica parallela, o PKM (*parallel kinematics machines*), sono costituiti, come definito da Merlet in [4], da una base fissa e da un end-effector con  $n$  gradi di libertà ,collegati da almeno due catene cinematiche indipendenti. Nel caso in cui le catene cinematiche siano esattamente  $n$  allora si parla di *fully parallel manipulators*. Queste macchine vengono utilizzate per applicazioni specifiche e di fatto risultano molto meno diffuse all'interno del mercato mondiale rispetto ai robot seriali. La progettazione infatti risulta molto più onerosa e specifica per ogni singolo caso. Esistono però delle eccezioni, infatti il robot *Delta* (Fig. 1.1(a)), ed in maniera minore il *Tricept* (Fig. 1.1(b)), sono riusciti a ritagliarsi una fetta di mercato per applicazioni di tipo pick-and-place.



(a) fanucM3iA



(b) Tricept

**Fig. 1.1:** Robot a cinematica parallela industriali: i più diffusi.

Rispetto ai robot seriali le PKM presentano degli aspetti vantaggiosi:

- *maggiore accuratezza*: nei robot di tipo seriale gli errori di posizionamento del TCP sono frutto della somma degli errori e dei giochi di ogni singolo giunto a monte, mentre nelle PKM il meccanismo con cui i giochi interni e

gli errori dei giunti influenzano la bontà del posizionamento del TCP è più complicata. Da un punto di vista questo permette alle PKM di avere una maggiore affidabilità, mentre dall'altro, a causa della cinematica complessa, gli errori di un singolo giunto si ripercuoteranno su ciascuno dei gradi di libertà del TCP. Per queste ragioni la fase di settaggio iniziale e di calibrazione dovrà essere particolarmente accurata, così come l'algoritmo di controllo necessario a compensare eventuali errori in fase di funzionamento.

- *maggiore rigidità e capacità di sopportare carichi*: in generale i link delle macchine a cinematica parallela sono identici fra loro, o comunque del tutto simili, per cui le forze che agiscono in corrispondenza del TCP tendono a ripartirsi in maniera equa lungo tutti i link. Di solito i link presentano una struttura affusolata e, per come sono progettate le macchine, le forze che si scaricano su di essi tendono ad essere prevalentemente assiali e quindi agenti lungo la direzione di massima rigidità col risultato di avere piccole deformazioni. Nel caso seriale invece le forze agenti sul TCP tendono a fare inflettere i link, e per ridurre le deformazioni, che si traducono in errori di posizionamento, è necessario ricorrere a corpi più tozzi che tuttavia vanno a gravare sui giunti più prossimi alla base fissa, in quanto nei robot seriali su ogni giunto si ripercuotono le azioni inerziali di tutti i link che si trovano a valle.
- *prestazioni dinamiche elevate*: essendo gli attuatori posizionati in prossimità della base o addirittura a terra, il carico inerziale che grava su di essi è inferiore rispetto al caso dei robot seriali, così come la loro taglia, e di conseguenza il costo.
- *modularità*: grazie alla struttura modulare dei componenti, si può pensare ad una riconfigurazione della macchina per poter svolgere più tipi di attività, tuttavia i tempi e le difficoltà di progetto, come la sintesi degli algoritmi di controllo e la calibrazione si presentano come un forte limite.

Se da un lato le PKM possiedono delle caratteristiche che le rendono più indicate per determinate applicazioni rispetto ai robot seriali, dall'altro lato presentano delle limitazioni:

- *spazio di lavoro ridotto*: lo spazio di lavoro che la macchina riesce a coprire a parità di volume occupato risulta molto più ridotto rispetto al caso dei robot seriali, a causa dei limiti imposti dai range di mobilità dei giunti, dall'interferenza fra gli elementi della macchina e della presenza di configurazioni singolari non prevedibili.
- *elevata anisotropia*: le PKM sono fortemente anisotrope, cioè la loro rigidità ed il loro rapporto di trasmissione generalizzato di forze e velocità varia a seconda della posa assunta dalla macchina. L'isotropia di queste macchine è oggetto di studio di molti ricercatori. Le principali problematiche nel definire

un indice che caratterizzi il grado di isotropia di una macchina nascono dalle difficoltà nel mettere in relazioni i gradi di libertà traslazionali con quelli rotazionali.

- *criticità dei giunti passivi*: con il termine giunti passivi si indicano quei tipi di giunto che non sono direttamente attuati. I più comuni sono: i giunti che consentono un unico grado di libertà, ad esempio i giunti prismatici e le cerniere; i giunti che consentono due o più gradi di libertà, come ad esempio i giunti cardanici e quelli sferici. Il range di mobilità di questi giunti influenza direttamente le dimensioni dello spazio di lavoro raggiungibile dal TCP con qualsiasi orientazione. Altro aspetto critico è la capacità di carico che sono in grado di sopportare, così come l'entità dei giochi interni che influenza direttamente la precisione di posizionamento del TCP.

Molto spesso si cerca di sfruttare i pregi di entrambi i tipi di robot realizzando delle macchine ibride. Ad esempio, così come accade nel robot *Delta* e nel *Tricept*, sulla piattaforma mobile del manipolatore a cinematica parallela vengono posizionati degli ulteriori elementi in serie al fine di garantire una maggiore libertà nell'orientazione della parte terminale.

## 1.2 Metodologie di progettazione

È molto difficile trovare un indice che caratterizzi le prestazioni di una macchina a cinematica parallela. Non è possibile mettere a confronto due macchine caratterizzate da architetture diverse. Questo aspetto viene messo in evidenza da Merlet [4], e da M. Weck e D. Staimer in [3], dove sottolineano che gli step più importanti della fase di progettazione di una macchina a cinematica parallela siano: la scelta di una topologia cinematica appropriata; ma soprattutto l'ottimizzazione dei parametri geometrici che caratterizzano quella determinata topologia. Per il primo dei due step si può ricorrere all'esperienza, scegliendo una delle architetture fra quelle già note in base a quelle che sono le proprie esigenze e ad i pregi e difetti che ogni tipologia di macchina presenta rispetto alle altre. Alcuni ricercatori sono all'opera per cercare di creare dei metodi di sintesi di carattere universale che abbiano l'obiettivo di indirizzare il progettista verso la soluzione architeturale più consona. In prima analisi, per il calcolo degli  $M$  gradi di libertà di una macchina, si ricorre alla formula di Grubler:

$$M = b(n - g - 1) + \sum_{i=1}^g f_i \quad (1.1)$$

dove  $b$ , detto *mobility number*, assumerà valore pari a 3 o 6, a seconda che il TCP debba muoversi in piano o nello spazio. Con  $n$  è stato indicato il numero di corpi rigidi che compongono la macchina contando anche la base, con  $g$  si è indicato il numero di giunti e con  $f_i$  i gradi di libertà concessi da ciascun vincolo. Questa formula non ha carattere generale, in quanto non tiene conto di come

sono posizionati gli assi dei giunti. Le ipotesi di validità di questa formula sono descritte in [10]. Passando al secondo step, esistono vari metodi per determinare i parametri caratteristici della macchina. In questo lavoro di tesi si è utilizzato un tipo di ottimizzazione a singolo obiettivo con funzione di costo.

### 1.3 PKM a 6-gdl

Le PKM a sei gradi di libertà di tipo *fully parallel manipulators* possono essere ricondotti a tre grandi famiglie così come indicato da I. Bonev in [8]. Questo tipo di macchine presenta sei catene cinematiche strutturate tutte in maniera identica. Ciascuna famiglia si distingue dalle altre per il modo in cui i giunti, attivi e passivi, si susseguono all'interno di una singola catena cinematica. Le tre famiglie, raffigurate in Fig. 1.2, sono:  $6 - \underline{RRPS}$ ,  $6 - \underline{RRRS}$ , e  $6 - \underline{PRRS}$ , dove  $R$  indica un giunto rotativo,  $P$  uno prismatico, e  $S$  uno sferico, mentre la lettera sottolineata indica quale dei giunti sia quello attivo.

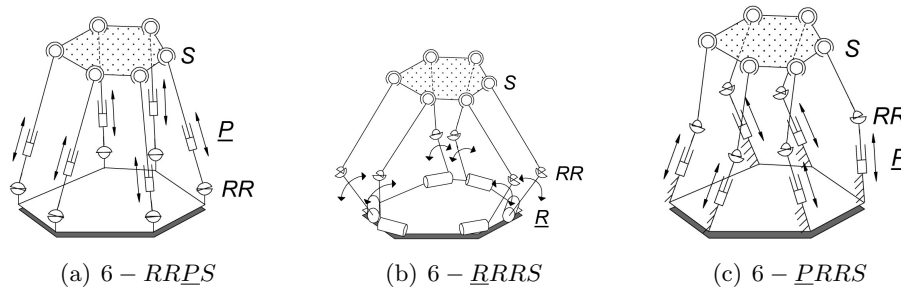


Fig. 1.2: *fully parallel manipulators* a 6-gdl.

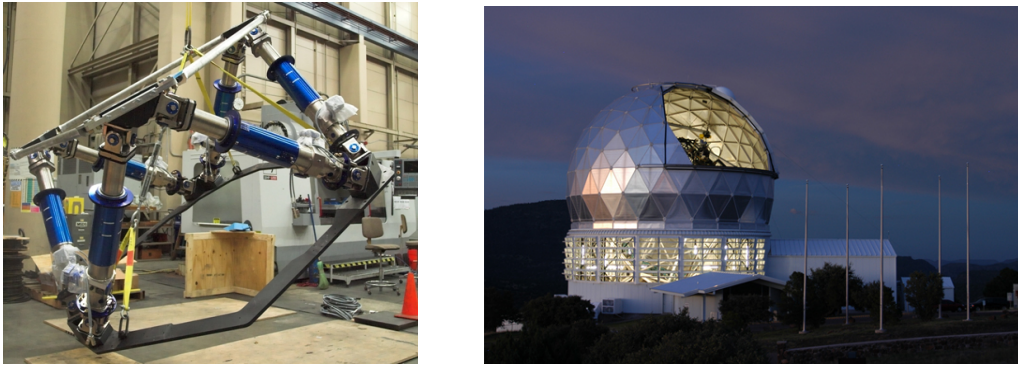
Applicando la formula di Grubler si nota che tutte e tre le famiglie presentano lo stesso numero di gradi di libertà:

$$M = b(n - g - 1) + \sum_{i=1}^g f_i = 6(14 - 18 - 1) + 36 = 6 \quad (1.2)$$

Alcuni esempi di *fully parallel manipulators* sono:

- $6 - \underline{RRPS}$ : la macchina a cinematica parallela più conosciuta è sicuramente quella di tipo  $6 - \underline{RRPS}$ . Questo tipo di robot è stato realizzato per la prima volta da Gough nel 1947 e poi riproposta nel '65 da Stewart. La prima versione non era un vero e proprio robot, in quanto i link venivano azionati manualmente tramite martinetti idraulici. Grazie alla sua elevata precisione di posizionamento ed orientazione questa macchina viene utilizzata in campo astronomico per il posizionamento di telescopi di grosse dimensioni (Fig. 1.3). Altro ambito in cui questo tipo di architettura viene utilizzato è quello dei simulatori di volo (Fig. 1.4).





**Fig. 1.3:** Sistema di movimentazione dell'Hobby-Eberly Telescope (HET, Texas).



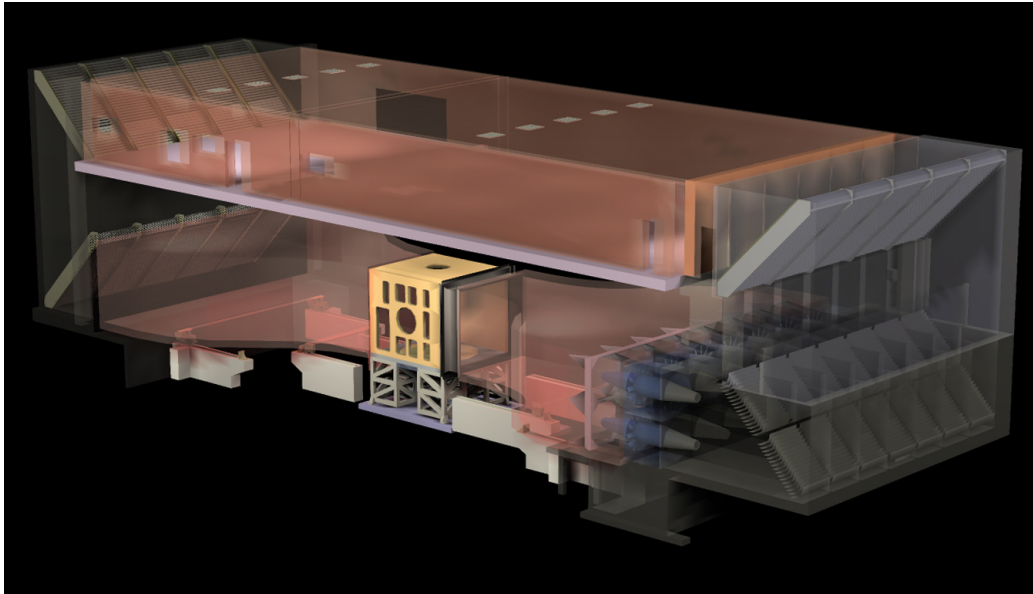
**Fig. 1.4:** Toyota-Lexus drive simulator.

- $6-RRRS$ : vanno citati il robot HEXA dell'università giapponese di Tohoku, che è in grado di fornire accelerazioni molto elevate a scapito di una scarsa capacità di carico e di una bassa rigidezza per il tipo di sollecitazione a cui sono sottoposti i link, e la Rotary-series della Mikrolar, che non presenta problemi relativi alla rigidezza, pur avendo prestazioni dinamiche inferiori rispetto al robot HEXA.
- $6-PRRS$ : fanno parte di questa categoria l'Hexaglide [5] con guide parallele, realizzato dall'ETH di Zurigo [6,7], e l'Hexaslide [9], in cui le guide rispettano una simmetria di tipo radiale. Entrambe le macchine si presentano come molto rigide in quanto i link vengono caricati solo assialmente, ed inoltre i motori sono a terra e di conseguenza il carico inerziale risulta ridotto.

## 1.4 Specifiche tecniche

Il compito che la macchina che si vuole progettare, dovrà assolvere è quello di simulare in galleria del vento l'azione delle onde marine su modelli in scala di generatori eolici off-shore e barche a vela, al fine di emulare nel modo più realistico

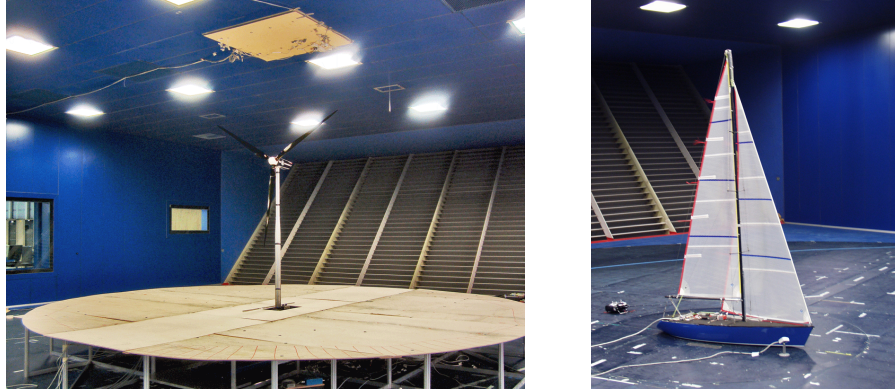
possibile le interazioni fra entrambi i fluidi (vento e mare) e la struttura [1, 2]. Il robot dovrà essere in grado di riprodurre le manovre di una barca a vela reale e vari tipi di ormeggio per la turbina eolica. La macchina dovrà quindi avere delle prestazioni dinamiche elevate e ricreare l'azione del mare grazie ad opportuni modelli matematici del moto ondoso in varie condizioni (mare calmo, mosso, etc.). Le azioni scambiate fra i modelli e la piattaforma vengono misurate tramite una bilancia a sei assi e permettono di risalire alla distribuzione delle forze fluidodinamiche agenti sui modelli. Come si può vedere dalla Fig. 1.5 la galleria del vento del Politecnico è provvista di due camere di prova: quella inferiore realizzata per applicazioni aerospaziali, e quella superiore ricavata nel circuito di ritorno per applicazioni civili ed ambientali.



**Fig. 1.5:** Spaccato della galleria del vento del Politecnico di Milano.

Una delle richieste che la macchina dovrà soddisfare è quella di avere un ingombro verticale ridotto: si vuole infatti evitare che la vena fluida investa il robot perchè questo causerebbe un'alterazione del profilo del vento. Inoltre la necessità di ridurre l'ingombro verticale è dettata dal fatto che in prossimità del soffitto il profilo della vena fluida è disturbato, e quindi è necessario che l'estremo superiore dei modelli di galleria si trovi sempre al di sotto da questa zona. Gli attuali set-up sperimentali sono raffigurati in Fig. 1.6.

Dopo vari studi sulla scalatura dei modelli aerodinamici e sui movimenti al vero di barche e turbine eoliche, c'è stata una ridefinizione delle specifiche di movimento che ha portato a prendere in esame altri tipi di architetture oltre a quelle che erano state prese in considerazione inizialmente. In Tab. 1.1 sono elencate le specifiche iniziali. Si può notare come lo spazio di lavoro iniziale presenti una direzione predominante lungo il verso di provenienza del vento, questa considerazione ha portato a scegliere fra le varie candidate un'architettura di tipo Hexaglide in



**Fig. 1.6:** Attuale set-up sperimentale delle barche a vela e dei generatori eolici nella Galleria delvento del Politecnico di Milano.

cui l'unico limite imposto agli spostamenti in direzione  $x$  è dettato dalla lunghezza delle unità lineari. Le specifiche richieste nel caso statico, in particolare quelle riguardanti l'imbarcata sono molto difficili da realizzare con gli ingombri a disposizione, si è quindi pensato di aggiungere un ulteriore grado di libertà montando sulla piattaforma mobile una piastra rotante.

	Turbina	Barca	
	Dinamica	Dinamica	Statica
$x$ surge	$\pm 500 \text{ mm}$	—	—
$y$ sway	$\pm 300 \text{ mm}$	—	—
$z$ heave	$\pm 250 \text{ mm}$	—	—
$\alpha$ roll	$\pm 15^\circ$	$\pm 10^\circ$	$\pm 30^\circ$
$\beta$ pitch	$\pm 15^\circ$	$\pm 10^\circ$	$\pm 0^\circ$
$\gamma$ yaw	$\pm 15^\circ$	$\pm 10^\circ$	$\pm 60^\circ$
freq. max	$0.7 \text{ Hz}$	$1.5 \text{ Hz}$	—

**Tab. 1.1:** Specifiche tecniche iniziali.

A seguito della ridefinizione le nuove specifiche sono diventate quelle indicate in Tab. 1.2.

Si nota subito come le dimensioni dello spazio di lavoro si siano ridotte e come si siano avvicinate ad una condizione di simmetria. A questo punto ci si è chiesti se fosse possibile considerare anche altri tipi di architettura, si è deciso quindi di prendere in esame e di analizzare due nuove macchine, la piattaforma di Gough-Stewart e l'Hexaslide, che non presentano una direzione privilegiata di movimento, ma che invece possiedono una simmetria di comportamento in direzione radiale. Sarà poi necessario prevedere due tipi di montaggio dato che le posizioni del TCP per i modelli di turbina eolica e barca a vela è diventata diversa.

	Turbina	Barca	
	Dinamica	Dinamica	Statica
$x$ surge	$\pm 120 \text{ mm}$	$\pm 100 \text{ mm}$	–
$y$ sway	$\pm 70 \text{ mm}$	$\pm 70 \text{ mm}$	–
$z$ heave	$\pm 100 \text{ mm}$	$\pm 150 \text{ mm}$	–
$\alpha$ roll	$\pm 5^\circ$	$\pm 5^\circ$	$\pm 30^\circ$
$\beta$ pitch	$\pm 16^\circ$	$\pm 15^\circ$	$\pm 0^\circ$
$\gamma$ yaw	$\pm 12^\circ$	$\pm 5^\circ$	$\pm 60^\circ$
freq. max	$0.7 \text{ Hz}$	$1.2 \text{ Hz}$	–

**Tab. 1.2:** Nuove specifiche tecniche.

## CAPITOLO 2

# CINEMATICA E CINETOSTATICA

## 2.1 Presentazione delle macchine

Le macchine analizzate appartengono alla famiglia degli esapodi. La differenza sostanziale si trova nel tipo di attuazione: mentre la piattaforma di Stewart è provvista di sei attuatori che costituiscono i link di lunghezza variabile, per quanto concerne l'Hexaslide e l'Hexaglide, i cui link sono di lunghezza fissa, l'attuazione è ottenuta grazie allo scorrimento di sei slider su altrettanti binari complanari, che nel caso dell'Hexaglide saranno tutti paralleli e diretti lungo l'asse  $x$ , mentre nell'Hexaslide saranno paralleli a coppie di due, e ciascuna coppia sarà sfasata di un angolo pari a  $120^\circ$  rispetto alle altre.

L'accoppiamento fra link e slider (o fra link e terreno nel caso dell'esapode di Stewart), e quello fra link e piattaforma può essere ottenuto tramite giunti di tipo sferico od universale, ciascuno dei quali presenta pregi e difetti che verranno illustrati in seguito.

Lo scorrimento degli slider lungo i binari costituisce di fatto un giunto prismatico attuato, in quanto la posa del robot viene modificata andando ad agire sull'avanzamento degli slider.

## 2.2 Cinematica inversa

Per poter muovere il robot nelle pose desiderate è di fondamentale importanza stabilire quali siano le relazioni che legano le coordinate del TCP (*tool center point*) a quelle dello spazio dei giunti, e cioè alla posizione degli slider sui binari. In generale per i robot a cinematica parallela risulta piú semplice risolvere la cinematica inversa, e cioè, nota la posa desiderata del robot, risalire al valore che devono assumere le coordinate di giunto per portarlo in tale configurazione. Al fine di risolvere la cinematica inversa si utilizzeranno due sistemi di riferimento, uno globale, i cui versori degli assi verranno denotati coi simboli  $\hat{x}$ ,  $\hat{y}$ ,  $\hat{z}$ , ed uno relativo, con origine nel TCP e solidale con la piattaforma, i cui versori degli assi verranno indicati coi simboli  $\hat{x}'$ ,  $\hat{y}'$ ,  $\hat{z}'$ .

Simbologia	Significato
$O-xyz$	Sistema di riferimento globale
$\hat{x}, \hat{y}, \hat{z}$	versori del sistema di riferimento globale
$C-xyz$	Sistema di riferimento locale, solidale con la piattaforma mobile
$\hat{x}', \hat{y}', \hat{z}'$	versori del sistema di riferimento locale
$C$	tool center point
$A_i$	$i$ -esimo centro giunto di base
$B_i$	$i$ -esimo centro giunto di piattaforma
$\mathbf{p} = \{x_C, y_C, z_C\}^T$	vettore coordinate TCP nel s.d.r. globale
$\Theta = \{\alpha, \beta, \gamma\}^T$	vettore orientazione piattaforma
$\mathbf{X} = \{\mathbf{p}; \Theta\}$	vettore posa del robot
$[R] = [R(\Theta)]$	matrice di rotazione (s.d.r. locale $\rightarrow$ globale)
$\mathbf{b}'_i$	vettore coordinate $i$ -esimo giunto di piattaforma nel s.d.r. locale
$\mathbf{a}_i = \{a_{i,x}, a_{i,y}, 0\}^T$	vettore coordinate $i$ -esimo giunto di base (piattaforma di Stewart)
$\mathbf{l}_i = l_i \hat{\mathbf{n}}$	vettore lunghezza $i$ -esimo link (modulo e versore)
$\mathbf{s}_i = \{s_{i,x}, s_{i,y}, 0\}^T$	vettore posizione $i$ -esima guida (Hexaglide ed Hexaslide)
$q_i$	coordinata dell' $i$ -esimo giunto
$\hat{\mathbf{u}}_i$	versore orientazione $i$ -esima guida espresso nel s.d.r. globale
$[J]$	matrice jacobiana (cinematica diretta)
$[J]^{-1}$	matrice jacobiana inversa (cinematica inversa)

Tab. 2.1: Grandezze utilizzate nel problema cineto-statico.

### 2.2.1 Piattaforma di Stewart

La strategia adottata consiste nel chiudere una catena cinematica per determinare la posizione dei giunti di piattaforma rispetto a quelli di base per risalire alla lunghezza dei singoli link.

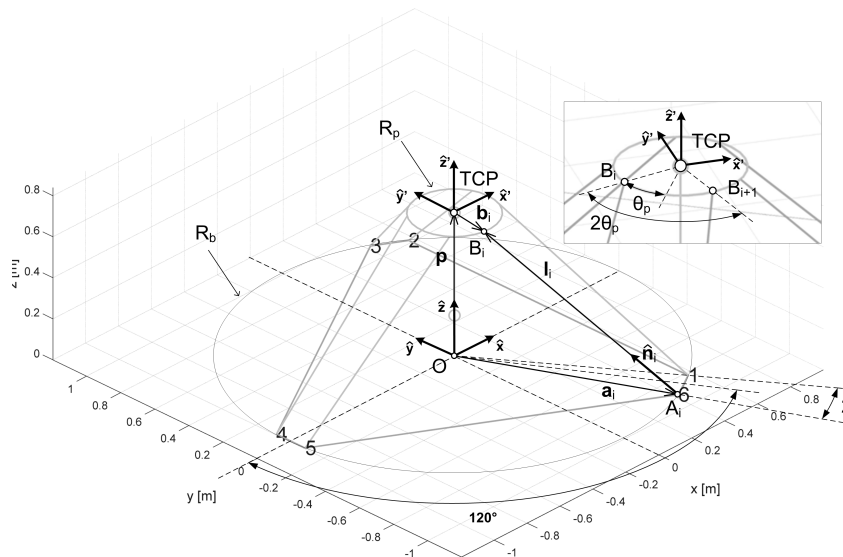


Fig. 2.1: Diagramma dell' $i$ -esima catena cinematica per la piattaforma di Gough-Stewart.

Con riferimento alla Tab. 2.1 e alla Fig. 2.1, per l' $i$ -esima catena cinematica si può scrivere:

$$\mathbf{l}_i = \mathbf{p} + [R]\mathbf{b}'_i - \mathbf{a}_i \quad (2.1)$$

e quindi, elevando al quadrato

$$\begin{aligned} l_i^2 &= \mathbf{l}_i^T \mathbf{l}_i = (\mathbf{p} + [R]\mathbf{b}'_i - \mathbf{a}_i)^T (\mathbf{p} + [R]\mathbf{b}'_i - \mathbf{a}_i) = \\ &= \mathbf{p}^T \mathbf{p} + \mathbf{b}'_i{}^T \mathbf{b}'_i + \mathbf{a}_i^T \mathbf{a}_i + 2\mathbf{p}^T [R]\mathbf{b}'_i - 2\mathbf{p}^T \mathbf{a}_i - 2\mathbf{a}_i^T [R]\mathbf{b}'_i \end{aligned} \quad (2.2)$$

Si nota come la lunghezza di ciascun link dipenda dall'intero vettore posa del robot, in quanto la matrice di rotazione  $[R]$  dipende dal vettore orientazione  $\Theta$ .

### 2.2.2 Hexaglide ed Hexaslide

Nel caso dell'Hexaglide e dell'Hexaslide si adatterà una metodologia del tutto simile a quella utilizzata per la piattaforma di Stewart; in questo caso però, sarà necessario chiudere due catene cinematiche, una per stabilire la posizione dei giunti di piattaforma rispetto alle guide, e l'altra per determinare la posizione degli slider sulle guide. Con riferimento alla Fig. 2.2, per l' $i$ -esima guida lineare si potrà scrivere  $\mathbf{l}_i = \mathbf{d}_i - q_i \hat{\mathbf{u}}_i$ , e quindi elevando al quadrato anche in questo caso, si ottiene:

$$l_i^2 = \mathbf{l}_i^T \mathbf{l}_i = (\mathbf{d}_i - q_i \hat{\mathbf{u}}_i)^T (\mathbf{d}_i - q_i \hat{\mathbf{u}}_i) = \mathbf{d}_i^T \mathbf{d}_i - 2\mathbf{d}_i^T \hat{\mathbf{u}}_i q_i + q_i^2 \quad (2.3)$$

dove

$$\mathbf{d}_i = \mathbf{p} + [R]\mathbf{b}'_i - \mathbf{s}_i \quad (2.4)$$

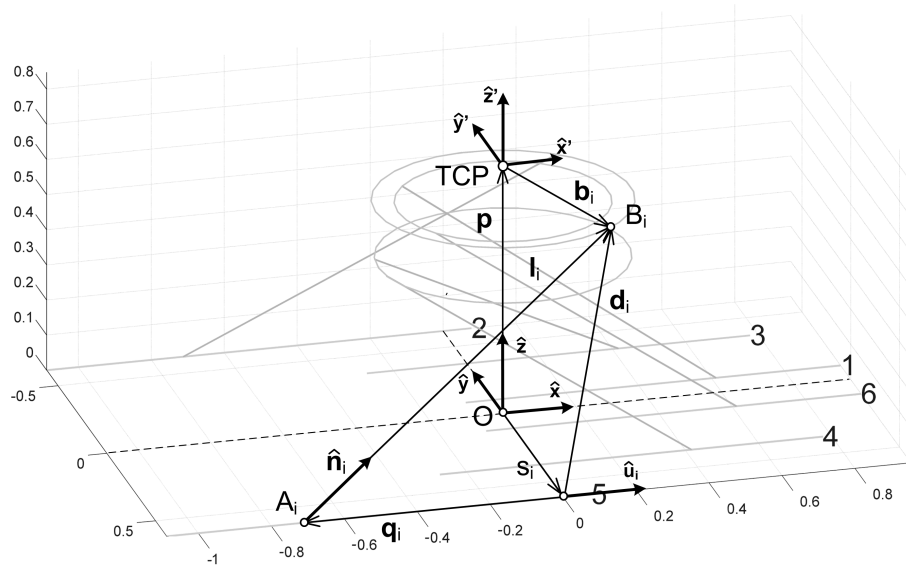
A differenza dell'Hexaglide, dove le guide sono tutte parallele e dirette lungo l'asse  $x$  ( $\hat{\mathbf{u}}_i = \{1, 0, 0\}^T$ ), le guide dell'Hexaslide avranno orientazioni differenti le une dalle altre, così come indicato in Tab. 2.2.

$\hat{\mathbf{u}}_1$	$\hat{\mathbf{u}}_2$	$\hat{\mathbf{u}}_3$	$\hat{\mathbf{u}}_4$	$\hat{\mathbf{u}}_5$	$\hat{\mathbf{u}}_6$
1	1	-1/2	-1/2	-1/2	-1/2
0	0	$\sqrt{3}/2$	$\sqrt{3}/2$	$-\sqrt{3}/2$	$-\sqrt{3}/2$
0	0	0	0	0	0

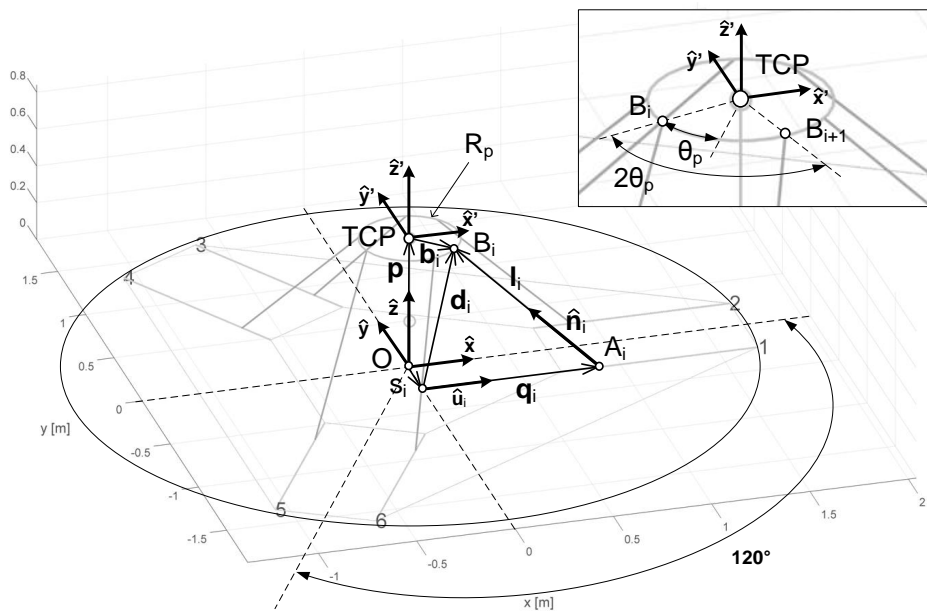
**Tab. 2.2:** Componenti dei versori delle guide dell'Hexaslide espresse nel s.d.r. globale.

Con semplici passaggi matematici si arriva ad un'equazione di secondo grado con incognita  $q_i$ :

$$q_i^2 - 2\mathbf{d}_i^T \hat{\mathbf{u}}_i q_i + \mathbf{d}_i^T \mathbf{d}_i - l_i^2 = 0 \quad (2.5)$$



(a) Hexaglide



(b) Hexaslide

**Fig. 2.2:** Diagramma dell' $i$ -esima catena cinematica per le due macchine.

la cui soluzione sarà:

$$q_i = \mathbf{d}_i^T \hat{\mathbf{u}}_i + h_i \sqrt{\mathbf{d}_i^T (\hat{\mathbf{u}}_i \hat{\mathbf{u}}_i^T - [I]) \mathbf{d}_i + l_i^2} \quad (2.6)$$

dove i parametri  $h_i$  assumeranno valore pari a  $\pm 1$  a seconda di come sono



disposti i link, in particolare nell'Hexaslide sarà  $h_i = +1 \quad \forall i$ .

## 2.3 Analisi di velocità

Per trovare il legame fra la velocità del TCP e quelle delle coordinate di giunto è necessario calcolare la matrice Jacobiana  $[J]$ , o meglio, per le convenzioni scelte, la sua inversa. Infatti, indicando con  $\mathbf{W}$  il vettore delle componenti di velocità del TCP, si può scrivere:

$$\mathbf{W} = [J]\dot{\mathbf{q}} \quad \text{e quindi} \quad \dot{\mathbf{q}} = [J]^{-1}\mathbf{W} \quad (2.7)$$

### 2.3.1 Piattaforma di Stewart

Derivando l'Eq. 2.1 rispetto al tempo e considerando le componenti di velocità dirette secondo il versore dell' $i$ -esimo link  $\hat{\mathbf{n}}_i$  si ottiene:

$$\dot{l}_i = \hat{\mathbf{n}}_i^T(\dot{\mathbf{p}} + [\dot{R}]\mathbf{b}'_i) = \hat{\mathbf{n}}_i^T(\mathbf{v}_C + [\dot{R}][R]^{-1}\mathbf{b}_i) = \hat{\mathbf{n}}_i^T(\mathbf{v}_C + [\Omega]\mathbf{b}_i) \quad (2.8)$$

dove  $[\Omega]$  è la matrice emisimmetrica contenente le componenti di velocità angolare del TCP. Sapendo che la relazione  $[\Omega]\mathbf{b}_i$  può anche essere scritta in forma vettoriale come  $\boldsymbol{\omega} \times \mathbf{b}_i$ , e che  $\hat{\mathbf{n}}_i \cdot (\boldsymbol{\omega} \times \mathbf{b}_i) = (\mathbf{b}_i \times \hat{\mathbf{n}}_i) \cdot \boldsymbol{\omega}$  si può riformulare l'equazione precedente nel seguente modo:

$$\dot{l}_i = \hat{\mathbf{n}}_i^T \mathbf{v}_C + (\mathbf{b}_i \times \hat{\mathbf{n}}_i)^T \boldsymbol{\omega} = [\hat{\mathbf{n}}_i^T \quad (\mathbf{b}_i \times \hat{\mathbf{n}}_i)^T] \begin{Bmatrix} \mathbf{v}_C \\ \boldsymbol{\omega} \end{Bmatrix} = [\hat{\mathbf{n}}_i^T \quad (\mathbf{b}_i \times \hat{\mathbf{n}}_i)^T] \mathbf{W} \quad (2.9)$$

Considerando ora tutti e sei i link si ottiene:

$$\dot{\mathbf{l}} = [J_{gs}]^{-1}\mathbf{W} \quad \text{dove} \quad [J_{gs}]^{-1} = \begin{bmatrix} \hat{\mathbf{n}}_1^T & (\mathbf{b}_1 \times \hat{\mathbf{n}}_1)^T \\ \hat{\mathbf{n}}_2^T & (\mathbf{b}_2 \times \hat{\mathbf{n}}_2)^T \\ \vdots & \vdots \\ \hat{\mathbf{n}}_6^T & (\mathbf{b}_6 \times \hat{\mathbf{n}}_6)^T \end{bmatrix} \quad (2.10)$$

Essendo in questo caso la lunghezza dei link coincidente con le coordinate di giunto, si può notare che  $\dot{\mathbf{l}} = \dot{\mathbf{q}}$ . Inoltre si è adottato il pedice  $gs$ , che denota la matrice Jacobiana della piattaforma di Gough-Stewart, per distinguerla dalle altre due macchine analizzate. Va sottolineato il fatto che le componenti di velocità angolare del TCP contenute nel vettore  $\mathbf{W}$  non coincidono con le derivate temporali degli angoli  $\alpha, \beta, \gamma$ , ma esiste una relazione del tipo:

$$\boldsymbol{\omega} = [H(\alpha, \beta, \gamma)]\dot{\boldsymbol{\Theta}} \quad (2.11)$$

### 2.3.2 Hexaglide ed Hexaslide

Come nel caso precedente si deriva rispetto al tempo l'Eq. 2.3 ricavando:

$$2\mathbf{d}_i^T \dot{\mathbf{d}}_i - 2\mathbf{d}_i^T \hat{\mathbf{u}}_i \dot{q}_i - 2q_i \hat{\mathbf{u}}_i^T \dot{\mathbf{d}}_i + 2\dot{q}_i q_i = 0 \quad (2.12)$$

raccogliendo a fattor comune il termine  $(\mathbf{d}_i - \hat{\mathbf{u}}_i q_i)^T$  si arriva alla scrittura:

$$(\mathbf{d}_i - \hat{\mathbf{u}}_i q_i)^T (\dot{\mathbf{d}}_i - \hat{\mathbf{u}}_i \dot{q}_i) = 0 \quad (2.13)$$

Esprimendo il termine  $\dot{\mathbf{d}}_i$  come:

$$\dot{\mathbf{d}}_i = \dot{\mathbf{p}} + [\dot{R}] \mathbf{b}'_i = \mathbf{v}_C + [\dot{R}][R]^{-1} \mathbf{b}_i = \mathbf{v}_C + [\Omega] \mathbf{b}_i \quad (2.14)$$

e sostituendo  $(\mathbf{d}_i - \hat{\mathbf{u}}_i q_i)$  con  $\mathbf{l}_i$  si giunge alla seguente espressione:

$$\mathbf{l}_i^T (\hat{\mathbf{u}}_i \dot{q}_i - \mathbf{v}_C - [\Omega] \mathbf{b}_i) = 0 \quad (2.15)$$

che, scomponendo il vettore  $\mathbf{l}_i$  come modulo e versore, e dividendo entrambi i membri dell'equazione per  $l_i$ , diventa:

$$\hat{\mathbf{n}}_i^T \hat{\mathbf{u}}_i \dot{q}_i - \hat{\mathbf{n}}_i^T (\mathbf{v}_C + [\Omega] \mathbf{b}_i) = 0 \quad (2.16)$$

Con passaggi analoghi a quelli fatti per la piattaforma di Gough-Stewart si ricava:

$$\hat{\mathbf{n}}_i^T \hat{\mathbf{u}}_i \dot{q}_i - [\hat{\mathbf{n}}_i^T \quad (\mathbf{b}_i \times \hat{\mathbf{n}}_i)^T] \begin{Bmatrix} \mathbf{v}_C \\ \boldsymbol{\omega} \end{Bmatrix} = 0 \quad (2.17)$$

Considerando i sei link si ottiene una forma matriciale compatta:

$$diag(\hat{\mathbf{n}}_1^T \hat{\mathbf{u}}_1, \dots, \hat{\mathbf{n}}_6^T \hat{\mathbf{u}}_6) \dot{\mathbf{q}} - [J_{gs}]^{-1} \mathbf{W} = \mathbf{0} \quad (2.18)$$

Nel secondo termine infatti si può riconoscere una matrice identica a quella che è stata definita come matrice Jacobiana inversa della piattaforma di Gough-Stewart. Ribattezzando ora la matrice diagonale  $diag(\hat{\mathbf{n}}_1^T \hat{\mathbf{u}}_1, \dots, \hat{\mathbf{n}}_6^T \hat{\mathbf{u}}_6)$  come  $[J_q]$ , si arriva ad avere:

$$[J_q] \dot{\mathbf{q}} - [J_{gs}]^{-1} \mathbf{W} = \mathbf{0} \quad \text{e quindi} \quad \dot{\mathbf{q}} = [J_q]^{-1} [J_{gs}]^{-1} \mathbf{W} = [J]^{-1} \mathbf{W} \quad (2.19)$$

## 2.4 Cinetostatica

La matrice Jacobiana è di fondamentale importanza anche per il problema cinetostatico. Infatti applicando il principio dei lavori virtuali si riesce a risalire a quali debbano essere le forze di attuazione  $\tau_{actuators}$  necessarie ad equilibrare quelle esterne agenti sul TCP  $\mathbf{f}_{e,C}$ :

$$\delta \mathbf{X}^T \mathbf{f}_{e,C} + \delta \mathbf{q}^T \tau_{actuators} = 0 \quad (2.20)$$

Sapendo che il legame fra le variazioni virtuali delle coordinate di giunto e quelle del TCP è:

$$\delta \mathbf{X} = [J] \delta \mathbf{q} \quad (2.21)$$

si può riformulare il principio dei lavori virtuali come segue:

$$\delta \mathbf{q}^T ([J]^T \mathbf{f}_{e,C} + \tau_{actuators}) = 0 \quad (2.22)$$

Dovendo valere tale relazione per qualsiasi variazione virtuale delle coordinate di giunto, si ricava:

$$[J]^T \mathbf{f}_{e,C} + \tau_{actuators} = 0 \quad \text{e quindi} \quad \tau_{actuators} = -[J]^T \mathbf{f}_{e,C} \quad (2.23)$$

Potrebbe essere interessante studiare come le forze di attuazione si trasformano e in quali tipi di forze e coppie in corrispondenza del TCP si traducano. In questo caso si arriva ad una relazione del tipo:

$$\tau_{actuators} = [J]^T \mathbf{f}_C \quad (2.24)$$

In questo caso la matrice Jacobiana assume il ruolo di un rapporto di trasmissione generalizzato. Nel caso in cui si volessero valutare le forze che si scaricano sui link invece che sui giunti attuati, si potrebbe ricorrere ad uno Jacobiano intermedio rappresentato da quella che è stata chiamata  $[J_{gs}]$ :

$$\tau_{links} = [J_{gs}]^T \mathbf{f}_C \quad (2.25)$$

Per quanto riguarda la piattaforma di Gough-Stewart, dove i link costituiscono gli attuatori del robot, le due formule vengono a coincidere.

## 2.5 Rigidezza cinematica

Gli elementi che compongono il robot, i link, i giunti, gli organi di trasmissione, gli attuatori, non sono infinitamente rigidi; per tale motivo, le forze applicate dall'esterno e quelle di attuazione causeranno una deflessione del robot spostando il TCP da quella che era la posa desiderata. Per fare un'analisi accurata e cogliere al meglio tali aspetti, è necessario utilizzare un modello agli elementi finiti (vedi Cap.4), ma in prima approssimazione si può pensare che tutte le inflessioni del robot siano da imputare agli elementi di attuazione. In tal caso si potrà trovare una rigidezza equivalente che lega ciascuna forza di attuazione alla variazione della corrispondente coordinata di giunto:

$$\tau_{i,actuators} = k_{i,q} dq_i \quad (2.26)$$

Essendo ogni coordinata di giunto dipendente solo ed unicamente dalla corrispondente forza di attuazione e viceversa, la relazione di sopra potrà essere riscritta in forma matriciale, e le rigidezze dei vari elementi di attuazione verranno inseriti in una matrice diagonale  $[K_q]$ :

$$\tau_{actuators} = [K_q] d\mathbf{q} \quad (2.27)$$

Ricordando che  $d\mathbf{q} = [J]^{-1} d\mathbf{X}$ , e che  $\tau_{actuators} = [J]^T \mathbf{f}_C$ , si può riscrivere la precedente equazione come:

$$\mathbf{f}_C = [J]^{-T} [K_q] [J]^{-1} d\mathbf{X} \quad (2.28)$$

Nel caso in cui gli attuatori fossero tutti uguali, gli elementi diagonali della matrice di rigidezza sarebbero identici e pari a  $k_q$ , per cui la relazione appena trovata diventerebbe:

$$\mathbf{f}_C = k_q [J]^{-T} [J]^{-1} d\mathbf{X} \quad \text{ovvero} \quad \mathbf{f}_C = k_q ([J][J]^T)^{-1} d\mathbf{X} = [K_X] \mathbf{X} \quad (2.29)$$

Dall'analisi degli autovalori e degli autovettori di  $[K_X]$ , che rappresentano rispettivamente i coefficienti di rigidezza e le direzioni principali, si possono fare interessanti considerazioni riguardo l'isotropia di rigidezza del robot ( [12], [13]). A tale scopo si possono calcolare i valori massimi e minimi di tali coefficienti:

$$k_{min} = k_q \lambda_{min}((J J^T)^{-1}) \quad k_{max} = k_q \lambda_{max}((J J^T)^{-1}) \quad (2.30)$$

La matrice Jacobiana può essere suddivisa in una componente traslazionale ed in una rotazionale:

$$[J] = \begin{bmatrix} [J_F] \\ [J_T] \end{bmatrix} \quad (2.31)$$

per cui sarà:

$$([J][J]^T)^{-1} = \begin{bmatrix} [J_F]^{-T} [J_F]^{-1} & [J_F]^{-T} [J_T]^{-1} \\ [J_T]^{-T} [J_F]^{-1} & [J_T]^{-T} [J_T]^{-1} \end{bmatrix} \quad (2.32)$$

Per fare in modo che tutti gli elementi della matrice Jacobiana inversa siano adimensionali è opportuno dividere la componente rotazionale, che di fatto ha le dimensioni di una lunghezza, per un fattore di scala definito lunghezza caratteristica  $L_c$ . Si ottiene così una matrice Jacobiana normalizzata e risulta:

$$([J_n][J_n]^T)^{-1} = \begin{bmatrix} [J_F]^{-T} [J_F]^{-1} & \frac{1}{L_c} [J_F]^{-T} [J_T]^{-1} \\ \frac{1}{L_c} [J_T]^{-T} [J_F]^{-1} & \frac{1}{L_c^2} [J_T]^{-T} [J_T]^{-1} \end{bmatrix} \quad (2.33)$$

Per la lunghezza caratteristica, Fassi e Legnani in [11] danno la seguente definizione:

$$L_c = \sqrt{\frac{tr(J_T^{-T} J_T^{-1})}{tr(J_F^{-T} J_F^{-1})}} \quad (2.34)$$

## 2.6 Singolarità

Vengono definite configurazioni singolari quelle pose del robot tali per cui vale:

$$\det([J]) = 0 \quad \vee \quad \det([J]) = \infty \quad (2.35)$$

Nel paragrafo dell'analisi di velocità si è visto come la matrice Jacobiana inversa possa essere scritta come:

$$[J]^{-1} = [J_q]^{-1}[J_{gs}]^{-1} \quad (2.36)$$

dove  $[J_q]$  è una matrice diagonale i cui elementi rappresentano la proiezione dei versori dei link lungo le direzioni di attuazione; tale matrice, nel caso della piattaforma di Gough-Stewart diverrà una matrice identità, in quanto i versori dei link andranno a coincidere con quelli di attuazione. Noto che:

$$\det([J]) = \det([J_q])\det([J_{gs}]) \quad (2.37)$$

il raggiungimento di una configurazione singolare potrà essere imputabile tanto alla matrice  $[J_q]$  quanto alla matrice  $[J_{gs}]$ ; si individuano quindi tre tipi di singolarità così come indicato in Tab. 2.3.

Tipo	Determinante			Descrizione
1° tipo	$\det([J_q]) = 0$	$\vee$	$\det([J_{gs}]^{-1}) = \infty$	direzioni a rigidità infinita
2° tipo	$\det([J_q]) = \infty$	$\vee$	$\det([J_{gs}]^{-1}) = 0$	direzioni a rigidità nulla
3° tipo	$\det([J_q]) = 0$	$\wedge$	$\det([J_{gs}]^{-1}) = 0$	dir. a rigidità infinita e nulla

**Tab. 2.3:** Tre tipologie di singolarità.

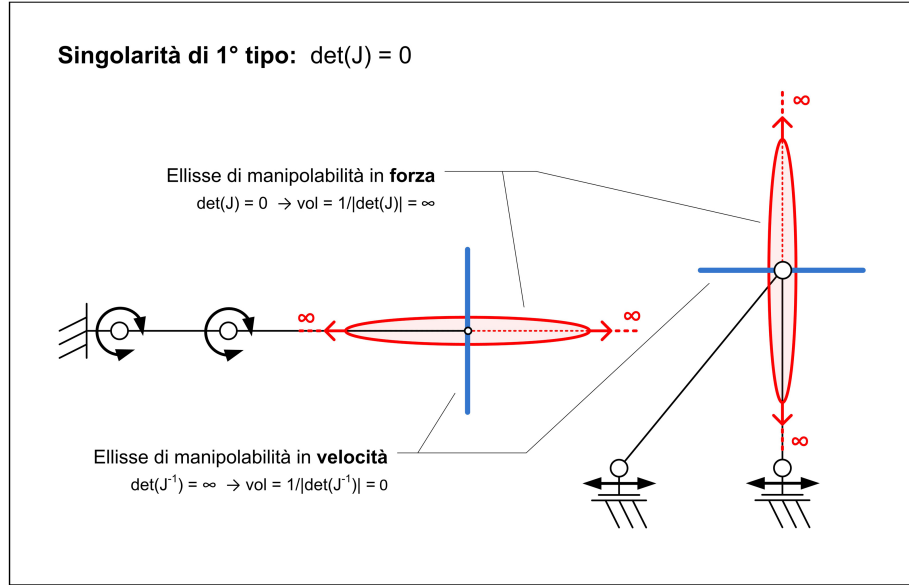
### 2.6.1 Singolarità del 1° tipo

Si parla di singolarità del 1° tipo quando si annulla il determinante della matrice Jacobiana, o analogamente quando il determinante della matrice Jacobiana inversa tende all'infinito:

$$\det([J]) = 0 \quad \vee \quad \det([J]^{-1}) = \infty \quad (2.38)$$

Nel caso in cui sia  $\det([J_{gs}]^{-1}) = \infty$  non si riesce a dare una spiegazione fisica di quello che stia accadendo. Al contrario le cause dell'annullamento del determinante della matrice  $[J_q]$  sono dovute all'annullamento di uno dei suoi elementi diagonali: in questo caso almeno uno dei versori dei link è perpendicolare al versore della corrispondente coordinata di giunto. Ciò comporta due effetti: (1) la perdita di un grado di libertà, in quanto l'attuatore non è in grado di fornire componenti di velocità lungo la direzione individuata del versore del corrispondente link, perché ad esso perpendicolare; (2) il raggiungimento di un valore infinito di rigidità lungo il link in quanto le forze agenti in tale direzione verranno completamente compensate dalle reazioni vincolari. Un esempio di questo tipo di singolarità per un semplice manipolatore a 2-gdl è mostrato in Fig. 2.3.

