# **POLITECNICO DI TORINO**

Corso di Laurea Magistrale in Ingegneria Meccanica

Tesi di Laurea Magistrale

Modellazione multibody su ADAMS di vite a ricircolo di sfere per identificazione modalità di guasto progressivo per finalità di prognostica di attuatori elettromeccanici per comandi primari di volo



Relatori: Prof. Massimo Sorli

Candidato: Parisi Gianluca

Ing. Antonio Carlo Bertolino

Anno Accademico 2018/2019

# RINGRAZIAMENTI

Ringrazio innanzitutto il prof. Massimo Sorli che mi ha permesso di affrontare questa attività di tesi e di lavorare sempre con i giusti mezzi, consentendomi inoltre di imparare un programma del tutto nuovo per me. Ringrazio l'ing. Antonio Carlo Bertolino che mi ha fornito il suo aiuto e il suo sostegno tutte le volte che ne avevo bisogno a tutte le ore del giorno. Ringrazio l'ing. Daniele Catelani della MSC Software per la disponibilità e per la collaborazione che mi ha permesso di realizzare questo modello in ADAMS.

Ringrazio la mia famiglia che mi è sempre stata vicino, per me è la cosa più preziosa che ho ed è la fonte della mia felicità, non potevo chiedere di meglio in questa vita. Ringrazio gli amici di sempre, quelli con cui ho condiviso tante emozioni, sin dalla più tenera età fino ad oggi, abbiamo affrontato qualsiasi situazione insieme e ci siamo sempre sostenuti l'un l'altro. Ringrazio gli amici che ho conosciuto all'interno del Politecnico di Torino, delle persone fantastiche con cui ho studiato e con cui ho sostenuto ogni singolo esame insieme, il mio percorso di studio è proseguito anche grazie a loro in quanto ci siamo sempre aiutati con appunti, spiegazioni e consigli. Ringrazio gli amici del calcetto e della palestra con cui mi sono sempre allenato e con cui mi sono sempre divertito, mi è sempre piaciuto fare attività sportiva con loro. Chiedo scusa se non ho citato nessuno ma non voglio fare esclusioni, sono molte le persone importanti e non voglio dimenticarne nessuna.

# SOMMARIO

| 1   | Intro  | oduzione1       |   |     |
|---|--|-----------------|---|-----|
| 2 Architettura EMA                              |  |                 | ra EMA  | . 3 |
|   | 2.1  | Moto            | re Brushless a corrente continua (BLDC)         | . 6 |
|   | 2.2  | Ridut           | ttore epicicloidale                             | . 7 |
|   | 2.3  | Frizic          | one   | . 7 |
| 2.4 Sistema di controllo                        |  | ma di controllo | . 8   |     |
|   | 2.5  | Vite a          | a ricircolo di sfere                            | . 8 |
|   | 2.5.   | 1 (             | Geometria del profilo                           | . 9 |
|   | 2.5.   | 2 1             | Ricircolo delle sfere                           | 10  |
|   | 2.5.   | 3               | Vite a ricircolo di sfere scelta per il modello | 11  |
| 3   | Def  | inizion         | ne dei parametri principali del modello         | 13  |
|   | 3.1  | Centr           | i delle sfere                                   | 13  |
|   | 3.2  | Rigid           | lezza di contatto                               | 21  |
|   | 3.3  | Ango            | lo di contatto                                  | 26  |
| 4   | Moo  | dellazi         | one in MSC ADAMS                                | 29  |
|   | 4.1  | Strutt          | tura dei files e comandi di default del modello | 29  |
|   | 4.2  | Strutt          | tura del modello in ADAMS                       | 32  |
|   | 4.3  | Macr            | 0   | 33  |
|   | 4.3.   | 1 (             | Generate_spheres_xyz                            | 34  |
|   | 4.3.   | 2 (             | Generate_contact_force_screwshaft_sphere        | 38  |
|   | 4.3.   | 3 (             | Generate_contact_force_sphere_nut               | 42  |
|   | 4.3.   | 4 0             | Generate_contact_force_sphere_sphere            | 44  |
|   | 4.3.   | 5 (             | Generate_requests                               | 46  |
|   | 4.4  | Impo            | rtazione e creazione delle parti                | 49  |
| <ul><li>4.5 Forze</li><li>4.6 Vincoli</li></ul> |  | 2               | 51  |     |
|   |  | oli             | 56  |     |
|   | <ul> <li>4.7 Motion</li> <li>4.8 Marker per l'angolo di contatto</li> <li>4.9 Measure per il rendimento</li> </ul> |                 | on  | 60  |
|   |  |                 | er per l'angolo di contatto                     | 61  |
|   |  |                 | 62  |     |
|   | 4.10 Script di simulazione   |                 | t di simulazione                                | 63  |
|   | 4.11   | Requ            | ests  | 64  |
| 5   | Sett   | ings de         | el modello                                      | 65  |
|   | 5.1  | Dyna            | mics  | 66  |
|   | 5.2  | Conta           | acts  | 68  |
| 5.3 Executable                                  |  | utable          | 70  |     |