Per quanto detto sopra, tale scenario potrà realizzarsi solo ed unicamente nell'Hexaglide e nell'Hexaslide. Inoltre in tali macchine nella situazione appena descritta potrebbe verificarsi il passaggio da un tipo di montaggio ad un altro: infatti si dimostra che sotto tali ipotesi il radicando dell'Eq. 2.6 si annulla:



**Fig. 2.3:** Rappresentazione di una singolarità di 1° tipo per un robot scara ed un robot parallelo a 2-gdl.

$$\Delta_i = \mathbf{d}_i^T (\hat{\mathbf{u}}_i \hat{\mathbf{u}}_i^T - [I]) \mathbf{d}_i + l_i^2 = 0 \quad (2.39)$$

e quindi le soluzioni dei due tipi di montaggio diventano coincidenti e pertanto potrebbe verificarsi il passaggio indesiderato da una soluzione costruttiva all'altra. Per evitare tale scenario è opportuno prevedere dei vincoli sul range di mobilità dei giunti.

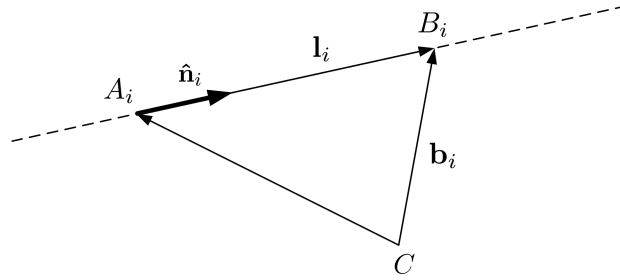
## 2.6.2 Singolarità del 2° tipo

Le singolarità del 2° tipo sono quelle in cui il determinante della matrice Jacobiana tende all'infinito, o equivalentemente il determinante della matrice Jacobiana inversa si annulla:

$$\det([J]) = \infty \quad \vee \quad \det([J]^{-1}) = 0 \quad (2.40)$$

In questo caso la responsabilità dell'annullamento del determinante dello Jacobiano inverso è da attribuirsi interamente alla matrice  $[J_{gs}]^{-1}$ , in quanto gli elementi diagonali della matrice  $[J_q]$  non potranno mai assumere valore infinito essendo dei prodotti scalari. In questo caso un'interpretazione fisica di ciò che accade in queste configurazioni può essere data ricorrendo all'uso della geometria di Grassmann così come indicato da Merlet in. Con riferimento alla Fig. 2.4 si definisce vettore di Plücker il seguente vettore:

$$\mathbf{p} = [\mathbf{AB}, \mathbf{CA} \times \mathbf{CB}]_i = [\mathbf{AB}, \mathbf{CB} \times \mathbf{AB}]_i \quad (2.41)$$



**Fig. 2.4:** Vettore di Plücker.

Tale notazione permette di identificare in maniera univoca la linea passante per i punti  $A_i$  e  $B_i$ , che altro non è che il link  $i$ -esimo. A questo punto si può normalizzare il vettore di Plücker ottenendo:

$$\mathbf{p}_n = \left[ \frac{\mathbf{l}_i}{\|\mathbf{l}_i\|}, \frac{\mathbf{l}_i \times \mathbf{b}_i}{\|\mathbf{l}_i\|} \right] = [\hat{\mathbf{n}}_i, \hat{\mathbf{n}}_i \times \mathbf{b}_i] \quad (2.42)$$

Se si analizzano le righe delle matrici  $[J_{gs}]^{-1}$  ci si accorge che queste sono esattamente i vettori di Plücker ora descritti. E' noto che se il determinante di una matrice è nullo, significa che almeno una delle sue righe è linearmente dipendente dalle altre, perciò, nel caso di singolarità di 2° tipo almeno una delle righe di  $[J_{gs}]^{-1}$  è linearmente dipendente dalle altre. Questo tipo di singolarità comporta due effetti indesiderati: (1) lungo una determinata direzione la rigidezza cinematica del robot diventa nulla, e quindi gli attuatori non sono in grado di contrastare eventuali forze agenti lungo tale direzione; (2) lungo questa stessa direzione il robot acquista un grado di libertà dato che sono possibili spostamenti infinitesimi nonostante gli attuatori siano bloccati, il sistema diventa quindi labile. Questi effetti possono essere visti in Fig. 2.5.

### 2.6.3 Singolarità del 3° tipo

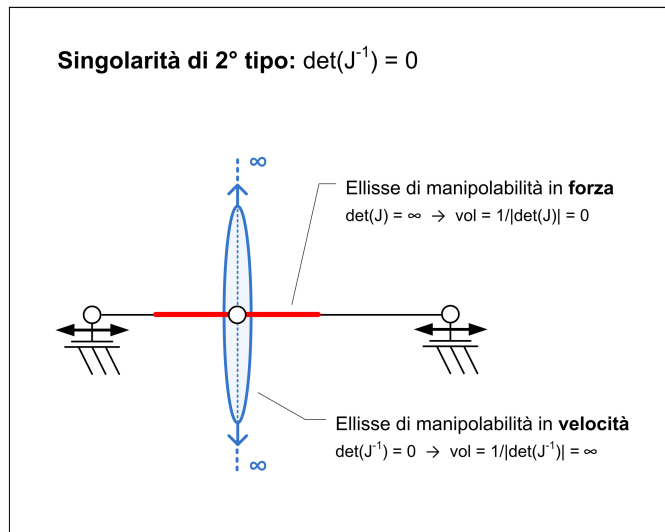
Nel caso particolare in cui si verifichi contemporaneamente:

$$\det([J_q]) = 0 \quad \wedge \quad \det([J_{gs}]^{-1}) = 0 \quad (2.43)$$

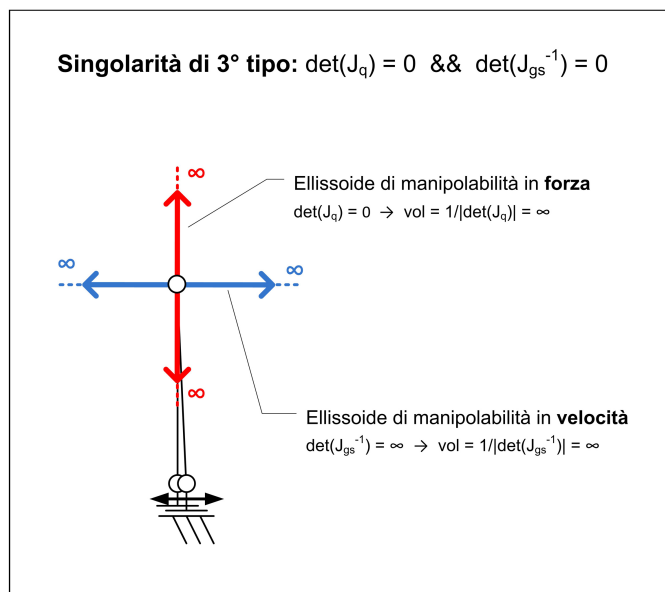
si parla di singolarità del 3° tipo. In questa situazione coesisteranno direzioni a rigidezza cinematica infinita e direzioni a rigidezza nulla come si vede in Fig. 2.6.

## 2.7 Ellipsoidi di manipolabilità ed isotropia

Grazie alla matrice  $[J]$  è possibile avere un'idea di come le velocità e le forze nello spazio dei giunti si trasformino in quelle nello spazio di lavoro. Considerando infatti un'ipersfera  $n$ -dimensionale di modulo unitario, dove  $n$  rappresenta il numero



**Fig. 2.5:** Rappresentazione di una singularità di 2° tipo per un robot planare a 2-gdl.



**Fig. 2.6:** Rappresentazione di una singularità di 3° tipo per un robot planare a 2-gdl.

di coordinate di giunto, è possibile vedere come tale ipersfera si trasformi in un iperellissoide nello spazio di lavoro per effetto della matrice Jacobiana. In questo modo si ha un'indicazione visiva della qualità del robot in una determinata posa, poichè ellissoidi troppo deformati sono indice di una forte anisotropia e spesso della vicinanza ad una configurazione singolare.



### 2.7.1 Ellissoidi di manipolabilità in velocità

Si consideri l'ipersfera  $n$ -dimensionale unitaria i cui assi rappresentano le velocità delle coordinate di giunto; si potrà scrivere:

$$\|\dot{\mathbf{q}}\| \leq 1 \quad (2.44)$$

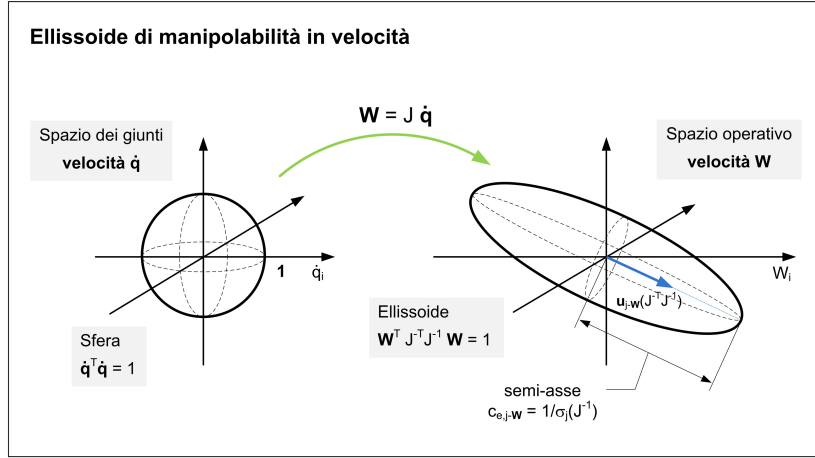
dove come norma è stata scelta quella euclidea:

$$\|\dot{\mathbf{q}}\|_2 = \dot{\mathbf{q}}^T \dot{\mathbf{q}} \leq 1 \quad (2.45)$$

Se si utilizza la relazione trovata nell'analisi di velocità, si può scrivere:

$$\mathbf{W}^T [\mathbf{J}]^{-T} [\mathbf{J}]^{-1} \mathbf{W} \leq 1 \quad (2.46)$$

Questa definisce un iperellissoide in uno spazio i cui assi sono rappresentati dalle velocità dello spazio di lavoro che prende il nome di iperellissoide di manipolabilità in velocità (Fig. 2.7).



**Fig. 2.7:** Ellissoide di manipolabilità in velocità.

I suoi semiassi saranno diretti lungo gli autovettori  $\mathbf{u}_j$  della matrice  $([\mathbf{J}][\mathbf{J}]^T)$  e avranno modulo pari ai suoi autovalori:

$$c_{e,j-W} = \lambda_j(\mathbf{J}\mathbf{J}^T) = \sigma_j(\mathbf{J}) \quad (2.47)$$

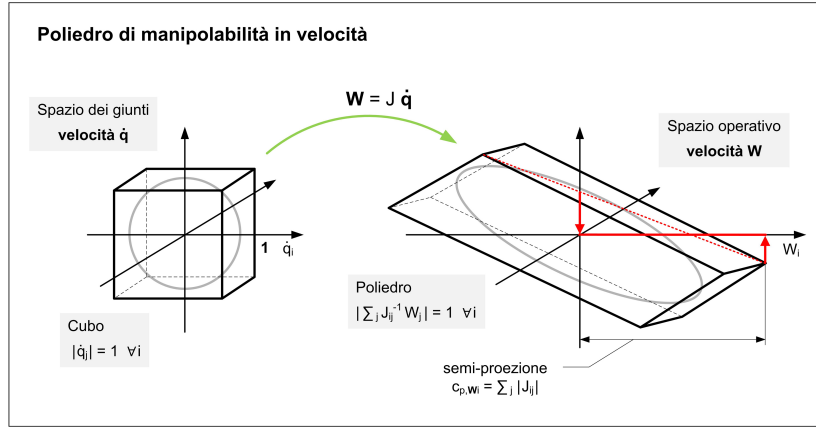
Il volume di un iperellissoide è proporzionale alla produttoria dei suoi semiassi:

$$vol_{e,W} \propto \prod_{j=1}^n c_{e,j-W} = \prod_{j=1}^n \sigma_j(\mathbf{J}) = |\det(\mathbf{J})| \quad (2.48)$$

In caso di singolarità di 1° o 2° tipo il determinante di  $[\mathbf{J}]$ , e di conseguenza il volume dell'iperellissoide, assumerà valore rispettivamente nullo o al limite infinito.

Una trattazione di questo tipo presuppone che se una delle velocità delle coordinate di giunto raggiunge il valore limite, pari ad 1, le altre componenti debbano

essere nulle. Questo scenario non è molto rappresentativo della realtà, dove ogni attuatore è indipendente dagli altri e di conseguenza ogni componente di velocità può assumere il valore limite indipendentemente dalle altre. Si passa quindi a considerare un ipercubo di semi-lato unitario che verrà mappato, grazie alla matrice  $[J]$ , in un iperpoliedro (Fig. 2.8).



**Fig. 2.8:** Poliedro di manipolabilità in velocità.

Per fare ciò si utilizzerà la norma infinito in luogo di quella euclidea:

$$\|\dot{\mathbf{q}}\|_{\infty} \leq 1 \Rightarrow |\dot{q}_i| \leq 1 \forall i = [1, n] \quad (2.49)$$

o equivalentemente

$$\left| \sum_{j=1}^n J_{ij}^{-1} W_j \right| \leq 1 \forall i = [1, n] \quad (2.50)$$

### 2.7.2 Ellissodi di manipolabilità in forza

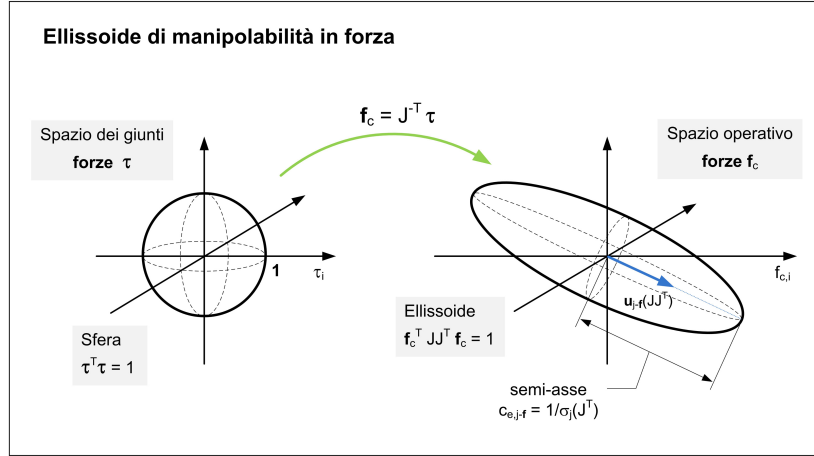
Analogamente a quanto fatto per le velocità, si può vedere in che modo le forze di attuazione si trasformino in quelle trasmesse alla piattaforma. A tal proposito si prenderà in esame l'ipersfera unitaria nello spazio delle forze di attuazione, ottenuta utilizzando la norma euclidea:

$$\|\tau\|_2 \leq 1 \Leftrightarrow \tau^T \tau \leq 1 \quad (2.51)$$

Ricordando dalla cinetostatica che  $\tau = [J]^T \mathbf{f}_C$  l'espressione precedente diventa:

$$\mathbf{f}_C^T [J][J]^T \mathbf{f}_C \leq 1 \quad (2.52)$$

La relazione appena trovata identifica un iperellissoide nello spazio i cui assi sono rappresentati dalle componenti delle forze e coppie trasmesse alla piattaforma mobile (Fig. 2.9).



**Fig. 2.9:** Ellissoide di manipolabilità in forza.

L'iperellissoide in questione avrà semi-assi diretti lungo i versori della matrice  $([J][J]^T)^{-1}$  e di modulo:

$$c_{e,j-f} = \lambda_j((JJ^T)^{-1}) = \frac{1}{\lambda_j(JJ^T)} = \frac{1}{\sigma_j(J^T)} \quad (2.53)$$

La prima cosa che si nota è che il modulo dei semi-assi dell'iperellissoide di manipolabilità in forza è il reciproco di quello dei corrispondenti dell'iperellissoide di manipolabilità in velocità. Inoltre, essendo le matrici che permettono di mappare i due iperellissoidi una l'inversa dell'altra, le direzioni dei semi-assi sarà la medesima, avendo le due matrici gli stessi autovettori. Questo aspetto evidenzia che in una determinata posa, lungo le direzioni in cui si avrà moltiplica statica delle forze massima le velocità trasmesse al TCP saranno minime e viceversa. Per sottolineare tale aspetto, si può ricavare la relazione che lega il volume dell'iperellissoide al determinante della matrice  $[J]$ :

$$vol_{e,f} \propto \prod_{j=1}^n c_{e,j-f} = \frac{1}{\prod_{j=1}^n \sigma_j(J^T)} = \frac{1}{|\det(J^T)|} = \frac{1}{|\det(J)|} \quad (2.54)$$

Si nota subito che tale volume è il reciproco di quello dell'ellissoide di velocità. Per quanto riguarda le configurazioni singolari valgono considerazioni analoghe a quelle del caso precedente. Anche per le forze vale un discorso analogo a quello delle velocità, e per tale motivo si utilizzerà la norma infinito invece di quella euclidea perché più rappresentativa della realtà:

$$\|\tau\|_\infty \leq 1 \Rightarrow |\tau_i| \leq 1 \forall i = [1, n] \quad (2.55)$$

e di conseguenza

$$\left| \sum_{j=1}^n J_{ij}^T f_j \right| \leq 1 \forall i = [1, n] \quad (2.56)$$

Il risultato sarà l'iperpoliedro di Fig. 2.10.

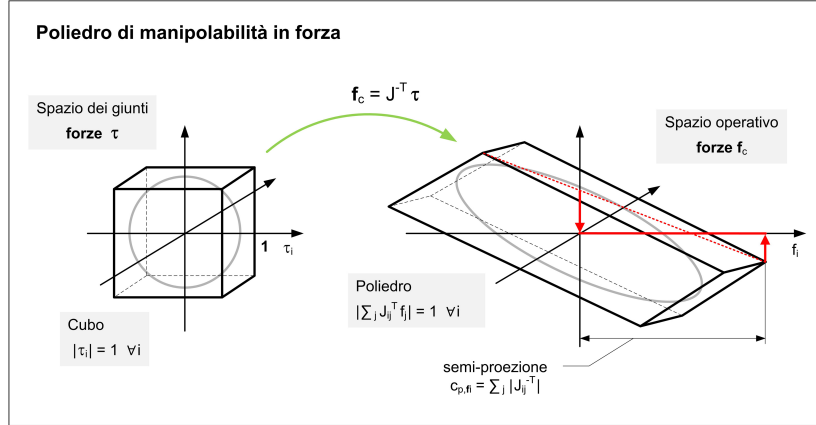


Fig. 2.10: Poliedro di manipolabilità in forza.

## 2.8 Forze di attuazione massime

Può essere utile avere un'idea di come le forze applicate alla piattaforma si trasmettano lungo i link fino agli attuatori. In pratica si tratta di applicare al contrario il procedimento sopra descritto. Per cui anche in questo caso si assume che:

$$\|\mathbf{f}\| \leq 1 \quad (2.57)$$

Usando come norma quella euclidea, e ricordando la relazione  $\mathbf{f} = [J]^{-T} \boldsymbol{\tau}$ , l'ipersfera unitaria nello spazio di lavoro  $\mathbf{f}^T \mathbf{f} \leq 1$  viene mappata nell'iperellissoide nello spazio delle forze di attuazione:

$$\boldsymbol{\tau}^T J^{-1} J^{-T} \boldsymbol{\tau} \leq 1 \quad (2.58)$$

Se invece della norma euclidea si usa la norma infinito si ottiene una mappatura dell'iperpoliedro delle forze applicate al TCP  $\|\mathbf{f}\|_\infty \leq 1$  nell'iperpoliedro nello spazio delle forze di attuazione:

$$-1 \leq \sum_{j=1}^n J_{ij}^{-T} \tau_j \leq 1 \quad (2.59)$$

partendo da questa equazione si dimostra che:

$$-\sum_{j=1}^n |J_{ij}^T| \leq \tau_i \leq \sum_{j=1}^n |J_{ij}^T| \quad (2.60)$$

dove  $\sum_{j=1}^n |J_{ij}^T|$  altro non è che la semi-proiezione dell'iperpoliedro sull'asse  $\tau_i$ . Il valore della semi-proiezione massima coincide con la norma infinito della matrice  $[J]^T$ :

$$\|[J]^T\|_\infty = \max_i \left( \sum_{j=1}^n |J_{ij}^T| \right) \quad (2.61)$$

Un discorso analogo può essere fatto anche nel caso delle velocità.



## OTTIMIZZAZIONE CINETOSTATICA

In questo capitolo verranno descritti due processi di sintesi: il primo utilizzato per sintetizzare la piattaforma di Gough-Stewart e l'Hexaslide; il secondo usato per la sintesi dell'Hexaglide. Per cominciare si descriverà la porzione di volume all'interno del quale il TCP dovrà essere in grado di raggiungere qualsiasi posizione con qualsiasi orientazione compresa entro certi limiti  $\Theta \in [\pm\alpha_{lim}, \pm\beta_{lim}, \pm\gamma_{lim}]$ . Si procederà poi ad illustrare la parametrizzazione delle due macchine e l'algoritmo utilizzato per la loro ottimizzazione cinetostatica, indicando gli obiettivi ed i vincoli che sono stati imposti.

### 3.1 Spazio di lavoro

La porzione di spazio all'interno della quale il TCP dovrà essere libero di muoversi è identificata da un parallelepipedo il cui centro si trova in un punto le cui coordinate calcolate rispetto alla terna globale sono:  $C_{center} = \{0, 0, z_{Wsd}\}^T$  dove  $z_{Wsd}$  assumerà un valore differente a seconda che si parli di modellini di barche o di generatori eolici: nel primo caso tale valore verrà fissato pari a  $700\text{ mm}$ , mentre nel secondo sarà pari a  $250\text{ mm}$ . Si definisce *home position* la posa del robot tale per cui il TCP si trova nel punto sopra descritto con orientazione nulla della terna mobile rispetto a quella globale. Oltre al punto centrale, anche le dimensioni e la forma del parallelepipedo sono differenti nel caso delle barche e dei generatori eolici:

- *barche*: a partire dal punto appena descritto il TCP dovrà potersi spostare di  $\pm 0.1\text{ m}$  lungo la direzione  $x$ , di  $\pm 0.07\text{ m}$  lungo  $y$ , e di  $\pm 0.15\text{ m}$  lungo  $z$ . Il parallelepipedo così descritto avrà quindi dimensioni  $0.2 \times 0.14 \times 0.3\text{ m}^3$ .
- *generatori eolici*: il parallelepipedo avrà dimensioni  $0.24 \times 0.14 \times 0.2\text{ m}^3$  e centro in  $C_{center}$ .

Per l'orientazione del TCP e quindi della piattaforma mobile si è scelto di utilizzare gli angoli nautici:  $\alpha$  rollio,  $\beta$  beccheggio, e  $\gamma$  imbardata. I valori limite per i tre angoli nei due casi valgono:

- *barche*:
 
$$\alpha \in [-5^\circ, +5^\circ] \quad \beta \in [-15^\circ, +15^\circ] \quad \gamma \in [-5^\circ, +5^\circ] \quad (3.1)$$

- *generatori eolici*:

$$\alpha \in [-5^\circ, +5^\circ] \quad \beta \in [-16^\circ, +16^\circ] \quad \gamma \in [-12^\circ, +12^\circ] \quad (3.2)$$

## 3.2 Parametrizzazione delle macchine

Sfruttando la simmetria ogni  $120^\circ$  che caratterizza sia l'Hexaslide che la piattaforma di Gough-Stewart è stato possibile utilizzare una parametrizzazione a quattro variabili. Per quanto concerne l'Hexaglide invece è necessario utilizzare un numero più elevato di parametri.

### 3.2.1 Piattaforma di Gough-Stewart

L'esapode di Gough-Stewart è costituito da una piattaforma mobile su cui si trovano tre coppie di giunti sferici. Ciascuna coppia è sfasata dalle altre di  $120^\circ$  rispetto all'asse  $z$  della terna mobile solidale con la piattaforma. Sulla base si ritrova una disposizione dei giunti del tutto analoga a quella della piattaforma, ma in questo caso si farà riferimento direttamente alla terna globale. Per finire i link costituiti da pistoni idraulici e quindi di lunghezza variabile collegano ciascun giunto di base col corrispondente giunto di piattaforma. Con riferimento alla Fig. ?? si definiscono le quattro variabili scelte per parametrizzare la piattaforma di Stewart:

- $R_b$ : raggio della circonferenza di base, lungo la quale si collocano i giunti di base  $A_i$ .
- $\theta_b$ : semi-apertura angolare fra i due segmenti che congiungono l'origine  $O$  con una coppia di giunti di base  $A_i$  contigui.
- $R_p$ : raggio della circonferenza della piattaforma, lungo la quale si collocano i giunti di piattaforma  $B_i$ .
- $\theta_p$ : semi-apertura angolare fra i due segmenti che congiungono il TCP (origine della terna mobile) con una coppia di giunti di piattaforma  $B_i$  contigui.

Come accade di solito nella piattaforma di Stewart le coppie di giunti di piattaforma contigui saranno:  $[(1, 2)(3, 4)(5, 6)]$ ; mentre quelle dei giunti di base:  $[(1, 6)(2, 3)(4, 5)]$ .

### 3.2.2 Hexaslide

Anche nell'Hexaslide l'elemento principe è rappresentato da una piattaforma mobile, la cui descrizione è identica a quella dell'esapode di Stewart. Ciò che cambia è che in questo caso i link saranno di lunghezza fissa ed uguali fra loro. Alla base saranno presenti tre coppie di guide parallele, sfasate rispetto all'asse  $z$  della terna globale di  $120^\circ$  le une dalle altre. La parte terminale dei link è vincolata tramite un giunto sferico ad uno slider che potrà scorrere lungo le guide. I quattro parametri scelti per l'ottimizzazione in questo caso saranno:



- $s$ : semidistanza fra due guide parallele;
- $l_{link}$ : lunghezza dei sei link;
- $R_p$ : raggio della circonferenza di piattaforma su cui si collocano i giunti  $B_i$ ;
- $\theta_p$ : semi-apertura angolare fra i due segmenti che congiungono il TCP (origine della terna mobile) con una coppia di giunti di piattaforma  $B_i$  contigui.

### 3.2.3 Hexaglide

La piattaforma dell'Hexaglide è differente rispetto a quella delle macchine precedenti. In questo caso i giunti di piattaforma non giacciono tutti su uno stesso piano e l'apertura angolare fra una coppia di giunti è diversa da quella delle altre coppie. Inoltre la lunghezza dei sei link non è uguale per tutti, ma ci saranno tre coppie di link di lunghezza diversa. Per l'ottimizzazione quindi è stata scelta una parametrizzazione a 16 parametri riportata in Tab. 3.1.

Simbolo	Parametro
$l_{01}$	lunghezza link 1 e 6
$l_{02}$	lunghezza link 2 e 5
$l_{03}$	lunghezza link 3 e 4
$s_{01}$	semidistanza lungo $y$ delle guide 1 e 6
$s_{02}$	semidistanza lungo $y$ delle guide 2 e 5
$s_{03}$	semidistanza lungo $y$ delle guide 3 e 4
$\beta_{01}$	semi-apertura angolare fra i giunti di piattaforma 1 e 6
$\beta_{02}$	semi-apertura angolare fra i giunti di piattaforma 2 e 5
$\beta_{03}$	semi-apertura angolare fra i giunti di piattaforma 3 e 4
$r_{01}$	distanza radiale fra TCP e giunti di piattaforma 1 e 6
$r_{02}$	distanza radiale fra TCP e giunti di piattaforma 2 e 5
$r_{03}$	distanza radiale fra TCP e giunti di piattaforma 3 e 4
$t_{01}$	distanza lungo $z$ fra TCP e giunti di piattaforma 1 e 6
$t_{02}$	distanza lungo $z$ fra TCP e giunti di piattaforma 2 e 5
$t_{03}$	distanza lungo $z$ fra TCP e giunti di piattaforma 3 e 4
$z_{hp}$	posizione lungo $z$ del TCP rispetto alla terna globale

**Tab. 3.1:** Parametri utilizzati per l'ottimizzazione dell'Hexaglide.

## 3.3 Algoritmo genetico a singolo obiettivo

L'algoritmo di ottimizzazione utilizzato per l'Hexaslide e la piattaforma di Stewart è un algoritmo di tipo genetico a singolo obiettivo. Tale algoritmo ha lo scopo di ricercare il set di parametri che annulli o quantomeno minimizzi una specifica funzione di costo rappresentativa dell'obiettivo prefissato. I passi che lo caratterizzano sono:

1. Scelta di una popolazione di individui di numero sufficiente da rappresentare un campione statistico significativo. Tale popolazione costituirà una singola generazione di individui. I geni degli individui saranno le variabili scelte per parametrizzare le macchine. Per ciascun individuo le quattro variabili saranno libere di assumere un valore compreso fra un limite minimo ed un limite massimo.
2. Si procede per ciascun individuo della generazione alla valutazione della funzione di costo che sarà dipendente dalle variabili di parametrizzazione.
3. Sulla base del valore assunto dalla funzione di costo si sceglie un gruppo di individui che costituirà l'*élite* e che passerà immutato alla successiva generazione.
4. Si passa quindi alla creazione di una nuova generazione. I meccanismi secondo i quali vengono generati i nuovi individui, oltre a quello elitario, sono due: (1) *crossover*, i nuovi individui nascono per combinazione casuale dei geni degli individui elitari; (2) *mutation*, i nuovi individui nascono dalla mutazione casuale dei geni di un singolo individuo elitario. La percentuale di individui che verranno generati secondo un meccanismo piuttosto che un altro è un parametro che deve essere impostato tra le opzioni dell'algoritmo.
5. Per la nuova generazione si calcola nuovamente la funzione di costo e si procede così in maniera iterativa fin quando non si rispetta uno dei criteri di stop.

### 3.3.1 Criteri di stop

Possono essere impostati vari criteri di stop, ad esempio: (1) raggiungimento di un numero massimo di generazioni; (2) stallo, per un determinato numero di generazioni, del valore assunto dalla funzione di costo del migliore individuo; (3) stallo del miglior valore assunto dalla funzione di costo per un determinato intervallo di tempo; (4) la media dei valori assunti dalla funzione di costo per una generazione si discosta dalla media della generazione precedente per un valore inferiore ad una tolleranza imposta; (5) stop manuale.

## 3.4 Algoritmo genetico multiobiettivo

L'ottimizzazione dell'Hexaglide è stata condotta utilizzando un algoritmo genetico multi-obiettivo ([14]). A differenza dell'ottimizzazione a singolo obiettivo non esiste un'unica soluzione ottima, ma una serie di soluzioni Pareto-ottime. Per stabilire quali individui appartengano alla frontiera di Pareto si confrontano fra di loro. Un individuo  $\mathbf{x}^{(1)}$  viene definito dominante rispetto ad un individuo  $\mathbf{x}^{(2)}$  se valgono le due seguenti ipotesi:

$$\begin{cases} f_j(\mathbf{x}^{(1)}) \leq f_j(\mathbf{x}^{(2)}) \quad \forall j \quad \text{con } j = 1, \dots, M \\ \exists \bar{m} : f_{\bar{m}}(\mathbf{x}^{(1)}) < f_{\bar{m}}(\mathbf{x}^{(2)}) \end{cases} \quad (3.3)$$

Il meccanismo di creazione degli individui della generazione successiva è identico a quello del caso a singolo obiettivo, in questo caso però, non essendoci un'unica soluzione, per valutare quali siano gli elementi che costituiranno la frontiera di Pareto si uniscono gli individui che costituiscono la nuova generazione con quelli della generazione precedente e si valuta quali siano quelli dominanti.

## 3.5 Funzioni di costo

### 3.5.1 Copertura del volume di lavoro

La funzione di costo per un determinato set di variabili prevede il calcolo della porzione di volume non raggiungibile dello spazio di lavoro desiderato. Per fare ciò lo spazio di lavoro desiderato  $Wsd$  è stato suddiviso utilizzando una griglia tridimensionale di punti: definite  $L_{Wsd,x}, L_{Wsd,y}, L_{Wsd,z}$  le dimensioni del parallelepipedo, e scelto un numero di punti  $N_{grid}$  in cui suddividere ogni lato, lo spazio di lavoro è stato suddiviso in  $N_{grid}^3$  porzioni, ciascuna di volume pari a:

$$\Delta v = \frac{L_{Wsd,x} \times L_{Wsd,y} \times L_{Wsd,z}}{N_{grid}^3} \quad (3.4)$$

Fissata ora un'orientazione  $\bar{\Theta} = [\bar{\alpha}, \bar{\beta}, \bar{\gamma}]$  si calcola il volume non coperto dello spazio di lavoro desiderato per quella determinata orientazione: si valuta cioè per quanti dei punti in cui è stato frazionato lo spazio di lavoro non vengano rispettati i vincoli imposti

$$v_{nc}(\bar{\Theta}) = \sum_i k_i \Delta v \quad (3.5)$$

dove  $k_i$  assumerà valore nullo se l' $i$ -esimo punto dello spazio di lavoro rispetta tutti i vincoli, altrimenti assumerà valore unitario.

Ripetendo tale procedura per ciascuna combinazione dei parametri angolari  $\alpha, \beta, \gamma$  si arriva alla definizione del volume totale non coperto:

$$V_{nc} = \sum_j v_{nc}(\bar{\Theta}_j) \quad (3.6)$$

### 3.5.2 Copertura area di lavoro

Analogamente a quanto fatto per le altre due macchine si è cercato di minimizzare, o al limite annullare lo spazio di lavoro non coperto. Questa volta però, sfruttando il fatto che l'Hexaglide può spostarsi liberamente lungo la direzione  $x$ , come funzione di costo è stata considerata la differenza fra l'area individuata dalla proiezione dello spazio di lavoro desiderato sul piano  $yz$  e la porzione di tale piano che il TCP riesce a raggiungere con un'orientazione qualsiasi.

### 3.5.3 Ingombro longitudinale

l'ingombro longitudinale necessario ad esplorare un generico piano  $yz$  viene calcolato come:

$$size_x = \max_{\mathbf{X},i}(q_i) - \min_{\mathbf{X},i}(q_i) \quad (3.7)$$

Le dimensioni totali lungo la direzione  $x$ , e quindi le dimensioni delle guide, si ottengono per semplice traslazione. Per tale motivo ridurre l'ingombro longitudinale di per esplorare un singolo piano, automaticamente si traduce nella riduzione delle dimensioni longitudinali totali della macchina.

## 3.6 Vincoli

Per far sì che le macchine soddisfacessero tutte le richieste, sono stati imposti dei vincoli. Questi vincoli servono per assicurare che la macchina sia effettivamente in grado di raggiungere tutti i punti dello spazio di lavoro desiderato, ed inoltre assicurano che vengano rispettati i limiti di moltiplica statica delle forze e di distanza minima fra i componenti della macchina per evitare collisioni durante il funzionamento. Le tipologie di vincoli possono essere suddivise nella seguente maniera:

1. **Vincoli cinematici:** sono quei vincoli che servono a garantire il raggiungimento di tutti i punti dello spazio di lavoro. In particolare si è imposto:
  - $\Delta_i \geq 0$  (solo *Hexaslide* ed *Hexaglide*): per ciascun link si verificherà se  $\Delta_i = (\mathbf{d}_i^T \hat{\mathbf{u}}_i)^2 - \mathbf{d}_i^T \mathbf{d}_i + l_i^2$  sia maggiore, o al più uguale, a 0. Se così non fosse, significherebbe che la distanza fra l' $i$ -esimo giunto di piattaforma ed il corrispondente giunto di base è maggiore della lunghezza del link.
  - $q_{min} \leq q_i \leq q_{max}$ : per ciascuna coordinata di giunto si verifica che questa sia compresa all'interno di un certo range di variazione. Se  $q_i$  assumesse un valore al di fuori di questo range la macchina assumerebbe dimensioni eccessive.
  - $\theta_{A,i} \leq \theta_{A,lim}$  e  $\theta_{B,i} \leq \theta_{B,lim}$ : si verifica che ogni giunto, sia di piattaforma che di base, rispetti il range di mobilità fornito dal costruttore. Per effettuare questa verifica si calcolano i versori di ciascun link nel seguente modo:

$$\hat{\mathbf{n}}_i = \frac{B_i - A_i}{\|B_i - A_i\|_2} \quad (3.8)$$

A questo punto si calcola lo sfasamento angolare fra il versore dei link appena calcolato e il versore dei giunti alle sue estremità:

$$\theta_{A_i} = \text{acos}(\hat{\mathbf{n}}_i^T \hat{\mathbf{j}}_{A,i}) \quad \mathbf{e} \quad \theta_{B_i} = \text{acos}(-\hat{\mathbf{n}}_i^T [R] \hat{\mathbf{j}}_{B,i}) \quad (3.9)$$

I versori dei giunti  $\hat{\mathbf{j}}_{A,i}$  e  $\hat{\mathbf{j}}_{B,i}$  vengono calcolati imponendo che, quando il robot si trova nella *homeposition*, i versori di ciascun link coincidano

con quelli dei giunti corrispondenti, e quindi:

$$\hat{\mathbf{j}}_{A,i} = \frac{B_{i,hp} - A_{i,hp}}{\|B_{i,hp} - A_{i,hp}\|_2} \quad \text{e} \quad \hat{\mathbf{j}}'_{B,i} = -\hat{\mathbf{j}}_{A,i} \quad (3.10)$$

Infine si procede confrontando gli angoli di snodo  $\theta_{A,i}$  e  $\theta_{B,i}$  con i relativi valori limite  $\theta_{A,lim}$  e  $\theta_{B,lim}$ .

2. **Vincolo cinetostatico:** se tutti i vincoli cinemantici vengono rispettati si passa al calcolo della moltiplica statica delle forze per una data posa del robot per ciascuna forza di attuazione:

$$\tau_{max,i}(\mathbf{X}) = \sum_j | [J_{n,ij}(\mathbf{X})]^T | \quad (3.11)$$

Fra le sei moltipliche statiche delle forze appena calcolate si prende la massima:

$$\tau_{MAX}(\mathbf{X}) = \max_i (\tau_{max,i}) = \| [J_n(\mathbf{X})]^T \|_\infty \quad (3.12)$$

Si può notare che per il calcolo di  $\tau_{MAX}$  è stata utilizzata la versione normalizzata della matrice Jacobiana così come definito nel capitolo della cinematica e cinetostatica. L'ultimo passo è il confronto diretto con il limite imposto sulla moltiplica statica delle forze:

$$\tau_{MAX}(\mathbf{X}) \leq \tau_{MAX,lim} \quad (3.13)$$

3. **Vincoli geometrici:** per vincoli geometrici si intende il rispetto delle distanze minime fra i link e fra i link e la piattaforma. La distanza minima fra un elemento e l'altro è stata calcolata come distanza minima fra due segmenti nello spazio. Essendo un'analisi preliminare, e non avendo quindi a disposizione un disegno costruttivo della piattaforma, è stata schematizzata come un cilindro di 50cm con asse perpendicolare al piano su cui giacciono i giunti di piattaforma. I valori di distanza minima sono stati scelti in maniera tale da garantire un margine di sicurezza. Le verifiche che verranno fatte per ciascun punto dello spazio di lavoro saranno quindi:

$$\begin{cases} d_{link,ij} \geq d_{link,min} & \forall i,j \quad \text{con} \quad i \neq j \\ d_{link-platform,i} \geq d_{link-platform,min} & \forall i \end{cases} \quad (3.14)$$

Affinchè un qualsiasi punto, con un'orientazione qualsiasi sia verificato è necessario che ogni condizione di vincolo venga rispettata, altrimenti la porzione di volume associata a tale punto verrà considerata come non coperta.

### 3.7 Risultati dell'ottimizzazione

Per ottimizzare le macchine devono essere imposti dei limiti inferiori e superiori di variazione dei parametri, oltre ai limiti che servono per verificare che i vincoli imposti siano rispettati o meno. Le strategie adottate per il processo di ottimizzazione sono state le seguenti:

- **riduzione graduale della soglia di moltiplica statica delle forze :** Si comincia scegliendo un valore di  $\tau_{max}$  elevato e lo si riduce man mano fintanto che la copertura dello spazio di lavoro risulta pressochè totale. Ci si arresta quando le dimensioni dello spazio di lavoro esplorabile dalla macchina diventano inaccettabili.
- **riduzione del limite superiore della lunghezza dei link:** si è notato che l'algoritmo genetico converge sempre ad un risultato in cui la dimensione dei link si attesta attorno al limite superiore imposto. Questa dimensione è un indice di quello che sarà l'ingombro finale della macchina, quindi sarebbe auspicabile che fosse il più basso possibile. Una volta trovata la soglia minima di moltiplica statica delle forze che permette di avere una copertura decente dello spazio di lavoro, si procede riducendo il valore massimo di lunghezza dei link, e anche in questo caso ci si arresta quando le dimensioni dello spazio coperto non risultano più accettabili.

Per le macchine a cinematica parallela uno dei fattori limitanti è rappresentato dal range di mobilità dei giunti, soprattutto per quanto riguarda i giunti di piattaforma. Per tutte le macchine sono stati presi in esame i giunti sferici della Hephaist Seiko con un range di mobilità di  $\pm 25^\circ$ , tuttavia essi non permettono di avere una copertura sufficiente dello spazio di lavoro pur imponendo delle soglie di moltiplica statica elevate. Si è deciso quindi di considerare l'utilizzo di giunti cardanici che, pur presentando maggiori problemi di giochi rispetto agli sferici, presentano un range di mobilità più ampio. Per le analisi sono stati considerati dei giunti cardanici comuni con un range di mobilità pari a  $\pm 40^\circ$  su entrambi gli assi di rotazione della crociera.

### 3.7.1 Piattaforma di Gough-Stewart

In Tab. 3.2 vengono indicati i valori minimi e massimi che i parametri della piattaforma di Gough-Stewart possono assumere.

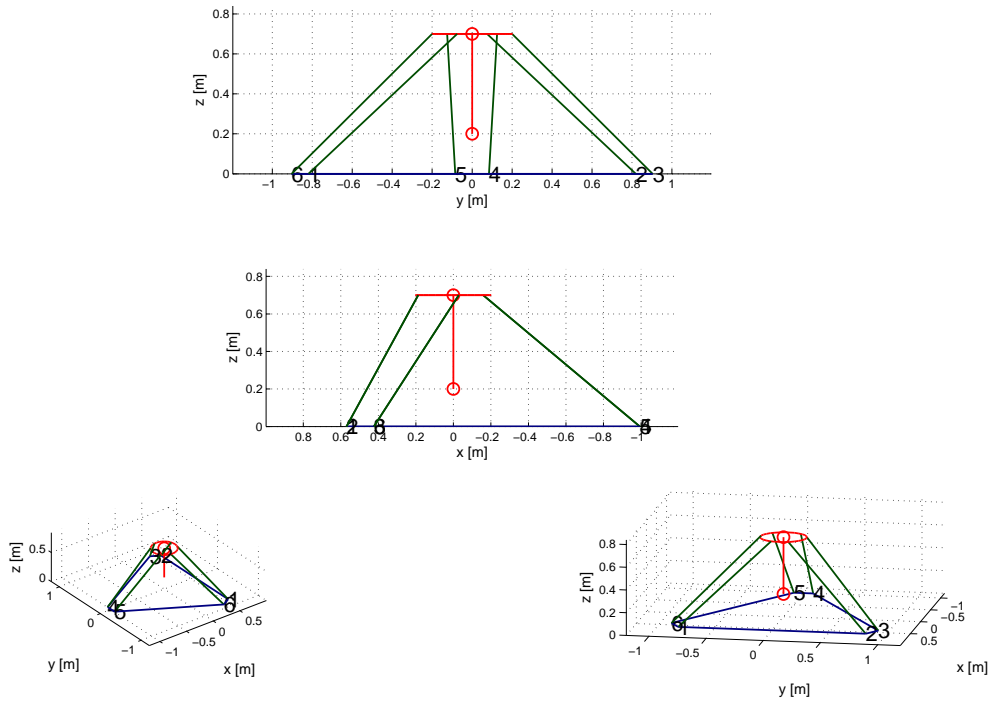
Variabile	unità di misura	Range di variazione
$R_b$	[m]	0.6 ÷ 1
$\theta_b$	[°]	0 ÷ 60
$R_p$	[m]	0.2 ÷ 0.6
$\theta_p$	[°]	0 ÷ 60

**Tab. 3.2:** Range di variazione dei parametri della piattaforma di Gough-Stewart.

Dopo aver visto che con un range di mobilità dei giunti pari a  $25^\circ$  la copertura dello spazio di lavoro non era accettabile, si è passati ad un range di mobilità di  $40^\circ$ . In questo ultimo caso la copertura dello spazio di lavoro non è totale, ma risulta molto migliore rispetto al caso precedente. Da una ricerca bibliografica è risultato che in genere la lunghezza massima di ogni link debba essere pari a

circa una volta e mezza la lunghezza corrispondente ad una situazione in cui lo stelo è completamente rientrato. Per questa ragione i limiti superiore ed inferiore della lunghezza dei pistoni sono stati inizialmente imposti pari a  $1.3\text{ m}$  e  $0.9\text{ m}$ . La copertura migliore dello spazio di lavoro si ha per un valore di  $\tau$  pari a 5. Successivamente si è provato, senza successo, ad usare dei pistoni di dimensioni inferiori, per cui i limiti di lunghezza sono diventati  $1\text{ m}$  e  $0.7\text{ m}$ . L'andamento della funzione di costo al variare dei diversi parametri è riportata in Fig. 3.10, insieme ai dati relativi all'Hexaslide.

Il disegno costruttivo della macchina è rappresentato in Fig. 3.1

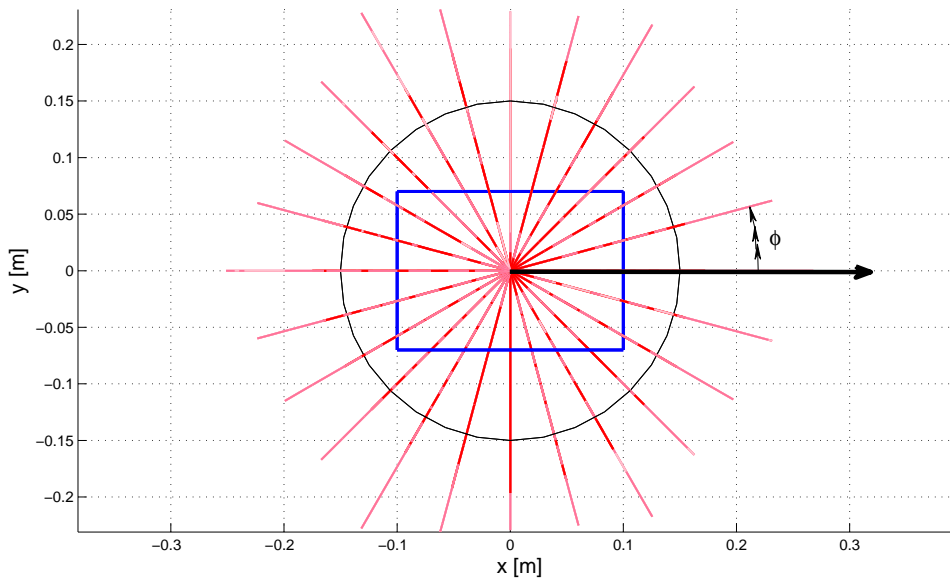


**Fig. 3.1:** Struttura finale della piattaforma di Gough-Stewart.

In Fig. 3.3 vengono messi a confronto lo spazio di lavoro desiderato con quello raggiungibile dalla macchina per vari valori di range di mobilità dei giunti.

Dopo aver sezionato lo spazio di lavoro con dei piani così come mostrato in Fig. 3.2, si riportano per ciascuno di essi:

- mappe della distanza minima fra i link Fig. 3.4, e Fig. 3.5.
- mappe della distanza minima fra link e piattaforma Fig. 3.6, e Fig. 3.7.
- mappe della moltiplica statica delle forze Fig. 3.8, e Fig. 3.9.



**Fig. 3.2:** Sezionamento dello spazio di lavoro con piani verticali.

### 3.7.2 Hexaslide

Nel caso dell'Hexaslide i range di variazione dei parametri sono quelli riportati in Tab. 3.3.

Variabile	unità di misura	Range di variazione
$s$	[m]	$0.1 \div 0.6$
$l_{link}$	[m]	$0.6 \div l_{max}$
$R_p$	[m]	$0.2 \div 0.6$
$\theta_p$	[°]	$0 \div 60$

**Tab. 3.3:** Range di variazione dei parametri dell'Hexaslide.

Il grafico in Fig. 3.10 mostra il confronto fra l'Hexaslide e la piattaforma di Stewart al variare della moltiplica statica e per diversi limiti di lunghezza dei link e di range di mobilità dei giunti.

Si considera dapprima il caso che prevede l'uso di giunti sferici. Viene imposto come valore iniziale  $\tau_{max} = 9$  e si mette in atto la strategia di ottimizzazione sopra descritta. Non riuscendo ad avere una copertura decente dello spazio di lavoro nemmeno per valori così elevati di  $\tau_{max}$ , si capisce che la causa di tale risultato è da attribuirsi ai limiti troppo stringenti imposti dai giunti sferici. Si passa quindi a considerare il caso con i giunti cardanici. A questo punto si cerca di abbassare il valore di  $l_{max}$ , precedentemente posto pari a  $1.3m$ . Dal grafico in Fig. 3.10 si può notare che il minimo valore di soglia di moltiplica statica delle forze per cui si riesce ad avere una copertura totale dello spazio di lavoro è 3. Tuttavia



portando il valore di  $l_{max}$  ad  $1\text{ m}$ , il volume di lavoro non risulta completamente coperto. Si prende quindi in esame il caso con  $\tau_{max} = 5$  e  $l_{max} = 1\text{ m}$ , accettando di avere valori di moltiplica statica più elevati pur di ridurre le dimensioni finali della macchina. In Fig. 3.11 si può vedere la forma che assumerà la macchina.

In Fig. 3.12 viene riportato il volume ad orientazione totale raggiungibile dalla macchina al variare del range di mobilità dei giunti. Si vede come solo per i range più ampi si riesca ad avere una copertura totale dello spazio di lavoro desiderato, sia nel caso relativo alle barche a vela, sia in quello relativo ai generatori eolici.

Si riportano infine le altre informazioni frutto dell'ottimizzazione cinetostatica per vari piani ottenuti sezionando lo spazio di lavoro ad orientazione totale analogamente a quanto fatto per la piattaforma di Gough-Stewart:

- mappe della distanza minima fra i link Fig. 3.13, e Fig. 3.14.
- mappe della distanza minima fra link e piattaforma Fig. 3.15, e Fig. 3.16.
- mappe della moltiplica statica delle forze Fig. 3.17, e Fig. 3.18.

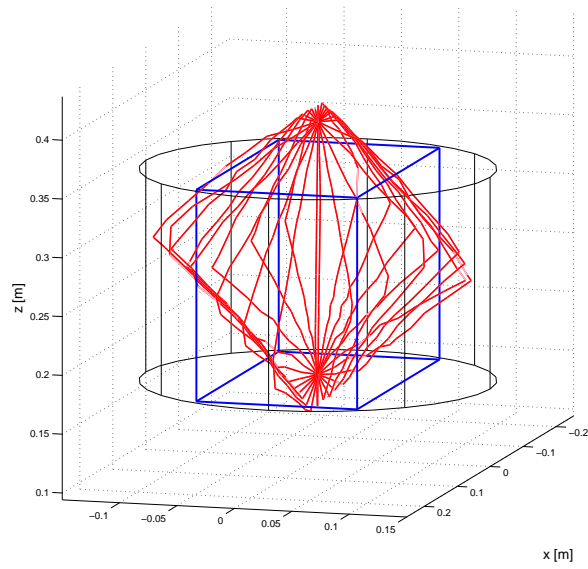
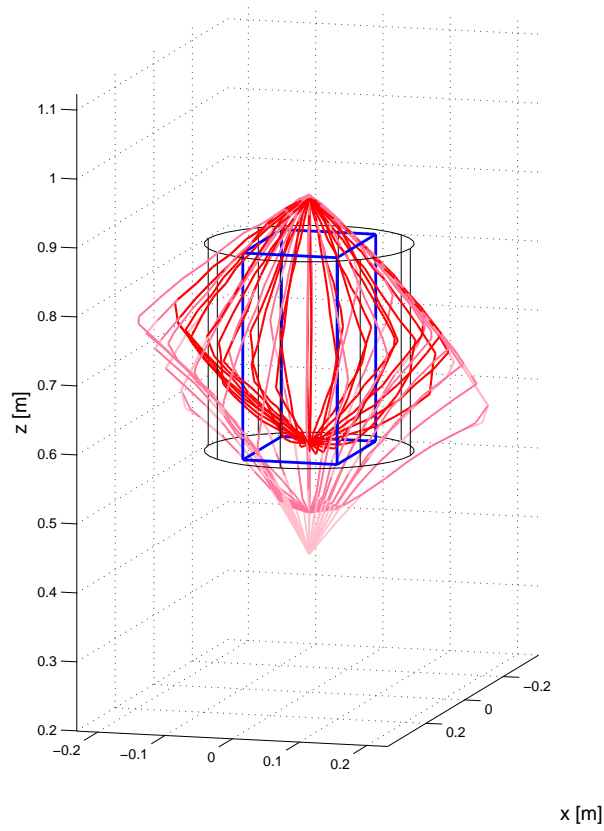
### 3.7.3 Hexaglide

Nel caso dell'Hexaglide, la moltiplica statica delle forze è stata calcolata dapprima utilizzando la matrice Jacobiana  $[J]$ , e successivamente la matrice Jacobiana normalizzata  $[J_n]$ , introducendo la lunghezza caratteristica come descritto nel Cap. 2. In Fig. 3.19 vengono riportati i valori massimi delle azioni di trazione e compressione dei link per diversi valori di  $\tau_{max}$ . Si nota che per  $\tau_{max} = 10$ , le macchine ottimizzate usando la lunghezza caratteristica sono soggette a sollecitazioni maggiori, mentre per  $\tau_{max} = 5$  accade il contrario. L'ottimizzazione senza lunghezza caratteristica è stata condotta solo per  $\tau_{max} = 10$  in quanto per valori inferiori di soglia di moltiplica statica si ha una scarsa copertura dello spazio di lavoro.

Per  $\tau_{max} = 3$ , pur utilizzando la lunghezza caratteristica, il volume di lavoro non coperto dalla macchina risulta troppo ampio, pertanto la soluzione presa in considerazione è quella con  $\tau_{max} = 5$ , la cui frontiera di Pareto è riportata in Fig. 3.20.

Tra tutti gli individui che si collocano sulla frontiera di Pareto è stato scelto quello che permette di avere una copertura totale dello spazio di lavoro, il cui disegno è riportato in Fig. 3.21.

In Fig. 3.22, fino a 3.24, sono infine riportati sul generico piano  $yz$  rispettivamente le mappe relative a distanza fra i link, distanza link-piattaforma, forza massima lungo i link.



**Fig. 3.3:** Volume ad orientazione totale raggiungibile dalla piattaforma di Gough-Stewart.

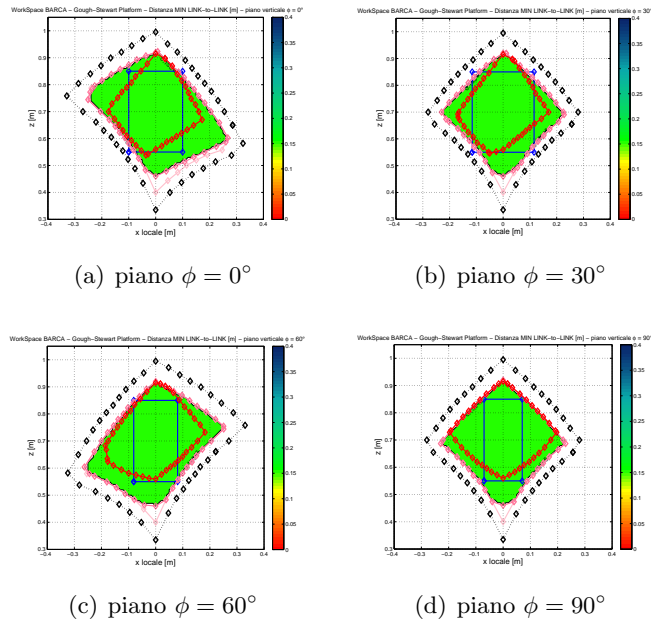


Fig. 3.4: Piattaforma di Gough-Stewart: distanza fra i link per vari piani, caso barche.

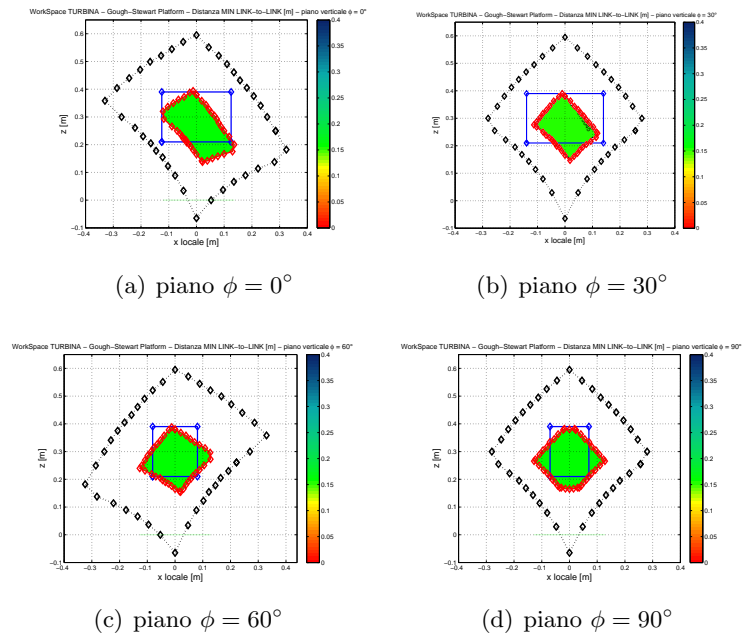
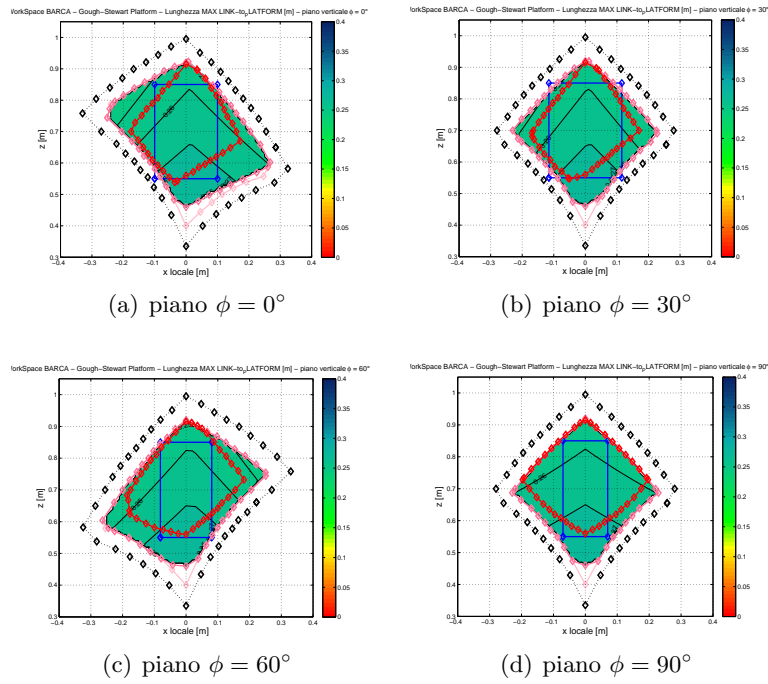
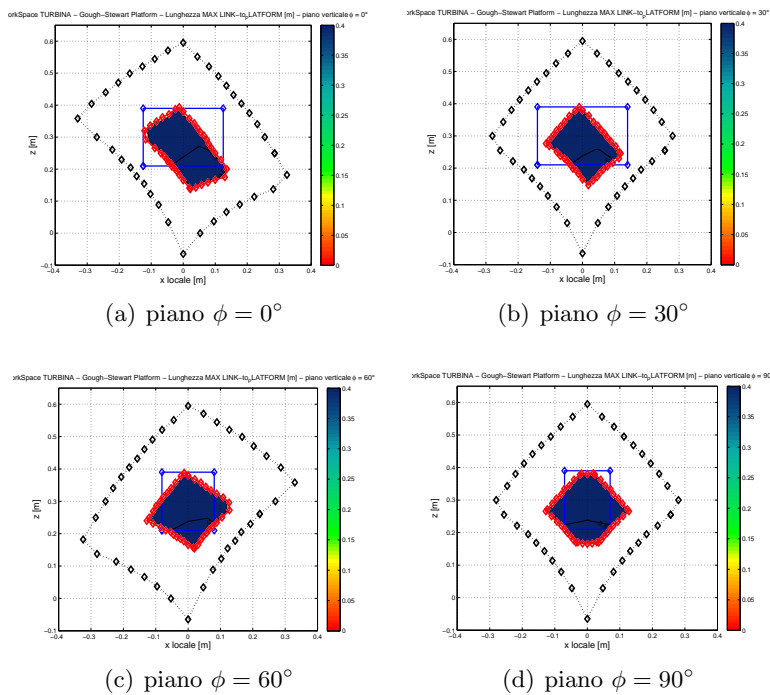


Fig. 3.5: Piattaforma di Gough-Stewart: distanza fra i link per vari piani, caso turbine.



**Fig. 3.6:** Piattaforma di Gough-Stewart: distanza link-piattaforma, caso barche.



**Fig. 3.7:** Piattaforma di Gough-Stewart: distanza link-piattaforma, caso turbine.

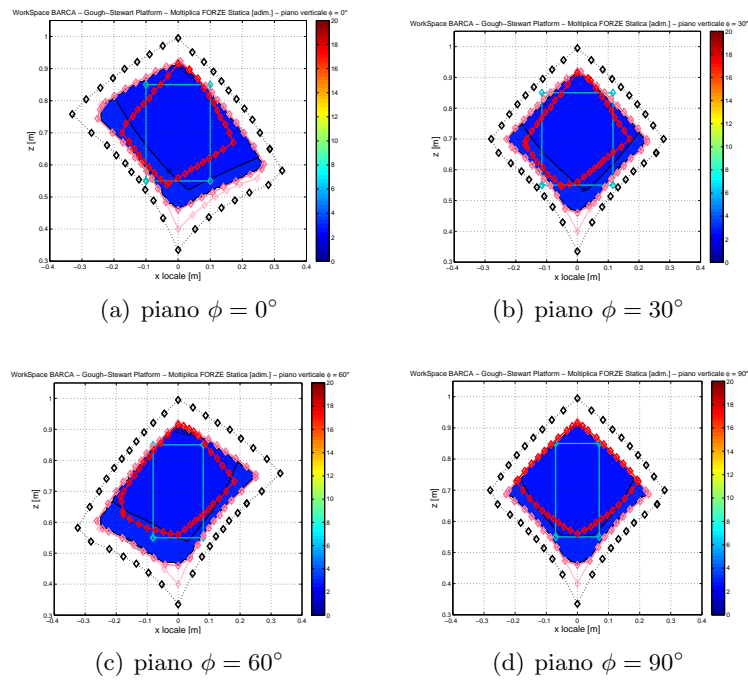


Fig. 3.8: Piattaforma di Gough-Stewart: moltiplica delle forze, caso barche.

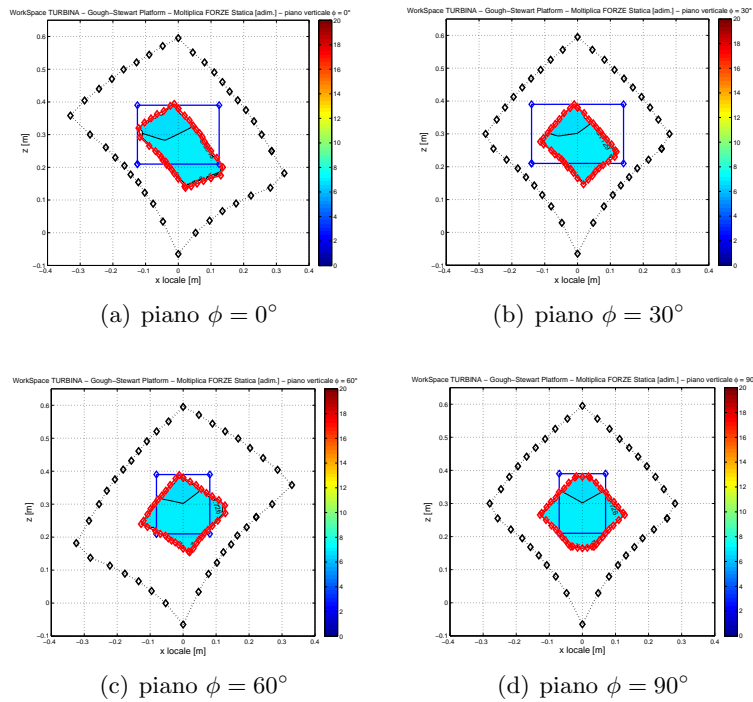
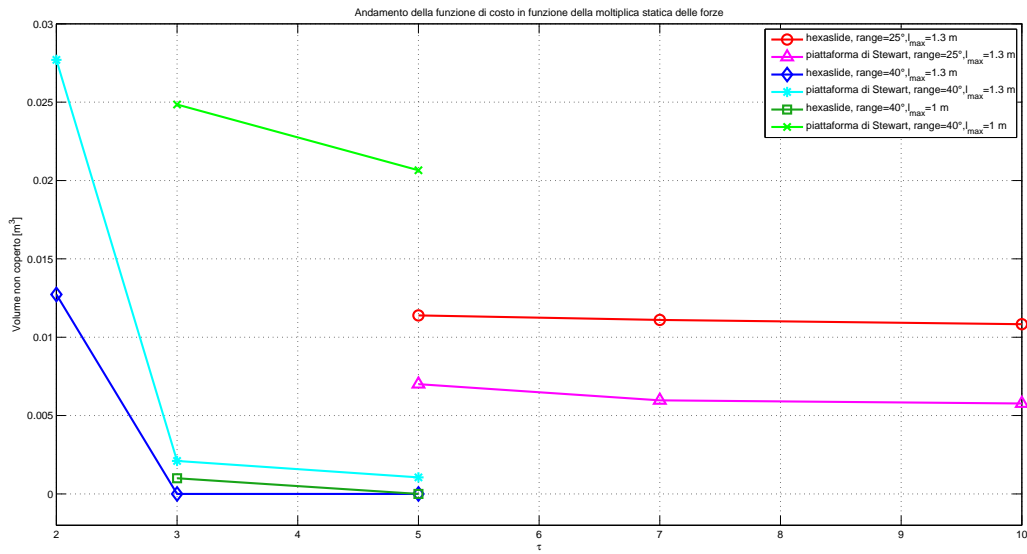
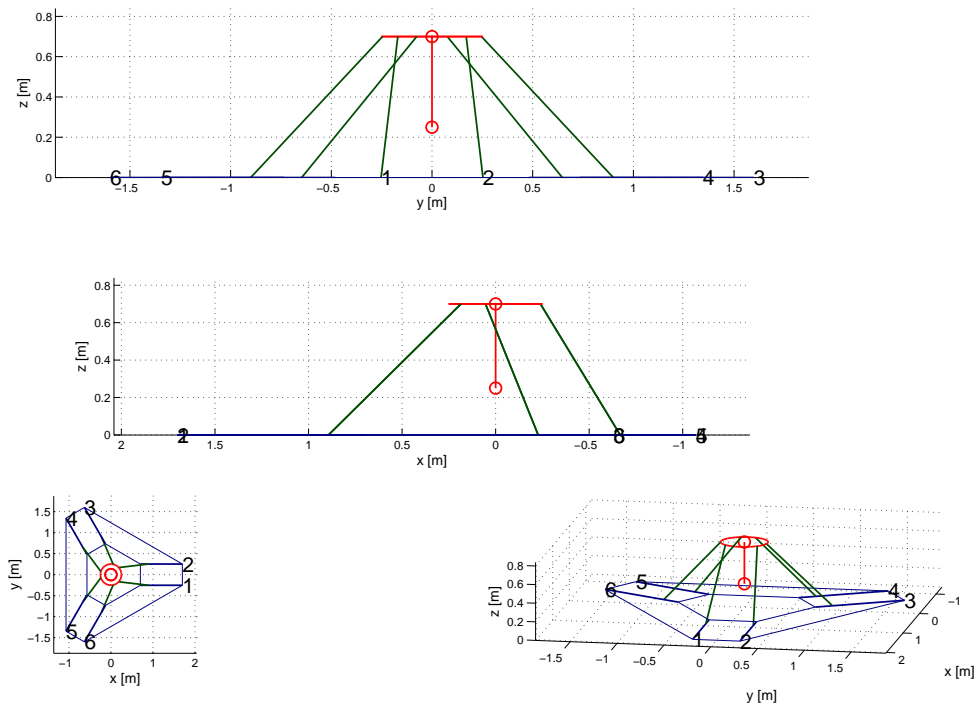


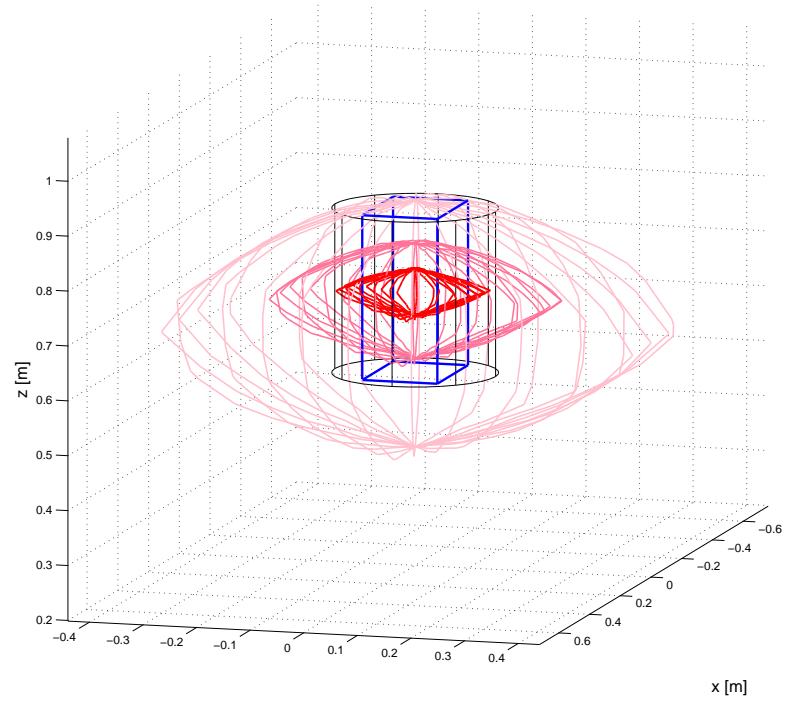
Fig. 3.9: Piattaforma di Gough-Stewart: moltiplica delle forze, caso turbine.



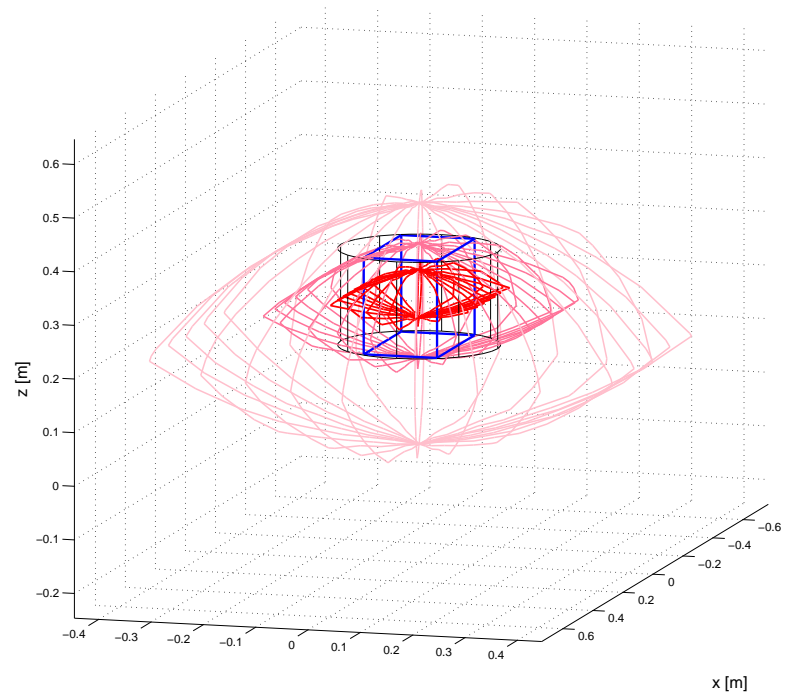
**Fig. 3.10:** Volume di lavoro non coperto al variare di  $\tau$  per Hexaslide e piattaforma di Stewart



**Fig. 3.11:** Struttura finale dell'Hexaslide



(a) WS barche



(b) WS turbine

**Fig. 3.12:** Volume ad orientazione totale raggiungibile dall'Hexaslide.

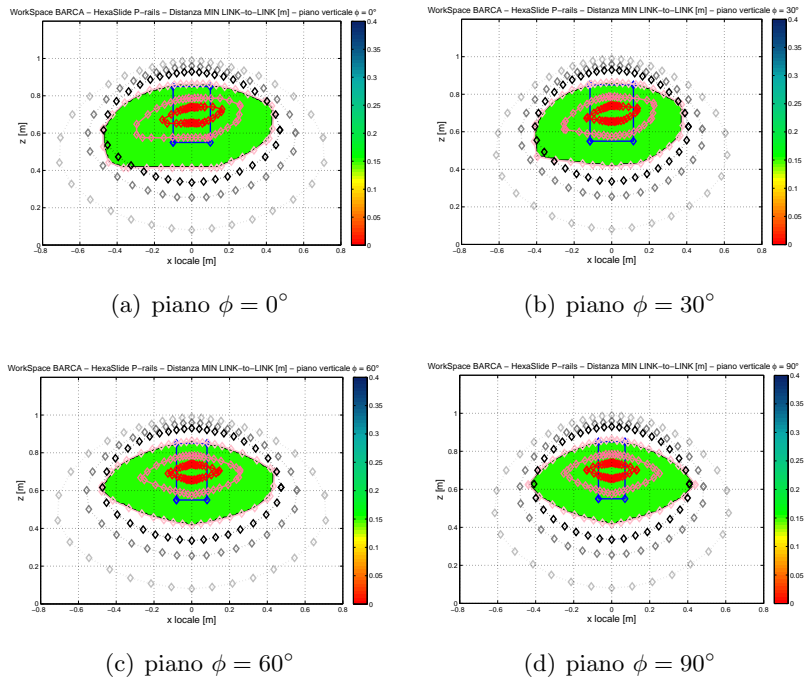


Fig. 3.13: Distanza fra i link per vari piani, caso barche.

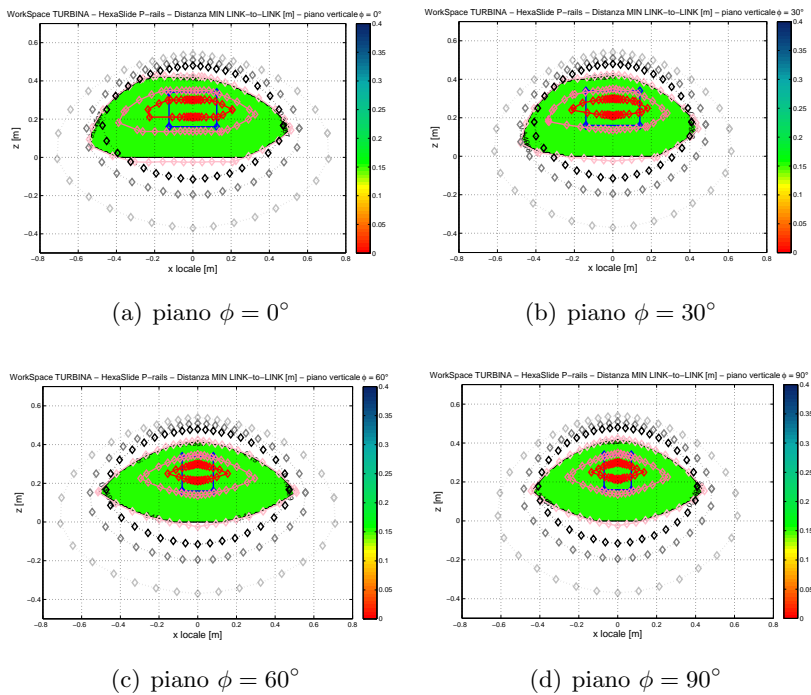


Fig. 3.14: Distanza fra i link per vari piani, caso turbine.



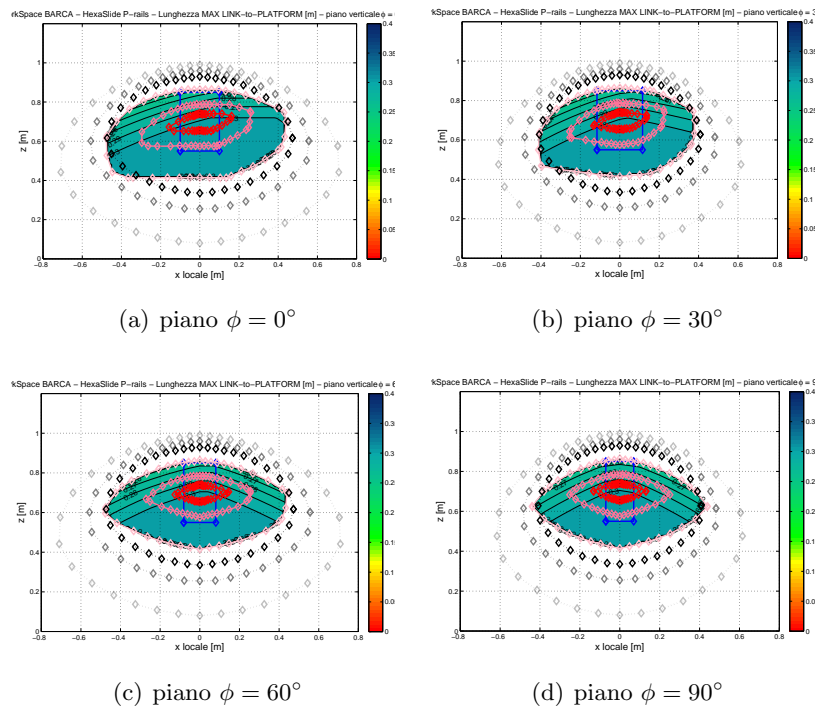


Fig. 3.15: Distanza fra i link e la piattaforma per vari piani, caso barche.

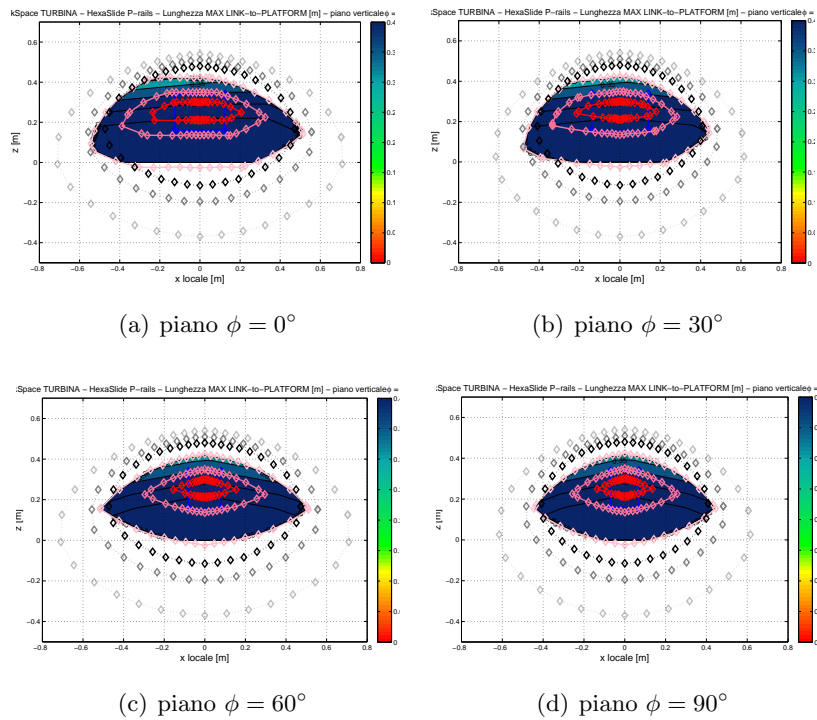


Fig. 3.16: Distanza fra i link e la piattaforma per vari piani, caso turbine.

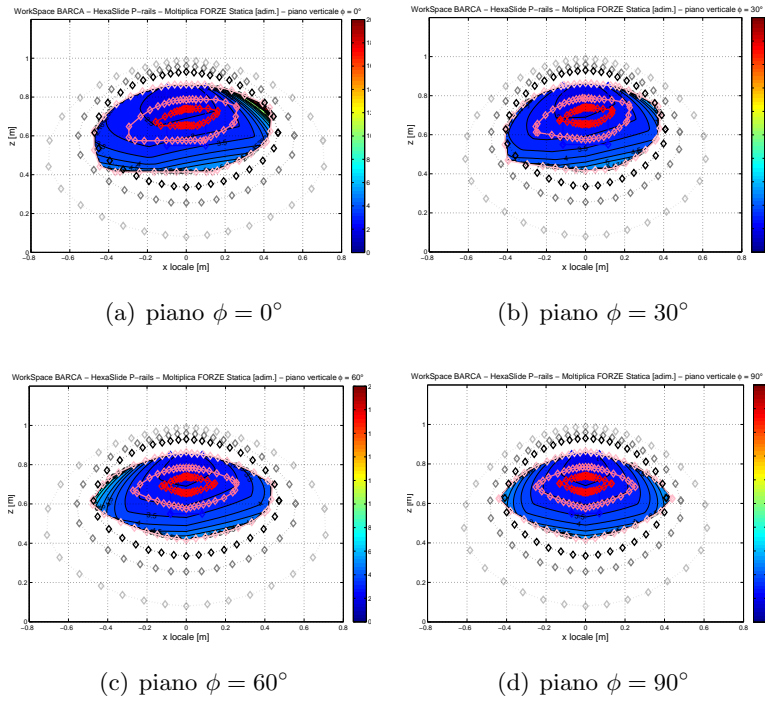


Fig. 3.17: Moltiplica statica delle forze per vari piani, caso barche.

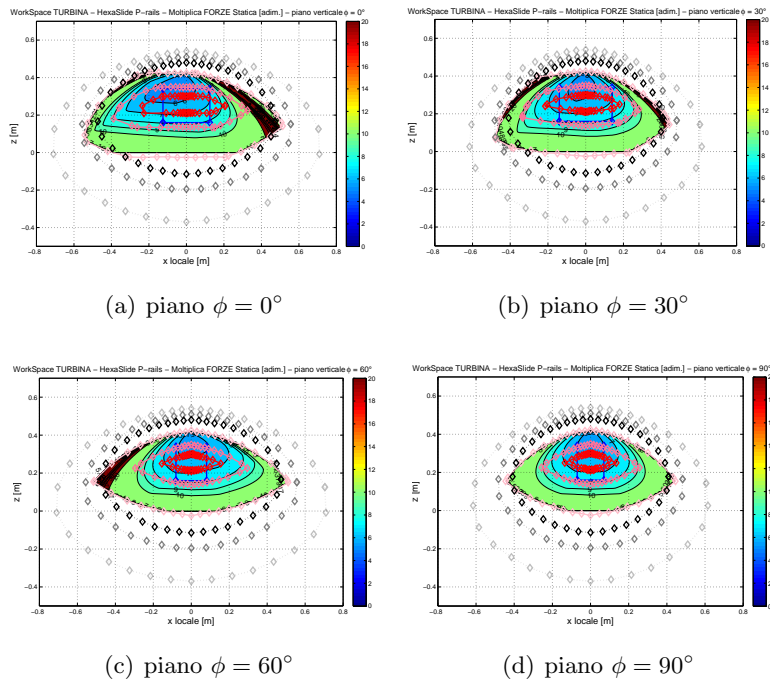


Fig. 3.18: Moltiplica statica delle forze per vari piani, caso turbine.

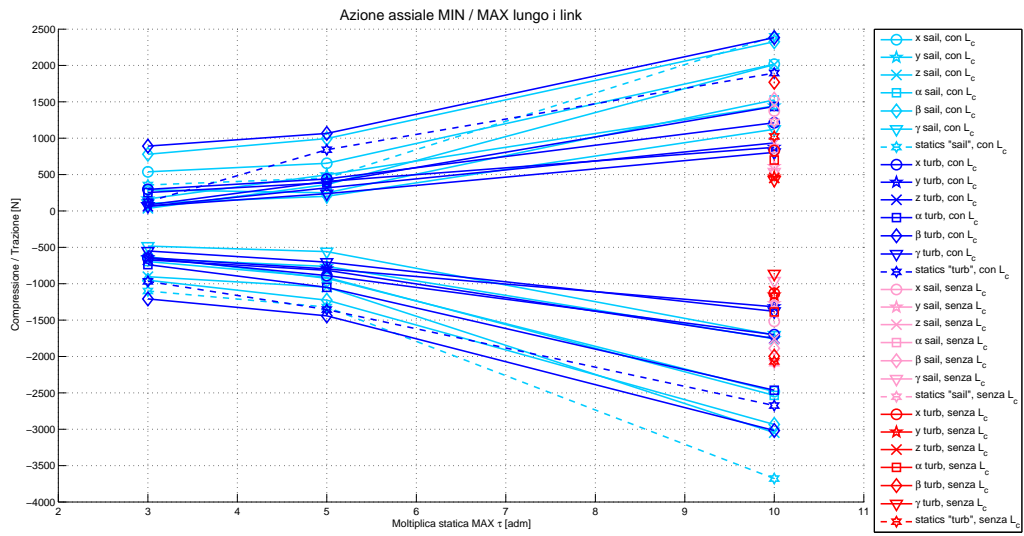


Fig. 3.19: Azioni interne dei link per macchine ottimizzate con diversi valori di  $\tau_{max}$ .

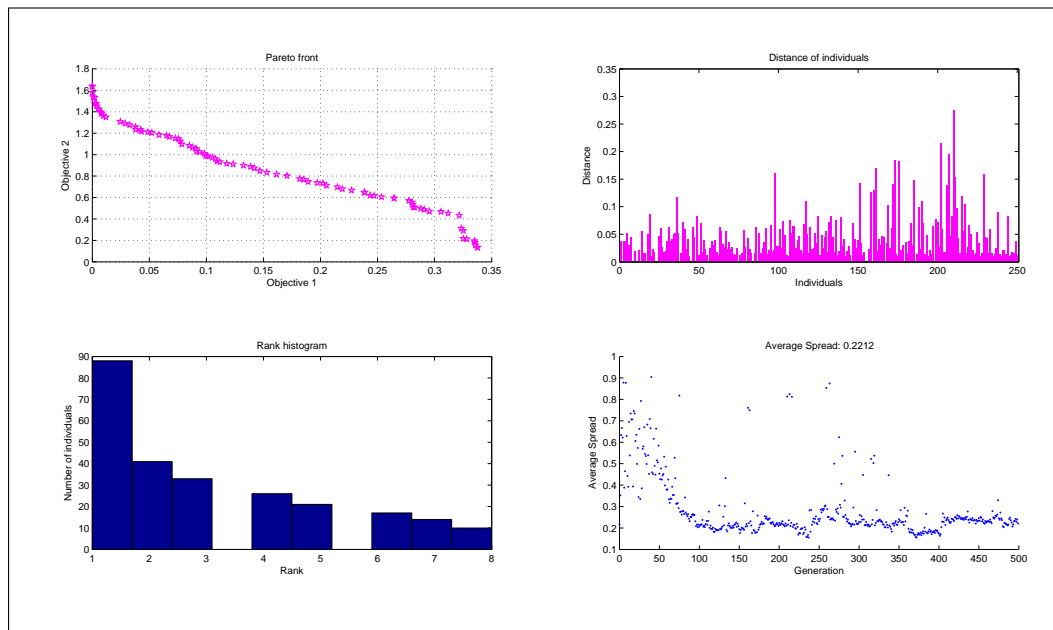
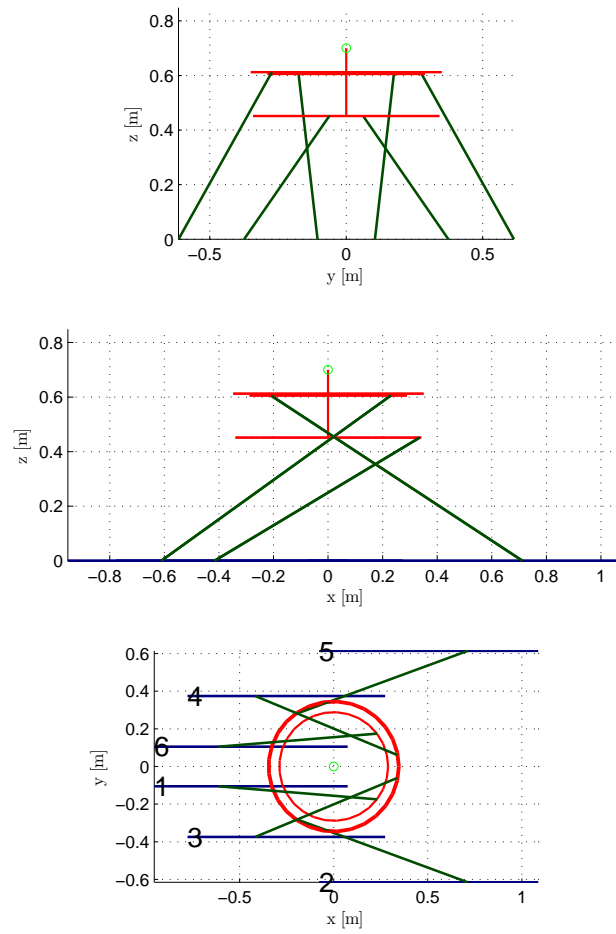
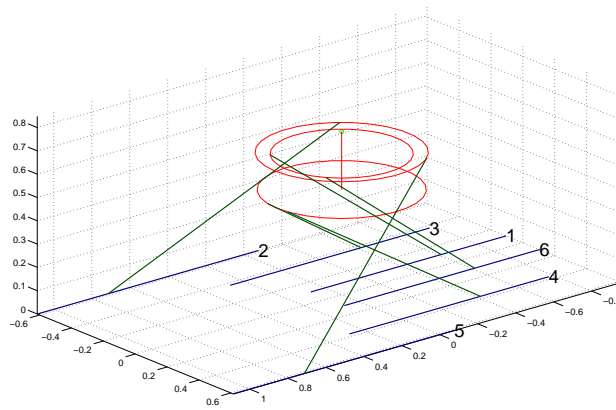


Fig. 3.20: Frontiera di Pareto per il caso  $\tau_{max} = 5$  con  $L_c$ .



(a) Vista frontale, laterale, e dall'alto



(b) Vista 3D

**Fig. 3.21:** Volume ad orientazione totale raggiungibile dall'Hexaslide.

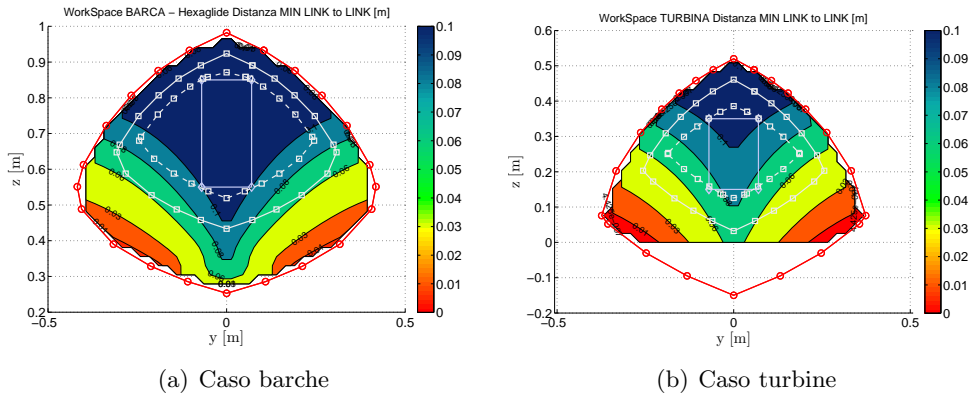


Fig. 3.22: Distanza minima fra i link.

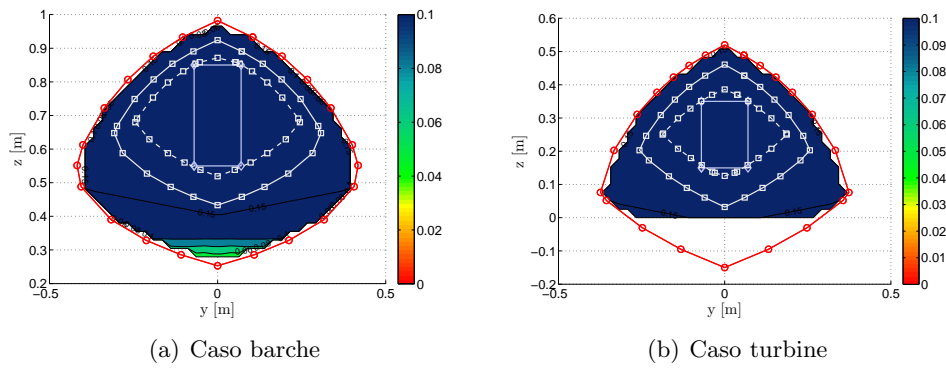


Fig. 3.23: Distanza minima fra i link.

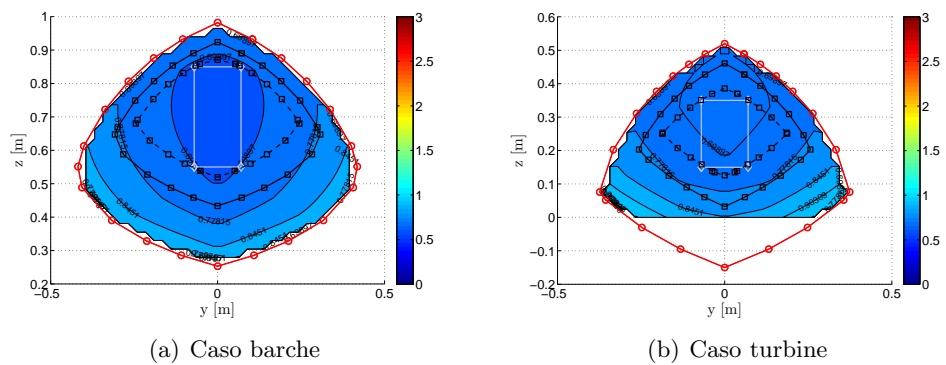


Fig. 3.24: Distanza minima fra i link.



## CAPITOLO 4

### ANALISI DI VIBRAZIONI

Il passo successivo è rappresentato dall'analisi vibrazioni fatta sulle macchine Hexaglide ed Hexaslide. Alla fine del processo di sintesi ci si è chiesti se la flessibilità degli elementi delle macchine avrebbe inficiato sul loro corretto funzionamento. Nel caso particolare di Hexaglide ed Hexaslide si è voluto verificare che i link, costituiti da cilindri cavi e snelli, non fossero fonte di vibrazioni eccessive durante le fasi di movimentazione della piattaforma. Non è stato ritenuto interessante lo sviluppo di un modello flessibile per i pistoni idraulici che costituiscono i link della piattaforma di Gough-Stewart. Questi infatti si presentano come cilindri tozzi e quindi meno soggetti a vibrazione di tipo flessionale, senza contare il fatto che gli aspetti idro-dinamici dell'olio in pressione presente al loro interno non possono essere trascurati e quindi richiedono una modellazione accurata. In letteratura sono presenti numerose trattazioni riguardanti la piattaforma di Stewart dove i pistoni vengono schematizzati come delle molle considerando il contributo puramente assiale di rigidità di tali elementi (vedi [15]). Al contrario ci si aspetta che i modi di vibrare alle più basse frequenze delle altre due macchine coinvolgano le rigidità di tipo flessionale. Per queste ragioni è stato creato un programma in ambiente ADAMS che valuti come cambino le prime frequenze proprie della struttura al variare delle pose assunte dal robot. Va sottolineato il fatto che le analisi vengono effettuate imponendo condizioni di velocità ed accelerazione nulle, sia per i giunti attuati che per la piattaforma mobile. Per entrambe le macchine sono stati considerati tre scenari: (1) trasmissioni infinitamente rigide; (2) trasmissioni flessibili a cinghia; (3) trasmissioni flessibili a vite a ricircolo di sfere.

Il codice completo relativo al caso dell'Hexaslide con trasmissione a vite è riportato in Appendice B.

#### 4.1 Marker di riferimento e loro parametrizzazione

Sia per la creazione del modello che per la successiva movimentazione sono stati creati dei marker di riferimento. Un marker costituisce un sistema di riferimento, e pertanto sarà caratterizzato da un'origine, le cui coordinate rispetto al sistema di riferimento globale o rispetto ad un altro marker in ADAMS sono indicate in *location*, e da un'orientazione degli assi, i cui parametri angolari sono indicati in *orientation*. In particolare è stato fissato un marker denominato *Origine*, che

rappresentasse la terna del sistema di riferimento globale, ed uno denominato *TCP*, che rappresentasse la terna mobile solidale con la piattaforma. Successivamente sono stati creati, sfruttando la cinematica inversa, i marker  $A_i$  e  $B_i$  corrispondenti ai centri dei giunti di base e di piattaforma. Entrambi i tipi di marker sono stati parametrizzati rispetto al TCP, in modo tale che cambiando la posizione e l'orientazione di quest'ultimo marker, automaticamente si riposizionassero anche tutti gli altri: a tale scopo è stata utilizzata la seguente sintassi di ADAMS:

```
marker create marker_name=M_1 &
location=loc_relative_to ({x_P, y_P, z_P}, M_ref)
```

dove  $x_P, y_P, z_P$  esprimono le coordinate del generico punto  $P$ , espresse nel sistema di riferimento costituito dal marker di riferimento  $M_{ref}$ . Per i marker  $B_i$ , fissi rispetto alla terna mobile si è adottato una formulazione del tipo

```
marker create marker_name=B_i &
location=loc_relative_to ({x_B_i, y_B_i, z_B_i}, TCP)
```

dove le coordinate sono state calcolate una volta noti i parametri caratteristici delle macchine. Nel caso dei marker  $A_i$  invece, si è scelto come marker di riferimento l'Origine, e le coordinate sono state calcolate considerando che:

$$\overline{\mathbf{A}_i \mathbf{O}} = \mathbf{s}_i + q_i \hat{\mathbf{u}}_i \quad (4.1)$$

Essendo  $q_i$  un parametro che cambia al variare della posa del robot, la posizione dei giunti di base rispetto all'origine sarà variabile e quindi sarà necessario ricorrere a delle variabili di appoggio di tipo reale. Per i soli giunti di base è utile impostare oltre alla posizione anche l'orientazione dei marker. Analogamente a quanto fatto per la posizione, si ricorrerà ad una funzione di parametrizzazione. Per motivi che verranno chiariti in seguito, è utile che l'asse  $z$  dei marker  $A_i$  si trovi lungo la congiungente dei centri dei giunti di base e di piattaforma, cioè lungo il versore dell' $i$ -esimo link. La sintassi usata in questo caso è:

```
marker modify marker_name=M_1
orientation=(ori_along_axis (M_2, M_1, "z"))
```

in cui  $M_1$  ed  $M_2$  sono i punti (o meglio marker) che individuano l'unica retta passante per essi, e tra apici è indicato quale asse debba giacere su tale retta. Riadattando la funzione al caso in esame, si potrà scrivere:

```
marker modify marker_name=A_i
orientation=(ori_along_axis (B_i, A_i, "z"))
```

In seguito per agevolare la modellazione della macchina verranno creati dei nuovi marker, ed anche essi verranno parametrizzati rispetto ai marker di cui parlato finora.

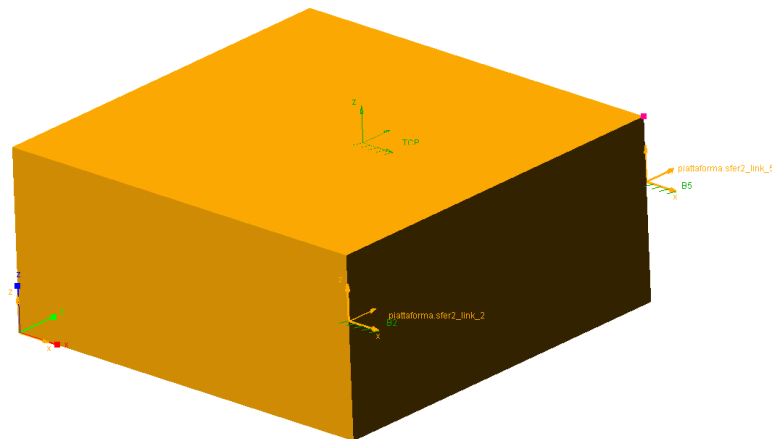
## 4.2 Modellazione piattaforma

Per entrambe le macchine le piattaforme sono state modellate come un corpo rigido. Questa scelta deriva dal fatto che non è ancora presente un disegno definitivo



della geometria che le piattaforme dovranno assumere. Inoltre si presuppone che vengano progettate in maniera tale da risultare molto rigide pur rimanendo leggere, l'idea infatti è quella di avere una struttura composta da piastre con opportune nervature posizionate in modo da contrastare l'inflessione delle piastre stesse. In ogni caso una volta che si avrà a disposizione una geometria definitiva si potrà creare un modello agli elementi finiti. In ADAMS è possibile, una volta definito un corpo rigido con geometria assegnata, renderlo flessibile grazie al pacchetto *viewflex*, tuttavia la mesh risultante è abbastanza grezza, in quanto offre la possibilità di utilizzare unicamente elementi di tipo tetraedrico (TET4 e TET10) ed inoltre non permette di infittire la mesh nelle zone più critiche. Di conseguenza sarà necessario ricorrere a programmi nati apposta per la modellazione agli elementi finiti come PATRAN e NASTRAN per poi importare la geometria meshata in ADAMS. In ogni caso il contributo vibrazionale di una piattaforma flessibile dovrebbe risultare trascurabile rispetto a quello dei link. Nei due casi le piattaforme sono state così modellate:

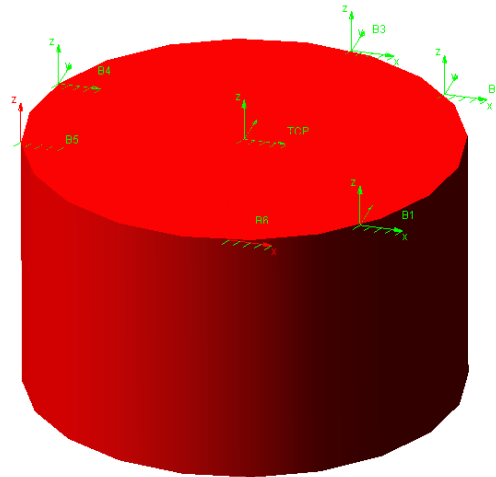
- **Hexaglide:** per la particolare conformazione assunta dalla macchina, si è deciso di modellare la piattaforma dell'Hexaglide come un parallelepipedo di dimensioni tali da contenere al suo interno i centri dei giunti di piattaforma (Fig. 4.1).



**Fig. 4.1:** Disegno della piattaforma dell'Hexaglide.

Si è ipotizzato una massa di  $50\text{kg}$  per la sola piattaforma, più un contributo aggiuntivo di  $11.64\text{kg}$  dovuti alla bilancia. Come centro di massa è stato scelto il centro del parallelepipedo, ed è stato anche esso parametrizzato in modo tale da mantenere sempre la stessa posizione ed orientazione rispetto al TCP.

- **Hexaslide:** per la simmetria che caratterizza l'Hexaslide, la piattaforma mobile è stata modellata come un cilindro, la cui circonferenza fosse coincidente con quella su cui giacciono i giunti di piattaforma, e con un'altezza di  $50\text{cm}$  (Fig. 4.2).



**Fig. 4.2:** Disegno della piattaforma dell'Hexaslide.

Anche in questo caso il centro di massa è stato preso coincidente con il centro del cilindro e la massa della piattaforma è stata scelta uguale a quella dell'Hexaglide.

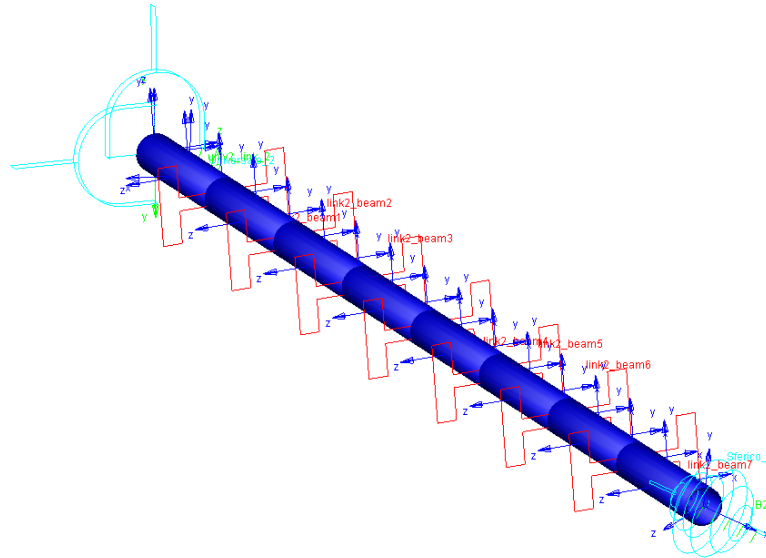
### 4.3 Modellazione dei link

Come già detto i link sono costituiti da cilindri cavi di acciaio. Essendo la loro lunghezza molto maggiore rispetto al diametro esterno, si è deciso di modellarli come degli elementi beam discreti. Questo tipo di elementi si presenta come una serie di segmenti identici fra loro che di fatto sono dei corpi rigidi, che sono legati gli uni agli altri da campi di forze che rappresentano le rigidità assiali, flessionali e torsionali della trave, così come mostrato in Fig. 4.3.

In Adams è presente un tool facente parte del pacchetto *viewflex* che definisce la geometria della sezione, il materiale, le rigidità, e i punti di inizio e di fine della beam, automaticamente crea questo tipo di corpi flessibili. Tuttavia, sfruttando questo pacchetto, non è possibile parametrizzare i marker appartenenti ai vari corpi, per cui è stato necessario creare un codice che ripercorresse tutti i passaggi del *viewflex*, ma dove ogni singolo marker fosse parametrizzato rispetto a quelli posizionati agli estremi della trave.

#### 4.3.1 Creazione segmenti

Ciascun segmento che compone un link è un corpo rigido, in particolare nel caso in esame sarà un cilindro cavo e quindi verrà creato come solido di rivoluzione. In Adams per creare un solido di rivoluzione bisogna definire dei punti che rappresentino i vertici di un poligono regolare. L'area di tale poligono rappresenta la sezione trasversale del solido che verrà fatta ruotare attorno al suo asse di rivoluzione. I vertici del poligono verranno definiti rispetto ad un marker che nei codici è stato



**Fig. 4.3:** Disegno dei link.

definito  $gm_j$ , dove  $j$  rappresenta il numero dell'elemento del singolo link; inoltre l'asse  $z$  di questo marker rappresenterà l'asse di rivoluzione del solido. Il marker  $gm_j$  quindi definisce completamente il solido di rivoluzione ad esso associato, per cui sarà sufficiente parametrizzare la sua posizione ed orientazione per far sì che il solido si sposti al cambiare della posa del robot. Per la parametrizzazione sono state usate le seguenti funzioni:

```
marker create marker_name=gm_j &
location=loc_along_line(A_i, B_i, L_seg*(j-1)) &
orientation=ori_along_axis(A_i, B_i, "z")
```

#### 4.3.2 Creazione campi di forze di tipo beam

Le varie rigidzze che legano un segmento al successivo sono state introdotte per mezzo di campi di forze di tipo *element like beam*. Questa forza che rappresenta il legame elastico della trave viene applicata fra due marker che devono avere l'asse  $x$  giacente sulla stessa retta e gli assi  $y$  e  $z$  paralleli. A tal proposito sono stati creati i marker  $CG_j$  e  $CG_{j+1}$ , appartenenti rispettivamente al  $j$ -esimo elemento ed al  $(j+1)$ -esimo, ciascun dei quali avrà l'asse  $x$  diretto come il versore del link a cui appartengono gli elementi. La parametrizzazione dei marker  $CG$  è stata fatta nel seguente modo:

```
marker create marker_name=CG_j &
location=loc_along_line(A_i, B_i, L_seg*(j-0.5)) &
orientation=ori_along_axis(A_i, B_i, "z")
```

di conseguenza si troveranno esattamente al centro del cilindro cavo. Oltre ai due marker descritti prima, per creare il campo di forza è necessario fornire le informazioni relative al materiale di cui è costituita la trave ed i valori dei momenti d'inerzia di superficie. Avendo scelto l'asse  $x$  come asse della trave, i momenti di superficie relativi ai tre assi sono stati calcolati come:

$$I_{yy} = I_{zz} = \frac{\pi(R^4 - r^4)}{4} \quad I_{xx} = \frac{\pi(R^4 - r^4)}{2} \quad (4.2)$$

I dati relativi al materiale sono indicati in Tab. 4.1.

Nome	Valore
Modulo di Young (E)	206000 [MPa]
Coefficiente di Poisson ( $\nu$ )	0.29
Modulo di elasticità tangenziale (G)	79845 [MPa]

**Tab. 4.1:** Dati relativi al materiale dei link.

Il parametro  $L_{seg}$  che compare all'interno delle funzioni di parametrizzazione rappresenta la lunghezza di un singolo elemento e viene calcolata dividendo la lunghezza dell'intero link per il numero di segmenti  $n_{seg}$  in cui si è deciso di dividerlo. Questo parametro viene richiesto una volta lanciata la macro ed è preimpostato ad un valore pari a 8 segmenti.

## 4.4 Slider

Nel caso infinitamente rigido la modellazione delle macchine si arresta alla piattaforma ed ai link, in quanto, considerando le trasmissioni infinitamente rigide, gli slider rimarranno fermi e quindi il loro contributo inerziale sarà nullo. Diverso è il caso delle trasmissioni non ideali dove gli slider si comporteranno come delle masse traslanti. In questo secondo caso sono stati modellati come dei piccoli parallelepipedi il cui centro è stato preso coincidente con il centro dei giunti di base. Le forze che un singolo slider è in grado di scambiare col sistema di trasmissione, sia esso a cinghia o a vite senza fine, sono dirette lungo l'asse della guida corrispondente. Ciò è garantito dalla presenza di supporti che hanno il compito di scaricare direttamente a terra eventuali componenti di forze dirette perpendicolarmente all'asse delle guide ed i momenti derivanti dai link e dalla piattaforma mobile. Il contributo di forza visto dalle trasmissioni sarà quindi: di tipo assiale nelle cinghie; torsionale nelle sistema a vite. In questo secondo caso la trasformazione della forza assiale in una coppia torsionale è dovuto all'accoppiamento vite-madrevite.

## 4.5 Modellazione delle cinghie

Le trasmissioni a cinghia sono costituite da una fascia di poliuretano avvolta attorno a due puleggie, una delle quali è motorizzata. A questa fascia è poi agganciato

lo slider con il suo supporto. Per quanto detto l'unico tipo di forzante a cui sono soggette è quella assiale. All'interno della macro il loro contributo inerziale è stato trascurato, in quanto la fascia è di piccolo spessore e di bassa densità specifica, senza contare il fatto che tale contributo è di tipo traslante. Per queste ragioni, per simulare l'effetto delle cinghie si è fatto ricorso a delle forze di tipo elastico con rigidità variabile. Per cominciare in Adams sono stati definiti due marker per ciascuna guida appartenenti al *ground*; per la loro designazione sono stati scelti i nomi  $P_i$  e  $Q_i$ . La distanza fra i due è pari a  $1.2m$  ed entrambi si trovano sull'asse della guida ad una distanza di  $\pm 0.6m$  dalla posizione assunta dall' $i$ -esimo giunto di base nella *home-position*. In seguito sono state definite due forze nella seguente maniera:

$$F_i = c_{spec} \frac{l_i - l_0(q_i)}{l_0(q_i)} = k_{eq}(q_i)(l_i - l_0(q_i)) \quad (4.3)$$

dove  $l_0$  rappresenta la lunghezza della porzione di cinghia che va dallo slider ad uno dei marker  $P_i$  o  $Q_i$  calcolata prima che cominci l'analisi dinamica, mentre  $l_i$  indica la distanza effettiva fra slider e uno dei due marker a seguito di piccoli spostamenti dello slider. La costante  $c_{spec}$  rappresenta un valore equivalente del prodotto fra il modulo di Young della cinghia, che non sarà costante in quanto costituita da strati di materiali diversi, e la sezione equivalente della cinghia, che varierà a causa della presenza dei tasselli che servono per ingranare sulle puleggie. Questa costante avrà quindi le dimensioni di un  $[N]$ . Per il calcolo di  $l_0$  è stata definita una variabile di tipo reale definita nel seguente modo:

```
variable set variable_name=l_0 real=(eval(dm(slider_i.cm, P_i)))
```

La funzione *dm* calcola la distanza fra due marker utilizzando la norma euclidea. Si noti che il comando *eval* serve a valutare la funzione in un determinato istante, in questo caso prima che cominci l'analisi di vibrazioni, e a mantenere tale valore costante. La variabile  $l_0$  verrà aggiornata prima di ogni analisi dopo che il robot si è spostato nella nuova posa. È stato definito in seguito un nuovo marker ausiliario appartenente allo slider denominato *force-marker* posizionato in corrispondenza del centro di massa dello slider e con orientazione nulla rispetto al marker *Origine*. Infine è stata creata una forza di tipo *direct single\_component\_force* fra i marker *force-marker* e  $P_i$  con funzione:

```
function=c_spec/l_0*(dm(slider_i.force_marker, P_i)-l_0)
```

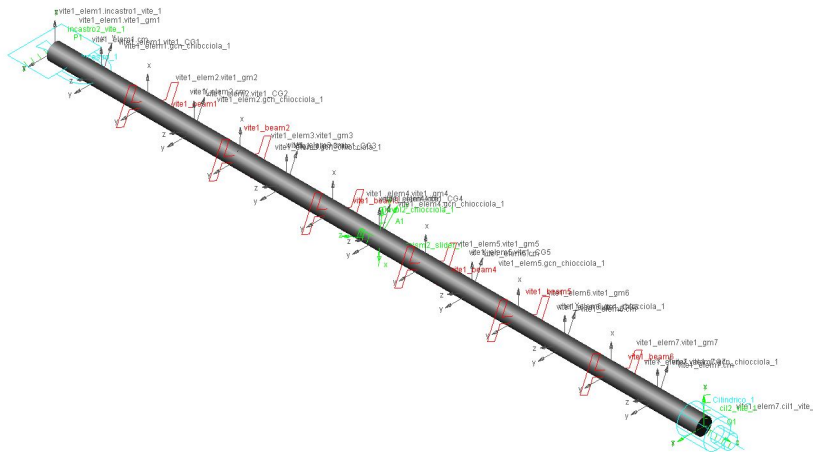
Ovviamente lo stesso procedimento viene ripetuto anche per la quota parte di cinghia compresa fra lo slider ed il marker  $Q_i$ .

In ADAMS è disponibile un tool all'interno del pacchetto *machinery* che permette di creare tutto il sistema di trasmissione a cinghia, comprese le puleggie. La cinghia in questo caso viene schematizzata come una serie di tasselli che interagiscono tra di loro per mezzo di campi di forza denominati *fields*. La ragione per cui non si è fatto ricorso a tale strumento risiede nel fatto che per creare una singola cinghia della lunghezza di  $1,2m$ , è necessario circa un centinaio di tasselli e di campi di forza. Ogni tassello di fatto si presenta come un corpo rigido a sé

stante, ed un numero così elevato di corpi rende i tempi di analisi e la quantità di memoria richiesta inaccettabili.

## 4.6 Modellazione vite senza fine

Il sistema di trasmissione a vite a ricircolo di sfere è costituito da barre d'acciaio filettate e da un elemento detto madrevite a filettatura concava. Tra i due elementi vengono inserite delle sferette che scorrono all'interno delle cave dei filetti. La rotazione della vite quindi mette in rotazione le sferette che a loro volta provocano un avanzamento della madrevite. Tale sistema risulta molto preciso, presenta giochi molto ridotti e possiede un elevato rendimento. Agli estremi della vite sono presenti dei cuscinetti che hanno il compito di assorbire eventuali azioni di tipo radiale e momenti. Uno degli estremi poi viene collegato per mezzo di un cuscinetto reggispinta all'albero di un motore elettrico. Infine un supporto in corrispondenza della madrevite, come già detto in precedenza, si occupa di scaricare sul telaio le forze provenienti dai link che non sono dirette lungo l'asse della vite. Il disegno finale in ADAMS sarà quello rappresentato in Fig. 4.4.



**Fig. 4.4:** Disegno delle vite senza fine.

Analogamente a quanto fatto per le cinghie, anche in questo caso sono stati creati i marker  $P_i$  e  $Q_i$ . In questo caso il marker  $P_i$  corrisponderà all'estremo della vite dove è presente il motore, che durante le analisi sarà a rotore bloccato, di conseguenza in corrispondenza di questo marker verrà imposto un vincolo di incastro rappresentato dall'unione del motore bloccato e del cuscinetto. In corrispondenza del marker  $Q_i$  invece verrà posizionato un vincolo di tipo cilindrico, ovvero un vincolo che lascia libera la traslazione e la rotazione lungo un determinato asse, in questo caso quello della vite. I passi seguiti per la modellazione dell'elemento flessibile sono identici a quelli del caso dei link, solo che in questo caso i segmenti saranno dei cilindri pieni di acciaio, inoltre, essendo i marker di riferimento  $P_i$  e  $Q_i$  fissi, non sarà necessaria una parametrizzazione accurata di ogni singolo marker.

I momenti di inerzia superficiali diventeranno in questo caso:

$$I_{yy} = I_{zz} = \frac{\pi R^4}{4} \quad I_{xx} = \frac{\pi R^4}{2} \quad (4.4)$$

A differenza delle cinghie, si è ritenuto che in questo caso il contributo inerziale della vite non potesse essere trascurato, in quanto rappresentato da un cilindro pieno in rotazione ad un elevato numero di giri. Ovviamente non si è tenuto conto della filettatura della vite e delle forze di contatto fra le sfere e la vite e fra le sfere e la madrevite. La ragione per aver scelto un modello semplificato sta nel fatto che lo scopo di questo lavoro è quello di avere un'idea di come cambino le frequenze proprie del sistema considerando la flessibilità della trasmissione e non quello di avere una descrizione minuziosa di quello che accade all'interno dell'accoppiamento vite-madrevite per cui si rimanda a [16].

## 4.7 Vincoli

I vincoli che sono stati imposti sono quelli classici che si trovano nella finestra constraint, tuttavia anche in questo caso i marker necessari alla loro creazione sono stati parametrizzati. Merita particolare attenzione il modo in cui è stato modellato l'accoppiamento fra vite e madrevite nel caso di trasmissione non ideale.

### 4.7.1 Giunti di piattaforma e di base

Per rappresentare questi giunti sono stati utilizzati dei vincoli sferici per i giunti di piattaforma e dei giunti cardanici di tipo universale per quelli di base. E' stato usato questo espediente in quanto, utilizzando per entrambi dei giunti sferici, la rotazione dei link attorno al proprio asse sarebbe rimasta libera e ciò avrebbe portato ad una labilità in Adams. Per realizzare un vincolo sferico fra due corpi è sufficiente creare due marker coincidenti ciascuno appartenente ad uno dei due corpi, infatti Adams farà in modo che la distanza fra le origini delle due terne rimanga nulla mentre consentirà la rotazione relativa fra le due lungo qualsiasi direzione. Nella macro i due marker, uno appartenente alla piattaforma ed uno all' $i$ -esimo link sono stati parametrizzati in maniera tale da trovarsi sempre in corrispondenza dei marker  $B_i$ . Per il giunto cardanico di tipo universale il discorso è un po più complicato: in questo caso si dovranno creare, come nel caso precedente, due marker con origine comune, ma questa volta Adams consentirà delle rotazioni solo lungo l'asse  $z$  delle due terne. Volendo bloccare solo la rotazione di ciascun link lungo il proprio asse, l'asse  $z$  di ciascun marker dovrà essere allo stesso tempo perpendicolare all'asse del link ed all'asse  $z$  dell'altro marker. La parametrizzazione di questi marker è stata fatta in maniera tale da avere l'origine delle due terne coincidente con quella dei marker  $A_i$ , ed inoltre, avendo questi ultimi l'asse  $z$  rivolto lungo il link, si è fatto in modo che risultasse perpendicolare a quello degli altri due. Nel caso di trasmissione infinitamente rigida, i giunti universali sono stati creati fra il primo segmento di ciascun link ed il *ground*. Al contrario nel caso di trasmissione flessibile sono stati creati fra i link e gli slider.

### 4.7.2 Prismatici

Fra gli slider ed il *ground* è stato inserito un vincolo prismatico che lasciasse come unico movimento consentito fra i due la traslazione lungo un determinato asse. Come al solito è necessario creare due marker con la stessa orientazione, uno appartenente al *ground* e l'altro all'*i*-esimo slider. Tra i due potrà esserci uno scorrimento relativo lungo l'asse  $x$  che in questo caso giacerà sull'asse delle guide.

### 4.7.3 Accoppiamento vite-madrevite

La modellazione dell'accoppiamento fra gli slider e le aste d'acciaio che rappresentano le viti è piuttosto complicata: la difficoltà sta nel fatto che uno dei due corpi fra cui è applicato il vincolo cambia, in altre parole, a seconda di dove si trova lo slider, il segmento di vite interessato varierà. Per far fronte a tale difficoltà si è pensato di ricorrere all'introduzione di un corpo fittizio con le seguenti caratteristiche e funzionalità: tale corpo modellato come un cilindro pieno, ha il proprio centro coincidente con quello dello slider, ed il suo asse di rivoluzione coincide con l'asse di avanzamento di quest'ultimo; tra questo corpo e lo slider è stato poi applicato un vincolo di tipo *screw* che trasforma uno spostamento dello slider  $\Delta q_i$  in una rotazione  $\Delta \theta_i$  del cilindro attorno al proprio asse di rivoluzione secondo una legge del tipo:

$$\Delta \theta_i = k_\theta \Delta q_i \quad (4.5)$$

dove  $k_\theta$  è un parametro che dipende dal passo della vite, in particolare varrà  $k_\theta = 2\pi/p_h$ , dove  $p_h$  è il passo della vite. Il passo successivo è quello di fare in modo che ad una rotazione del corpo fittizio corrisponda una rotazione identica del segmento di vite in corrispondenze del quale si trova lo slider. Per raggiungere tale obiettivo si è pensato di usare un *general constraint*. In adams i vincoli geometrici vengono trattati come un set di equazioni del tipo:

$$\Phi_k(\mathbf{q}) = 0 \quad \text{con} \quad k = 1, \dots, n_v \quad (4.6)$$

dove  $n_v$  rappresenta il numero di vincoli imposti. Ad esempio un giunto sferico può essere visto come un set di tre equazioni del tipo:

$$\begin{cases} x_{marker_1} - x_{marker_2} = 0 \\ y_{marker_1} - y_{marker_2} = 0 \\ z_{marker_1} - z_{marker_2} = 0 \end{cases} \quad (4.7)$$

Allo stesso modo un *general constraint* permette di imporre un qualsiasi vincolo geometrico scrivendo una funzione che Adams manterrà nulla durante le analisi. Nel caso specifico è stato creato un vincolo di questo tipo per ciascuna vite e come funzione da annullare si è imposto:

```
function=az(M_gcn_corpo_fittizio, M_gcn_j)
```

dove il comando *az* restituisce il valore dello sfasamento angolare lungo l'asse  $z$  del marker  $M_{gcn,corpo-fittizio}$ , appartenente al corpo fittizio, rispetto al marker  $M_{gcn,j}$



appartenente al  $j$ -esimo segmento di vite. Questo ultimo marker è stato parametrizzato in modo tale da trovarsi in corrispondenza del centro del corrispondente cilindro nel caso in cui lo slider si trovi in corrispondenza di un altro segmento, altrimenti sarà coincidente con il marker  $M_{gcn,corpo-fittizio}$ . Tale risultato è stato raggiunto con l'utilizzo di due funzioni *step*, in questo modo è stata creata una funzione ausiliaria del tipo:

$$h(\xi) = \begin{cases} 1 & \text{se } -\frac{L_{seg}}{2} \leq \xi \leq \frac{L_{seg}}{2} \\ 0 & \text{altrove} \end{cases} \quad (4.8)$$

dove con  $\xi$  si è indicata la distanza relativa fra il marker  $M_{gcn,corpo-fittizio}$  ed il centro del cilindro che costituisce il  $j$ -esimo segmento. Finora è stato descritto come creare l'accoppiamento fra lo slider ed un generico segmento della vite, ma non è stato definito in che modo viene scelto il  $j$ -esimo elemento da inserire nella funzione del *general constraint*. La soluzione trovata prevede di valutare in corrispondenza di quale segmento si trova lo slider e di aggiornare di conseguenza il *general constraint* prima di effettuare l'analisi. Si è fatto ricorso quindi a delle variabili di appoggio di tipo reale definite nel seguente modo:

$$n_i = floor\left(\frac{q_i - q_{i,0} + L_{vite}/2}{L_{vite}}\right)n_{segmenti} + 1 \quad (4.9)$$

dove  $q_{i,0}$  rappresenta il valore assunto dall' $i$ -esima coordinata di giunto nella *home-position*. In definitiva la funzione del *general constraint* sarà:

`function=az ( M_gcn_corpo_fittizio }, M_gcn_n_i )}`

A questo punto una traslazione dello slider comporterà una rotazione del corpo fittizio, che a sua volta causerà una rotazione dell' $n_i$ -esimo segmento di vite. Si può affermare quindi che alle variazioni di posizione dello slider si oppone una forza elastica direttamente proporzionale alla rigidità torsionale della vite.

## 4.8 Analisi e generazione dei file di testo

Una volta completato il modello, ogni elemento della macchina sarà parametrizzato in maniera diretta o indiretta rispetto alle coordinate del TCP, di conseguenza sarà sufficiente modificare la *location* e l'orientazione di questo marker per riorientare l'intero robot. Per valutare come cambino le frequenze all'interno dello spazio di lavoro, sono stati scelti dei piani sfasati rispetto all'asse  $z$  globale e sono stati suddivisi in un numero discreto di punti dove effettuare l'analisi di vibrazione. A questo punto, fissata una determinata orientazione  $\Theta = \{\alpha, \beta, \gamma\}^T$ , la posizione del TCP è stata cambiata in modo da andare ad occupare ogni punto della griglia discreta di punti in cui è stato suddiviso il piano scelto. Per ciascun punto poi è stata effettuata un'analisi di tipo *static equilibrium* e successivamente sono stati calcolati gli autovalori del sistema in questa configurazione. Il tipo di analisi scelta rispecchia la volontà di analizzare la macchina in condizioni di velocità ed

accelerazione nulle, altrimenti si sarebbe dovuto ricorrere ad un'analisi di tipo dinamico. I valori della prima frequenza propria sono stati salvati in una serie di file di testo. E' stato generato un file per ogni orientazione  $\Theta$ , ciascuno di essi contenente i file relativi alla griglia di punti del piano in esame. Si è scelto di utilizzare una risoluzione di  $5^\circ$  per cui  $\alpha$  e  $\gamma$  potranno assumere i valori  $[-5^\circ, 0^\circ, 5^\circ]$ , mentre  $\beta$  potrà assumere uno dei sette valori compresi fra  $-15^\circ$  e  $15^\circ$ . Combinando tutti i possibili casi si avranno 63 orientazioni da esplorare, e quindi altrettanti saranno i file di testo generati.

## 4.9 Risultati

I piani su cui sono state condotte le analisi sono stati suddivisi in griglie di 11 elementi per 11. Inizialmete era stata scelta una griglia più fitta, ma si è visto che il gradiente delle frequenze proprie delle due macchine non era così elevato e che anzi una mesh più rada sarebbe stata sufficiente per cogliere a pieno tutti gli aspetti del problema in esame. Anche per quanto riguarda il numero di segmenti che costituiscono i link e le viti a ricircolo di sfere, è stato scelto un valore iniziale e si è visto che pur suddividendo i link in un numero maggiore di elementi il risultato rimaneva pressochè identico, con variazioni del decimo di  $Hz$ . Per le analisi definitive sono stati utilizzati 8 elementi per ciascun link e 7 per ciascuna vite. In Adams aumentare, seppur di poco, il numero di corpi che costituiscono il sistema in esame porta ad un aumento significativo dei tempi di calcolo, e questi valori, così come la risoluzione spaziale scelta, rappresentano un ottimo compromesso fra tempo totale di analisi e qualità dei risultati.

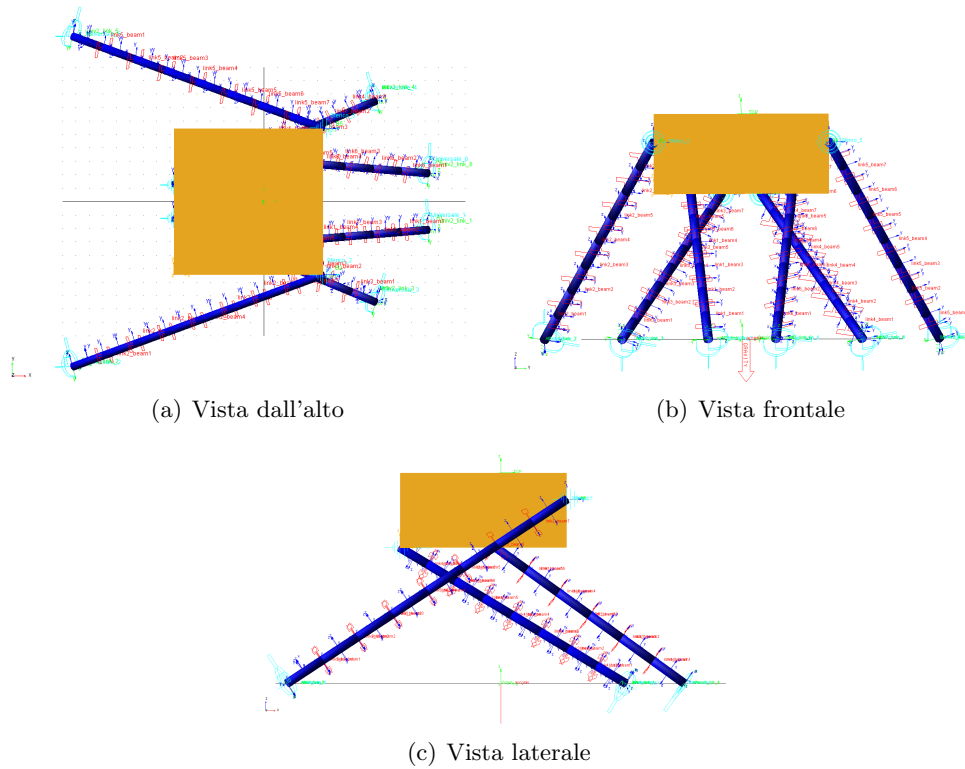
### 4.9.1 Hexaglide

Il disegno della macchina in ADAMS si presenta come quello di Fig. 4.5.

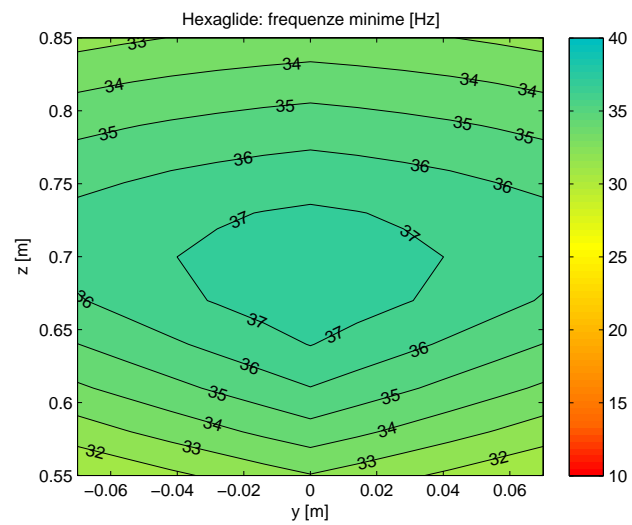
Per l'Hexaglide, così come già fatto nell'ottimizzazione cinematica, è stato analizzato un unico piano, e cioè quello  $yz$ . Dato l'elevato numero di analisi la memoria allocata ad ADAMS non è sufficiente, e richiedere al programma di generare tutti i 63 file lo manderebbe in crash. Per questo motivo sono stati generati tre gruppi di 21 file ciascuno, ottenuti fissando l'angolo di rollio  $\alpha$ . All'interno dei file generati i dati sono stati organizzati in modo da avere una struttura che rispecchiasse la griglia con cui è stato suddiviso il piano analizzato. Grazie a questa struttura risulta semplice importare i dati in MATLAB per generare delle mappe che mostrino l'andamento della prima frequenza propria in tutto il piano per un'orientazione fissata. Infine per ciascun punto della griglia sono stati considerati i valori assunti dalla frequenza propria per ciascuna orientazione e se ne è preso quello minimo, ottenendo una mappa delle frequenze minime a prescindere dall'orientazione.

In Fig. 4.6 si riporta la mappa delle frequenze minime nel caso con trasmissione infinitamente rigida.

In Fig. 4.7 ed in Fig. 4.8 vengono mostrate invece le mappe delle frequenze minime relative rispettivamente al caso con trasmissione a cinghia e a vite.

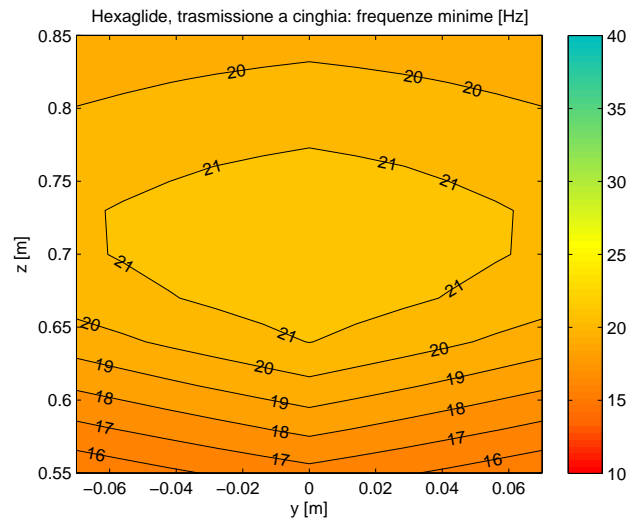


**Fig. 4.5:** Vista dall'alto, frontale e laterale in ADAMS dell'Hexaglide.

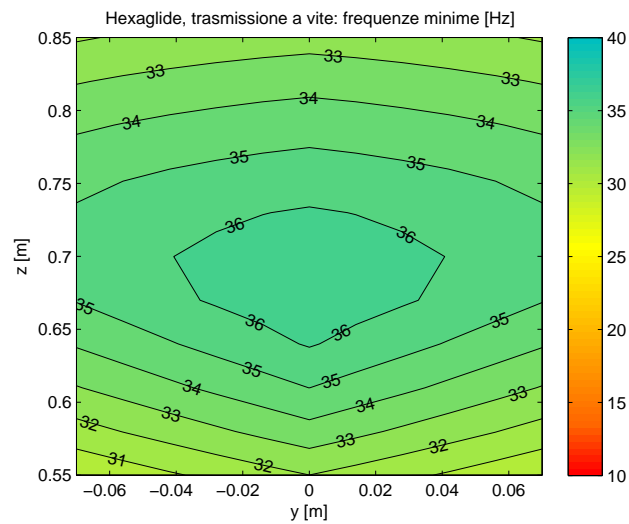


**Fig. 4.6:** Mappa delle frequenze minime dell'Hexaglide sul piano  $yz$  nel caso con trasmissione infinitamente rigida.

Si vede come nel passare da una trasmissione infinitamente rigida ad una trasmissione flessibile la situazione peggiori in quanto le frequenze proprie del sistema



**Fig. 4.7:** Mappa delle frequenze minime dell'Hexaglide sul piano  $yz$  nel caso con trasmissione a cinghia.

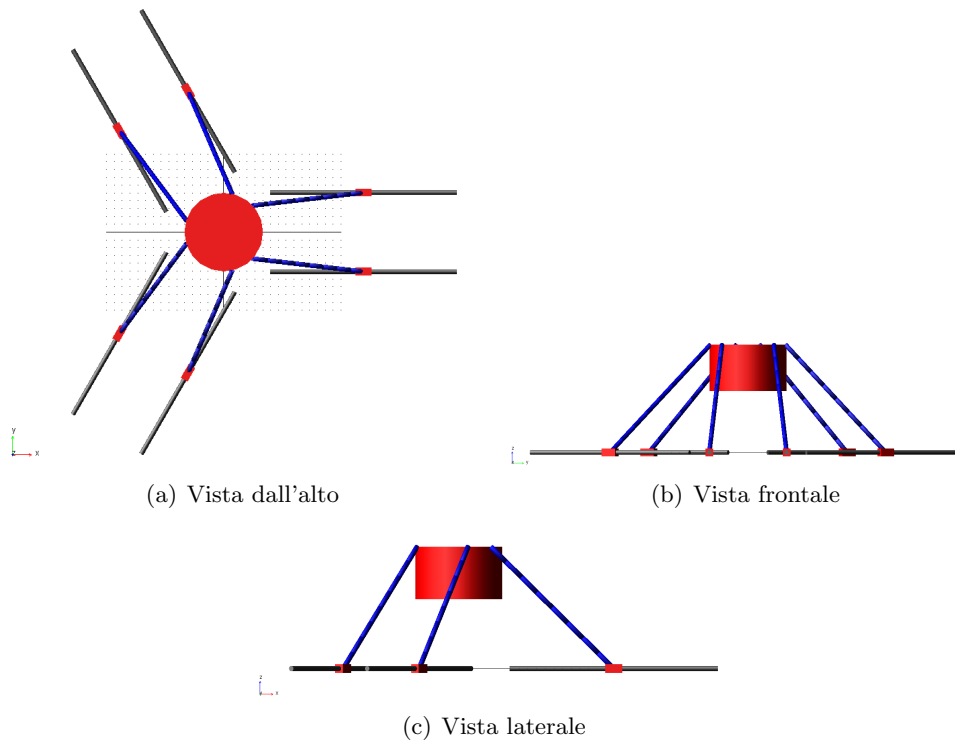


**Fig. 4.8:** Mappa delle frequenze minime dell'Hexaglide sul piano  $yz$  nel caso con trasmissione a vite.

si abbassano. Tale peggioramento è decisamente più evidente nel caso della trasmissione a cinghia in quanto si ha una riduzione di circa  $15\text{ Hz}$  in tutto il piano, mentre nel caso delle viti tale riduzione è quasi impercettibile.

#### 4.9.2 Hexaslide

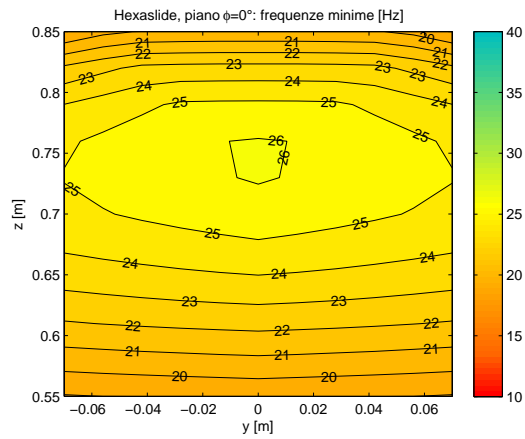
La macchina comprensiva di viti a ricircolo di sfere si presenta come in Fig. 4.9.



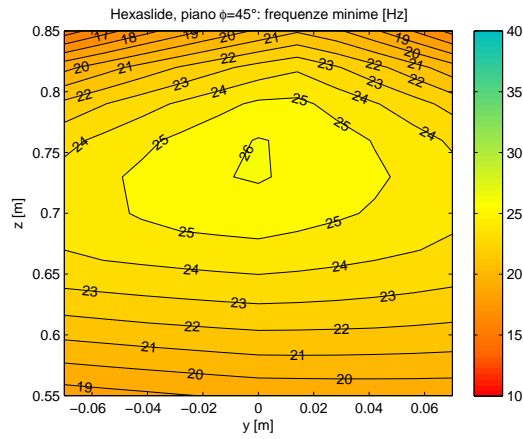
**Fig. 4.9:** Vista dall'alto, frontale e laterale in ADAMS dell'Hexaslide.

A differenza dell'Hexaglide, nel caso dell'Hexaslide non è sufficiente analizzare un unico piano, per cui lo spazio di lavoro è stato sezionato con dei piani così come già fatto nel Cap. 3 e per ciascuno di essi si è analizzato il comportamento della macchina. Le analisi sono state condotte per valori di  $\phi$  pari a  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ , ma in questo caso l'asse a  $0^\circ$  è stato preso diretto come l'asse  $y$  ma nel verso opposto. La mappa delle frequenze su questi tre piani nel caso con trasmissione infinitamente rigida è riportata in Fig. 4.10.

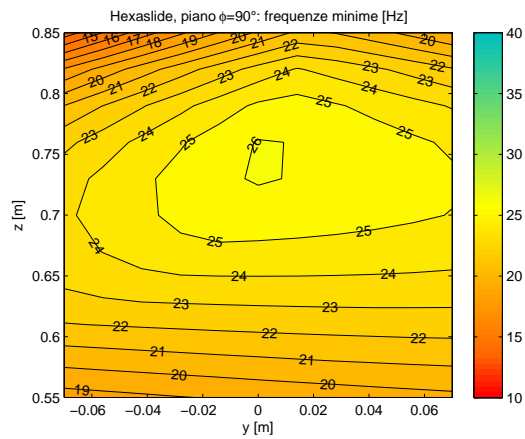
Anche in questo caso vengono riportati i casi con trasmissione a cinghia (Fig. 4.11), e trasmissione a vite (Fig. 4.12). Così come accade per l'Hexaglide le frequenze proprie nel caso con trasmissione a cinghia sono le peggiori, tuttavia in questo caso la riduzione dei valori di frequenza è inferiore, circa  $8 \div 10 Hz$ .



(a) piano  $\phi = 0^\circ$

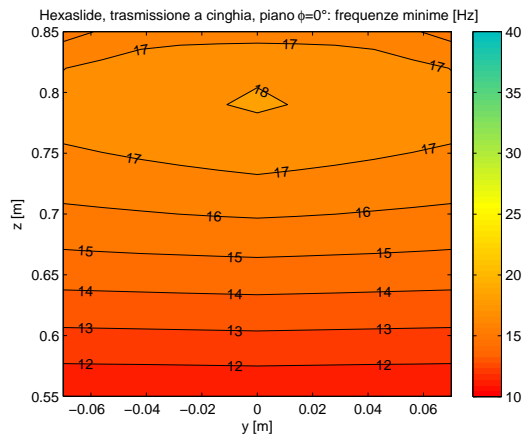
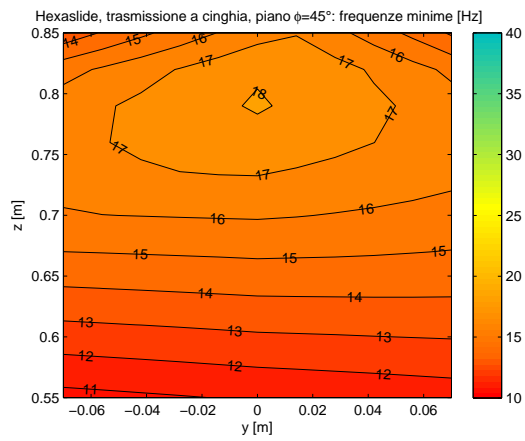
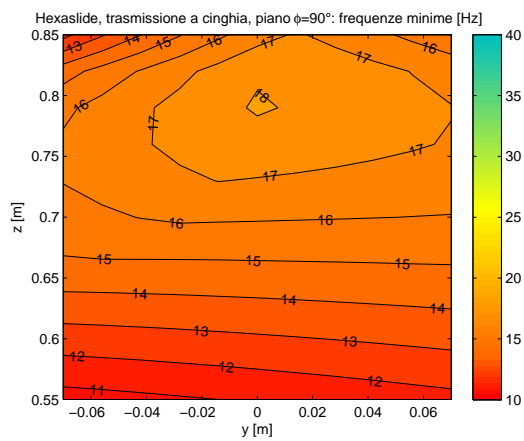


(b) piano  $\phi = 45^\circ$

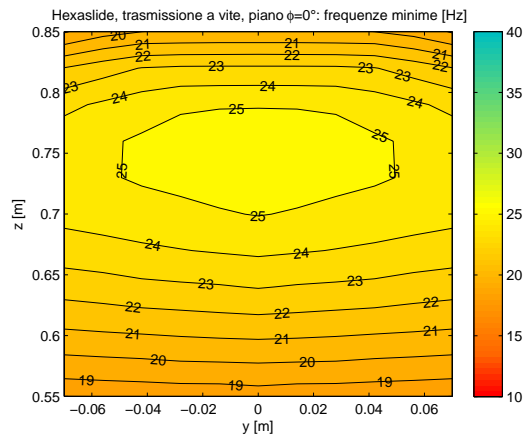


(c) piano  $\phi = 90^\circ$

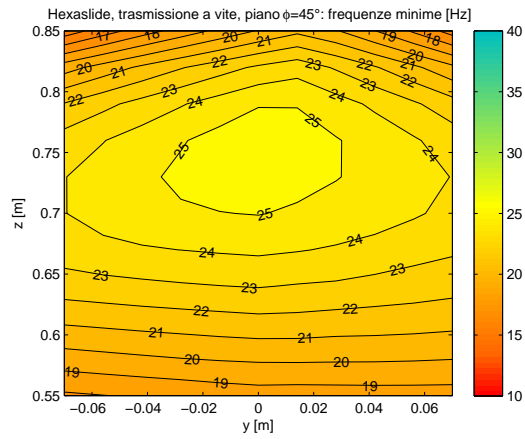
**Fig. 4.10:** Mappa delle frequenze minime dell'Hexaslide su vari piani nel caso con trasmissione a vite.

(a) piano  $\phi = 0^\circ$ (b) piano  $\phi = 45^\circ$ (c) piano  $\phi = 90^\circ$ 

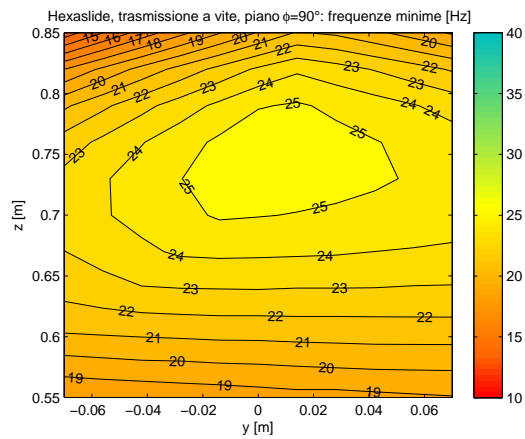
**Fig. 4.11:** Mappa delle frequenze minime dell'Hexaslide con trasmissione a cinghia su vari piani.



(a) piano  $\phi = 0^\circ$



(b) piano  $\phi = 45^\circ$



(c) piano  $\phi = 90^\circ$

**Fig. 4.12:** Mappa delle frequenze minime dell'Hexaslide con trasmissione a vite su vari piani.



## CAPITOLO 5

### CLEARANCE

I risultati dell'ottimizzazione cinetostatica tengono conto della distanza relativa fra i link e di quella fra link e piattaforma. Quest'ultima è stata schematizzata come un cilindro di altezza pari a 50 cm e con un raggio di base tale da contenere i giunti di piattaforma. Tuttavia, quando sarà presente un disegno costruttivo definitivo di questo elemento, sarà necessario verificare che la geometria reale non vada ad interferire con nessuno degli altri elementi. A tale scopo è stata creata una macro in Adams volta a calcolare in ciascuna posa del robot la distanza minima fra link e piattaforma reale ricorrendo alle *clearance function*. Il caso analizzato è quello dell'Hexaglide, dove in un primo momento, quando le specifiche richiedevano uno spazio di lavoro che si sviluppasse prevalentemente lungo la direzione del vento, era stato realizzato un primo disegno costruttivo. Essendo poi cambiate le specifiche ed il relativo spazio di lavoro, tale disegno è stato abbandonato con l'intenzione di realizzarne uno nuovo che si addicesse maggiormente alle nuove richieste. Anche se la piattaforma reale non avrà le sembianze del disegno iniziale, tuttavia questo verrà utilizzato per illustrare le funzionalità della macro, che comunque resta un tool valido e che verrà riutilizzato una volta che si avrà a disposizione il disegno finale.

#### 5.1 Clearance function

Questo tipo di funzioni permette di calcolare la minima distanza fra due corpi. In Adams è possibile definire un corpo come semplice *parte* senza definirne la forma, indicando unicamente le sue proprietà inerziali, e cioè la posizione del suo centro di massa, degli assi principali di inerzia, nonché i valori della massa e dei momenti d'inerzia. Tuttavia per poter utilizzare le *clearance function* è necessario che entrambi i corpi possiedano una geometria. Inoltre è possibile assegnare ad una parte più di una geometria, ad esempio se il corpo in esame ha una struttura modulare. In tal caso la *clearance* calcolerà il valore di distanza minima fra ciascuna geometria appartenente alla prima parte e le geometrie appartenenti alla seconda e ne restituirà il valore minimo. La sintassi utilizzata per dichiarare una *clearance* è la seguente:

**Listing 5.1:** Esempio di clearance function

```
clearance create clearance_name=distanza &  
run_time=yes &  
i_geometry=.modello.corpo_1.geometria &  
j_geometry=.modello.corpo_2.geometria &  
threshold=0.0
```

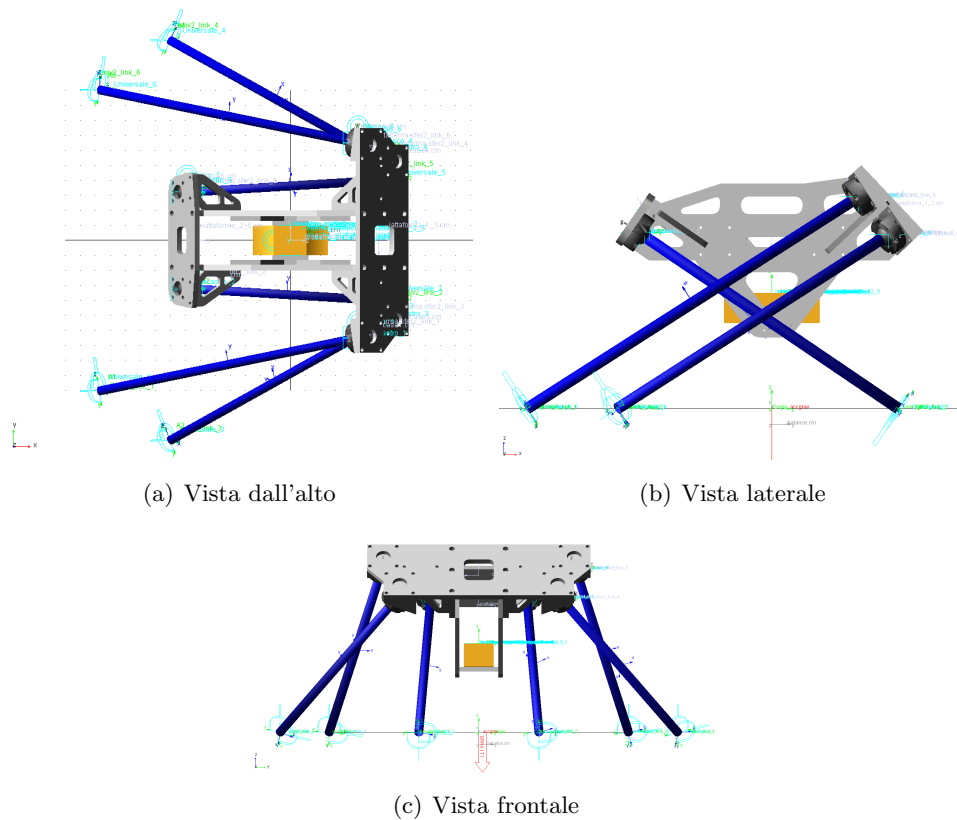
Nell'esempio è stato impostato un valore di *threshold* pari a 0, quindi non appena la distanza fra i due corpi si annulla, il valore di distanza minima viene mantenuto fisso anche se avviene una compenetrazione fra i due corpi.

## 5.2 Descrizione della piattaforma

Per avere una piattaforma che risultasse allo stesso tempo rigida ed abbastanza leggera per non sovraccaricare i link ed i giunti attuati, nonché quelli passivi, si è pensato di realizzare una struttura modulare, costituita da un insieme di piastre di alluminio tenute assieme da giunzioni bullonate, così come si può notare dalla Fig. 5.1. Il disegno è stato realizzato in Solidworks, e l'insieme delle parti è stato salvato in un file di tipo *parasolid*, estensione che garantisce l'interazione fra vari programmi. Al fine di ottenere un risultato che fosse il più preciso possibile, oltre alla pura e semplice piattaforma sono stati inseriti: la bilancia, la cui forma è assimilabile a quella di un parallelepipedo; l'involucro esterno dei giunti di piattaforma, costituiti da cilindri molto schiacciati con i fori necessari per posizionare i bulloni e un incavo centrale all'interno del quale è alloggiata la testa del giunto sferico. I singoli componenti della piattaforma sono stati dapprima creati come semplici parti senza alcuna forma, e successivamente sono state loro assegnate le geometrie corrispondenti presenti nel file *parasolid*. La piattaforma intesa come puro e semplice insieme di piastre è stata suddivisa in tre corpi distinti: corpo centrale; piastra porta-giunti relativa ai giunti 2 e 5; piastra porta-giunti relativa ai giunti 1, 3, 4, e 6. Questa suddivisione è stata necessaria in quanto, se fosse stata modellata come un corpo unico, nella maggior parte dei casi la distanza minima sarebbe risultata quella fra i link e le piastre porta-giunto, mentre l'obiettivo è quello di avere una misura di quanto ciascun link sia lontano da una condizione di collisione col corpo centrale.

## 5.3 Descrizione link

A differenza dell'analisi di vibrazioni dove i link sono stati modellati come dei corpi flessibili, in questo caso sono stati trattati come un unico corpo. Le ragioni che hanno portato a tale scelta sono due: è ragionevole pensare che le vibrazioni della struttura causino delle inflessioni di piccola entità e che quindi non siano rilevanti ai fini del calcolo della distanza fra gli elementi che compongono la struttura; nell'analisi di vibrazioni ogni segmento dei link è stato creato come singola parte e non come un unico corpo con assegnate più geometrie, e questo rende impossibile l'utilizzo delle *clearance*, a meno che non si crei una *clearance* distinta fra ogni segmento di un link ed il corpo centrale della piattaforma. I link sono stati modellati



**Fig. 5.1:** Disegno costruttivo della piattaforma.

dunque come cilindri cavi di diametro esterno pari a  $3\text{ cm}$  e con uno spessore di  $2\text{ mm}$ .

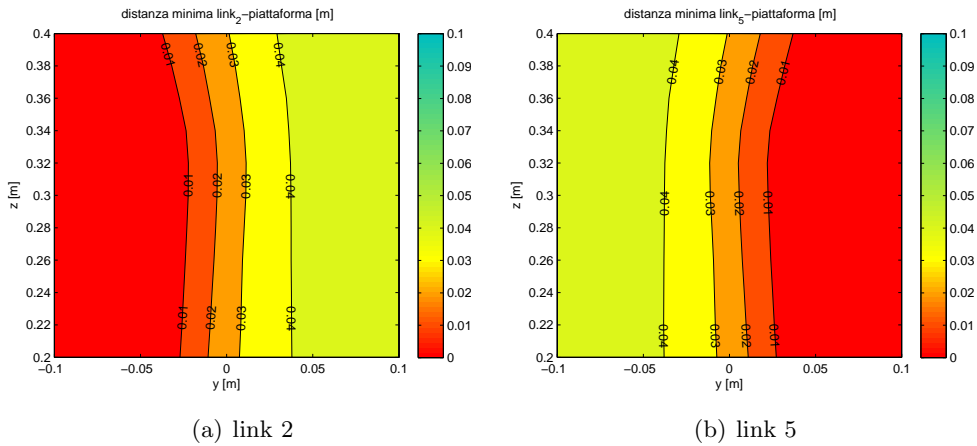
## 5.4 File generati

Nella macro sono state create dodici clearance, sei per il calcolo della distanza fra ciascun link ed il corpo centrale della piattaforma e sei per il calcolo della distanza fra i link. La simmetria della macchina e la sua particolare conformazione consentono di evitare il calcolo della distanza fra un link ed i tre che si trovano dalla parte opposta rispetto al piano  $xz$  in quanto nelle configurazioni analizzate non verranno mai in contatto fra di loro. In realtà l'ottimizzazione cinetostatica assicura che la distanza fra i link sia maggiore di un valore minimo imposto. Il motivo della creazione di queste altre sei clearance risiede nel fatto che in seguito alle prime analisi, si è notato che i link 2 e 5 andavano ad interferire con il corpo centrale della piattaforma e quindi si è pensato di allargarla lungo la direzione  $y$  senza ritornare alla fase di ottimizzazione. Analogamente a quanto fatto per l'analisi di vibrazioni, fissata un'orientazione  $\Theta$  il piano  $yz$  è stato suddiviso in un numero discreto di punti e per ciascuno di esso sono state calcolate le *clearance*. I

risultati vengono salvati in file di testo di tipo iso-orientazione. Alla fine di tutte le analisi, i file di testo saranno 27 per ciascuna clearance, e quindi in definitiva saranno 324.

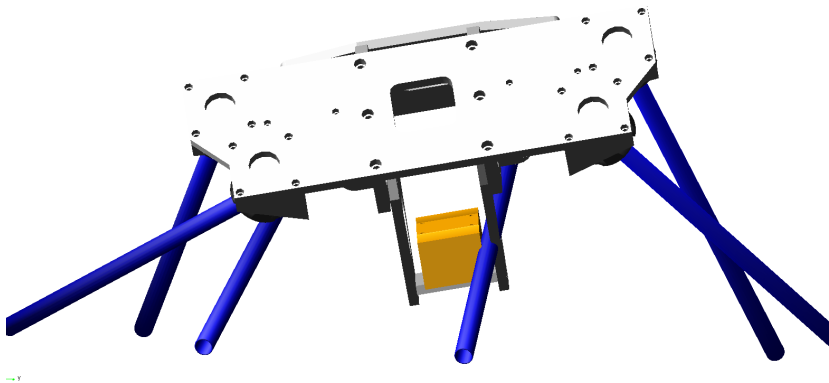
## 5.5 Risultati

Le analisi effettuate sono relative al caso in cui le specifiche tecniche richiedevano uno spazio di lavoro più grande e dove le richieste sulle rotazioni erano di  $\pm 10^\circ$  su tutti e tre gli assi. La risoluzione adottata è di  $10^\circ$  per ciascun asse. La Fig. 5.2 mostra le mappe di distanza minima fra i link 2 e 5.



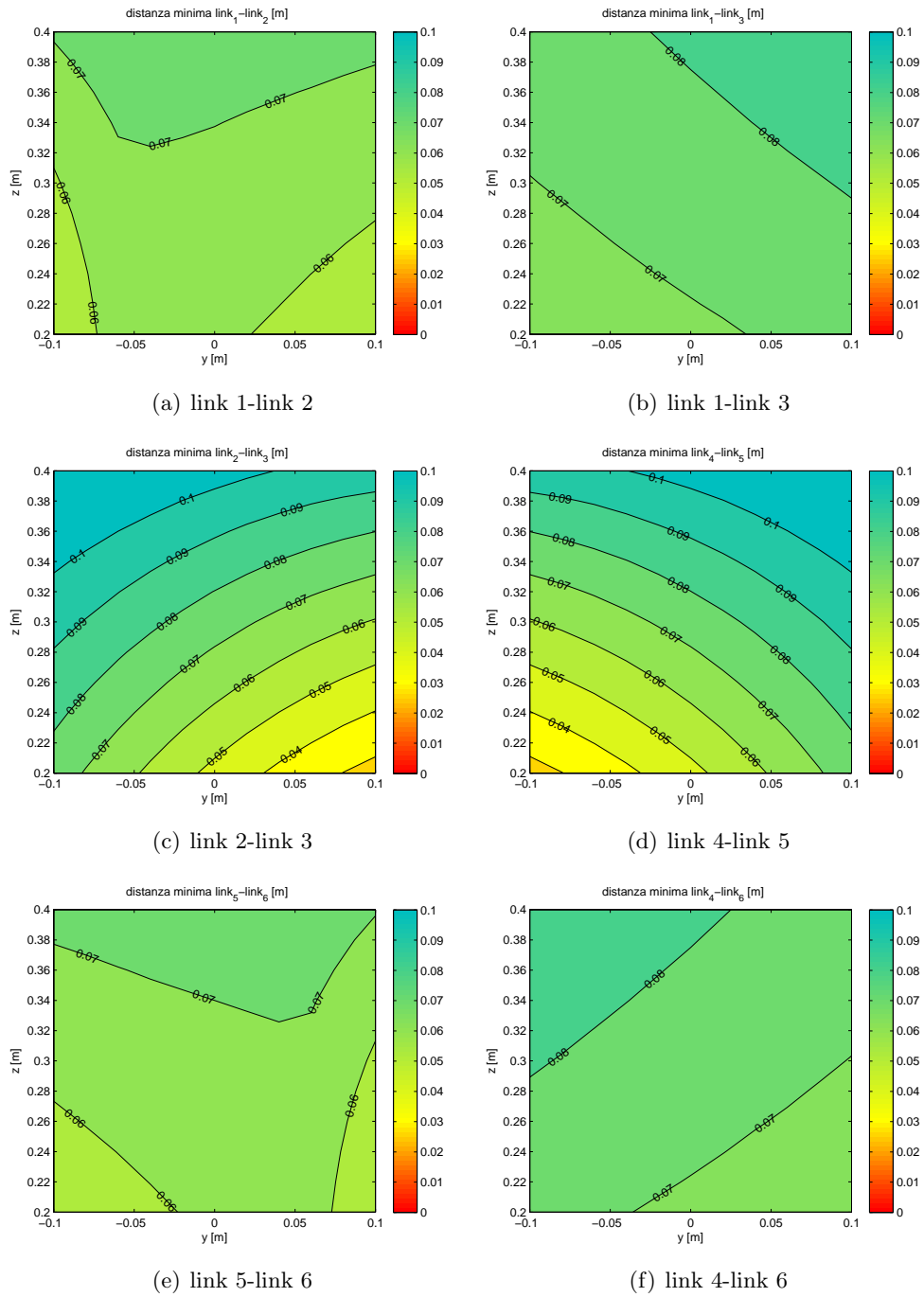
**Fig. 5.2:** Mappe di distanza minima fra i link 2 e 5 e la piattaforma.

Si vede subito che in alcuni casi la distanza minima raggiunge il valore nullo, situazione che viene mostrata in Fig. 5.3. Si è deciso quindi di allargare la piattaforma di 10 *cm* in direzione trasversale per evitare che i link in questione (2 e 5) andassero a collidere con essa.

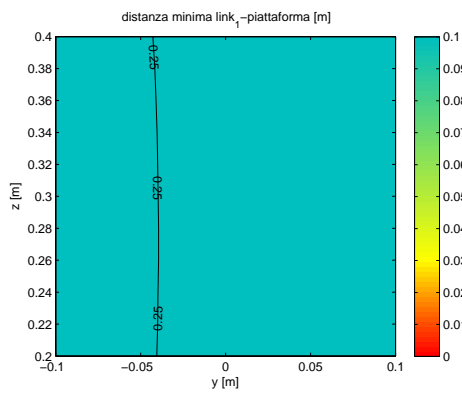


**Fig. 5.3:** Comp penetrazione link-piattaforma per una particolare posa.

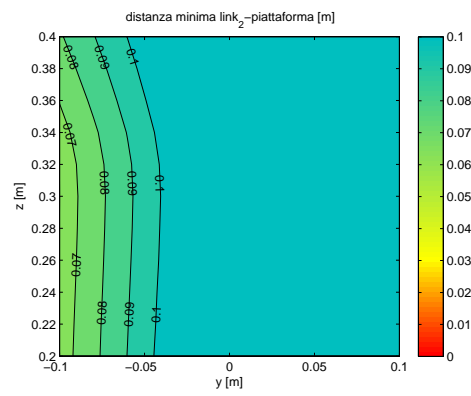
La nuove mappe relative alla piattaforma allargata sono rappresentate in Fig. 5.4.



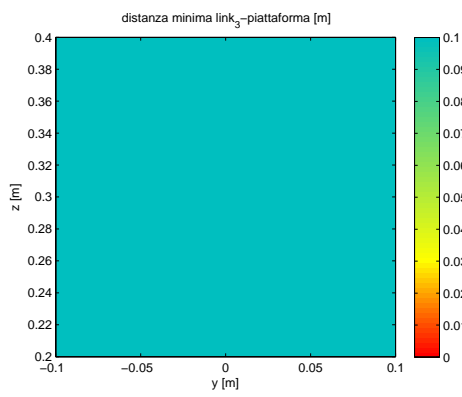
**Fig. 5.4:** Mappe della distanza fra i link con piattaforma allargata.



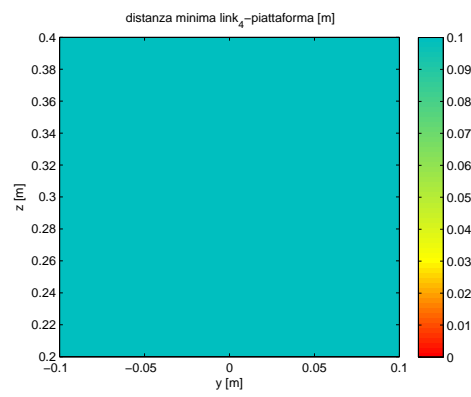
(a) link 1-piattaforma



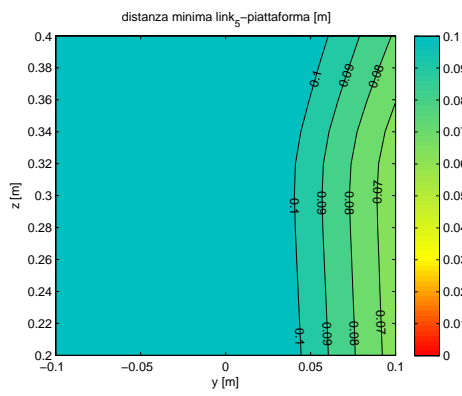
(b) link 2-piattaforma



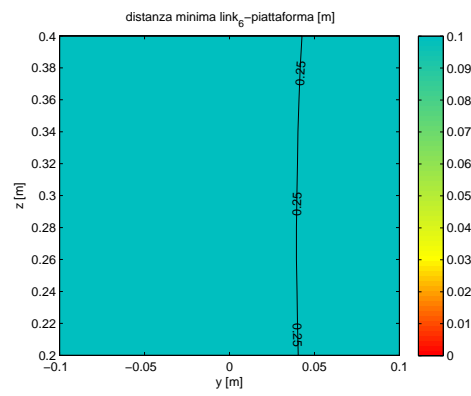
(c) link 3-piattaforma



(d) link 4-piattaforma



(e) link 5-piattaforma



(f) link 6-piattaforma

**Fig. 5.5:** Mappe della distanza fra i link e la piattaforma con piattaforma allargata.

## CAPITOLO 6

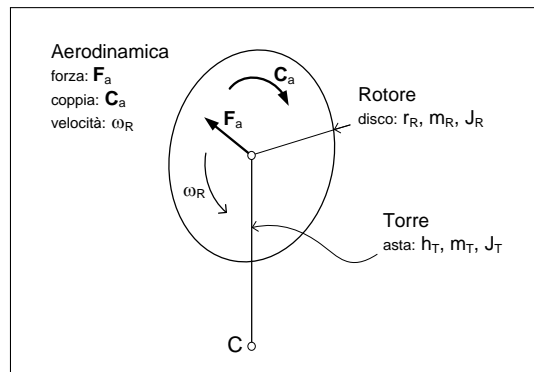
# ANALISI DINAMICA E SCELTA DEL GRUPPO MOTORIDUTTORE

La macchina scelta a valle del processo di ottimizzazione è l'Hexaslide. Questa macchina infatti permette di avere una copertura totale dello spazio di lavoro desiderato, ed il suo ingombro totale risultata inferiore rispetto alle altre due macchine. Infatti nonostante l'Hexaglide abbia un andamento delle frequenze proprie migliore su tutto lo spazio di lavoro, per evitare collisioni fra i link e la piattaforma, quest'ultima deve essere allargata in maniera spropositata, andando ad occupare uno spazio maggiore rispetto a quello disponibile. L'Hexaslide invece, per come sono disposti i suoi link, lascia grande libertà di movimento al corpo centrale della piattaforma senza alcun rischio di collisioni. Si è passati quindi al dimensionamento del gruppo motore-riduttore. È necessario per cui valutare quale sia l'andamento in funzione del tempo della coppia richiesta al motore durante un possibile ciclo di lavoro. Per questo motivo è stato creato un modello Simmechanics della macchina.

### 6.1 Modello Simmechanics

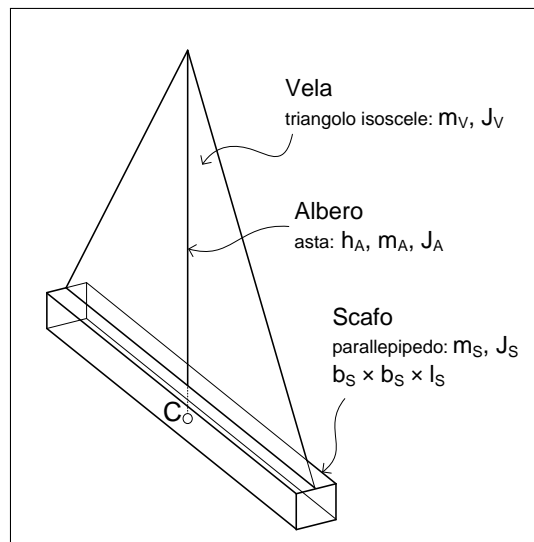
Per la costruzione del modello si è ricorsi all'uso del pacchetto Simmechanics di Matlab, in particolare è stato utilizzato il gruppo First-Generation. A differenza del modello Adams i corpi sono stati considerati come rigidi e sono stati aggiunti i contributi dei modelli di turbina e barca a vela che il robot dovrà provvedere a movimentare all'interno dello spazio di lavoro. I valori della massa e dei momenti di inerzia dei link e della piattaforma sono analoghi a quelli utilizzati all'interno del modello Adams, così come la posizione del centro di massa e l'orientazione degli assi principali d'inerzia. Per quanto riguarda gli elementi aggiuntivi invece, essi sono stati modellati nel seguente modo:

- *generatore eolico*: come si può vedere dalla Fig. 6.1, la turbina eolica è stata schematizzata come un'asta di lunghezza  $l_P$  che rappresenta il pilone che sorregge il gruppo rotore-statore, ed un disco di raggio  $r_R$  che rappresenta il gruppo rotorico con annesse le pale. Oltre al puro e semplice contributo inerziale, è stata aggiunta la possibilità di mettere in rotazione il disco ad una velocità costante  $\omega_{rotore}$ .



**Fig. 6.1:** Schema turbina.

- *barca a vela*: in questo caso il modello raffigurato in Fig. 6.2, è costituito da un parallelepipedo di dimensioni  $b_B \times b_B \times l_B$  volto a rappresentare lo scafo della barca, da un'asta lunga  $l_A$  atta a simulare l'albero, e da una vela schematizzata come un triangolo isoscele.



**Fig. 6.2:** Schema barca.

Per risalire alla coppia resistente agente sui motori è necessario calcolare le forze che si scaricano in corrispondenza dei giunti di base, sono quindi stati imposti i moti dei giunti attuati attraverso i blocchetti *joint actuator* e sono state calcolate le componenti di tali forze attraverso i *joint sensor*. I movimenti imposti al robot sono stati scelti in maniera da esplorare l'intero spazio di lavoro. Scelta una posizione dalla griglia di pose in cui il volume di lavoro è stato suddiviso, sono stati imposti dei movimenti sinusoidali lungo una singola coordinata cartesiana od angolare.



Sinteticamente si può scrivere:

$$\begin{Bmatrix} \mathbf{p}(t) \\ \Theta(t) \end{Bmatrix} = \begin{Bmatrix} \mathbf{p}_0 \\ \Theta_0 \end{Bmatrix} + \begin{Bmatrix} \Delta \mathbf{p} \\ \Delta \Theta \end{Bmatrix} \sin(\bar{\omega}t) \quad (6.1)$$

Dove  $\mathbf{p}_0$ ,  $\Theta_0$ ,  $\Delta \mathbf{p}$ , e  $\Delta \Theta$  assumeranno i valori riportati in Tab. 6.1 e Tab. 6.2 in base all'asse lungo il quale si è scelto di movimentare il robot.

	movimento lungo x	movimento lungo y	movimento lungo z
$x_0$	0	$x_{lim,-} \div x_{lim,+}$	$x_{lim,-} \div x_{lim,+}$
$y_0$	$y_{lim,-} \div y_{lim,+}$	0	$y_{lim,-} \div y_{lim,+}$
$z_0$	$z_{lim,-} \div z_{lim,+}$	$z_{lim,-} \div z_{lim,+}$	0
$\alpha_0$	$\alpha_{lim,-} \div \alpha_{lim,+}$	$\alpha_{lim,-} \div \alpha_{lim,+}$	$\alpha_{lim,-} \div \alpha_{lim,+}$
$\beta_0$	$\beta_{lim,-} \div \beta_{lim,+}$	$\beta_{lim,-} \div \beta_{lim,+}$	$\beta_{lim,-} \div \beta_{lim,+}$
$\gamma_0$	$\gamma_{lim,-} \div \gamma_{lim,+}$	$\gamma_{lim,-} \div \gamma_{lim,+}$	$\gamma_{lim,-} \div \gamma_{lim,+}$
	rotazione $\alpha$	rotazione $\beta$	rotazione $\gamma$
$x_0$	$x_{lim,-} \div x_{lim,+}$	$x_{lim,-} \div x_{lim,+}$	$x_{lim,-} \div x_{lim,+}$
$y_0$	$y_{lim,-} \div y_{lim,+}$	$y_{lim,-} \div y_{lim,+}$	$y_{lim,-} \div y_{lim,+}$
$z_0$	$z_{lim,-} \div z_{lim,+}$	$z_{lim,-} \div z_{lim,+}$	$z_{lim,-} \div z_{lim,+}$
$\alpha_0$	0	$\alpha_{lim,-} \div \alpha_{lim,+}$	$\alpha_{lim,-} \div \alpha_{lim,+}$
$\beta_0$	$\beta_{lim,-} \div \beta_{lim,+}$	0	$\beta_{lim,-} \div \beta_{lim,+}$
$\gamma_0$	$\gamma_{lim,-} \div \gamma_{lim,+}$	$\gamma_{lim,-} \div \gamma_{lim,+}$	0

**Tab. 6.1:** Posizioni di partenza attorno alle quali si sviluppano i moti sinusoidali.

	movimento lungo x	movimento lungo y	movimento lungo z
$\Delta x$	$\frac{LW_{sd,x}}{2}$	0	0
$\Delta y$	0	$\frac{LW_{sd,y}}{2}$	0
$\Delta z$	0	0	$\frac{LW_{sd,z}}{2}$
$\Delta \alpha$	0	0	0
$\Delta \beta$	0	0	0
$\Delta \gamma$	0	0	0
	rotazione $\alpha$	rotazione $\beta$	rotazione $\gamma$
$\Delta x$	0	0	0
$\Delta y$	0	0	0
$\Delta z$	0	0	0
$\Delta \alpha$	$\alpha_{lim}$	0	0
$\Delta \beta$	0	$\beta_{lim}$	0
$\Delta \gamma$	0	0	$\gamma_{lim}$

**Tab. 6.2:** Ampiezza delle sinusoidi.

Per risalire poi ad i movimenti da imporre alle coordinate di giunto è sufficiente risolvere la cinematica inversa.

## 6.2 Scelta del gruppo motore-riduttore

Il campo di funzionamento dei motori elettrici è definito da una curva caratteristica in condizioni di funzionamento nominale e da una curva limite che definisce la coppia massima erogabile in funzione della velocità di rotazione. I limiti sulla coppia nominale  $C_{m,nom}$ , e cioè sulla coppia che può essere erogata in maniera continuativa, sono dettati dalla capacità del motore di smaltire il calore generato per dissipazione. Sull'asse delle velocità sono degni di nota i valori di velocità massima  $\omega_{max}$  che stabilisce quale sia il limite tollerabile dai componenti meccanici, e quella nominale  $\omega_{nom}$  che definisce il punto di passaggio da una zona a coppia costante ad una a potenza costante. Le verifiche da effettuare per ciascun gruppo motore-riduttore sono:

- rispetto dei limiti di coppia massima:

$$\max|C_m(t)| < C_{m,max} \quad (6.2)$$

- rispetto dei limiti sulla coppia nominale:

$$C_{m,rms}^* < C_{m,nom} \quad (6.3)$$

- rispetto dei limiti di velocità massima:

$$\max|\omega_r(t)| < \tau\omega_{m,max} \quad (6.4)$$

dove con  $\omega_m$  e  $\omega_r$  sono state indicate rispettivamente le velocità di rotazione lato motore e lato utilizzatore, e con  $\tau$  il rapporto di trasmissione che le lega. È utile riesprimere la coppia motrice necessaria in funzione della coppia resistente e di quella d'inerzia del motore, il cui andamento temporale può essere ricavato facilmente dai risultati della simulazione Simmechanics. Attraverso un bilancio di potenze dal lato dell'utilizzatore si arriva alla seguente espressione della coppia motrice:

$$C_m = \tau C_r^* + J_m \frac{\dot{\omega}_r}{\tau} \quad (6.5)$$

dove  $C_r^*$  rappresenta un termine di coppia resistente generalizzata, in cui sono compresi i contributi relativi agli attriti presenti fra i vari organi della macchina e quelli inerziali. Seguendo la metodologia illustrata da H. Giberti ([17], [18], [19]), per la verifica termica si arriva ad una scrittura del tipo:

$$\alpha > \beta + \left[ C_{r,rms}^* \left( \frac{\tau_{rid}}{\sqrt{J_m}} \right) - \omega_{r,rms} \left( \frac{\sqrt{J_m}}{\tau_{rid}} \right) \right]^2 \quad (6.6)$$

dove  $\alpha$ , definito come *fattore accelerante* ha la seguente espressione:

$$\alpha = \frac{C_{m,nom}^2}{J_m} \quad (6.7)$$

Il termine  $\beta$  invece viene definito *fattore di carico*, ed è espresso come:

$$\beta = 2[\dot{\omega}_{r,rms}C_{r,rms}^* + (\dot{\omega}_r C_r^*)_{rms}] \quad (6.8)$$

I valori di coppia resistente ed accelerazione quadratiche medie sono definite come:

$$C_{r,rms}^* = \sqrt{\frac{1}{t_c} \int_0^{t_c} [C_r^*(t)]^2 dt} \quad \text{e} \quad \dot{\omega}_{r,rms} = \sqrt{\frac{1}{t_c} \int_0^{t_c} [\dot{\omega}_r(t)]^2 dt} \quad (6.9)$$

Avendo a disposizione i valori di  $\alpha$  e di  $\beta$ , noti una volta che si hanno a disposizione rispettivamente i dati dei motori e del ciclo di lavoro, sarà necessario trovare il range di valori di  $\tau$  che permette di soddisfare la disequazione 6.6. In prima analisi si potrebbe scegliere  $\tau$  in maniera tale che il termine quadratico si annulli. Il rapporto di trasmissione che realizza tale situazione viene definito rapporto di trasmissione ottimo, ed ha la seguente espressione:

$$\tau_{opt} = \sqrt{J_m \frac{\dot{\omega}_{r,rms}}{C_{r,rms}^*}} \quad (6.10)$$

A questo punto la disequazione diventa semplicemente  $\alpha > \beta$ , e quindi è sufficiente scegliere un motore con un *fattore accelerante* che al limite potrà essere uguale al *fattore di carico*. Il valore di rapporto di trasmissione appena trovato potrebbe non essere disponibile fra quelli forniti dal costruttore, esisteranno però due valori limite di  $\tau$  tali che per qualsiasi rapporto di trasmissione compreso fra di essi la disequazione risulti verificata. Sviluppando quindi il termine quadratico e moltiplicando entrambi i per  $\tau^2$  si ottiene:

$$\frac{C_{r,rms}^{*2}}{J_m} \tau^4 - (2C_{r,rms}^* \dot{\omega}_{r,rms} + \alpha - \beta) \tau^2 + \dot{\omega}_{r,rms}^2 J_m < 0 \quad (6.11)$$

Risolvendo l'equazione associata si ottiene:

$$\begin{aligned} \frac{\tau^2}{J_m} &= \frac{2C_{r,rms}^* \dot{\omega}_{r,rms} + \alpha - \beta \pm \sqrt{\alpha - \beta} \sqrt{\alpha - \beta + 4C_{r,rms}^* \dot{\omega}_{r,rms}}}{2C_{r,rms}^{*2}} = \\ &= \frac{4C_{r,rms}^* \dot{\omega}_{r,rms} + 2(\alpha - \beta) \pm 2\sqrt{\alpha - \beta} \sqrt{\alpha - \beta + 4C_{r,rms}^* \dot{\omega}_{r,rms}}}{4C_{r,rms}^{*2}} \end{aligned} \quad (6.12)$$

Nel numeratore si può riconoscere il quadrato del binomio  $(\sqrt{\alpha - \beta + 4C_{r,rms}^* \dot{\omega}_{r,rms}} \pm \sqrt{\alpha - \beta})$ , e quindi si può ricavare:

$$\tau_{min}, \tau_{max} = \sqrt{J_m} \frac{\sqrt{\alpha - \beta + 4C_{r,rms}^* \dot{\omega}_{r,rms}} \pm \sqrt{\alpha - \beta}}{2C_{r,rms}^*} \quad (6.13)$$

Si può notare che nel caso in cui si scelga  $\alpha = \beta$ , le due soluzioni diventano coincidenti e si ricade nel caso  $\tau = \tau_{opt}$ . Viene poi calcolato il rapporto fra la velocità massima richiesta dal ciclo di lavoro e la velocità massima del motore:

$$\tau_{lim} = \frac{\max|\omega_r(t)|}{\omega_{m,max}} \quad (6.14)$$

Ci si deve quindi assicurare che il rapporto di trasmissione scelto sia maggiore, poichè se così non fosse, la velocità di rotazione richiesta al motore per generare quella massima dell'utilizzatore risulterebbe maggiore di quella consentita. In definitiva il rapporto di trasmissione potrà spaziare all'interno del range:

$$\max(\tau_{lim}, \tau_{min}) < \tau < \tau_{max} \quad (6.15)$$

per cui verranno scartati a priori i motoriduttori con  $\tau_{lim} > \tau_{max}$ . A questo punto l'ultima verifica che resta da fare è quella sulla coppia massima.

### 6.3 Coppia resistente

I dati che vengono estrapolati dall'analisi dinamica devono essere rielaborati per il calcolo del *fattore di carico*  $\beta$ . Per l'andamento dell'accelerazione angolare lato utilizzatore di ciascuna unità lineare è sufficiente dividere l'accelerazione dell' $i$ -esima coordinata di giunto per il rapporto di trasmissione rotazionale:

$$\dot{\omega}_{r,i}(t) = \frac{\ddot{q}_i(t)}{\tau_{rot}} \quad (6.16)$$

Mentre per il calcolo della coppia resistente è necessario calcolare separatamente ogni singolo contributo per poi sommarli. Oltre ai termini inerziali, ad opporsi al motore ci saranno anche i contributi legati agli attriti e quelli relativi alle forze trasmesse dalla macchina:

$$C_{r,i}^*(t) = C_{inerziale,i}(t) + C_{attrito,i}(t) + C_{robot,i}(t) \quad (6.17)$$

I tre termini sono stati calcolati nel seguente modo:

1. *contributo inerziale*: una parte della coppia motrice servirà per accelerare le masse degli elementi che compongono le unità lineari, per cui questo contributo assumerà una formulazione del tipo:

$$C_{inerziale,i}(t) = \left( \tau_{rot} m_{tot} + \frac{J_{tot}}{\tau_{rot}} \right) \ddot{q}_i(t) \quad (6.18)$$

dove, a seconda del tipo di unità lineare usata, le masse, le inerzie, e i rapporti di trasmissione assumeranno i valori riportati in Tab. 6.3.

In cui  $p_h$  è il passo dell'elica della vite, mentre  $j_{specifico}$  è il momento d'inerzia per unità di lunghezza della vite fornito dal costruttore.

	$\tau_{rot}$	$m_{tot}$	$J_{tot}$
unità a cinghia	$\frac{d_{puleggia}}{2}$	$m_{slitta} + m_{cinghia}$	$2J_{puleggia}$
unità a vite	$\frac{p_h}{2\pi}$	$m_{slitta}$	$j_{specifico}L_{vite}$

**Tab. 6.3:** Dati delle inerzie dei due tipi di trasmissione.

2. *attriti*: il secondo contributo tiene conto della potenza dissipata a causa degli attriti interni all'unità di trasmissione e di quelli fra la slitta e la guida. Esso viene calcolato come:

$$C_{attrito,i}(t) = C_{r,zero}(|n(t)|)sign(\dot{q}_i(t)) + \mu\tau_{rot}|F_{A_i,z}(t) + m_{tot}g|sign(\dot{q}_i(t)) \quad (6.19)$$

dove  $\mu$  indica il coefficiente di attrito fra slitta e guida. Il coefficiente  $C_{r,zero}$  viene tabulato in funzione del numero di giri  $n(t)$  e è fornito dal costruttore delle guide lineari.

3. *forze esercitate dal robot*: a differenza delle componenti lungo l'asse  $y$  e  $z$  che vengono scaricati sui supporti, la componente di forza esercitata dal robot lungo l'asse  $x$  della slitta contribuisce ad aumentare o a diminuire, a seconda di come è diretta, la coppia richiesta al motore. La quotaparte di coppia resistente ad essa associata avrà la seguente espressione:

$$C_{robot,i}(t) = \tau_{rot}F_{A_i,x}(t) \quad (6.20)$$

A questo punto si può procedere col calcolo dei valori quadratici medi di coppia ed accelerazione angolare, e successivamente al calcolo di  $\beta$ .

## 6.4 Risultati

Le analisi dinamiche sono state condotte solo sulla macchina finale. Le situazioni analizzate sono le seguenti:

1. caso dinamico: piattaforma ( $m_P = 50 \text{ kg}$ ) + generatore eolico ( $m = 15 \text{ kg}$ );
2. caso dinamico: piattaforma ( $m_P = 50 \text{ kg}$ ) + barca a vela ( $m = 15 \text{ kg}$ );
3. caso statico: piattaforma ( $m_P = 50 \text{ kg}$  e  $m_P = 100 \text{ kg}$ ) + generatore eolico ( $m = 15 \text{ kg}$ );
4. caso statico: piattaforma ( $m_P = 50 \text{ kg}$  e  $m_P = 100 \text{ kg}$ ) + barca a vela ( $m = 15 \text{ kg}$ ).

Sono state quindi ricavate in ciascun caso i valori di velocità, accelerazione, e carichi massimi a cui vengono sottoposte le guide lineari. A questo punto è stato possibile scegliere da catalogo delle guide che soddisfacessero tutti i requisiti trovati. Per le guide a vite a ricircolo di sfere si è fatto riferimento al catalogo di Bosch Rexrot, mentre per quanto riguarda le guide a cinghia sono state prese in considerazione le guide di Rollon. La scelta è ricaduta sulle seguenti trasmissioni:

- guida a vite senza fine Bosch CKK 25-200, con guida a sfere su rotaia e con passo dell'elica massimo tra quelli resi disponibili da Bosch (passo  $p_h = 32 \text{ mm}$  e diametro  $d_{0,screw} = 32 \text{ mm}$ ).
- Rollon Robot SP 160, con guide a ricircolo di sfere.

Gli estratti dei cataloghi relativi a queste due trasmissioni si trovano nelle Appendici C e D. In Tab. 6.4 vengono riportati i dati relative ai due sistemi di trasmissione scelti, insieme alle prestazioni richieste dalla macchina.

		Unità di trasmissione		
		Bosch CKK25-200 $p_h = 32\text{mm}$	Rollon SP160	Valori richiesti
$v_{\max}$	[m/s]	1.6	5	1.7
$a_{\max}$	[m/s <sup>2</sup> ]	–	50	13.02
$F_x$	[N]	3090	3090	2111
$F_y$	[N]	8350	9600	564
$F_z$	[N]	8350	9600	1212

**Tab. 6.4:** Selezione unità di trasmissione: prestazioni fornite vs. richieste.

Si nota che nel caso della trasmissione a vite pur adottando il passo più elevato fra quelli disponibili non si riesce a soddisfare le richieste in termini di velocità massime. Va però considerato che i valori riportati nel catalogo dal costruttore si riferiscono a macchine in servizio continuativo, e quindi opportunamente ridotti per assicurare un'elevata vita a fatica. Le applicazioni per cui è stato progettato l'Hexaslide invece richiedono poche ore di funzionamento ed un numero di cicli di lavoro decisamente inferiore, per questo motivo si ritiene che le trasmissioni CKK25-200 possano comunque essere adoperate. In ogni caso nel momento della costruzione della macchina verrà contattato il produttore per indagare le effettive potenzialità della trasmissione. In Fig. 6.3 viene mostrato il confronto fra i valori massimi e minimi di azione assiale lungo i link per i vari tipi di movimentazione. Le azioni assiali di trazione e di compressione massime rilevate nel caso della rotazione di beccheggio sono riportate in Fig. 6.4. In Fig. 6.5 invece vengono riportati i valori massimi in valore assoluto delle forze che si scaricano in corrispondenza dei giunti universali posti sulle guide lineari. Si può procedere ora con la mappatura di  $\beta$ . Benchè la situazione peggiore in termini di coppia massima richiesta si registri per la rotazione di beccheggio, la condizione di carico più gravosa in termini di valori

quadratici medi di coppia ed accelerazione angolare si verifica nel caso relativo alla movimentazione lungo l'asse  $z$ . In Fig. 6.6 vengono confrontati i valori di  $\beta_{\max}$  e coppia massima per le varie movimentazioni.

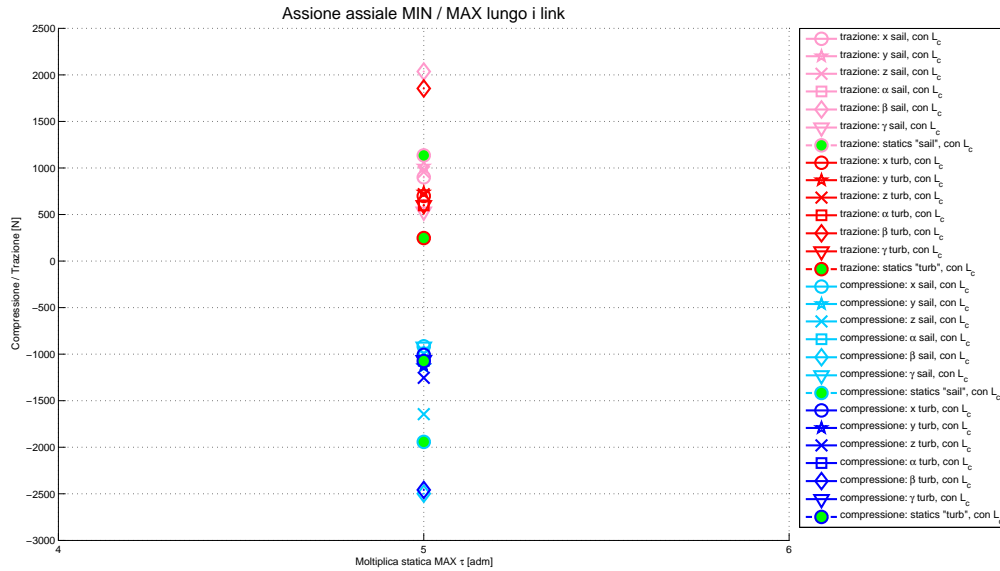
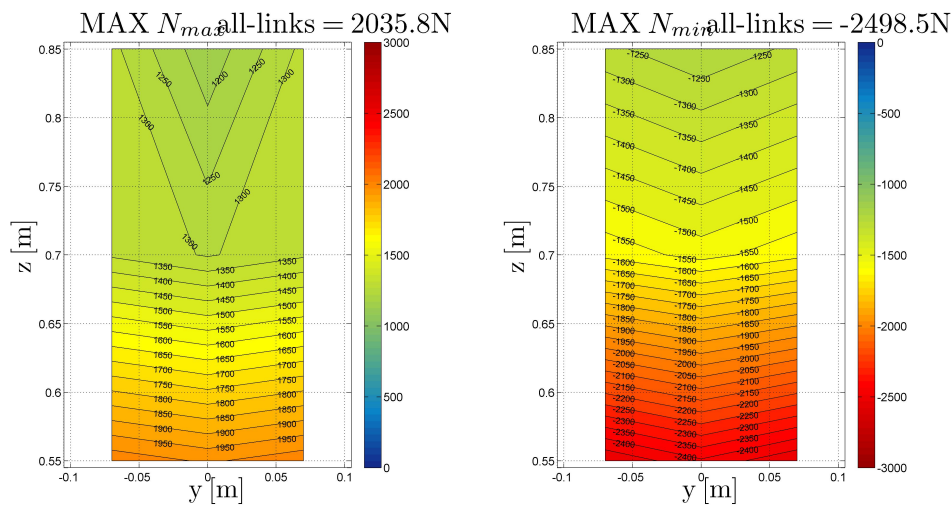
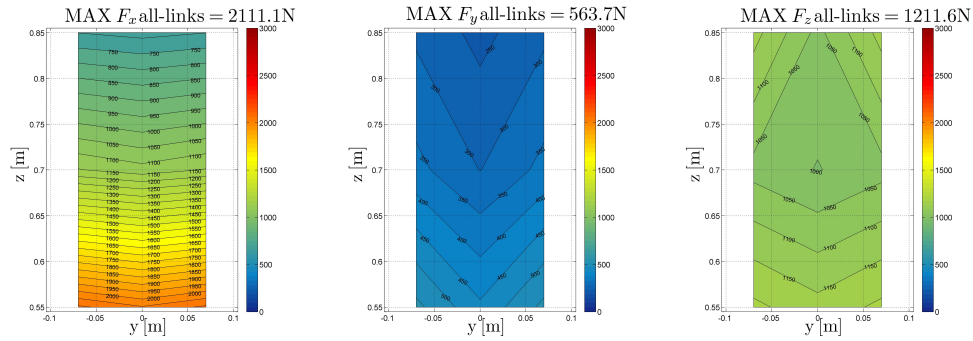


Fig. 6.3: Confronto fra i vari valori di azione assiale lungo i link per varie movimentazioni.



(a) Azione assiale di trazione massima (b) Azione assiale di compressione minima durante la rotazione di beccheggio.

Fig. 6.4: Forze massime in direzione assiale lungo i link.



(a) Forza diretta lungo l'asse  $x$ , ovvero quello della guida. (b) Forza diretta lungo l'asse  $y$ , ovvero quello trasversale. (c) Forza diretta lungo l'asse  $z$ , ovvero quello verticale.

**Fig. 6.5:** Forze massime lungo gli assi delle guide.

A questo punto sarà necessario utilizzare la procedura sopra descritta per la scelta del rapporto di riduzione e del motore, e successivamente sarà necessario verificare che la coppia massima richiesta nel caso della rotazione di beccheggio sia inferiore rispetto a quella fruibile dal motore scelto. Le mappe di  $\beta$  sia per la trasmissione a vite che per quella a cinghia sono riportate in Fig. 6.7.

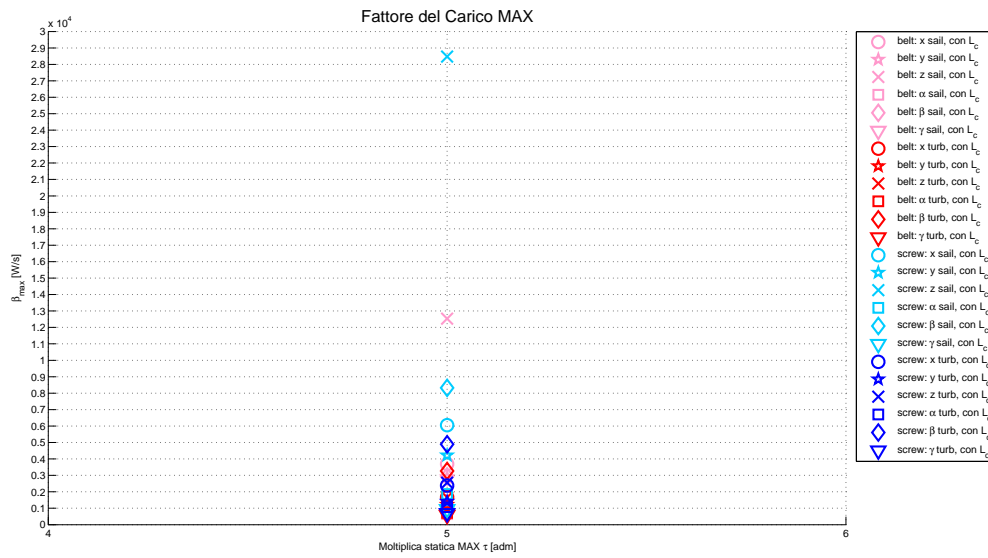
Si può ora valutare per entrambi i casi quale sia la massima forza assiale a cui sono soggette le guide e quale sia la percentuale di tale forza dovuta alle inerzie della vite e del gruppo cinghia-puleggie. In Tab. 6.5 viene calcolata la massa traslante equivalente nei due casi e la forza necessaria ad imprimere a tale massa un'accelerazione di circa  $1g$ .

Massa traslante equivalente [kg]						
Unità lineare: Bosch CKK25-200 (vite)						
$p_h$ [mm]	$\rightarrow$	$\frac{J_{screw}}{[p_h/(2\pi)]^2}$	$+ m_{slider}$	$= m_{CKK25-200,eq}$	$  F_{1g}$ [N]	$\frac{F_{1g}}{\max F_{A_i,x}}$ [%]
32		52.9	12.5	65.4	641	46%
Unità lineare: Rollon Robot SP160 (cinghia)						
$\frac{J_{pul}}{(d_0/2)^2}$	$+$	$m_{belt}$	$+ m_{slider}$	$= m_{SP160,eq}$	$  F_{1g}$ [N]	$\frac{F_{1g}}{\max F_{A_i,x}}$ [%]
0.9		1.64	12.6	15.14	149	11%

**Tab. 6.5:** Contributo inerziale dato dall'unità di trasmissione, ridotto alla slitta.  $F_{1g}$  è la forza per accelerare  $m_{eq}$  di  $1g$ .  $\max F_{A_i,x} = 1388N$  è la forza di traslazione massima richiesta nel caso di "barca a vela" per un'accelerazione  $\ddot{q}_i \approx 1g$ .

Avendo a disposizione i dati del fattore di carico, della coppia resistente e dell'accelerazione angolare (riportati in Tab. 6.6), si può procedere alla scelta del gruppo motore-riduttore. Si è deciso di focalizzare l'attenzione sui motori della Kollmorgen, marchio che fa riferimento alla multinazionale Danaher, così come i



(a) Confronto fra i massimi valori di  $\beta$ .(b) Confronto fra i massimi valori di  $C_{r,max}^*$ .

**Fig. 6.6:** Confronto fra i vari tipi di movimentazione in termini di massima coppia resistente e massimo fattore di carico.

riduttori della Micron che sono stati presi in esame, i cui cataloghi e la numerazione adottata sono riportati in Appendice E.

		Bosch CKK25-200 (vite)	Rollon SP160 (cinghia)
$\beta_{max}$	[kW/s]	28.5	12.5
$C_{r,rms}^*$	[Nm]	4.3	10.1
$\dot{\omega}_{r,rms}$	[rad/s <sup>2</sup> ]	1972	264

**Tab. 6.6:** Valori di fattore di carico, coppia resistente ed accelerazione angolare per le due trasmissioni.

Si riportano quindi i grafici con il confronto fra  $\alpha$  e  $\beta_{max}$  (Fig. 6.8 e Fig. 6.9), e quelli relativi alla scelta del rapporto di trasmissione (Fig. 6.10 e Fig. 6.11).

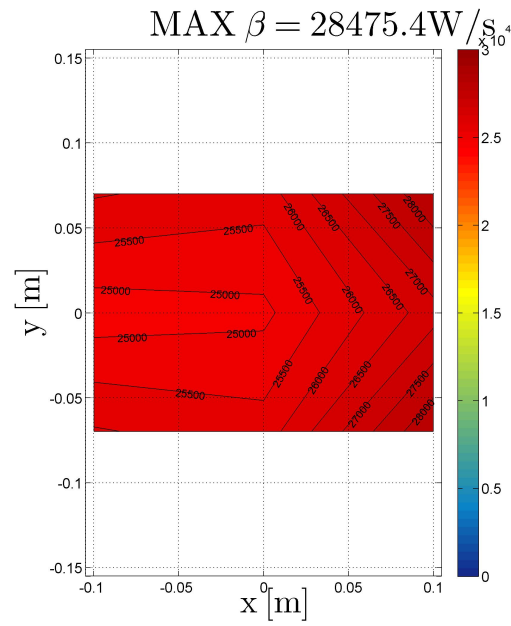
Si può vedere che con la trasmissione Bosch CKK25-200 il primo motore utile è il 32 con accoppiamento diretto, mentre nel caso di trasmissione Rollon SP160 sarà il 14 con rapporto di trasmissione 1:5. In Tab. 6.7 e Tab. 6.8 viene riportata la verifica sulla coppia massima, considerando sia il caso con movimento lungo  $z$ , sia quello relativo alla rotazione di beccheggio.

Verifica coppia massima - trasm. a vite CKK25-200		
$\tau_{rot} = \frac{p_h}{2\pi} = 5.1 \cdot 10^{-3} \text{m/rad}$		
		Barca a vela
		$z$ mov.
$C_r^*$	[Nm]	17.6 (max)
$\dot{\omega}_r$	[rad/s <sup>2</sup> ]	4755 (max)
1) Gruppo motore-riduttore: AKM5-2L (mot. n.32) con "collegamento diretto"		
$J_m$ con freno	[kg-cm <sup>2</sup> ]	6.37
$C_m$ max. richiesta	[Nm]	20.63
$C_m$ max. disponibile	[Nm]	22

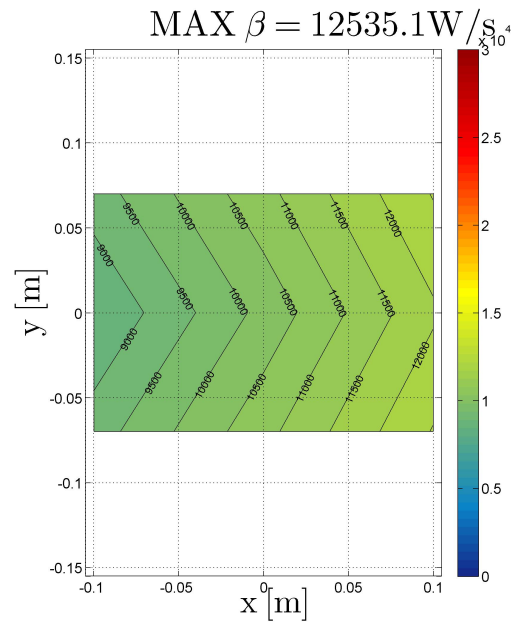
**Tab. 6.7:** Verifica coppia motrice massima: richiesta vs. disponibile, nel caso di movimento lungo  $z$ .

Verifica coppia massima - trasm. a cinghia SP160			
$\tau_{rot} = \frac{d_0}{2} = 3.19 \cdot 10^{-2} \text{m/rad}$			
		Barca a vela	Barca a vela
		rot. $\beta$	z mov.
$C_r^*$	[Nm]	73.1 (max)	47.6
$\dot{\omega}_r$	[rad/s <sup>2</sup> ]	332.81	760.44 (max)
1) Gruppo motore-riduttore: AKM3-3H (mot. n.15) con $\tau = 0.1$			
$J_m$ con freno	[kg·cm <sup>2</sup> ]	0.862	0.862
$C_m$ max. richiesta	[Nm]	7.6	5.42
$C_m$ max. disponibile	[Nm]	10.22	10.22

**Tab. 6.8:** Verifica coppia motrice massima: richiesta vs. disponibile, nei casi di: rotazione di beccheggio, movimento lungo  $z$ .



(a)  $\beta_{max}$  con trasmissione a vite Bosch CKK25-200, movimentazione lungo l'asse  $z$  con barca a vela.



(b)  $\beta_{max}$  con trasmissione a cinghia Rollon SP160, movimentazione lungo l'asse  $z$  con barca a vela.

**Fig. 6.7:** Massimo fattore di carico per entrambi i tipi di trasmissione.

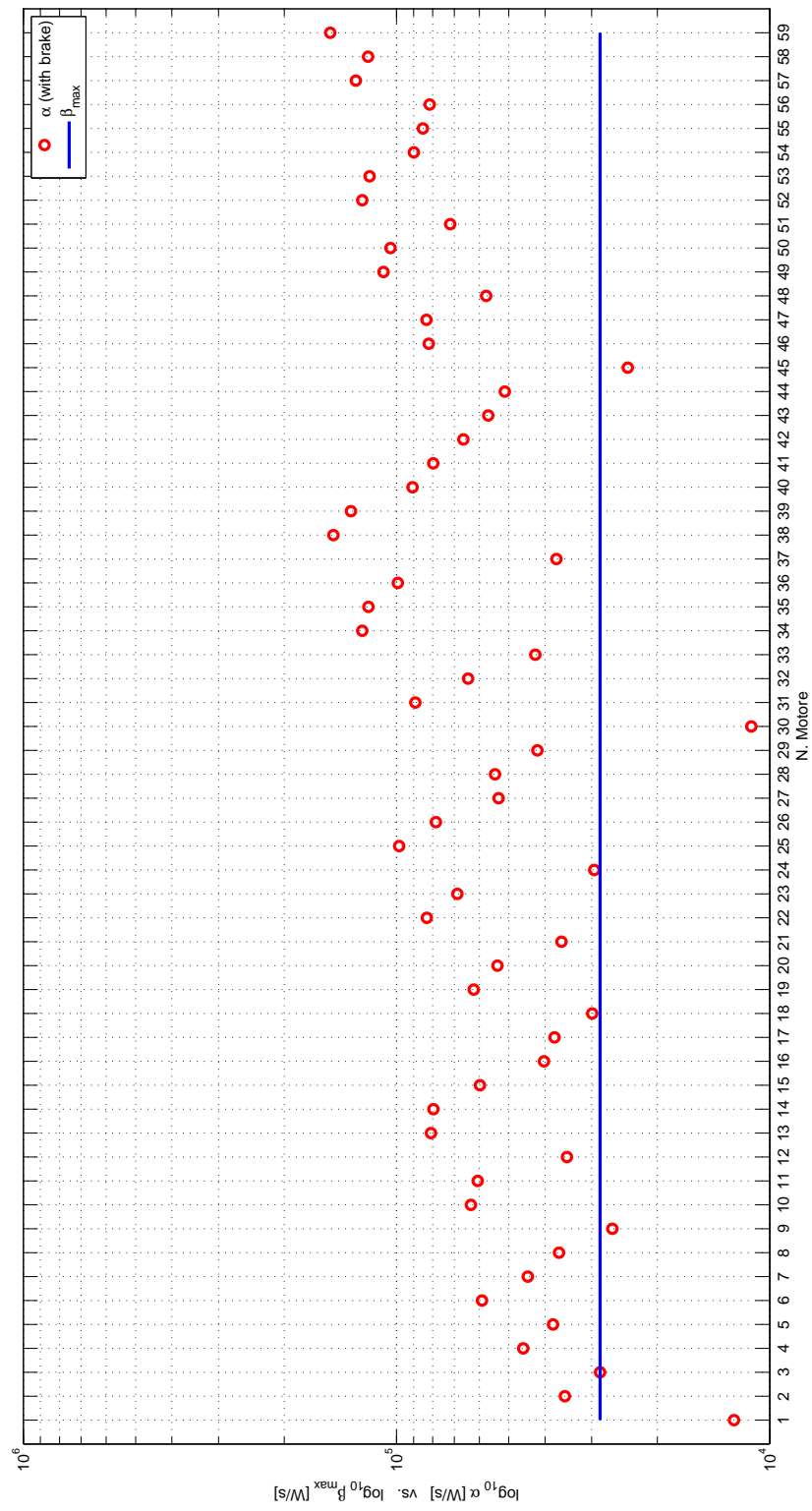


Fig. 6.8:  $\alpha$  vs.  $\beta$  nel caso con trasmissione a vite.

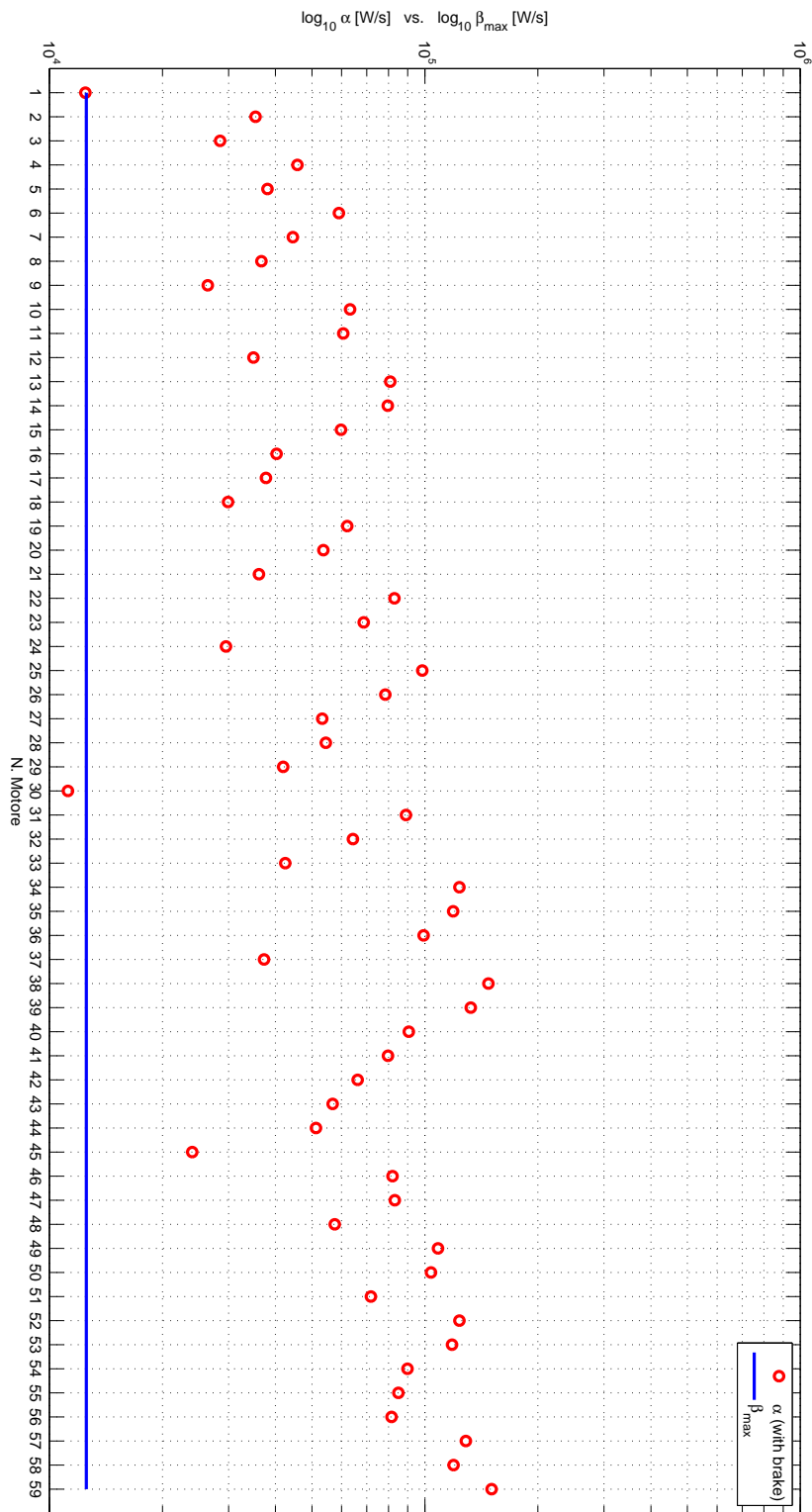


Fig. 6.9:  $\alpha$  vs.  $\beta$  nel caso con trasmissione a cinghia.

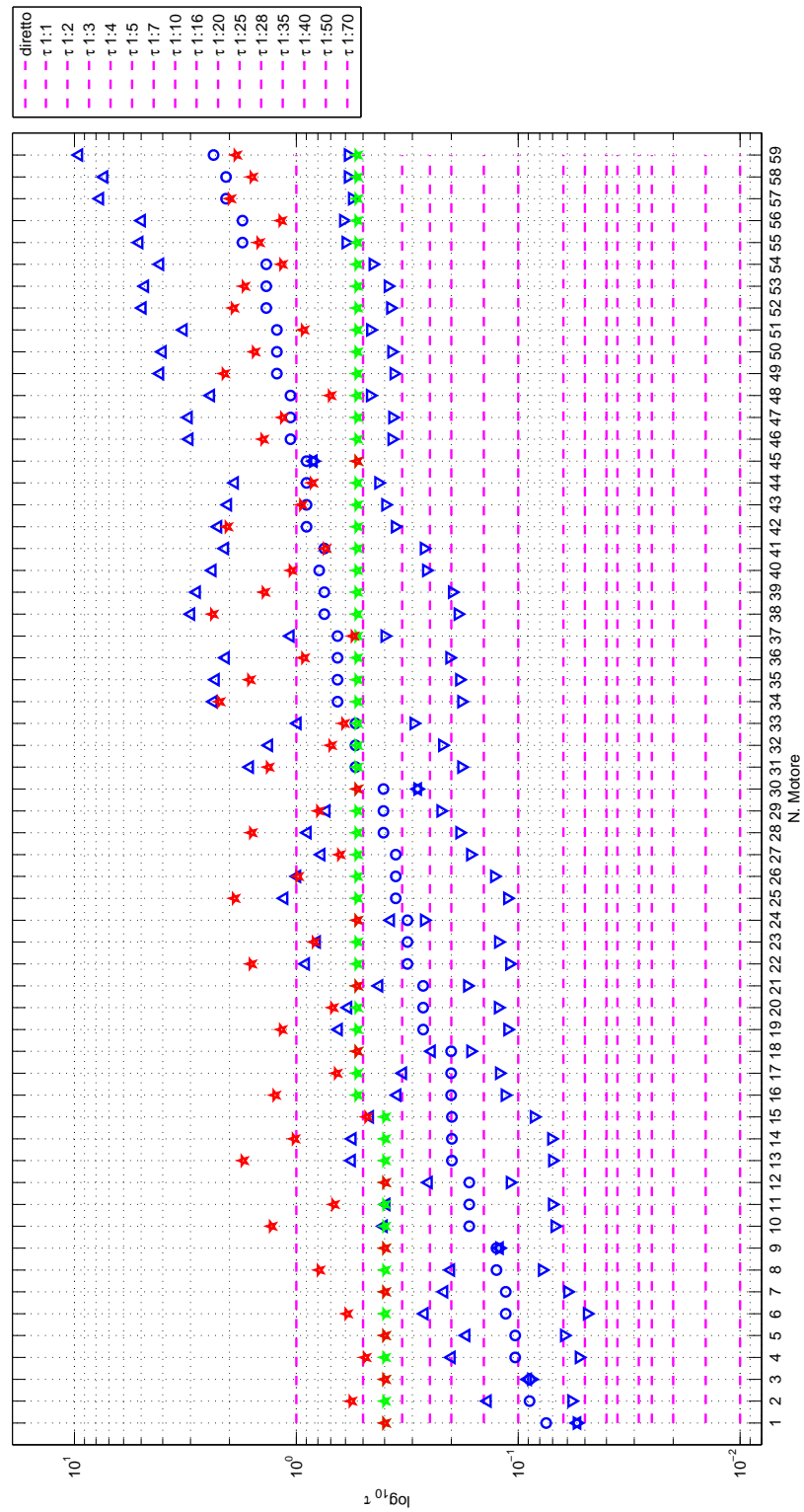


Fig. 6.10: Rapporto di trasmissione per vari motori nel caso con trasmissione a vite.

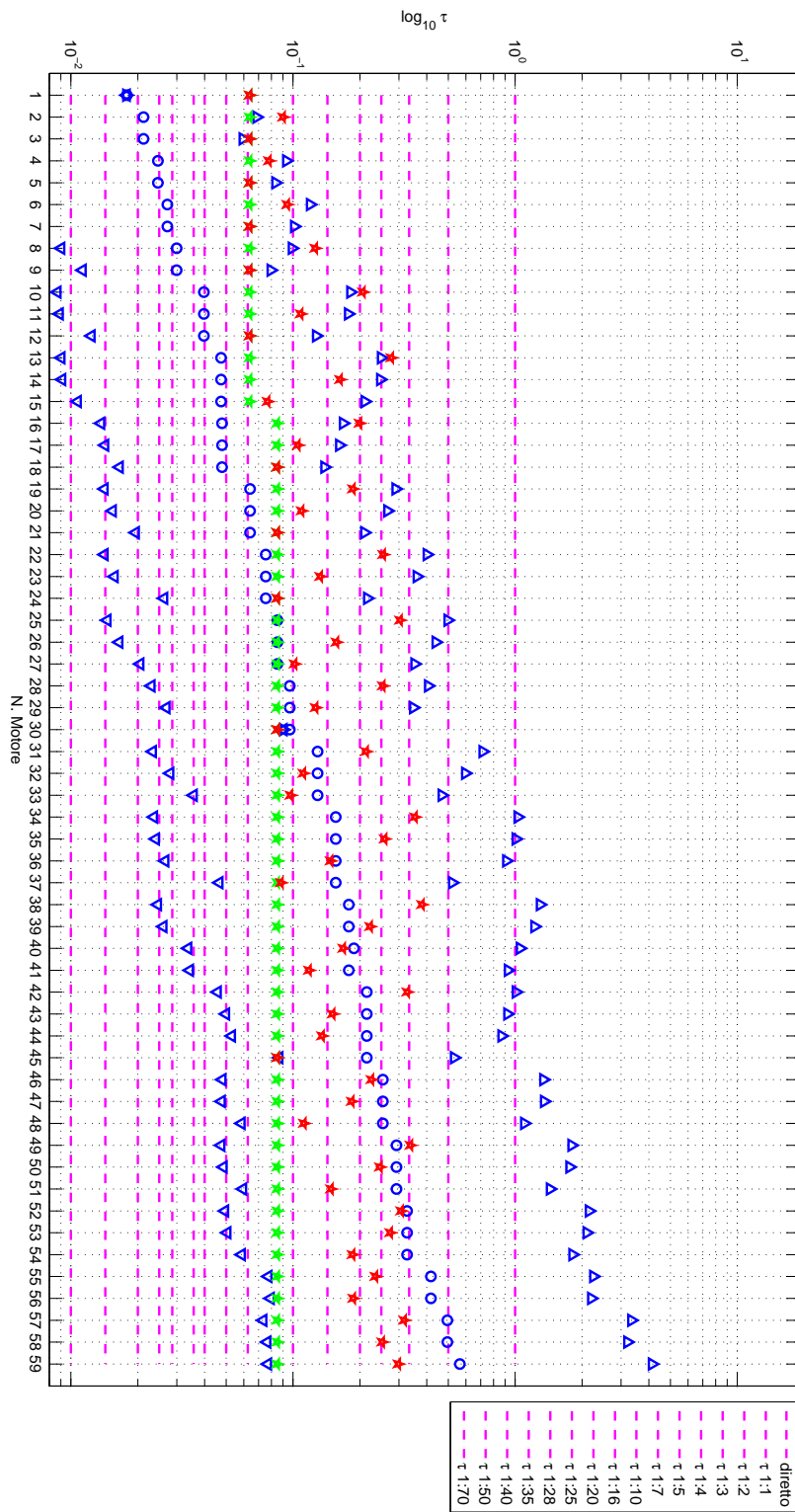


Fig. 6.11: Rapporto di trasmissione per vari motori nel caso con trasmissione a cinghia.



## CONCLUSIONI

Questo lavoro di tesi presenta la metodologia utilizzata per il progetto preliminare di una macchina per applicazioni di tipo hardware-in-the-loop in galleria del vento. Dopo le considerazioni fatte per scegliere l'architettura che si addicesse di più alle specifiche fornite dai ricercatori della galleria del vento del Politecnico di Milano, vengono descritti gli algoritmi genetici usati per la sintesi cinetostatica ed i modelli multibody flessibili costruiti per mappare l'andamento delle prime frequenze proprie delle macchine all'interno dello spazio di lavoro. Sia i risultati dell'ottimizzazione cinetostatica, sia quelli ottenuti dall'analisi dinamica in piccolo sono stati di fondamentale importanza per la scelta di quella che dovrà essere la macchina finale. Infine è stata necessaria la costruzione di un modello multibody rigido per l'analisi dinamica in grande, al fine di avere un'idea chiara di quelle che potessero essere le forze in gioco durante le reali condizioni di funzionamento. A questo punto, note le richieste che il sistema di attuazione dovrà essere in grado di soddisfare, è stato possibile dimensionare il gruppo motore-riduttore. I metodi di progetto utilizzati assumono un carattere generale in quanto sia gli algoritmi genetici che le macro di ADAMS possono essere riadattati facilmente per altri tipi di macchine ed applicazioni.

I possibili sviluppi futuri potrebbero essere:

1. Costruzione di un modello agli elementi finiti della piattaforma, sia per valutare come cambino le frequenze proprie del sistema meccanico rispetto al caso rigido, sia per verificare che la soluzione costruttiva trovata riesca a sopportare le sollecitazioni a cui verrà sottoposta in fase operativa. Sarà quindi necessaria una prima fase di progettazione del solo corpo centrale della piattaforma, in modo da avere una struttura che sia allo stesso tempo leggera e rigida.
2. Calcolo delle distanze fra gli elementi della macchina considerando non solo il caso statico ma anche la movimentazione durante un task tipico.
3. la collaborazione con i ricercatori della galleria del vento per la definizione di una legge di moto che rispecchi il modello matematico del mare coinvolgendo contemporaneamente tutte le coordinate dello spazio di lavoro. In questo modo sarà possibile avere delle informazioni più dettagliate sulle richieste in termini di coppia ed accelerazione, e si potrà procedere quindi ad una verifica del dimensionamento degli attuatori.

4. Confronto del moto in grande del robot nel caso infinitamente rigido e nel caso flessibile, in modo da poter valutare in maniera preliminare di quanto si discosti il comportamento reale della macchina dal caso ideale.
5. Sviluppo degli algoritmi di controllo, con integrati i modelli delle trasmissioni e dei motori elettrici. Si potrà valutare se sia opportuno adottare uno schema di controllo di tipo centralizzato o se sia sufficiente un controllo decentralizzato. Oltre alla compensazione di eventuali errori di posizionamento, bisognerà prevedere delle strategie di arresto in caso di pericolo che permettano alla macchina di arrestarsi in condizioni di sicurezza.
6. Realizzazione di un eventuale modello in scala su cui poter fare la messa a punto dell'hardware-in-the-loop prima di entrare in galleria del vento.

## APPENDICE A

### Matrice di Rotazione e Velocità Angolare

Per descrivere l'orientazione di un corpo nello spazio si può descrivere la relazione che intercorre fra una terna mobile solidale col corpo ed una terna globale fissa. Si può passare dalla terna fissa a quella mobile, e viceversa, attraverso una serie di tre rotazioni successive attorno a tre assi distinti. Ogni rotazione è descritta da una matrice di rotazione e la rotazione complessiva risulta quindi dal prodotto di tre matrici come illustrato di seguito.

Si definisce con  $\mathbf{b}_i$  un vettore nel sistema di riferimento  $i$ -esimo e con  $\mathbf{b}_j$  lo stesso vettore ma espresso nel sistema di riferimento  $j$ -esimo. La matrice di rotazione  $[R_{ij}]$  permette di passare dal sistema di riferimento  $i$ -esimo al sistema  $j$ -esimo. Si potrà quindi scrivere:

$$\mathbf{b}_i = [R_{ij}]\mathbf{b}_j \quad (\text{A.1})$$

Le colonne di  $[R_{ij}]$  rappresentano i coseni direttori degli assi della terna  $j$ -esima rispetto alla terna  $i$ -esima. Indicando con  $\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{z}}_i$  i versori della terna  $i$ -esima, e con  $\hat{\mathbf{x}}_j, \hat{\mathbf{y}}_j, \hat{\mathbf{z}}_j$  quelli della terna  $j$ -esima, la matrice  $[R_{ij}]$  potrà essere riscritta come:

$$[R_{ij}] = \begin{bmatrix} \hat{\mathbf{x}}_j \cdot \hat{\mathbf{x}}_i & \hat{\mathbf{y}}_j \cdot \hat{\mathbf{x}}_i & \hat{\mathbf{z}}_j \cdot \hat{\mathbf{x}}_i \\ \hat{\mathbf{x}}_j \cdot \hat{\mathbf{y}}_i & \hat{\mathbf{y}}_j \cdot \hat{\mathbf{y}}_i & \hat{\mathbf{z}}_j \cdot \hat{\mathbf{y}}_i \\ \hat{\mathbf{x}}_j \cdot \hat{\mathbf{z}}_i & \hat{\mathbf{y}}_j \cdot \hat{\mathbf{z}}_i & \hat{\mathbf{z}}_j \cdot \hat{\mathbf{z}}_i \end{bmatrix} \quad (\text{A.2})$$

Se si considerano tre rotazioni successive attorno a tre assi coordinati, per passare dal sistema di riferimento (3) al sistema di riferimento (0), si potrà scrivere:

$$\mathbf{b} = [R_{01}][R_{12}][R_{23}]\mathbf{b}' = [R_{03}]\mathbf{b}' = [R]\mathbf{b}' \quad (\text{A.3})$$

dove  $\mathbf{b}'$  è espresso nel sistema di riferimento (3). La matrice di rotazione inoltre è ortogonale, e quindi varrà:

$$R^{-1} = R^T \quad (\text{A.4})$$

#### A.1 Angoli Nautici XYZ

In questo lavoro di tesi vengono utilizzati gli angoli nautici o di Cardano. Vengono usati spesso utilizzati in ambito aerospaziale e sono particolarmente adatti a descrivere il problema in esame. Vengono definiti come:

- $\alpha$  rollio: rotazione attorno all'asse  $x$  della prima terna.
- $\beta$  beccheggio: rotazione attorno all'asse  $y$  della seconda terna, ovvero quella ottenuta dopo la rotazione di rollio.
- $\gamma$  imbardata: rotazione attorno all'asse  $y$  della terza terna, ovvero quella ottenuta dopo la rotazione di beccheggio.

Le tre matrici di rotazione che descrivono le singole rotazioni sono:

$$R_{01} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \quad (\text{A.5})$$

$$R_{12} = \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \quad (\text{A.6})$$

$$R_{23} = \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A.7})$$

Mentre la matrice di rotazione che permette di passare da una terna mobile a quella fissa è:

$$R = \begin{bmatrix} c\beta c\gamma & -c\beta s\gamma & s\beta \\ s\alpha s\beta c\gamma + c\alpha s\gamma & -s\alpha s\beta s\gamma + c\alpha c\gamma & -s\alpha c\beta \\ -c\alpha s\beta c\gamma + s\alpha s\gamma & c\alpha s\beta s\gamma + s\alpha c\gamma & c\alpha c\beta \end{bmatrix} \quad (\text{A.8})$$

Nel caso in cui valga  $\cos(\beta) = 0$  la matrice di rotazione diventa singolare.

## A.2 Velocità angolare

Si riporta la definizione del vettore  $\mathbf{d}$ :

$$\mathbf{d} = \mathbf{p} + [R]\mathbf{b}' \quad (\text{A.9})$$

Derivando rispetto al tempo si ottiene:

$$\mathbf{v} = \mathbf{v}_C + \frac{d}{dt}([R])\mathbf{b}' = \mathbf{v}_C + \frac{d}{dt}([R])[R]^T\mathbf{b} \quad (\text{A.10})$$

in cui si riconosce la matrice di velocità angolare:

$$[\Omega] = \frac{d}{dt}([R])[R]^T \quad (\text{A.11})$$

se si esprime la sua trasposta:

$$[\Omega]^T = [R]\left(\frac{d}{dt}([R])\right)^T = [R]\left(\frac{d}{dt}([R]^T)\right) \quad (\text{A.12})$$

si può dimostrare che si tratta di una matrice emisimmetrica, infatti:

$$\frac{d}{dt}([I_6]) = \frac{d}{dt}([R][R]^T) = \frac{d}{dt}([R])[R]^T + [R]\frac{d}{dt}([R]^T) \quad (\text{A.13})$$

$$\frac{d}{dt}([R])[R]^T = -[R]\frac{d}{dt}([R]^T) \quad (\text{A.14})$$

e quindi:

$$[\Omega] = -[\Omega]^T \quad (\text{A.15})$$

la sua struttura quindi sarà del tipo:

$$[\Omega] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (\text{A.16})$$

Alla forma matriciale si può sostituire una forma vettoriale, e viceversa:

$$[\Omega]\mathbf{b} = \boldsymbol{\omega} \times \mathbf{b} \quad (\text{A.17})$$

Nel caso di angoli Nautici XYZ la matrice  $[\Omega]$  diventa:

$$[\Omega] = \begin{bmatrix} 0 & -\dot{\beta}s(\alpha) - \dot{\gamma}c(\alpha)c(\beta) & \dot{\beta}c(\alpha) - \dot{\gamma}c(\beta)s(\alpha) \\ \dot{\beta}s(\alpha) + \dot{\gamma}c(\alpha)c(\beta) & 0 & -\dot{\alpha} - \dot{\gamma}s(\beta) \\ -\dot{\beta}c(\alpha) + \dot{\gamma}c(\beta)s(\alpha) & \dot{\alpha} + \dot{\gamma}s(\beta) & 0 \end{bmatrix} \quad (\text{A.18})$$

da questa espressione si può ricavare il legame che intercorre fra le derivate degli angoli nautici ed il vettore velocità angolare:

$$\begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix} = \begin{bmatrix} 1 & 0 & s(\beta) \\ 0 & c(\alpha) & -s(\alpha)c(\beta) \\ 0 & s(\alpha) & c(\alpha)c(\beta) \end{bmatrix} \begin{Bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{Bmatrix} \quad (\text{A.19})$$



## APPENDICE B

### Codici ADAMS relativi al calcolo delle frequenze proprie

Si riportano di seguito i frammenti di codice in ambiente ADAMS usati per la creazione del modello multibody e per le analisi nel caso relativo all' Hexaslide con trasmissione a vite a ricircolo di sfere.

#### B.1 Creazione del modello e dei marker di riferimento

```
model create model=.hexaslide
!
default units length=meter mass=kg force=newton time=Second
angle=degrees frequency=hz
!
force create body gravitational gravity_field_name=accgrav &
x_component_gravity=0.0 &
y_component_gravity=0.0 &
z_component_gravity=-9.80665
!
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!CREAZIONE TCP
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
marker create marker_name=.ground.TCP location= 0.0, 0.0, ($z_wsd) &
orientation=0.0, 0.0, 0.0
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!CREAZIONE ORIGINE
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
marker create marker_name=.ground.Origin &
location= 0.0, 0.0, 0.0 &
orientation=0.0, 0.0, 0.0
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!CREAZIONE MARKER B
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
marker create marker_name=.Hexaslide.ground.B1 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p)),-($r_p*sin($theta_p)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
marker create marker_name=.Hexaslide.ground.B2 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p)),$r_p*sin($theta_p)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
marker create marker_name=.Hexaslide.ground.B3 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p-120.0d)),-($r_p*sin($theta_p-120.0d)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
marker create marker_name=.Hexaslide.ground.B4 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p+120.0d)),$r_p*sin($theta_p+120.0d)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
marker create marker_name=.Hexaslide.ground.B5 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p+120.0d)),-($r_p*sin($theta_p+120.0d)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
marker create marker_name=.Hexaslide.ground.B6 &
location=(LOC_RELATIVE_TO({($r_p*cos($theta_p-120.0d)),$r_p*sin($theta_p-120.0d)),$h_z}),.TCP) &
orientation=0.0,0.0,0.0
```

```

!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!CREAZIONE MARKER A
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
var set var=d1_q &
real=((dx(.B1,.Origin)**2+(dy(.B1,.Origin)+$s)**2+(dz(.B1,.Origin)**2)
var set var=d2_q &
real=((dx(.B2,.Origin)**2+(dy(.B2,.Origin)+$s)**2+(dz(.B2,.Origin)**2)
var set var=d3_q &
real=((dx(.B3,.Origin)**2+(dy(.B3,.Origin)+$s)**2+(dz(.B3,.Origin)**2)
var set var=d4_q &
real=((dx(.B4,.Origin)**2+(dy(.B4,.Origin)+$s)**2+(dz(.B4,.Origin)**2)
var set var=d5_q &
real=((dx(.B5,.Origin)**2+(dy(.B5,.Origin)+$s)**2+(dz(.B5,.Origin)**2)
var set var=d6_q &
real=((dx(.B6,.Origin)**2+(dy(.B6,.Origin)+$s)**2+(dz(.B6,.Origin)**2)
!
var set var=d1_u real=(dx(.B1,.Origin))
var set var=d2_u real=(dx(.B2,.Origin))
var set var=d3_u real=(-0.5*(dx(.B3,.Origin))+sqrt(3)*0.5*(dy(.B3,.Origin)))
var set var=d4_u real=(-0.5*(dx(.B4,.Origin))+sqrt(3)*0.5*(dy(.B4,.Origin)))
var set var=d5_u real=(-0.5*(dx(.B5,.Origin))-sqrt(3)*0.5*(dy(.B5,.Origin)))
var set var=d6_u real=(-0.5*(dx(.B6,.Origin))-sqrt(3)*0.5*(dy(.B6,.Origin)))
!
!
var set var=delta1 real=($lunghezza**2+(.hexaslide.d1_u)**2-(.hexaslide.d1_q))
var set var=delta2 real=($lunghezza**2+(.hexaslide.d2_u)**2-(.hexaslide.d2_q))
var set var=delta3 real=($lunghezza**2+(.hexaslide.d3_u)**2-(.hexaslide.d3_q))
var set var=delta4 real=($lunghezza**2+(.hexaslide.d4_u)**2-(.hexaslide.d4_q))
var set var=delta5 real=($lunghezza**2+(.hexaslide.d5_u)**2-(.hexaslide.d5_q))
var set var=delta6 real=($lunghezza**2+(.hexaslide.d6_u)**2-(.hexaslide.d6_q))
!
var set var=q1 real=((.hexaslide.d1_u)+sqrt(.hexaslide.delta1))
var set var=q2 real=((.hexaslide.d2_u)+sqrt(.hexaslide.delta2))
var set var=q3 real=((.hexaslide.d3_u)+sqrt(.hexaslide.delta3))
var set var=q4 real=((.hexaslide.d4_u)+sqrt(.hexaslide.delta4))
var set var=q5 real=((.hexaslide.d5_u)+sqrt(.hexaslide.delta5))
var set var=q6 real=((.hexaslide.d6_u)+sqrt(.hexaslide.delta6))
!
marker create marker_name=.hexaslide.ground.A1 &
location =(LOC_RELATIVE_TO({(.hexaslide.q1),-($s),0.0},.hexaslide.ground.Origin))
marker create marker_name=.hexaslide.ground.A2 &
location =(LOC_RELATIVE_TO({(.hexaslide.q2),($s),0.0},.hexaslide.ground.Origin))
marker create marker_name=.hexaslide.ground.A3 &
location =(LOC_RELATIVE_TO({-0.5*(.q3)+sqrt(3)*0.5*($s),sqrt(3)*0.5*(.q3)+0.5*($s),0.0},.Origin))
marker create marker_name=.hexaslide.ground.A4 &
location =(LOC_RELATIVE_TO({-0.5*(.q4)-sqrt(3)*0.5*($s),sqrt(3)*0.5*(.q4)-0.5*($s),0.0},.Origin))
marker create marker_name=.hexaslide.ground.A5 &
location =(LOC_RELATIVE_TO({-0.5*(.q5)-sqrt(3)*0.5*($s),-sqrt(3)*0.5*(.q5)+0.5*($s),0.0},.Origin))
marker create marker_name=.hexaslide.ground.A6 &
location =(LOC_RELATIVE_TO({-0.5*(.q6)+sqrt(3)*0.5*($s),-sqrt(3)*0.5*(.q6)-0.5*($s),0.0},.Origin))
!
marker modify marker_name=.hexaslide.ground.A1 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A1,.hexaslide.ground.B1,"z"))
marker modify marker_name=.hexaslide.ground.A2 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A2,.hexaslide.ground.B2,"z"))
marker modify marker_name=.hexaslide.ground.A3 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A3,.hexaslide.ground.B3,"z"))
marker modify marker_name=.hexaslide.ground.A4 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A4,.hexaslide.ground.B4,"z"))
marker modify marker_name=.hexaslide.ground.A5 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A5,.hexaslide.ground.B5,"z"))
marker modify marker_name=.hexaslide.ground.A6 &
orientation = (ORI_ALONG_AXIS(.hexaslide.ground.A6,.hexaslide.ground.B6,"z"))

```

## B.2 Creazione piattaforma

```

part create rigid_body name_and_position part_name=.hexaslide.piattaforma
part modify rigid_body mass_properties part_name=.hexaslide.piattaforma material=.materials.steel
part attributes part_name=.hexaslide.piattaforma color=red name_vis=off
marker create marker=.hexaslide.piattaforma.C &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0},.hexaslide.ground.TCP)) &
orientation=(ORI_RELATIVE_TO({0,180,0},.hexaslide.ground.TCP))
geometry create shape cylinder &
cylinder_name=.hexaslide.piattaforma.cil &
length=0.3 &
radius=($r_p) &
angle = 360.0d &
center_marker=.hexaslide.piattaforma.C
!
part modify rigid mass_properties &
part_name = .hexaslide.piattaforma &

```



```

mass = 61.64 &
center_of_mass_marker = .hexaslide.piattaforma.cm
!
var set var=x_g &
real=(eval(dx(.hexaslide.piattaforma.cm,.hexaslide.ground.TCP,.hexaslide.ground.TCP)))
var set var=y_g &
real=(eval(dy(.hexaslide.piattaforma.cm,.hexaslide.ground.TCP,.hexaslide.ground.TCP)))
var set var=z_g &
real=(eval(dz(.hexaslide.piattaforma.cm,.hexaslide.ground.TCP,.hexaslide.ground.TCP)))
!
marker modify marker_name=.hexaslide.piattaforma.cm &
location=(loc_relative_to({(.hexaslide.x_g),(.hexaslide.y_g),(.hexaslide.z_g)},.TCP)) &
orientation=(ori_relative_to({-90.0,90.0,90.0},.hexaslide.ground.TCP))

```

## B.3 Creazione marker delle viti

```

marker create marker_name=.hexaslide.ground.P1 &
location = (eval(.A1.loc_x-0.6)),(eval(.A1.loc_y)),(eval(.A1.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.P2 &
location = (eval(.A2.loc_x-0.6)),(eval(.A2.loc_y)),(eval(.A2.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.P3 &
location = (eval(.A3.loc_x+0.3)),(eval(.A3.loc_y-0.3*sqrt(3))), (eval(.A3.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.P4 &
location = (eval(.A4.loc_x+0.3)),(eval(.A4.loc_y-0.3*sqrt(3))), (eval(.A4.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.P5 &
location = (eval(.A5.loc_x+0.3)),(eval(.A5.loc_y+0.3*sqrt(3))), (eval(.A5.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.P6 &
location = (eval(.A6.loc_x+0.3)),(eval(.A6.loc_y+0.3*sqrt(3))), (eval(.A6.loc_z)) &
relative_to=.hexaslide.ground.Origin
!
marker create marker_name=.hexaslide.ground.Q1 &
location = (eval(.A1.loc_x+0.6)),(eval(.A1.loc_y)),(eval(.A1.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.Q2 &
location = (eval(.A2.loc_x+0.6)),(eval(.A2.loc_y)),(eval(.A2.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.Q3 &
location = (eval(.A3.loc_x-0.3)),(eval(.A3.loc_y+0.3*sqrt(3))), (eval(.A3.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.Q4 &
location = (eval(.A4.loc_x-0.3)),(eval(.A4.loc_y+0.3*sqrt(3))), (eval(.A4.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.Q5 &
location = (eval(.A5.loc_x-0.3)),(eval(.A5.loc_y-0.3*sqrt(3))), (eval(.A5.loc_z)) &
relative_to=.hexaslide.ground.Origin
marker create marker_name=.hexaslide.ground.Q6 &
location = (eval(.A6.loc_x-0.3)),(eval(.A6.loc_y-0.3*sqrt(3))), (eval(.A6.loc_z)) &
relative_to=.hexaslide.ground.Origin
!
marker modify marker_name=.hexaslide.ground.P1 &
orientation=(ori_along_axis(.hexaslide.ground.P1,.hexaslide.ground.Q1,"x"))
marker modify marker_name=.hexaslide.ground.P2 &
orientation=(ori_along_axis(.hexaslide.ground.P2,.hexaslide.ground.Q2,"x"))
marker modify marker_name=.hexaslide.ground.P3 &
orientation=(ori_along_axis(.hexaslide.ground.P3,.hexaslide.ground.Q3,"x"))
marker modify marker_name=.hexaslide.ground.P4 &
orientation=(ori_along_axis(.hexaslide.ground.P4,.hexaslide.ground.Q4,"x"))
marker modify marker_name=.hexaslide.ground.P5 &
orientation=(ori_along_axis(.hexaslide.ground.P5,.hexaslide.ground.Q5,"x"))
marker modify marker_name=.hexaslide.ground.P6 &
orientation=(ori_along_axis(.hexaslide.ground.P6,.hexaslide.ground.Q6,"x"))
!
marker modify marker_name=.hexaslide.ground.Q1 &
orientation=(ori_along_axis(.hexaslide.ground.P1,.hexaslide.ground.Q1,"x"))
marker modify marker_name=.hexaslide.ground.Q2 &
orientation=(ori_along_axis(.hexaslide.ground.P2,.hexaslide.ground.Q2,"x"))
marker modify marker_name=.hexaslide.ground.Q3 &
orientation=(ori_along_axis(.hexaslide.ground.P3,.hexaslide.ground.Q3,"x"))
marker modify marker_name=.hexaslide.ground.Q4 &
orientation=(ori_along_axis(.hexaslide.ground.P4,.hexaslide.ground.Q4,"x"))
marker modify marker_name=.hexaslide.ground.Q5 &
orientation=(ori_along_axis(.hexaslide.ground.P5,.hexaslide.ground.Q5,"x"))
marker modify marker_name=.hexaslide.ground.Q6 &
orientation=(ori_along_axis(.hexaslide.ground.P6,.hexaslide.ground.Q6,"x"))

```

## B.4 Creazione link flessibili

```

var set var=Youngmodulus real=207E9
var set var=Shearmodulus real=((eval(.hexaslide.Youngmodulus))/2/(1+0.29))
var set var=Areasection real=(pi*(0.015**2-0.013**2))
var set var=I_xx real=((eval(.hexaslide.Areasection))*(0.015**2+0.013**2)/2)
var set var=I_yy real=((eval(.hexaslide.Areasection))*(0.015**2+0.013**2)/4)
var set var=I_zz real=((eval(.hexaslide.Areasection))*(0.015**2+0.013**2)/4)
!
for variable_name=nlink start_value=1 end_value=6 increment_value=1
!
var set var=L_seg &
real=(DM((eval(".hexaslide.ground.A"//RTOI(nlink))), (eval(".hexaslide.ground.B"//RTOI(nlink))))/nsegs)
!
var set var=mar1 string=(eval(".hexaslide.ground.A"//RTOI(nlink)))
var set var=mar2 string=(eval(".hexaslide.ground.B"//RTOI(nlink)))
!
part create rigid_body name_and_position part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1"))
part modify rigid_body mass_properties &
part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1")) material=.materials.steel
part attributes part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1")) color=blue name_vis=off
marker create marker=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.link"//RTOI(nlink)//"_gm1")) &
location=(LOC_ALONG_LINE((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), 0.0)) &
orientation=(ORI_ALONG_AXIS((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), "z"))
geometry create shape revolution &
revolution_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.rev1")) &
reference_marker=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.link"//RTOI(nlink)//"_gm1")) &
analytical=no &
angle = 360.0d &
points_for_profile= &
0.013, 0.0, 0.0, &
0.013, 0.0, (eval(.hexaslide.L_seg)), &
0.015, 0.0, (eval(.hexaslide.L_seg)), &
0.015, 0.0, 0.0, &
0.013, 0.0, 0.0 &
relative_to=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.link"//RTOI(nlink)//"_gm1"))
!
marker create &
marker_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.link"//RTOI(nlink)//"_CG1")) &
location=(LOC_ALONG_LINE((eval(.mar1)), (eval(.mar2)), ((eval(.L_seg))/2))) &
orientation=(ORI_ALONG_AXIS((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), "x"))
!
var set var=mar_cg string=(eval(".hexaslide.link"//RTOI(nlink)//"_elem1.link"//RTOI(nlink)//"_CG1"))
!
for variable_name=count start_value=2 end_value=nsegs increment_value=1
!
part create rigid_body name_and_position &
part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count)))
part modify rigid_body mass_properties &
part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count))) &
material=.materials.steel
part attributes part_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count))) &
color=blue name_vis=off
marker create &
marker=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)//".link"//RTOI(nlink)//"_gm"//RTOI(.count))) &
location=(LOC_ALONG_LINE((eval(.mar1)), (eval(.mar2)), ((eval(.count)-1)*eval(.L_seg)))) &
orientation=(ORI_ALONG_AXIS((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), "z"))
geometry create shape revolution &
revolution_name=(eval(".hexaslide.link"//RTOI(nlink)//"_elem"//RTOI(.count)//".rev"//RTOI(.count))) &
reference_marker=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)//".link"//RTOI(nlink)//"_gm"//RTOI(.count))) &
analytical=no &
angle = 360.0d &
points_for_profile= &
0.013, 0.0, 0.0, &
0.013, 0.0, (eval(.hexaslide.L_seg)), &
0.015, 0.0, (eval(.hexaslide.L_seg)), &
0.015, 0.0, 0.0, &
0.013, 0.0, 0.0 &
relative_to=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)//".link"//RTOI(nlink)//"_gm"//RTOI(.count)))
!
marker create &
marker_name=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)//".link"//RTOI(nlink)//"_CG"//RTOI(.count))) &
location=(LOC_ALONG_LINE((eval(.mar1)), (eval(.mar2)), ((eval(.count)-0.5)*eval(.L_seg)))) &
orientation=(ori_relative_to({0.0,0.0,0.0}, (eval(.hexaslide.mar_cg))))
!
force create element_like beam &
beam_name=(eval(".hexaslide.link"//RTOI(nlink)//"_beam"//RTOI(.hexaslide.count)-1)) &
i_marker_name=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)//".link"//RTOI(nlink)//"_CG"//RTOI(.count))) &
j_marker_name=(eval(".link"//RTOI(nlink)//"_elem"//RTOI(.count)-1)//".link"//RTOI(nlink)//"_CG"//RTOI(.count)-1)) &
length=(eval(.hexaslide.L_seg)) &
area_of_cross_section=(eval(.hexaslide.Areasection)) &
ixx=(eval(.hexaslide.I_xx)) &
iyy=(eval(.hexaslide.I_yy)) &
izz=(eval(.hexaslide.I_zz)) &
y_shear_area_ratio=1.0 &

```

```

z_shear_area_ratio=1.0 &
damping_ratio=1.0E-005 &
youngs_modulus=(eval(.hexaslide.Youngmodulus)) &
shear_modulus=(eval(.hexaslide.Shearmodulus))
!
end
end

```

## B.5 Creazione slider

```

var set var=$_self.LX_slider real=0.1
var set var=$_self.LY_slider real=0.05
var set var=$_self.LZ_slider real=0.05
!
part create rigid_body name_and_position part_name=.hexaslide.slider_1
part modify rigid_body mass_properties part_name=.hexaslide.slider_1 material=.materials.steel
part attributes part_name=.hexaslide.slider_1 color=red name_vis=off
marker create marker=.hexaslide.slider_1.C &
location=(LOC_RELATIVE_TO({(dx(.A1,.Origin,.Origin)-(eval($_self.LX_slider/2))),...
(dy(.A1,.Origin,.Origin)-(eval($_self.LY_slider/2))), -(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.ground.Origin))
geometry create shape block &
block_name=.hexaslide.slider_1.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_1.C
marker modify marker_name=.hexaslide.slider_1.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A1))
!
part create rigid_body name_and_position part_name=.hexaslide.slider_2
part modify rigid_body mass_properties part_name=.hexaslide.slider_2 material=.materials.steel
part attributes part_name=.hexaslide.slider_2 color=red name_vis=off
marker create marker=.hexaslide.slider_2.C &
location=(LOC_RELATIVE_TO({(dx(.A2,.Origin,.Origin)-(eval($_self.LX_slider/2))),...
(dy(.A2,.Origin,.Origin)-(eval($_self.LY_slider/2))), -(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.ground.Origin))
geometry create shape block &
block_name=.hexaslide.slider_2.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_2.C
marker modify marker_name=.hexaslide.slider_2.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A2))
!
part create rigid_body name_and_position part_name=.hexaslide.slider_3
part modify rigid_body mass_properties part_name=.hexaslide.slider_3 material=.materials.steel
part attributes part_name=.hexaslide.slider_3 color=red name_vis=off
marker create marker=.hexaslide.slider_3.C &
location=(LOC_RELATIVE_TO({(dx(.A3,.Origin,.Origin)+0.5*(eval($_self.LX_slider/2))+...
sqrt(3)*0.5*(eval($_self.LY_slider/2))), (dy(.A3,.Origin,.Origin)-...
sqrt(3)*0.5*(eval($_self.LX_slider/2))+ ...
0.5*(eval($_self.LY_slider/2))), -(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({120,0,0},.hexaslide.ground.Origin))
geometry create shape block &
block_name=.hexaslide.slider_3.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_3.C
marker modify marker_name=.hexaslide.slider_3.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A3))
!
part create rigid_body name_and_position part_name=.hexaslide.slider_4
part modify rigid_body mass_properties part_name=.hexaslide.slider_4 material=.materials.steel
part attributes part_name=.hexaslide.slider_4 color=red name_vis=off
marker create marker=.hexaslide.slider_4.C &
location=(LOC_RELATIVE_TO({(dx(.A4,.Origin,.Origin)+0.5*(eval($_self.LX_slider/2))+ ...
sqrt(3)*0.5*(eval($_self.LY_slider/2))), (dy(.A4,.Origin,.Origin)-sqrt(3)*0.5*(eval($_self.LX_slider/2))+ ...
0.5*(eval($_self.LY_slider/2))), ...
-(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({120,0,0},.hexaslide.ground.Origin))
geometry create shape block &
block_name=.hexaslide.slider_4.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_4.C
marker modify marker_name=.hexaslide.slider_4.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A4))
!
part create rigid_body name_and_position part_name=.hexaslide.slider_5
part modify rigid_body mass_properties part_name=.hexaslide.slider_5 material=.materials.steel
part attributes part_name=.hexaslide.slider_5 color=red name_vis=off
marker create marker=.hexaslide.slider_5.C &
location=(LOC_RELATIVE_TO({(dx(.A5,.Origin,.Origin)+0.5*(eval($_self.LX_slider/2)) ...
-sqrt(3)*0.5*(eval($_self.LY_slider/2))), (dy(.A5,.Origin,.Origin)+ ...
sqrt(3)*0.5*(eval($_self.LX_slider/2))+0.5*(eval($_self.LY_slider/2))), ...
-(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({240,0,0},.hexaslide.ground.Origin))

```

```

geometry create shape block &
block_name=.hexaslide.slider_5.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_5.C
marker modify marker_name=.hexaslide.slider_5.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A5))
!
part create rigid_body name_and_position part_name=.hexaslide.slider_6
part modify rigid_body mass_properties part_name=.hexaslide.slider_6 material=.materials.steel
part attributes part_name=.hexaslide.slider_6 color=red name_vis=off
marker create marker=.hexaslide.slider_6.C &
location=(LOC_RELATIVE_TO({(dx(.A6,.Origin,.Origin)+0.5*(eval($_self.LX_slider/2))...
-sqrt(3)*0.5*(eval($_self.LY_slider/2))), ...
(dy(.A6,.Origin,.Origin)+...
sqrt(3)*0.5*(eval($_self.LX_slider/2))+0.5*(eval($_self.LY_slider/2))), ...
-(eval($_self.LZ_slider/2))},.Origin)) &
orientation=(ORI_RELATIVE_TO({240,0,0},.hexaslide.ground.Origin))
geometry create shape block &
block_name=.hexaslide.slider_6.BOX_1 &
diag_corner_coords=(eval($_self.LX_slider)), (eval($_self.LY_slider)), (eval($_self.LZ_slider)) &
corner_marker=.hexaslide.slider_6.C
marker modify marker_name=.hexaslide.slider_6.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.ground.A6))!
!

```

## B.6 Creazione Viti

```

var set var=Youngmodulus real=207E9
var set var=Shearmodulus real=((eval(.hexaslide.Youngmodulus))/2/(1+0.29))
var set var=Areasection real=(pi*(0.015**2))
var set var=I_xx real=((eval(.hexaslide.Areasection))*(0.015**2)/2)
var set var=I_yy real=((eval(.hexaslide.Areasection))*(0.015**2)/4)
var set var=I_zz real=((eval(.hexaslide.Areasection))*(0.015**2)/4)
!
for variable_name=nlink start_value=1 end_value=6 increment_value=1
!
var set var=L_seg_viti &
real=(DM((eval("P"//RTOI(nlink))), (eval("Q"//RTOI(nlink))))/$nsegs_viti)
!
var set var=mar1 string=(eval(".hexaslide.ground.P"//RTOI(nlink)))
var set var=mar2 string=(eval(".hexaslide.ground.Q"//RTOI(nlink)))
!
part create rigid_body name_and_position part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1"))
part modify rigid_body mass_properties part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1")) &
material=.materials.steel
part attributes part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1")) color=dark_gray name_vis=off
marker create marker=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.vite"//RTOI(nlink)//"_gm1")) &
location=(LOC_ALONG_LINE((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), (0,0)) &
orientation=(ORI_RELATIVE_TO({90.0,90.0,0.0}, (eval(.hexaslide.mar1))))
geometry create shape cylinder &
cylinder_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.cyl1")) &
length=(eval(.hexaslide.L_seg_viti)) &
radius=0.015 &
angle = 360.0d &
center_marker=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.vite"//RTOI(nlink)//"_gm1"))
!
var set var=mar_ori string=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.vite"//RTOI(nlink)//"_gm1"))
!
marker modify marker_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.cm")) &
orientation=(ori_relative_to({0.0,0.0,0.0}, (eval(.hexaslide.mar_ori))))
!
marker create &
marker_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.vite"//RTOI(nlink)//"_CG1")) &
location=(LOC_ALONG_LINE((eval(.mar1)), (eval(.mar2)), ((eval(.L_seg_viti))/2))) &
orientation=(ORI_ALONG_AXIS((eval(.hexaslide.mar1)), (eval(.hexaslide.mar2)), "x"))
!
var set var=mar_cg string=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem1.vite"//RTOI(nlink)//"_CG1"))
!
for variable_name=count start_value=2 end_value=$nsegs_viti increment_value=1
!
part create &
rigid_body name_and_position &
part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count)))
part modify rigid_body mass_properties &
part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count))) &
material=.materials.steel
part attributes part_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count))) &
color=dark_gray name_vis=off
marker create &
marker=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".vite"//RTOI(nlink)//"_gm"//RTOI(.count))) &
location=(LOC_ALONG_LINE((eval(.mar1)), (eval(.mar2)), ((eval(.count)-1)*eval(.L_seg_viti))) &
orientation=(ORI_RELATIVE_TO({90.0,90.0,0.0}, (eval(.hexaslide.mar1))))
geometry create shape cylinder &

```

```

cylinder_name=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".cyl"//RTOI(.count))) &
length=(eval(.hexaslide.L_seg_viti)) &
radius=0.015 &
angle = 360.0d &
center_marker=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".vite"//RTOI(nlink)//"_gm"//RTOI(.count)))
!
var set var=mar_ori &
string=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".vite"//RTOI(nlink)//"_gm"//RTOI(.count)))
!
marker modify marker_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_elem"//RTOI(.hexaslide.count)//".cm")) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_ori)}))
!
!
marker create &
marker_name=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".vite"//RTOI(nlink)//"_CG"//RTOI(.count))) &
i_location=(LOC_ALONG_LINE((eval(.mar1)),(eval(.mar2)),((eval(.count)-0.5)*eval(.L_seg_viti)))) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_cg)}))
!
force create element_like beam &
beam_name=(eval(".hexaslide.vite"//RTOI(nlink)//"_beam"//RTOI(.hexaslide.count)-1)) &
i_marker_name=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)//".vite"//RTOI(nlink)//"_CG"//RTOI(.count))) &
j_marker_name=(eval(".vite"//RTOI(nlink)//"_elem"//RTOI(.count)-1)//".vite"//RTOI(nlink)//"_CG"//RTOI(.count)-1)) &
length=(eval(.hexaslide.L_seg_viti)) &
area_of_cross_section=(eval(.hexaslide.Areasection)) &
ixx=(eval(.hexaslide.I_xx)) &
iyy=(eval(.hexaslide.I_yy)) &
izz=(eval(.hexaslide.I_zz)) &
y_shear_area_ratio=1.0 &
z_shear_area_ratio=1.0 &
damping_ratio=1.0E-005 &
youngs_modulus=(eval(.hexaslide.Youngmodulus)) &
shear_modulus=(eval(.hexaslide.Shearmodulus))
!
end
end

```

## B.7 Creazione corpi fittizzi

```

var set var=$_self.L_chiocciola real=0.1
var set var=$_self.R_chiocciola real=0.025
!
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_1
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_1 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_1 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_1.C &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0},{-eval($_self.L_chiocciola)/2}),.hexaslide.slider_1.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_1.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_1.cyl_1 &
length=(eval($_self.L_chiocciola)) &
radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_1.C
marker modify marker_name=.hexaslide.chiocciola_1.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_1.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P1))
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_2
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_2 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_2 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_2.C &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0},{-eval($_self.L_chiocciola)/2}),.hexaslide.slider_2.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_2.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_2.cyl_1 &
length=(eval($_self.L_chiocciola)) &
radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_2.C
marker modify marker_name=.hexaslide.chiocciola_2.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_2.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P2))
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_3
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_3 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_3 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_3.C &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0},{-eval($_self.L_chiocciola)/2}),.hexaslide.slider_3.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_3.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_3.cyl_1 &
length=(eval($_self.L_chiocciola)) &

```

```

radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_3.C
marker modify marker_name=.hexaslide.chiocciola_3.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_3.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P3))
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_4
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_4 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_4 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_4.C &
location=(LOC_RELATIVE_TO({0.0,0.0,(-eval($_self.L_chiocciola))/2}),.hexaslide.slider_4.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_4.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_4.cyl_1 &
length=(eval($_self.L_chiocciola)) &
radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_4.C
marker modify marker_name=.hexaslide.chiocciola_4.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_4.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P4))
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_5
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_5 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_5 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_5.C &
location=(LOC_RELATIVE_TO({0.0,0.0,(-eval($_self.L_chiocciola))/2}),.hexaslide.slider_5.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_5.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_5.cyl_1 &
length=(eval($_self.L_chiocciola)) &
radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_5.C
marker modify marker_name=.hexaslide.chiocciola_5.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_5.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P5))
!
part create rigid_body name_and_position part_name=.hexaslide.chiocciola_6
part modify rigid_body mass_properties part_name=.hexaslide.chiocciola_6 material=.materials.steel
part attributes part_name=.hexaslide.chiocciola_6 color=gray name_vis=off
marker create marker=.hexaslide.chiocciola_6.C &
location=(LOC_RELATIVE_TO({0.0,0.0,(-eval($_self.L_chiocciola))/2}),.hexaslide.slider_6.cm)) &
orientation=(ORI_RELATIVE_TO({0,0,0},.hexaslide.slider_6.cm))
geometry create shape cylinder &
cylinder_name=.hexaslide.chiocciola_6.cyl_1 &
length=(eval($_self.L_chiocciola)) &
radius=(eval($_self.R_chiocciola)) &
angle = 360.0d &
center_marker=.hexaslide.chiocciola_6.C
marker modify marker_name=.hexaslide.chiocciola_6.cm &
location=(loc_relative_to({0.0,0.0,0.0},.hexaslide.slider_6.cm)) &
orientation=(ori_relative_to({90.0,90.0,0.0},.hexaslide.ground.P6))

```

## B.8 Vincoli

```

var set var=q_01 real=(eval(.hexaslide.q1))
var set var=q_02 real=(eval(.hexaslide.q2))
var set var=q_03 real=(eval(.hexaslide.q3))
var set var=q_04 real=(eval(.hexaslide.q4))
var set var=q_05 real=(eval(.hexaslide.q5))
var set var=q_06 real=(eval(.hexaslide.q6))
!
var set var=n1 integer=(floor(((.hexaslide.q1)-(.hexaslide.q_01)+0.6)/1.2*($nsegs_viti))+1)
var set var=n2 integer=(floor(((.hexaslide.q2)-(.hexaslide.q_02)+0.6)/1.2*($nsegs_viti))+1)
var set var=n3 integer=(floor(((.hexaslide.q3)-(.hexaslide.q_03)+0.6)/1.2*($nsegs_viti))+1)
var set var=n4 integer=(floor(((.hexaslide.q4)-(.hexaslide.q_04)+0.6)/1.2*($nsegs_viti))+1)
var set var=n5 integer=(floor(((.hexaslide.q5)-(.hexaslide.q_05)+0.6)/1.2*($nsegs_viti))+1)
var set var=n6 integer=(floor(((.hexaslide.q6)-(.hexaslide.q_06)+0.6)/1.2*($nsegs_viti))+1)
!
for variable_name=nlink start_value=1 increment_value=1 end_value=6
!
!!!!!!!!!!!!!!
!UNIVERSALI
!!!!!!!!!!!!!!
!
marker create marker_name=(eval(".hexaslide.link" // RTOI(nlink) // "_elem1.univ1_link_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.A" // RTOI(nlink)))) &
orientation=(ORI_RELATIVE_TO({0.0, -90.0, -90.0}, (eval(".hexaslide.ground.A" // RTOI(nlink))))))
!
marker create marker_name=(eval(".hexaslide.slider_"// RTOI(nlink) // "_univ2_link_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0},(eval(".link" // RTOI(nlink) // "_elem1.univ1_link_" // RTOI(nlink)))) &

```

```

orientation=(ORI_RELATIVE_TO({0.0, -90.0, 0.0}, (eval(".link" // RTOI(nlink) // "_elem1.univ1_link_" // RTOI(nlink))))
!
constraint create joint Universal joint_name=(eval(".hexaslide.Universale_" // RTOI(nlink))) &
i_marker_name=(eval(".hexaslide.link" // RTOI(nlink) // "_elem1.univ1_link_" // RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.slider_" // RTOI(nlink) // ".univ2_link_" // RTOI(nlink)))
!
!!!!!!!!!!!!!!
!SFERICI
!!!!!!!!!!!!!!
!
marker create &
marker_name=(eval(".link" // RTOI(nlink) // "_elem" // $nsegs // ".sfer1_link_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.B" // RTOI(nlink)))) &
orientation=(ORI_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.B" // RTOI(nlink))))
!
marker create marker_name=(eval(".hexaslide.piattaforma.sfer2_link_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.B" // RTOI(nlink)))) &
orientation=(ORI_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.B" // RTOI(nlink))))
!
constraint create joint Spherical joint_name=(eval(".hexaslide.Sferico_" // RTOI(nlink))) &
i_marker_name=(eval(".link" // RTOI(nlink) // "_elem" // $nsegs // ".sfer1_link_" // RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.piattaforma.sfer2_link_" // RTOI(nlink)))
!
!!!!!!!!!!!!!!
!INCASTRI
!!!!!!!!!!!!!!
!
marker create &
marker_name=(eval(".hexaslide.vite" // RTOI(nlink) // "_elem1.incastro1_vite_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.P" // RTOI(nlink)))) &
orientation=(ORI_RELATIVE_TO({0.0, -90.0, -90.0}, (eval(".hexaslide.ground.P" // RTOI(nlink))))
!
marker create marker_name=(eval(".hexaslide.ground.incastro2_vite_" // RTOI(nlink))) &
location=(LOC_RELATIVE_TO({0.0,0.0,0.0}, (eval(".hexaslide.ground.P" // RTOI(nlink)))) &
orientation=(ORI_RELATIVE_TO({0.0, -90.0, -90.0}, (eval(".hexaslide.ground.P" // RTOI(nlink))))
!
constraint create joint Fixed joint_name=(eval(".hexaslide.Incastro_" // RTOI(nlink))) &
i_marker_name=(eval(".hexaslide.vite" // RTOI(nlink) // "_elem1.incastro1_vite_" // RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.ground.incastro2_vite_" // RTOI(nlink)))
!
!!!!!!!!!!!!!!
!CILINDRICI
!!!!!!!!!!!!!!
!
marker create &
marker_name=(eval(".vite" // RTOI(nlink) // "_elem" // $nsegs_viti // ".cil1_vite_" // RTOI(nlink))) &
location=0.0,0.0,0.0 &
orientation=90.0, 90.0, 0.0 &
relative_to=(eval(".hexaslide.ground.Q" // RTOI(nlink)))
!
marker create marker_name=(eval(".hexaslide.ground.cil2_vite_" // RTOI(nlink))) &
location=0.0,0.0,0.0 &
orientation=90.0, 90.0, 0.0 &
relative_to=(eval(".hexaslide.ground.Q" // RTOI(nlink)))
!
constraint create joint Cylindrical joint_name=(eval(".hexaslide.Cilindrico_" // RTOI(nlink))) &
i_marker_name=(eval(".vite" // RTOI(nlink) // "_elem" // $nsegs_viti // ".cil1_vite_" // RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.ground.cil2_vite_" // RTOI(nlink)))
!
!!!!!!!!!!!!!!
!PRISMATICI SLIDER
!!!!!!!!!!!!!!
!
var set var=mar string=(eval(".hexaslide.slider_"//RTOI(nlink)//".cm"))
!
marker create &
marker_name=(eval(".hexaslide.slider_" // RTOI(nlink) // ".prism1_slider_" // RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},(eval(.hexaslide.mar)))) &
orientation=(ori_relative_to({0.0,0.0,0.0},(eval(.hexaslide.mar))))
!
marker create marker_name=(eval(".hexaslide.ground.prism2_slider_" // RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},(eval(.hexaslide.mar)))) &
orientation=(ori_relative_to({0.0,0.0,0.0},(eval(.hexaslide.mar))))
!
constraint create &
joint Translational joint_name=(eval(".hexaslide.Prismatico_slider_" // RTOI(nlink))) &
i_marker_name=(eval(".hexaslide.slider_" // RTOI(nlink) // ".prism1_slider_" // RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.ground.prism2_slider_" // RTOI(nlink)))
!
!!!!!!!!!!!!!!
!CERNIERE
!!!!!!!!!!!!!!
!
var set var=mar_rev string=(eval(".hexaslide.chiocciola_"//RTOI(nlink)//".cm"))
!
marker create &

```

```

marker_name=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".revol1_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)})) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
marker create marker_name=(eval(".hexaslide.ground.revol2_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)})) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
constraint create &
joint Revolute joint_name=(eval(".hexaslide.cerniera_chiocciola_">//RTOI(nlink))) &
i_marker_name=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".revol1_chiocciola_">//RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.ground.revol2_chiocciola_">//RTOI(nlink)))
!
!!!!!!!!!!!!!!!!!!!!!!
!SCREW
!!!!!!!!!!!!!!!!!!!!!!
!
marker create &
marker=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".screw1_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},{eval(.mar_rev)})) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
marker create &
marker=(eval(".hexaslide.slider_">//RTOI(nlink)//".screw2_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)})) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
constraint create joint Screw joint_name=(eval(".hexaslide.screw_chiocciola_">//RTOI(nlink))) &
i_marker_name=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".screw1_chiocciola_">//RTOI(nlink))) &
j_marker_name=(eval(".hexaslide.slider_">//RTOI(nlink)//".screw2_chiocciola_">//RTOI(nlink))) &
pitch=($passo)
!
!!!!!!!!!!!!!!!!!!!!!!
!GIUNTI USER-DEFINED
!!!!!!!!!!!!!!!!!!!!!!
!
marker create &
marker_name=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".gcn_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)})) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
for variable_name=kk start_value=1 increment_value=1 end_value=($nsegs_viti)
!
var set var=mar_kk string=(eval(".hexaslide.vite_">//RTOI(nlink)//".elem_">//RTOI(kk)//".cm"))
!
var set var=mar1 string=(eval(".hexaslide.chiocciola_">//RTOI(nlink)//".gcn_chiocciola_">//RTOI(nlink)))
!
marker create &
marker_name=(eval(".vite_">//RTOI(nlink)//".elem_">//RTOI(kk)//".gcn_chiocciola_">//RTOI(nlink))) &
location=(loc_relative_to({0.0,0.0,((step(dz((eval(.mar1)),(eval(.mar_kk)),(eval(.mar_kk))),...
(-(L_seg_viti)/2),0,(-(L_seg_viti)/2+0.001),1)+...
step(dz((eval(.mar1)),(eval(.mar_kk)),(eval(.mar_kk))),...
((L_seg_viti)/2+0.001),0,((L_seg_viti)/2,-1))*(dz((eval(.mar1)),(eval(.mar_kk)),(eval(.mar_kk))))),
(eval(.mar_kk)))))) &
orientation=(ori_relative_to({0.0,0.0,0.0},{eval(.hexaslide.mar_rev)}))
!
var set var=mar2 string=(eval(".vite_">//RTOI(nlink)//".elem_">//RTOI(kk)//".gcn_chiocciola_">//RTOI(nlink)))
!
end
end
!
data_element create variable variable_name=a_1 &
function=(eval("az(.chiocciola_1.gcn_chiocciola_1,.vite1_elem">//(.hexaslide.n1)//".gcn_chiocciola_1)"))
!
constraint create general_constraint &
general_constraint_name=Gcn_rot_1 &
i_marker_name=.hexaslide.chiocciola_1.gcn_chiocciola_1 &
function= "varval(.hexaslide.a_1)"
!
data_element create variable variable_name=a_2 &
function=(eval("az(.chiocciola_2.gcn_chiocciola_2,.vite2_elem">//(.hexaslide.n2)//".gcn_chiocciola_2)"))
!
constraint create general_constraint &
general_constraint_name=Gcn_rot_2 &
i_marker_name=.hexaslide.chiocciola_2.gcn_chiocciola_2 &
function= "varval(.hexaslide.a_2)"
!
data_element create variable variable_name=a_3 &
function=(eval("az(.chiocciola_3.gcn_chiocciola_3,.vite3_elem">//(.hexaslide.n3)//".gcn_chiocciola_3)"))
!
constraint create general_constraint &
general_constraint_name=Gcn_rot_3 &
i_marker_name=.hexaslide.chiocciola_3.gcn_chiocciola_3 &
function= "varval(.hexaslide.a_3)"
!
data_element create variable variable_name=a_4 &
function=(eval("az(.chiocciola_4.gcn_chiocciola_4,.vite4_elem">//(.hexaslide.n4)//".gcn_chiocciola_4)"))

```



```

!
constraint create general_constraint &
general_constraint_name=Gcn_rot_4 &
i_marker_name=.hexaslide.chiocciola_4.gcn_chiociola_4 &
function= "varval(.hexaslide.a_4)"
!
data_element create variable variable_name=a_5 &
function=(eval("az(.chiocciola_5.gcn_chiociola_5,.vite5_elem"//(.hexaslide.n5)//".gcn_chiociola_5)"))
!
constraint create general_constraint &
general_constraint_name=Gcn_rot_5 &
i_marker_name=.hexaslide.chiocciola_5.gcn_chiociola_5 &
function= "varval(.hexaslide.a_5)"
!
data_element create variable variable_name=a_6 &
function=(eval("az(.chiocciola_6.gcn_chiociola_6,.vite6_elem"//(.hexaslide.n6)//".gcn_chiociola_6)"))
!
constraint create general_constraint &
general_constraint_name=Gcn_rot_6 &
i_marker_name=.hexaslide.chiocciola_6.gcn_chiociola_6 &
function= "varval(.hexaslide.a_6)"
!

```

## B.9 Analisi

```

defaults units orientation_type=body123
!
!
var set var=.hexaslide.datetime string=(eval(STR_DATE("%d-%m-%Y_%H.%M")))
!
if cond=((alpha) == (-5.0))
!
var set var=creadir string=(mkdir("hexaslide_viti_piano_"//($phi)//"__"//(.hexaslide.datetime)))
!
var set var=dirname string=(eval("hexaslide_viti_piano_"//($phi)//"__"//(.hexaslide.datetime)))
!
else
!
var set var=dirname string=(eval($cartella))
!
end
!
if cond=((phi) < 90)
!
var set var=csi real=(0.07/cos(phi))
!
else
!
var set var=csi real=(0.1/sin(phi))
!
end
!
var set var=inc_csi real=((.hexaslide.csi)/5)
!
!
!
for variable_name=beta start_value=-15 end_value=15 increment_value=($inc_b)
for variable_name=gamma start_value=-5 end_value=5 increment_value=($inc_g)
!
var set var=titolo &
string=(eval((.hexaslide.dirname)//"/autovalori_a_"//($alpha)//"_b_"//(.hexaslide.beta)//"_g_"//(.hexaslide.gamma)//".txt"))
!
file text open file_name=(.hexaslide.titolo) open_mode=overwrite
file text write file_name=(.hexaslide.titolo) format_for_output="%s" values_for_output="" newline=no
file text close file_name=(.hexaslide.titolo)
!
!
for variable_name=posz start_value=0.85 end_value=0.55 increment_value=(-($inc_z))
for variable_name=poscsi start_value=(-(.csi)) end_value=(.csi) increment_value=(.inc_csi)
!
marker modify marker_name=.hexaslide.ground.TCP &
location=(eval(-(.poscsi)*sin(phi)), (eval((.poscsi)*cos(phi))), (eval(.posz)) &
orientation= (alpha),(eval(.hexaslide.beta)),(eval(.hexaslide.gamma)) relative_to=.hexaslide.ground.Origin
!
data_element modify variable variable_name=a_1 &
function=(eval("az(.chiocciola_1.gcn_chiociola_1,.vite1_elem"//(.n1)//".gcn_chiociola_1)"))
!
data_element modify variable variable_name=a_2 &
function=(eval("az(.chiocciola_2.gcn_chiociola_2,.vite2_elem"//(.n2)//".gcn_chiociola_2)"))
!
data_element modify variable variable_name=a_3 &

```

```
function=(eval("az(.chiocciola_3.gcn_chiocciola_3,.vite3_elem"//(.n3)//".gcn_chiocciola_3)"))
!
data_element modify variable variable_name=a_4 &
function=(eval("az(.chiocciola_4.gcn_chiocciola_4,.vite4_elem"//(.n4)//".gcn_chiocciola_4)"))
!
data_element modify variable variable_name=a_5 &
function=(eval("az(.chiocciola_5.gcn_chiocciola_5,.vite5_elem"//(.n5)//".gcn_chiocciola_5)"))
!
data_element modify variable variable_name=a_6 &
function=(eval("az(.chiocciola_6.gcn_chiocciola_6,.vite6_elem"//(.n6)//".gcn_chiocciola_6)"))
!
!
simulation single set update = none
simulation single equilibrium
!simulation single trans &
!type=auto_select &
!initial_static= no &
!end_time= 1.0 &
!number_of_steps= 10
!
simulation single eigensolution
!
var set var = $_self.name_eigen object = (eval(DB_CHILDREN(.hexaslide.last_run,"eigen")))
!
!lis var var = $_self.name_eigen
!
if cond=(.hexaslide.poscsi == (.hexaslide.csi) && .hexaslide.posz != (0.55))
file text open file_name=(.hexaslide.titolo) open_mode=append
file text write file_name=(.hexaslide.titolo) &
format_for_output="%10.4f_" values=(eval($_self.name_eigen.object_value.values[2])) newline=yes
file text close file_name=(.hexaslide.titolo)
!
!
elseif cond=(.hexaslide.poscsi != (.hexaslide.csi) || .hexaslide.posz == (0.55))
!
file text open file_name=(.hexaslide.titolo) open_mode=append
file text write file_name=(.hexaslide.titolo) &
format_for_output="%10.4f_" values=(eval($_self.name_eigen.object_value.values[2])) newline=no
file text close file_name=(.hexaslide.titolo)
!
end
!
simulation single reset
variable delete variable = (eval(DB_CHILDREN($_self,"variable")))
end
!
!variable delete variable_name=(.hexaslide.poscsi)
!
end
end
end
```

APPENDICE C

**BOSCH CKK25-200**

Linearmoduli Compact CKK

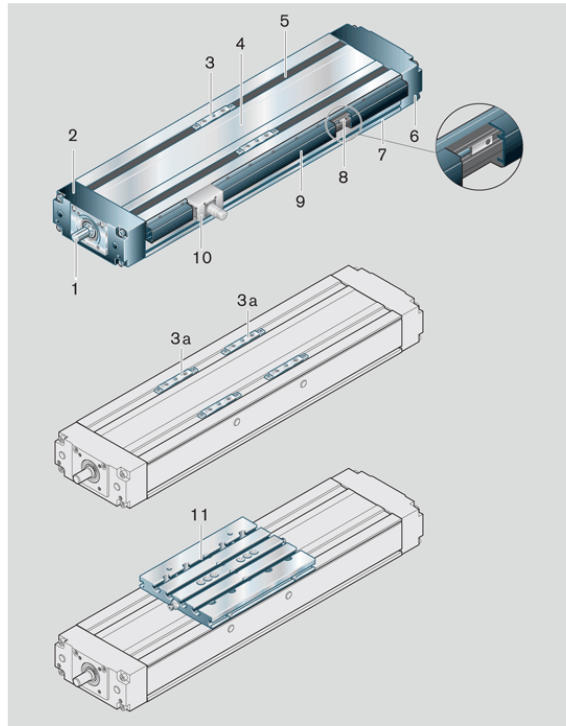
## Struttura

### Struttura CKK

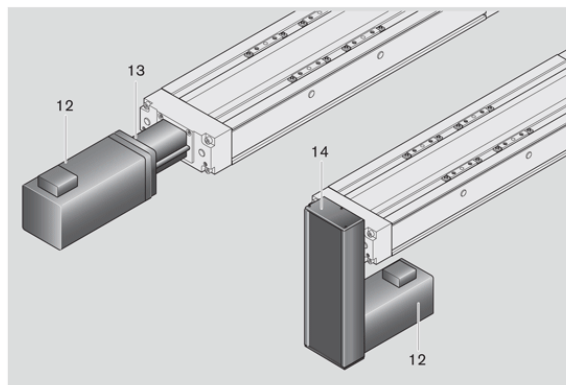
- 1 Vite a sfere con chiocciola singola cilindrica, senza gioco
- 2 Testata con cuscinetto di vincolo radiale
- 3 Tavola con due pattini integrati
- 3a Due tavole, ognuna con due pattini integrati
- 4 Copertura in alluminio
- 5 Nastro di protezione in poliuretano (scorrevole)
- 6 Testata con cuscinetto di vincolo assiale
- 7 Profilato di base

#### Accessori:

- 8 Sensore magnetico
- 9 Canalina di fissaggio
- 10 Presa/spina
- 11 Piastra d'accoppiamento



- 12 Motore
- 13 Flangia e giunto
- 14 Trasmissione a cinghia e puleggia



Linearmoduli Compact CKK

## Dati tecnici

### Dati tecnici generali

Fattori di carico e momenti dinamici

Grandezza	Numero tavole	Vite a sfere	Fattore di carico dinamico C (N)			Momenti dinamici		Momento d'inerzia della sezione		Lunghezza massima	Massa spostata
			Guida	Vite a sfere	Cuscinetto di vincolo assiale	$M_x$ (Nm)	$M_y$ (Nm)	$I_x$ (cm <sup>4</sup> )	$I_y$ (cm <sup>4</sup> )		
CKK 12-90	1	$d_0 \times P$	4620	2240	6900	125	16	14,32	124,4	750	0,36
		12 x 2									
		12 x 5									
	2 ( $l_m = 65$ mm)	12 x 2	7500	2240	6900	200	240	14,32	124,4	750	0,59
		12 x 5		3800							
		12 x 10		2500							
CKK 15-110	1	16 x 5	15600	12300	13400	515	80	37,74	318,7	1500	0,52
		16 x 10		9600							
		16 x 16		6300							
	2 ( $l_m = 85$ mm)	16 x 5	25340	12300	13400	835	1075	37,74	318,7	1500	0,86
		16 x 10		9600							
		16 x 16		6300							
CKK 20-145	1	20 x 5	37600	14300	17000	1650	255	114,10	986,4	1800	1,21
		20 x 20		9100							
		20 x 40		14000							
		25 x 10		15700							
	2 ( $l_m = 100$ mm)	20 x 5	61080	14300	17000	2685	3050	114,10	986,4	1800	2,06
		20 x 20		9100							
CKK 25-200	1	32 x 5	55000	21500	26000	3570	540	612,00	3008,0	2200 (con SPU 5500)*	3,18
		32 x 10		31700							
		32 x 20		19700							
		32 x 32		19500							
	2 ( $l_m = 175$ mm)	32 x 5	89340	21500	26000	5800	7810	612,00	3008,0	2200 (con SPU 5500)*	5,20
		32 x 10		31700							
		32 x 20		19700							
		32 x 32		19500							

 $l_m$  = interasse tra le tavole

\* Per lunghezze da 2200 a 5500 consultare il capitolo "Supporto vite per CKK 25-200"

### Carichi massimi ammissibili

Grandezza	Numero tavole	Forze massime ammissibili (N)			Momenti massimi ammissibili (Nm)		
		$F_{z1max}$	$F_{z2max}$	$F_{ymax}$	$M_{Lmax}$	$M_{Lmax}$	
CKK 12-90	1	4 620	4 620	2 490	125	16	
	2	7 500	7 500	4 050	200	240	
CKK 15-110	1	12 000	6 000	3 480	198	31	
	2	19 490	9 740	5 650	322	414	
CKK 20-145	1	29 000	14 500	8 410	638	100	
	2	47 110	23 550	13 660	1 030	1 180	
CKK 25-200	1	42 200	21 100	12 230	1 372	209	
	2	68 550	34 270	19 880	2 228	2 999	

**Carico appropriato**  
(valore raccomandato in base all'esperienza)

Per ottenere una buona durata nominale a fatica è bene, in linea generale, che i carichi effettivi non superino il 20% dei fattori di carico dinamico (C,  $M_x$ ,  $M_y$ ).

Per un corretto dimensionamento non si devono superare:

- i carichi massimi ammissibili,
- la coppia di azionamento ammissibile,
- la velocità ammissibile.

**Modulo di elasticità E**

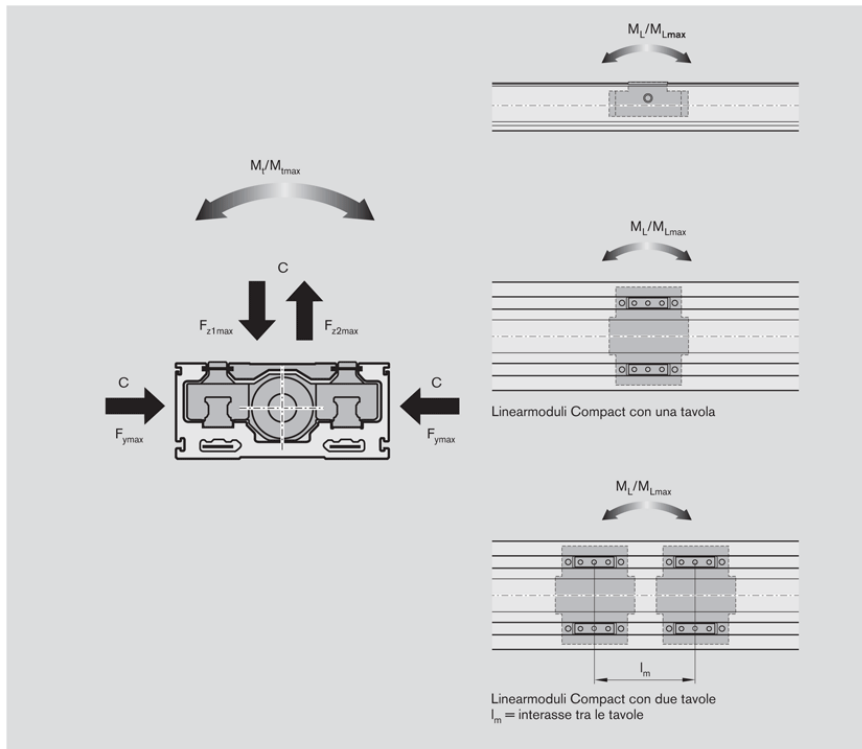
$E = 70\,000\text{ N/mm}^2$

**Peso**

Il peso è calcolato senza tener conto del motore e dell'interruttore.

Formula peso:  
 peso unitario (kg/mm) · lunghezza L (mm)  
 + peso di tutte le parti non dipendenti  
 (tavola, testate, ecc.) (kg)

Grandezza	Vite a sfere	Numero tavole	Peso (kg)
CKK 12-90	con	1	$0,0055 \cdot L + 0,9$
		2	$0,0055 \cdot L + 1,2$
CKK 15-110	con	1	$0,0092 \cdot L + 1,6$
		2	$0,0092 \cdot L + 2,0$
CKK 20-145	con	1	$0,0178 \cdot L + 3,0$
		2	$0,0178 \cdot L + 3,9$
CKK 25-200	con	1	$0,0299 \cdot L + 6,7$
		2	$0,0299 \cdot L + 8,7$



**Avvertenza in merito a fattori di carico dinamico e momenti dinamici**

I fattori di carico dinamico e i momenti dinamici sono calcolati sulla base di una percorrenza di 100 000 m. Tuttavia, di frequente si prendono come base soltanto 50 000 m. Pertanto, per il confronto vale quanto segue: moltiplicare per il coefficiente 1,26 i valori  $C$ ,  $M_l$  e  $M_l$  indicati nella tabella.

Linearmoduli Compact CKK

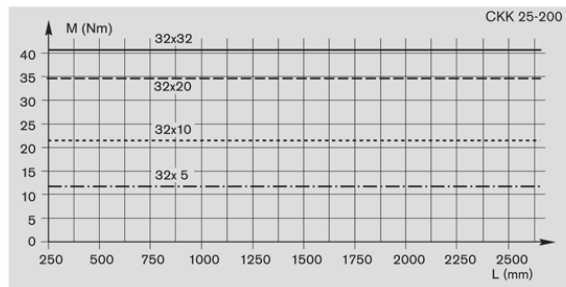
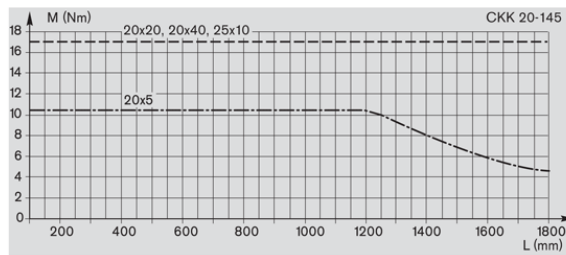
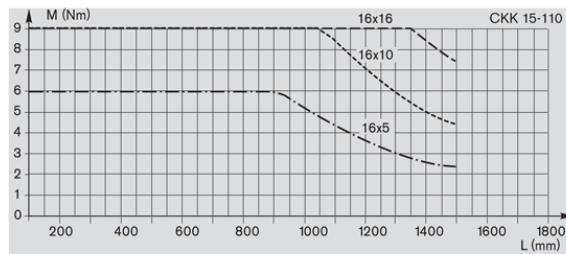
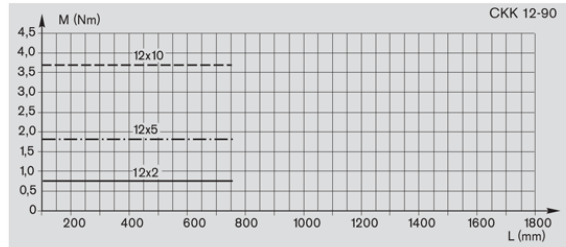
## Dati tecnici

### Coppia di azionamento ammissibile $M_{amm}$

I valori indicati di  $M_{amm}$  sono validi nelle seguenti condizioni:

- movimento orizzontale
- codoli della vite a sfere senza cava per chiave
- assenza di carico radiale sui codoli della vite a sfere

Rispettare il momento nominale del giunto utilizzato!



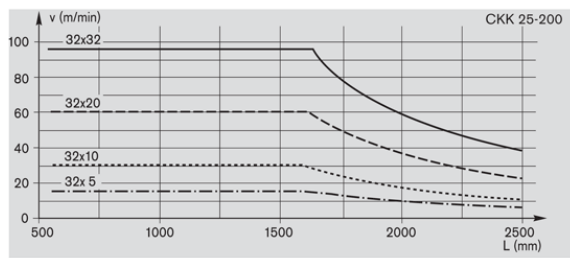
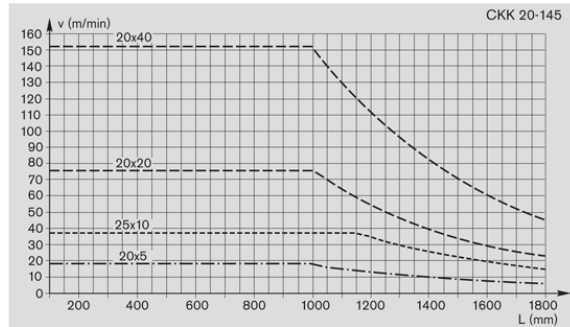
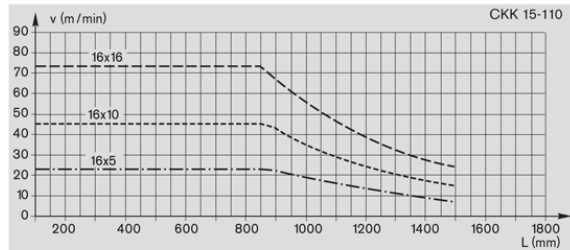
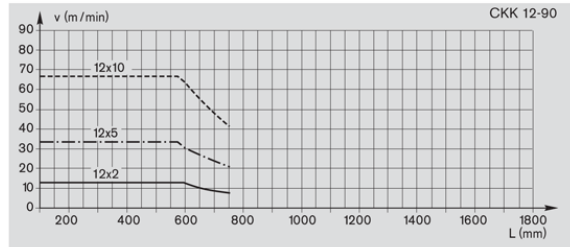
### Codolo della vite a sfere con cava per chiave

A causa dell'effetto di intaglio e della riduzione del diametro effettivo, tener conto dei seguenti valori massimi ammissibili per la coppia di azionamento!

Grandezza	$M_{amm}$ (Nm)
CKK 12-90	-
CKK 15-110	5,0
CKK 20-145	11,5
CKK 25-200	18,0

Per dati tecnici per lunghezze da 2200 a 5500 consultare il capitolo "Supporto vite per Linearmodulo Compact CKK 25-200".

**Velocità ammissibile v**  
 Osservare la velocità di rotazione  
 del motore!

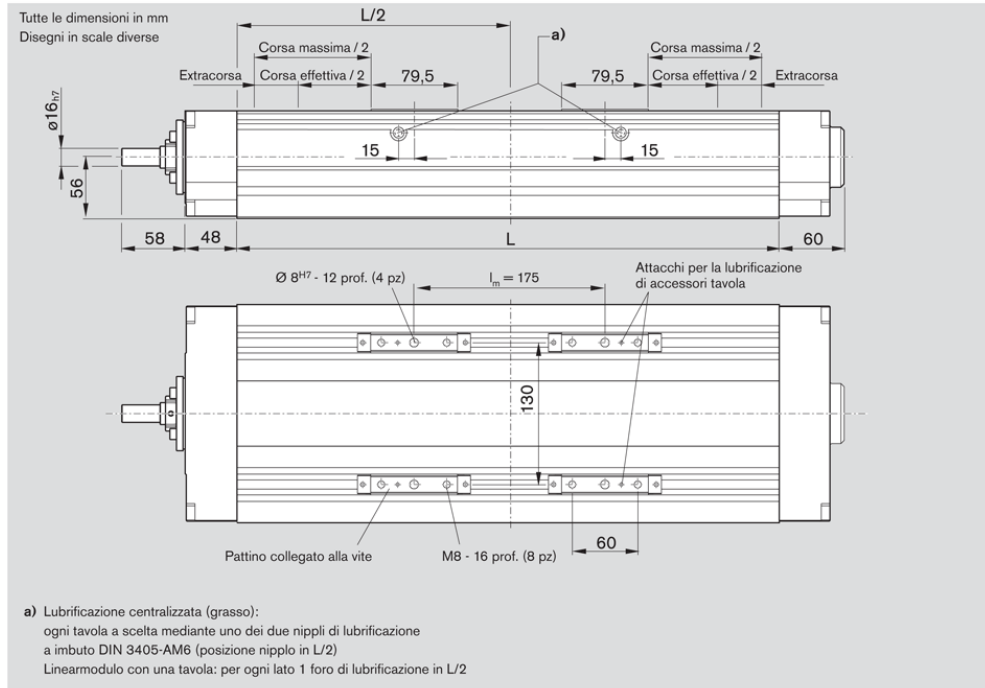


Per dati tecnici per lunghezze da  
 2200 a 5500 consultare il capitolo  
 "Supporto vite per Linearmodulo  
 Compact CKK 25-200".

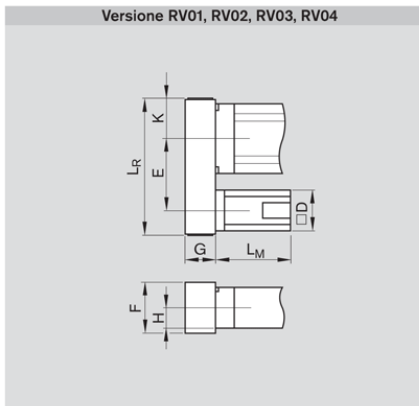
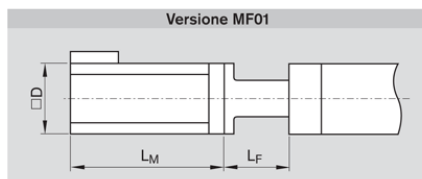
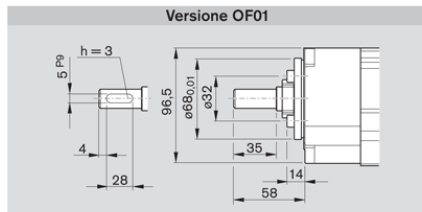


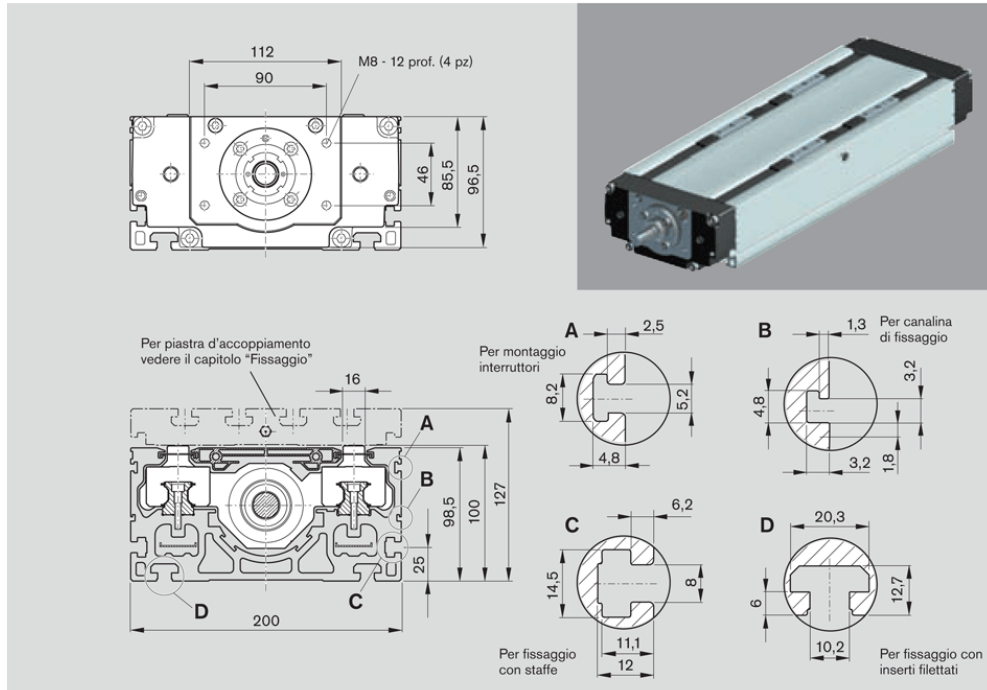
Linearmoduli Compact CKK

## CKK 25-200 Dimensioni



Per ulteriori informazioni e dimensioni vedere "Motori"





Versione	Motore	Dimensioni (mm)											
		D	i=1	E i=2	F	G	H	K	L <sub>F</sub>	senza freno	L <sub>M</sub> con freno	i=1	L <sub>R</sub> i=2
RV01/RV02	MSK 060C	116	267,5	265	116	66	56	59	-	-	-	403	403
RV03/RV04	MSK 060C	116	-	-	-	-	-	-	125	226	259	-	-
MF01	MSK 076C	140	-	-	-	-	-	-	133	-	292,5	-	-

APPENDICE D

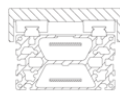
**ROLLON ROBOT SP160**

## 6 Vista generale e caratteristiche tecniche principali

### ROBOT 100

- Dimensioni del profilo: 55 x 100 mm
- Con cinghia di protezione

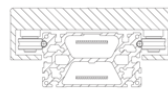
Con guide a ricircolo di sfere - *With ball bearing guides*



ROBOT 100 SP

- *Body dimensions: 55 x 100 mm*
- *With sealing strip*

Con guide a rotelle ad arco gotico - *With lancet arch bearing*

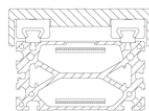


ROBOT 100 CE

### ROBOT 130

- Dimensioni del profilo: 70 x 130 mm
- Con cinghia di protezione

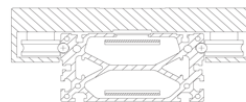
Con guide a ricircolo di sfere - *With ball bearing guides*



ROBOT 130 SP

- *Body dimensions: 70 x 130 mm*
- *With sealing strip*

Con guide a rotelle ad arco gotico - *With lancet arch bearing*

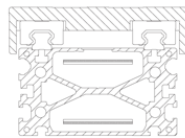


ROBOT 130 CE

### ROBOT 160

- Dimensioni del profilo: 90 x 160 mm
- Con cinghia di protezione

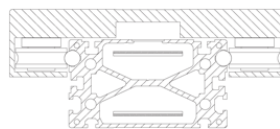
Con guide a ricircolo di sfere - *With ball bearing guides*



ROBOT 160 SP

- *Body dimensions: 90 x 160 mm*
- *With sealing strip*

Con guide a rotelle ad arco gotico - *With lancet arch bearing*

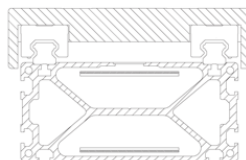


ROBOT 160 CE

### ROBOT 220

- Dimensioni del profilo: 100 x 220 mm
- Con cinghia di protezione

Con guide a ricircolo di sfere - *With ball bearing guides*



ROBOT 220 SP

- *Body dimensions: 100 x 220 mm*
- *With sealing strip*

© Nota: tutti i disegni riprodotti nel presente stampato sono disponibili sul sito [www.rollon.com](http://www.rollon.com) in formato CAD

## General view and main technical characteristics 7

Le sezioni riportate nella pagina precedente illustrano le caratteristiche principali e consentono una prima valutazione in funzione delle esigenze applicative. Per una definitiva valutazione consultare i dati e le dimensioni riportati nelle pagine specificate in fondo a ciascuna tabella.

The sectioned drawings on the previous page showing the main characteristics enable an initial selection to be made. For a final selection, refer to the data and dimensions in the pages listed at the end of each table.

Tipo	Carico radiale ( $F_z$ ) e laterale ( $F_y$ ) max. consigliato* Max permissible radial ( $F_z$ ) and side ( $F_y$ ) load**		Carico assiale ( $F_x$ ) max. consigliato Max. permissible axial load ( $F_x$ ) [N]	Velocità max. Max speed [m/s]	Accelerazione max. max. acceleration [m/s <sup>2</sup> ]	Ripetibilità max. di posizionamento Max positioning repeatability [mm]	Vedi pagina See pag.
	statico [N]	dinamico [N]					
ROBOT 100 SP	12500	2150	780	4,0	50	0,05	8
ROBOT 100 SP - 2C	12500	2150	390	4,0	50	0,05	9
ROBOT 100 CE	620	520	780	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	10
ROBOT 100 CE - 2C	620	520	390	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	11
ROBOT 130 SP	27000	3990	1870	5,0	50	0,05	12
ROBOT 130 SP - 2C	27000	3990	940	5,0	50	0,05	13
ROBOT 130 CE	1190	1110	1870	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	14
ROBOT 130 CE - 2C	1190	1110	940	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	15
ROBOT 160 SP	48800	9600	3090	5,0	50	0,05	16
ROBOT 160 SP - 2C	48800	9600	1410	5,0	50	0,05	17
ROBOT 160 CE	1900	1950	3090	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	18
ROBOT 160 CE - 2C	1900	1950	1410	1,5 <sup>2</sup>	1,5 <sup>2</sup>	0,05	19
ROBOT 220 SP	91000	15500	5510	5,0	50	0,05	20
ROBOT 220 SP - 2C	91000	15500	2210	5,0	50	0,05	21

\*1) Valori con i quali si ottengono una ragionevole durata ed una sufficiente sicurezza statica. Questi valori **non** corrispondono alle teoriche capacità di carico massime ammissibili del sistema di guida applicato.

\*2) Valore consigliato da Rollon

\*1) Reasonable operating life and system rigidity can be obtained from the values given. These values **do not** relate to the theoretical maximum permitted load capacity of the linear motion system.

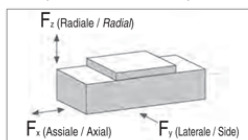
\*2) Permissible Rollon value

La capacità di carico delle unità lineari **Rollon** serie ROBOT dipende dal sistema di traslazione usato e può variare in funzione delle direzioni: radiale e laterale.

Il carico massimo per la direzione assiale dipende dal tipo di cinghia. I valori massimi ammissibili del carico radiale ( $F_z$ ) e laterale ( $F_y$ ) sopra riportati corrispondono al 20% della capacità statica e al 12% della capacità dinamica delle prestazioni relative alle guide a ricircolo di sfere.

Con questi valori, secondo la nostra esperienza, si ottengono sicurezza statica e durata sufficienti per la maggior parte delle applicazioni.

Per condizioni particolari (urti, vibrazioni, ambienti polverosi, accelerazioni, forze e momenti elevati, ecc) è necessario contattare **Rollon** per una ulteriore verifica tecnica. I valori massimi ammissibili della velocità, dell'accelerazione e della ripetibilità di posizionamento possono essere inferiori in caso di carichi elevati.



The capacity of **Rollon** ROBOT series linear units depends on the linear motion system used and may vary according to the loading direction: radial and side.

The maximum load in an axial direction depends on the type of driving belt used.

The maximum permissible values for radial load ( $F_z$ ) and side load ( $F_y$ ) indicated above correspond to 20% of the static capacity and 12% of the dynamic capacity of the performance of the ball bearing guides.

Experience shows that these values ensure safe static loading and adequate life for most applications.

For special conditions (such as shock, vibration, dusty environment, acceleration forces and high bending moments, etc), contact **Rollon** for further technical advice. Maximum permissible values for speed, acceleration and positioning repeatability can be lower in the presence of high loads

© All the drawings in this catalogue are available in CAD files on [www.rollon.com](http://www.rollon.com)

# 16 ROBOT 160 SP

**ROBOT 160 SP - Con guide a ricircolo di sfere**  
**Dati tecnici**

**ROBOT 160 SP - With ball bearings guide**  
**Technical data**

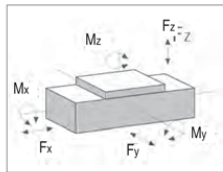
	ROBOT 160 SP
Lunghezza corsa utile min. [mm] - Min. useful stroke length [mm]	100
Lunghezza corsa utile max. [mm]*1 - Max. useful stroke length [mm]*1	6000
Ripetibilità max. di posizionamento [mm]*2 - Max. positioning repeatability [mm]*2	0,05
Velocità max. [m/s] - Max. speed [m/s]	5,0
Accelerazione max. [m/s <sup>2</sup> ] - Max. acceleration [m/s <sup>2</sup> ]	50
Tipo di cinghia - Type of belt	70 AT 10
Tipo di puleggia - Type of pulley	Ø 64 - Z 20 - Gioco 0
Spostamento carro per giro puleggia [mm] - Carriage displacement per pulley turn [mm]	200
Peso del carro [kg] - Carriage weight [kg]	5,3
Peso corsa zero [kg] - Zero travel weight [kg]	21
Peso per ogni 100 mm di corsa utile [kg] - Weight for 100 mm useful stroke [kg]	1,9

\*1) È possibile realizzare corse fino a 11000 mm tramite speciali giunzioni Rollon  
 \*1) It is possible to obtain strokes up to 11000 mm by means of special Rollon joints

\*2) La ripetibilità di posizionamento dipende dal tipo di trasmissione applicato  
 \*2) The positioning repeatability depends upon the type of transmission used

**ROBOT 160 SP - Carichi teorici massimi e consigliati / ROBOT 160 SP - Theoric and maximum permissible loads**

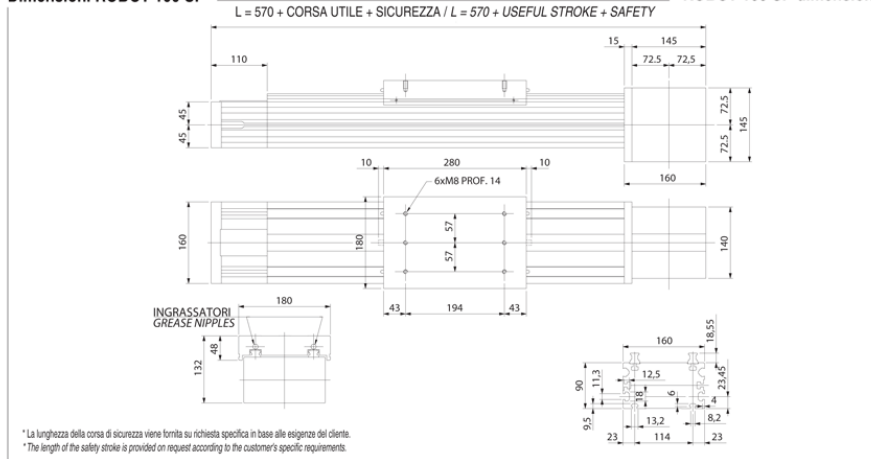
	Teorico - Theoric		Consigliato - Permissible*	
	stat.	din. / dyn.	stat.	din. / dyn.
Fx [N]	5150	3860	4120	3090
Fy [N]	86800	69600	17360	8350
Fz [N]	86800	69600	17360	8350
Mx [Nm]	4950	3970	990	476
My [Nm]	6900	5530	1380	664
Mz [Nm]	6900	5530	1380	664



\*) Con i valori riportati si ottengono una ragionevole durata ed una sufficiente sicurezza statica.  
 \*) Reasonable operating life and system rigidity can be obtained from the values given.

**Dimensioni ROBOT 160 SP**

**ROBOT 160 SP dimensions**



Ⓜ Nota: tutti i disegni riprodotti nel presente stampato sono disponibili sul sito [www.rollon.com](http://www.rollon.com) in formato CAD

## APPENDICE E

### **Dati tecnici motori**

In questa appendice viene riportata la documentazione tecnica, estratta dai cataloghi, riguardante i servomotori brushless Kollmorgen della serie AKM. In particolare, alle pag. 130-131 è indicata la numerazione che è stata utilizzata per i motori in fase di selezione.

# AKM Servomotor Series

## AKM Motors Offer Extremely High Torque Density and High Acceleration

The AKM high-performance motor series offers a wide range of mounting, connectivity, feedback and other options. These motors offer superb flexibility to meet application needs with:

- 8 frame sizes (40 to 260 mm)
- 28 frame-stack length combinations
- 117 'standard' windings

### Features

#### Torque

0.16 to 180 N-m continuous stall torque (1.4 to 1590 lb-in) in 28 frame/stack combinations. Specific torques are often available from multiple frame sizes to optimize mounting and inertia matching capabilities.

#### Speed

Speeds to 8000 rpm meet high speed application requirements. Windings tailored to lower speeds are also available.

#### Voltage

AKM motors can be applied to all standard global voltages. Windings are specifically tailored to 75 Vdc, 120, 240, 400 and 480 Vac.

#### Mounting

Multiple mounting standards are available to meet common European, North American, and Japanese standards.

#### Feedback

AKM motors include resolver, encoder (commutating), Sine-Absolute encoder or SFD (Smart Feedback Device) feedback options to meet specific application requirements.

#### Smoothness

Smooth performance results from low-cog, low-harmonic distortion magnetic designs.

#### Connectivity

Rugged, rotatable IP65 connectors or low cost IP20 Molex plugs are both available to provide flexibility. Single connectors/plugs (combined power and feedback) are also available to minimize motor and cable cost (SFD only).

#### Thermal

Windings are rated conservatively at 100°C rise over a 40°C ambient while using 155°C (class F) insulation materials. Motors meet applicable cURus and CE requirements and include thermistors. Thermal ratings at 60°C rise are also provided to meet the needs of specific applications.



### Kollmorgen Cables Offer the Complete Solution

The new Value Line cables provide a cost saving option for applications that don't require long distances or encounter extreme environmental conditions. Value Line is a composite cable that combines power and feedback in one cable to aid in faster machine commissioning. Contact Kollmorgen Customer Support to identify which cable option is best suited for your application.

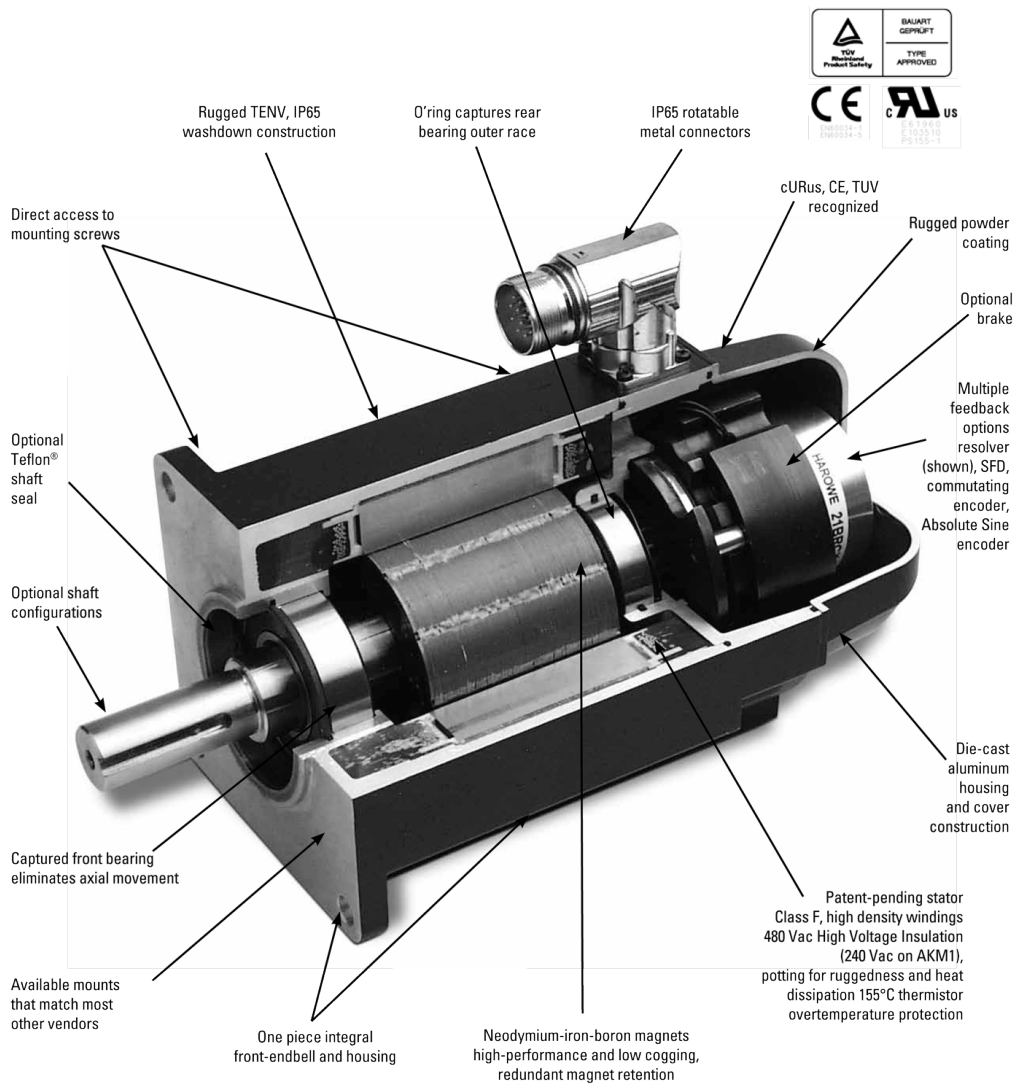
### Options

Additional options:

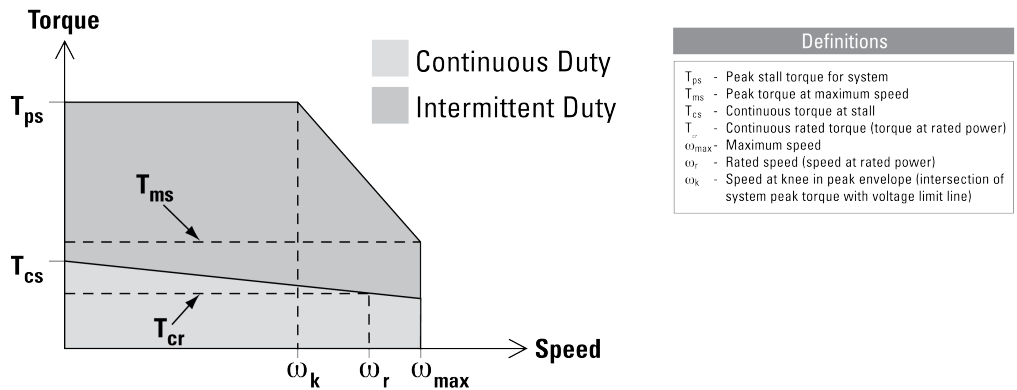
- Fail-safe brakes
- New, Teflon® shaft seals
- Feedback devices
- Shaft and mounting variations
- Custom windings
- Connectivity



### 3-D Model Shows Key Design Features



# AKM Systems Overview



## How to Build a Servo Drive and Motor System

System torque/speed information on the following pages is designed to help you select the optimum brushless servomotor/drives combination. The nominal values in this data illustrate performance for the recommended motor/controller systems.

## Drive and Motor Performance Curves

The performance characteristics of a brushless servo system (motor/drives combination) are described by a torque/speed operating envelope. As shown above, the shaded areas of the curve indicate the continuous duty and intermittent duty zones of the system.

### Continuous Duty Zone

The continuous duty zone is bordered by the maximum continuous torque line up to the intersection with the intermittent duty line. The continuous torque line is set by either the motor's maximum rated temperature, or the drives' rated continuous current output, whichever is less. The system voltage line is set by the voltage rating of the drives, the line voltage supplied, and the motor winding. The system can operate on a continuous basis anywhere within this area, assuming the ambient temperature is 40°C or less.

### Intermittent Duty Zone

The intermittent duty zone is bordered by the peak torque line and the system voltage line. The peak torque line is set by either the drives' peak current rating, which the drive can produce for a limited time, or the maximum rated peak current for the motor, whichever is less. Refer to the Rating Data on the pages that follow. Note: Higher torque levels may be achievable at higher power levels.

Consult Kollmorgen Customer Support for more details. The system voltage line is set by the voltage rating of the controller, the line voltage applied and the motor winding. Operation in the intermittent zone must be limited to a duty cycle that will produce an RMS system torque falling within the continuous duty area. The RMS torque value is a function of the magnitude of the intermittent torque and the percentage of the time spent at that torque.

# AKM Systems Overview

The AKM™ brushless servomotor stands alone in the marketplace in terms of flexibility and performance advantages. Kollmorgen's culture of continuous improvement has paid dividends again. The AKM servomotor's innovative design has been polished and optimized. With the new AKD amplifier, the venerable AKM servomotor sets a new standard of refined servo performance, designed to deliver precise motion and more power for your money. Nowhere else will you find a more versatile and complete servo family to meet your needs and exceed your expectations.

## Features

- Eight frame sizes (40 to 260 mm)
- 28 frame-stack length combinations
- 117 standard windings for low-voltage, 120/240/400/480 Vac operation
- Flexible flange mount and shaft options
- Industry-leading low-cogging contributing to extreme smoothness
- Wide feedback options for high-performance and precision or rugged environments
- Unmatched customization – special windings, special shafts, and much more



## AKD with AKM Plug-and-Play Feedback

These feedback devices include electronic motor nameplates allowing plug-and-play commissioning, eliminating the need for drive parameter set-up and servo loop tuning in most applications.

## Performance Data

	AKM Motor	Single-turn Absolute			Multi-turn Absolute		
		Accuracy (arc-min)	Resolution (bits)	Feedback Type	Accuracy (arc-min)	Resolution (bits)	Feedback Type
Value Line	AKM1	16	24	C	–	–	–
	AKM2-3	9	24	C	8	20	LB
	AKM4-8	9	24	C	4.66	21	LB
Performance Line	AKM2-4	1.0	27	DA	1.0	27	DB
	AKM5-8	0.333	27	DA	0.333	27	DB

Note: Additional plug-and-play feedback options covered in the feedback devices section on page 58.

Servomotori brushless Kollmorgen - serie AKM (230 Vac)					
N.	Motore	$J_m$ [kg·cm <sup>2</sup> ]	$C_n$ [Nm]	$C_{m,max}$ [Nm]	$\omega_{m,max}$ [rpm]
1	AKM21C	0.122	0.39	1.47	8000
2	AKM22C	0.172	0.77	2.73	5650
3	AKM22E	0.172	0.69	2.76	8000
4	AKM23D	0.232	1.01	3.84	6540
5	AKM23F	0.232	0.92	3.88	8000
6	AKM24D	0.282	1.24	4.76	5410
7	AKM24F	0.282	1.07	4.82	8000
8	AKM31C	0.342	1.12	3.88	4050
9	AKM31E	0.342	0.95	4.00	8000
10	AKM32C	0.602	1.90	6.92	2470
11	AKM32E	0.602	1.86	7.11	4710
12	AKM32H	0.602	1.40	7.26	8000
13	AKM33C	0.862	2.54	9.76	1840
14	AKM33E	0.862	2.52	9.96	3130
15	AKM33H	0.862	2.17	10.2	6640
16	AKM41C	0.878	1.78	6.12	2560
17	AKM41E	0.878	1.72	6.28	4850
18	AKM41H	0.878	1.52	6.36	6000
19	AKM42C	1.568	3.02	11.3	2740
20	AKM42C	1.568	2.80	11.5	4660
21	AKM42C	1.568	2.28	11.6	6000
22	AKM43C	2.168	4.04	15.9	2000
23	AKM43C	2.168	3.66	16.1	3850
24	AKM43C	2.168	2.33	16.0	6000
25	AKM44C	2.768	4.92	19.9	1680
26	AKM44C	2.768	4.36	20.2	3250
27	AKM44C	2.768	3.54	20.4	5010
28	AKM51E	3.570	4.26	11.6	2010
29	AKM51H	3.570	3.72	11.7	4030
30	AKM51L	3.570	1.85	12.0	6000

CONTINUA NELLA PAG. SUCCESSIVA ...

Nota 1:  $J_m$  comprende già l'inerzia del freno di stazionamento.

Nota 2:  $C_n$  è già diminuita per effetto del freno di stazionamento.

Nota 3:  $\omega_{m,max}$  dipende dal sistema di alimentazione ( $\omega_{m,max} \leq \omega_{m,mech}$ ).

... CONTINUA DALLA PAG. PRECEDENTE

Servomotori brushless Kollmorgen - serie AKM (230 Vac)					
N.	Motore	$J_m$ [kg·cm <sup>2</sup> ]	$C_n$ [Nm]	$C_{m,max}$ [Nm]	$\omega_{m,max}$ [rpm]
31	AKM52H	6.370	7.27	21.6	2390
32	AKM52L	6.370	6.14	22.0	4580
33	AKM52M	6.370	4.94	21.9	5230
34	AKM53G	9.270	10.4	29.7	1440
35	AKM53H	9.270	10.2	30.0	1970
36	AKM53L	9.270	9.24	30.3	3450
37	AKM53P	9.270	5.53	29.8	5770
38	AKM54H	12.17	13.0	37.5	1340
39	AKM54K	12.17	12.3	38.4	2290
40	AKM54L	12.17	11.1	37.5	3030
41	AKM54N	12.17	9.42	37.6	4320
42	AKM62H	17.61	10.3	29.6	1560
43	AKM62L	17.61	9.50	30.1	3380
44	AKM62M	17.61	9.00	30.2	3770
45	AKM62Q	17.61	6.00	30.9	6000
46	AKM63L	24.61	13.3	42.6	2260
47	AKM63M	24.61	13.4	43.0	2770
48	AKM63Q	24.61	11.0	51.9	4550
49	AKM64K	32.61	17.5	53.5	1510
50	AKM64L	32.61	17.1	54.1	2070
51	AKM64Q	32.61	14.0	53.2	3440
52	AKM65L	40.61	20.7	65.2	1660
53	AKM65M	40.61	20.2	65.2	1860
54	AKM65P	40.61	17.4	92.0	2750
55	AKM72P	66.64	21.8	78.5	2170
56	AKM72Q	66.64	21.3	78.4	2730
57	AKM73P	93.64	32.0	111	1610
58	AKM73Q	93.64	30.7	111	2020
59	AKM74Q	120.6	39.4	141	1710

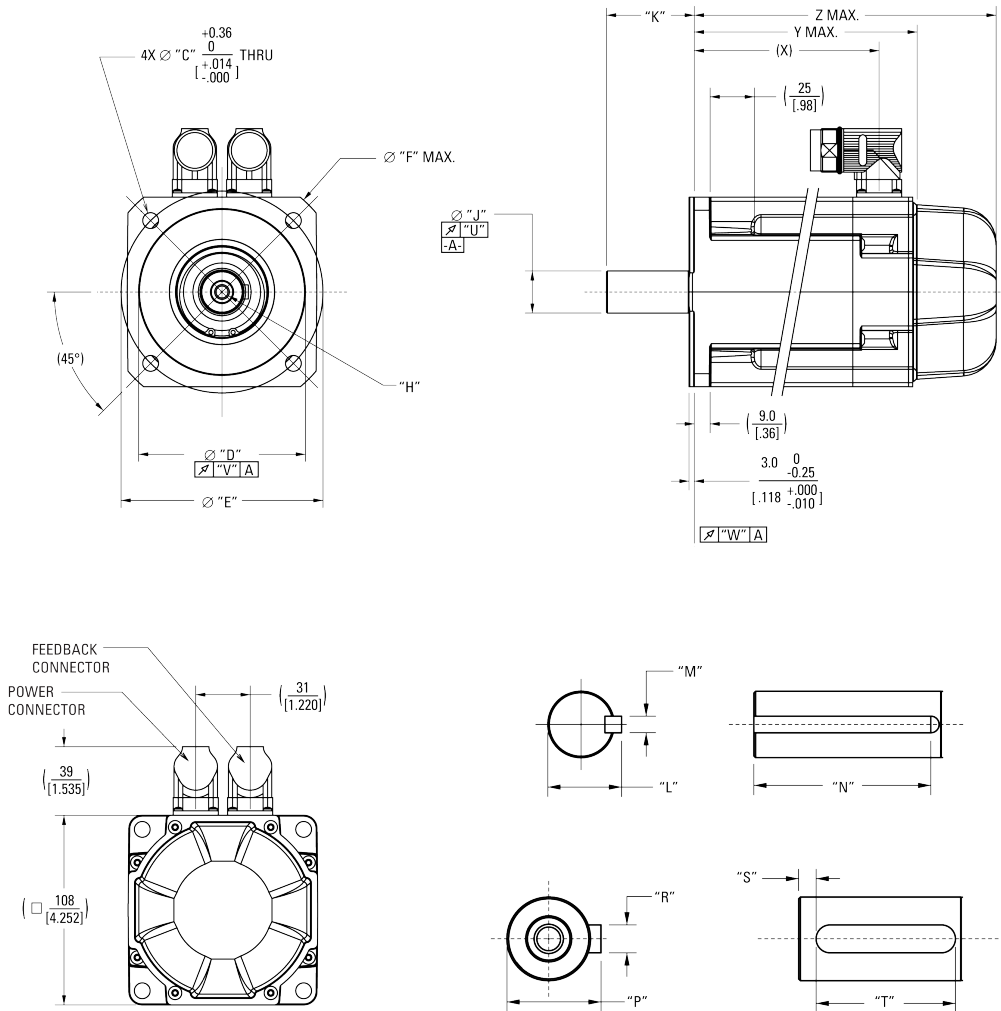
Nota 1:  $J_m$  comprende già l'inerzia del freno di stazionamento.

Nota 2:  $C_n$  è già diminuita per effetto del freno di stazionamento.

Nota 3:  $\omega_{m,max}$  dipende dal sistema di alimentazione ( $\omega_{m,max} \leq \omega_{m,mech}$ ).

# AKM5x Outline Drawings

## AKM5x Frame



# AKM5x Dimension Data

## AKM5x Dimension Data

Mounting Code	"C"	"D"	"E"	"F"	"H"	"J"	"K"	"L"
AC	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	–	D M8 DIN 332	24 <sup>+0.015</sup> +0.002 [.9449 <sup>+0.0006</sup> +0.0001]	{50.0} 1.97}	–
AN	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	–	D M8 DIN 332	24 <sup>+0.015</sup> +0.002 [.9449 <sup>+0.0006</sup> +0.0001]	{50.0} 1.97}	–
BK	8.33 [.328]	55.563 <sup>0</sup> -0.051 [2.1874 <sup>+0.0000</sup> -0.0020]	125.73 [4.950]	–	–	19.05 <sup>0</sup> +0.013 [.7500 <sup>+0.0000</sup> +0.0005]	{57.15 ± 0.79} [2.250 ± .031]}	21.15 <sup>0</sup> -0.43 [.83 <sup>+0.000</sup> -0.017]
CC	9 [.354]	95 <sup>+0.013</sup> -0.009 [3.7402 <sup>+0.0005</sup> -0.0003]	115 [4.528]	140 [5.512]	D M8 DIN 332	24 <sup>+0.015</sup> +0.002 [.9449 <sup>+0.0006</sup> +0.0001]	{50.0} 1.97}	–
CN	9 [.354]	95 <sup>+0.013</sup> -0.009 [3.7402 <sup>+0.0005</sup> -0.0003]	115 [4.528]	140 [5.512]	D M8 DIN 332	24 <sup>+0.015</sup> +0.002 [.9449 <sup>+0.0006</sup> +0.0001]	{50.0} 1.97}	–
DK	8.33 [.328]	63.5 <sup>0</sup> -0.05 [2.500 <sup>+0.000</sup> -0.002]	127 [5.000]	–	–	19.05 <sup>+0</sup> +0.013 [.7500 <sup>+0.0000</sup> +0.0005]	{57.15 ± 0.79} [2.250 ± .031]}	21.15 <sup>0</sup> -0.43 [.83 <sup>+0.000</sup> -0.017]
EK	8.33 [.328]	55.563 <sup>0</sup> -0.051 [2.1874 <sup>+0.0000</sup> -0.0020]	125.73 [4.950]	–	–	15.875 <sup>0</sup> +0.013 [.6250 <sup>+0.0000</sup> +0.0005]	{44.45} [1.750]}	17.91 <sup>0</sup> -0.43 [.705 <sup>+0.000</sup> -0.017]
GC	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	–	D M6 DIN 332	19 <sup>+0.015</sup> +0.002 [.7480 <sup>+0.0006</sup> +0.0001]	{40} [1.57]}	–
GN	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	–	D M6 DIN 332	19 <sup>+0.015</sup> +0.002 [.7480 <sup>+0.0006</sup> +0.0001]	{40.0} [1.57]}	–
HC	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	140 [5.512]	D M6 DIN 332	19 <sup>+0.015</sup> +0.002 [.7480 <sup>+0.0006</sup> +0.0001]	{40} [1.57]}	–
HN	9 [.354]	110 <sup>+0.013</sup> -0.009 [4.3307 <sup>+0.0005</sup> -0.0003]	130 [5.118]	140 [5.512]	D M6 DIN 332	19 <sup>+0.015</sup> +0.002 [.7480 <sup>+0.0006</sup> +0.0001]	{40.0} [1.57]}	–

Continued on the following page

MODEL	Z MAX SINE ENCODER (NO BRAKE)	Z MAX SINE ENCODER (W BRAKE)	(X)	Y MAX	Z MAX (W BRAKE)
AKM51	146.0 [5.75]	189.0 [7.44]	105.3 [4.15]	127.5 [5.02]	172.5 [6.79]
AKM52	177.0 [6.97]	220.0 [8.66]	136.3 [5.37]	158.5 [6.24]	203.5 [8.01]
AKM53	208.0 [8.19]	251.0 [9.88]	167.3 [6.59]	189.5 [7.46]	234.5 [9.23]
AKM54	239.0 [9.41]	282.0 [11.10]	198.3 [7.81]	220.5 [8.68]	265.5 [10.45]

Note 1: Dimensions are in mm [inches].

Note 2: Product designed in metric. English conversions provided for reference only.

# AKM5x Dimension Data

AKM5x Dimension Data (continued)

Mounting Code	"M"	"N"	"P"	"R"	"S"	"T"	"U"	"V"	"W"
AC	–	–	$27 \begin{matrix} 0 \\ -0.29 \\ +0.000 \\ [1.063 \text{ } -0.001] \end{matrix}$	$8 \begin{matrix} 0 \\ -0.036 \\ +0.000 \\ [0.3150 \text{ } -0.0014] \end{matrix}$ NS	5.00 [1.97]	$40 \begin{matrix} 0 \\ -0.30 \\ +0.000 \\ [1.575 \text{ } -0.012] \end{matrix}$	0.040 [.0015]	0.100 [.0039]	0.100 [.0039]
AN	–	–	–	–	–	–	0.040 [.0015]	0.100 [.0039]	0.100 [.0039]
BK	$4.763 \begin{matrix} 0 \\ -0.050 \\ +0.000 \\ [1.1875 \text{ } -0.0020] \end{matrix}$	$38.1 \pm 0.25$ [1.500 ± .010]	–	–	–	–	0.051 [.0020]	0.10 [.004]	0.10 [.004]
CC	–	–	$27 \begin{matrix} 0 \\ -0.29 \\ +0.000 \\ [1.063 \text{ } -0.011] \end{matrix}$	$8 \begin{matrix} 0 \\ -0.036 \\ +0.000 \\ [0.3150 \text{ } -0.0014] \end{matrix}$ NS	5.00 [1.97]	$40 \begin{matrix} 0 \\ -0.30 \\ +0.000 \\ [1.575 \text{ } -0.012] \end{matrix}$	0.040 [.0015]	0.080 [.0031]	0.080 [.0031]
CN	–	–	–	–	–	–	0.040 [.0015]	0.080 [.0031]	0.080 [.0031]
DK	$4.763 \begin{matrix} 0 \\ -0.050 \\ +0.000 \\ [1.1875 \text{ } -0.0020] \end{matrix}$	$34.93 \pm 0.25$ [1.375 ± .010]	–	–	–	–	0.051 [.0020]	0.05 [.002]	0.10 [.004]
EK	$4.763 \begin{matrix} 0 \\ -0.050 \\ +0.000 \\ [1.1875 \text{ } -0.0020] \end{matrix}$	$38.1 \pm 0.25$ [1.500 ± .010]	–	–	–	–	0.051 [.0020]	0.10 [.004]	0.10 [.004]
GC	–	–	$21.5 \begin{matrix} 0 \\ -0.13 \\ +0.000 \\ [0.846 \text{ } -0.005] \end{matrix}$	$6 \begin{matrix} 0 \\ -0.03 \\ +0.000 \\ [0.236 \text{ } -0.001] \end{matrix}$ NS	4.00 [1.57]	$32 \begin{matrix} 0 \\ -0.30 \\ +0.000 \\ [1.260 \text{ } -0.012] \end{matrix}$	0.040 [.0015]	0.080 [.0031]	0.080 [.0031]
GN	–	–	–	–	–	–	–	–	–
HC	–	–	$21.5 \begin{matrix} 0 \\ -0.13 \\ +0.000 \\ [0.846 \text{ } -0.005] \end{matrix}$	$6 \begin{matrix} 0 \\ -0.03 \\ +0.000 \\ [0.236 \text{ } -0.001] \end{matrix}$ NS	4.00 [1.57]	$32 \begin{matrix} 0 \\ -0.30 \\ +0.000 \\ [1.260 \text{ } -0.012] \end{matrix}$	0.040 [.0015]	0.080 [.0031]	0.080 [.0031]
HN	–	–	–	–	–	–	–	–	–



# AKM5x Performance Data

## AKM5x Performance Data – Up to 640 Vdc

Parameters	Tol	Sym	Units	AKM51			AKM52				AKM53				AKM54			
				E	H	L	E	H	L	M	G	H	L	P	H	K	L	N
Max Rated DC Bus Voltage	Max	V <sub>bus</sub>	Vdc	640	640	320	640	640	640	320	640	640	640	320	640	640	560	320
Continuous Torque (Stall) for ΔT winding = 100°C ①②⑦⑧⑨	Nom	T <sub>cs</sub>	N-m	4.70	4.79	4.89	8.34	8.48	8.67	8.60	11.4	11.5	11.6	11.4	14.2	14.4	14.1	14.1
			lb-in	41.6	42.4	43.3	73.8	75.1	76.7	76.1	101	102	103	101	126	127	125	125
Continuous Current (Stall) for ΔT winding = 100°C ①②⑦⑧⑨	Nom	I <sub>cs</sub>	A <sub>rms</sub>	2.75	6.0	11.9	2.99	5.9	11.6	13.1	4.77	6.6	11.8	19.1	5.5	9.7	12.5	17.8
Continuous Torque (Stall) for ΔT winding = 60°C ②	Nom	T <sub>cs</sub>	N-m	3.76	3.83	3.91	6.67	6.78	6.94	6.88	9.10	9.21	9.28	9.10	11.5	11.5	11.3	11.3
			lb-in	33.3	33.9	34.6	59.0	60.0	61.4	61.0	80.5	81.5	82.1	80.5	102	102	100	100
Max Mechanical Speed ⑤	Nom	N <sub>max</sub>	rpm	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000	6000
Peak Torque ①②	Nom	T <sub>p</sub>	N-m	11.6	11.7	12.0	21.3	21.6	22.0	21.9	29.7	30.0	30.3	29.8	37.5	38.4	37.5	37.6
			lb-in	103	104	106	189	191	195	194	263	266	268	264	332	340	332	333
Peak Current	Nom	I <sub>p</sub>	A <sub>rms</sub>	8.24	18.0	35.7	9.00	17.7	34.8	39.4	14.3	19.8	35.4	57.4	16.5	29.2	37.5	53.4
Rated Torque (speed) ①②⑦⑧⑨⑩	75 Vdc	T <sub>rtd</sub>	N-m	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
			lb-in	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Rated Speed		N <sub>rtd</sub>	rpm	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Rated Power (speed) ①②⑦⑧⑨	75 Vdc	P <sub>rtd</sub>	kW	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
			Hp	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Rated Torque (speed) ①②⑦⑧⑨⑩	160 Vdc	T <sub>rtd</sub>	N-m	-	4.46	3.95	-	-	7.89	-	-	-	13.0	-	-	-	-	-
			lb-in	-	39.4	35.0	-	-	69.9	-	-	-	115	-	-	-	-	-
Rated Speed	160 Vdc	N <sub>rtd</sub>	rpm	-	1200	3000	-	-	1500	-	-	-	1200	-	-	-	-	-
Rated Power (speed) ①②⑦⑧⑨	160 Vdc	P <sub>rtd</sub>	kW	-	0.56	1.24	-	-	1.24	-	-	-	1.63	-	-	-	-	-
			Hp	-	0.75	1.66	-	-	1.66	-	-	-	2.18	-	-	-	-	-
Rated Torque (speed) ①②⑦⑧⑨⑩	320 Vdc	T <sub>rtd</sub>	N-m	4.41	3.87	2.00	-	7.53	6.40	5.20	10.7	10.5	9.59	5.88	13.4	12.7	11.5	9.85
			lb-in	39.0	34.3	17.7	-	66.6	56.6	46.0	94.5	93.0	84.9	52.0	118	112	102	87.2
Rated Speed	320 Vdc	N <sub>rtd</sub>	rpm	1200	3000	6000	-	1800	3500	4500	1000	1500	2500	5000	1000	1800	2500	3500
Rated Power (speed) ①②⑦⑧⑨	320 Vdc	P <sub>rtd</sub>	kW	0.55	1.22	1.26	-	1.42	2.35	2.45	1.12	1.65	2.51	3.08	1.4	2.39	3.00	3.61
			Hp	0.74	1.63	1.69	-	1.90	3.15	3.28	1.50	2.21	3.36	4.13	1.87	3.20	4.03	4.84
Rated Torque (speed) ①②⑦⑧⑨⑩	560 Vdc	T <sub>rtd</sub>	N-m	3.98	1.97	-	7.61	6.26	3.27	-	9.85	8.83	6.00	-	12.6	10.05	8.13	-
			lb-in	35.2	17.4	-	67.3	55.4	2.89	-	87.2	78.2	53.1	-	112	88.9	72.0	-
Rated Speed	560 Vdc	N <sub>rtd</sub>	rpm	2500	6000	-	1500	3500	6000	-	2000	3000	5000	-	1800	3500	4500	-
Rated Power (speed) ①②⑦⑧⑨	560 Vdc	P <sub>rtd</sub>	kW	1.04	1.24	-	1.20	2.30	2.06	-	2.06	2.77	3.14	-	2.38	3.68	3.83	-
			Hp	1.40	1.66	-	1.60	3.08	2.76	-	2.77	3.72	4.21	-	3.18	4.94	5.14	-
Rated Torque (speed) ①②⑦⑧⑨⑩	640 Vdc	T <sub>rtd</sub>	N-m	3.80	1.97	-	7.28	5.77	3.27	-	9.50	8.82	4.05	-	12.2	9.25	-	-
			lb-in	33.6	17.4	-	64.4	51.1	28.9	-	84.0	78.0	35.8	-	108	81.9	-	-
Rated Speed	640 Vdc	N <sub>rtd</sub>	rpm	3000	6000	-	2000	4000	6000	-	2400	3000	6000	-	2000	4000	-	-
Rated Power (speed) ①②⑦⑧⑨	640 Vdc	P <sub>rtd</sub>	kW	1.19	1.24	-	1.52	2.42	2.06	-	2.39	2.77	2.55	-	2.56	3.87	-	-
			Hp	1.60	1.66	-	20.4	3.24	2.76	-	3.20	3.71	3.41	-	3.43	5.19	-	-

Note 1: See system data beginning on page 13 for typical torque/speed performance.  
 Note 2: Additional windings can be found through our online Motioneering sizing and selection software tool. See page 71 for more information.

# AKM5x Performance Data

Parameters	Tol	Sym	Units	AKM51				AKM52				AKM53				AKM54			
				E	H	L		E	H	L	M	G	H	L	P	H	K	L	N
Torque Constant ①	±10%	K <sub>t</sub>	N-m/ A <sub>rms</sub>	1.72	0.80	0.41	2.79	1.44	0.75	0.66	2.39	1.75	0.99	0.60	2.6	1.50	1.13	0.80	
			lb-in/ A <sub>rms</sub>	15.2	7.1	3.6	24.7	12.7	6.6	5.8	21.2	15.5	8.8	5.3	22.7	13.3	10.0	7.1	
Back EMF Constant ⑥	±10%	K <sub>e</sub>	V/krpm	110	51.3	26.6	179	92.7	48.3	42.4	154	112	63.6	38.4	166	96.6	72.9	51.3	
Resistance (line-line) ⑥	±10%	R <sub>m</sub>	ohm	8.98	1.97	0.56	8.96	2.35	0.61	0.49	3.97	2.1	0.69	0.28	3.2	1.08	0.65	0.33	
Inductance (line-line)		L	mH	36.6	7.9	2.1	44.7	11.9	3.24	2.5	21.3	11.4	3.64	1.3	18.3	6.2	3.5	1.8	
Inertia (includes Resolver feedback) ③		J <sub>m</sub>	kg-cm <sup>2</sup>	3.4				6.2				9.1				12			
			lb-in-s <sup>2</sup>	3.0E-03				5.5E-03				8.1E-03				0.011			
Optional Brake Inertia (additional)		J <sub>m</sub>	kg-cm <sup>2</sup>	0.17				0.17				0.17				0.17			
			lb-in-s <sup>2</sup>	1.5E-04				1.5E-04				1.5E-04				1.5E-04			
Weight		W	kg	4.2				5.8				7.4				9			
			lb	9.3				12.8				16.3				19.8			
Static Friction ①⑩		T <sub>f</sub>	N-m	0.022				0.04				0.058				0.077			
			lb-in	0.19				0.35				0.51				0.68			
Viscous Damping ①		K <sub>dv</sub>	N-m/ krpm	0.033				0.042				0.052				0.061			
			lb-in/ krpm	0.29				0.37				0.46				0.54			
Thermal Time Constant		TCT	minutes	20				24				28				31			
Thermal Resistance		R <sub>thw-a</sub>	°C/W	0.68				0.56				0.50				0.45			
Pole Pairs				5				5				5				5			
Heat Sink Size				12"x12"x1/2" Aluminum Plate				12"x12"x1/2" Aluminum Plate				12"x12"x1/2" Aluminum Plate				12"x12"x1/2" Aluminum Plate			

Notes:

- ① Motor winding temperature rise, ΔT=100°C, at 40°C ambient.
- ② All data referenced to sinusoidal commutation.
- ③ Add parking brake if applicable for total inertia.
- ④ Motor with standard heat sink.
- ⑤ May be limited at some values of V<sub>bus</sub>.
- ⑥ Measured at 25°C.
- ⑦ Brake motor option reduces continuous torque ratings by:  
AKM51 = 0.15 N-m AKM52 = 0.26 N-m AKM53 = 0.35 N-m AKM54 = 0.43 N-m
- ⑧ Non-Resolver feedback options reduce continuous torque ratings by:  
AKM51 = 0.15 N-m AKM52 = 0.34 N-m AKM53 = 0.58 N-m AKM54 = 0.86 N-m
- ⑨ Motors with non-resolver feedback and brake option, reduce continuous torque by:  
AKM51 = 0.39 N-m AKM52 = 0.76 N-m AKM53 = 1.13 N-m AKM54 = 1.55 N-m
- ⑩ For motors with optional shaft seal, reduce torque shown by 0.013 N-m (0.1.2 lb-in), and increase T<sub>f</sub> by the same amount.

# Brake Option

## Fail-safe, Holding Brake

The holding brake is designed to provide static holding torque to the motor shaft with the brake coil de-energized. The brake must first be released (coil energized) prior to commanding motor rotation as determined by its drop-out time. The brake is intended for holding or "parking" of a stationary motor. It is not intended for dynamic braking. There should be absolutely no motion of the rotor when power is removed from the brake coil.

### AKM Motor Brake Specifications

Motor Family	Minimum Static Torque @120°C		Weight		Power Consumption @24V, 20°C Watts ±7%	Current @24V, 20°C ADC	Inertia		Closing Time (engage) msec	Opening Time (release) msec	Backlash	
	N-m	lb-in	Kg	lb			kg-cm <sup>2</sup>	lb-in-sec <sup>2</sup>			Maximum deg.	Typical deg.
AKM2	1.42	12.6	0.27	0.59	8.4	0.35	0.011	0.97E-05	36	35	1.01	0.46
AKM3	2.5	22.1	0.35	0.77	10.1	0.42	0.011	0.97E-05	20	50	1.01	0.46
AKM4	5.3	46.9	0.63	1.39	12.8	0.53	0.068	6.02E-05	30	75	0.81	0.37
AKM5	14.5	128	1.1	2.42	19.5	0.82	0.173	1.53E-04	30	115	0.71	0.31
AKM6	25	221	2	4.4	25.7	1.07	0.605	5.35E-04	40	155	0.51	0.24
AKM7	53	469	2.9	6.38	35.6	1.48	1.64	1.46E-03	70	170	0.44	0.20
AKM8	150	1330	8	17.5	49	2.04	5.53	4.90E-03	100	300	0.44	0.20

Note 1: Contamination of the motor internal compartment by oil or other foreign materials will result in failure of the brake. Check the suitability of motor sealing for the working environment.

Note 2: Operating Voltage: 24 VDC +/- 10%.

Note 3: Maximum backlash is calculated using worst-case tolerancing, and typical backlash is calculated using statistical tolerancing.



## BIBLIOGRAFIA

- [1] Bayati, I., Belloli, M., Facchinetti, A., Giappino, S., 2013, Wind Tunnel Tests on Floating Offshore Wind Turbines: A Proposal for Hardware-in-the-Loop Approach to Validate Numerical Codes, *Wind Engineering Journal*, **37**(6), pp. 557-568.
- [2] Bayati, I., Belloli, M., Ferrari, D., Fossati, F., Giberti, H., 2014, "Design of a 6-DoF Robotic Platform for Wind Tunnel Tests of Floating Wind Turbines, *Proceedings of 11<sup>th</sup> Deep Sea Off-shore Wind R&D Conference - EERA DeepWind'2014*, January 22-24, Trondheim, Norway.
- [3] Weck, M., Staimer, D., 2002, "Parallel Kinematic Machine Tools: Current State and Future Potentials, *CIRP Annals, Manufacturing Technology - Elsevier Journal*, **51**, pp. 671-683.
- [4] Merlet, J.P., 2006, "Parallel Robots, 2<sup>nd</sup> ed., Springer.
- [5] Honegger, M., Codourey, A., Burdet, E., 1997, "Adaptive Control of the Hexaglide, a 6-DoF Parallel Manipulator, *Proceedings of IEEE International Conference on Robotics and Automation*, **1**, pp. 543-548.
- [6] Honegger, M., 1999, "Konzept einer Steuerung mit Adaptiver Nichtlinearer Regelung für einen Parallelmanipulator, Ph.D. thesis, ETH, Zürich.
- [7] Hebsacker, M., 2000, "Entwurf und Bewertung Paralleler Werkzeugmaschinen - das Hexaglide, Ph.D. thesis, ETH, Zürich.
- [8] Bonev, I.A., 1998, "Analysis and Design of a 6-DOF 6-PRRS Parallel Manipulators, M.Sc. thesis, GIST, South Korea.
- [9] Koteswara Rao, A.B., Rao, P.V.M., Saha, S.K., 2005, "Dimensional Design of Hexaslides for Optimal Workspace and Dexterity, *IEEE Transactions On Robotics*, **21**(3), pp. 444-449.
- [10] Gogu, Grigore. "Structural Synthesis of Parallel Robots. Part 1: Methodology. Springer, 2008.
- [11] Fassi, I., Legnani, G., Tosi, D., 2005, "Geometrical Conditions for the Design of Partial or Full Isotropic Hexapods, *Journal of Robotic Systems*, **22**(10), pp. 507-518.

- [12] Legnani, G., Tosi, D., Fassi, I., Giberti, H., Cinquemani, S., 2010, “The Point of Isotropy and other Properties of Serial and Parallel Manipulators, *Mechanism and Machine Theory*, **45**(10), pp. 1407-1423.
- [13] Legnani, G., Fassi, I., Giberti, H., Cinquemani, S., Tosi, D., 2012, “A New Isotropic and Decoupled 6-DoF Parallel Manipulator, *Mechanism and Machine Theory*, **58**, pp. 64-81.
- [14] Kalyanmoy, D., 2001, “Multi-Objective Optimization using Evolutionary Algorithms, Wiley.
- [15] Behrouz Afzali-Far , Per Lidström , Kristina Nilsson, 2013, “Parametric damped vibrations of Gough?Stewart platforms for symmetric configurations, Elsevier.
- [16] Gallerani F., 2012, “Analisi numerico sperimentale del rendimento nelle viti a ricircolo di sfere.
- [17] Giberti, H., Cinquemani, S., Legnani, G., 2011, “A practical approach to the selection of the motor-reducer unit in electric drive systems”, *Mechanics Based Design of Structures and Machines*, **39**(3), pp. 303-319.
- [18] Giberti, H., Cinquemani, S., Legnani, G., 2010, “Effects of Transmission Mechanical Characteristics on the Choice of a Motor-Reducer, *Mechatronics - Elsevier Journal*, **20**, pp. 604 - 610.
- [19] Giberti, H., Clerici, A., Cinquemani, S., Available online 16 December 2013, “Specific Accelerating Factor: One More Tool in Motor Sizing Projects”, *Mechatronics - Elsevier Journal*, ISSN 0957-4158, <http://dx.doi.org/10.1016/j.mechatronics.2013.11.007>.
- [20] Kollmorgen-Danaher, 2013, “Automation and Motion Solutions Catalog, [www.kollmorgen.com](http://www.kollmorgen.com).
- [21] Micron-Danaher, 2012, “True Planetary Gearheads, [www.thomsonlinear.com](http://www.thomsonlinear.com).