| 5.4              | Equilibrium                            | 71 |  |  |  |
|------------------|--|----|--|--|--|
| 5.5              | Output                                 | 72 |  |  |  |
| 5.6              | Display                                | 72 |  |  |  |
| 6 Pos            | st-processing e risultati              | 73 |  |  |  |
| 6.1              | Rendimento                             | 74 |  |  |  |
| 6.2              | Angolo di contatto                     | 77 |  |  |  |
| 6.3              | Velocità di rivoluzione e di rotazione | 80 |  |  |  |
| 6.4              | Velocità di strisciamento              | 85 |  |  |  |
| 6.5              | Forze di contatto                      | 88 |  |  |  |
| 6.6              | Accelerazione della madrevite          |    |  |  |  |
| 7 Conclusioni    |  |    |  |  |  |
| Nomenclatura     |  |    |  |  |  |
| Design Variables |  |    |  |  |  |
| Bibliografia     |  |    |  |  |  |
|                  |  |    |  |  |  |

# **1 INTRODUZIONE**

Gli attuatori elettromeccanici (EMA) trovano spazio in numerose applicazioni in vari campi industriali, come ad esempio la robotica per la movimentazione dei gradi di libertà del manipolatore, nelle macchine utensili e nelle macchine di test per la gestione dei vari assi lineari o rotatori oppure in altri utilizzi a livello industriale. Recentemente stanno trovando spazio anche nel campo aereonautico: in particolare la ricerca si sta focalizzando attualmente sulla realizzazione e sullo sviluppo di aerei "more electric", dove il punto chiave è quello di evolvere il sistema di controllo delle superfici di volo dei velivoli, passando dal tradizionale sistema di attuazione elettro-idraulico a quello elettromeccanico.

L'utilizzo di questa tecnologia ha dei vantaggi e degli svantaggi rispetto al sistema tradizionale: la presenza degli attuatori elettromeccanici permette di eliminare le problematiche legate alle perdite nel circuito idraulico e consente di ottenere un minor costo di installazione e manutenzione; inoltre questo consente di ottenere un alleggerimento del peso complessivo del velivolo, di migliorare il rapporto peso-potenza e di ridurre l'impatto ambientale. Allo stesso tempo, gli EMA hanno una probabilità di grippaggio più elevata e ciò porta ad una riduzione dell'affidabilità del sistema. Questo è il problema principale che impedisce la diffusione di questo tipo di attuatori per la movimentazione di superfici safety-critical come i comandi di volo primari in quanto si compromette il buon funzionamento del velivolo ma soprattutto si metterebbe a rischio la vite delle persone a bordo.

Negli anni sono stati innumerevoli gli studi fatti da aziende, università e centri di ricerca per ovviare al problema principale degli attuatori elettromeccanici, trovando diverse soluzioni come ad esempio il by-pass dell'attuatore grippato con uno di backup, che può essere una buona soluzione ma allo stesso tempo aumenta la complessità, il peso e l'ingombro del sistema, oppure l'utilizzo di un unico attuatore affiancato da un sistema di prognostica del sistema definendo indicatori di salute e monitorando l'intero funzionamento dell'attuatore. Per quest'ultima soluzione, risulta di rilevante importanza la realizzazione di un modello numerico molto accurato e preciso che riproduca le condizioni di funzionamento in assenza e presenza di difetti, in maniera da osservare i fenomeni a cui è soggetto il sistema e per estrapolare i vari dati relativi agli indicatori di salute, da poter confrontare e sfruttare nell'attività di prognostica.

La seguente tesi è incentrata proprio sulla definizione del modello numerico dell'EMA ad alta fedeltà, in particolare si focalizza su un elemento del sistema, ossia la vite a ricircolo di sfere, realizzando uno studio dinamico in ambiente MSC ADAMS, software per la simulazione dinamica multibody. Le viti a ricircolo di sfere sono un sistema meccanico che permettono la trasformazione del moto rotatorio in un moto lineare o viceversa. Gli elementi che compongono il seguente sistema sono un albero filettato, una madrevite detta anche chiocciola, delle sfere che si interpongono tra vite e madrevite, degli elementi di ricircolo e degli elementi di tenuta, se necessari. Una caratteristica fondamentale di questa tipologia di trasmissione è quella di avere elevate prestazioni grazie alla presenza di un attrito volvente garantito dalle sfere in acciaio, che ruotano all'interno della pista elicoidale.

Il modello sviluppato della vite a ricircolo di sfere permette di studiare nella maniera più completa e realistica possibile il sistema, tenendo conto per esempio del ricircolo delle sfere, cosa che in altre modellazioni risulta difficile da realizzare. La particolarità di questo modello è quella di essere stato costruito interamente mediante script .cmd (Script di comandi Windows) e sfruttando la funzionalità delle macro che si possono generare all'interno di ADAMS. Questo ha permesso di creare una struttura modulare del modello, in maniera da dare la possibilità all'utente di studiare viti a ricircolo di sfere di qualsiasi tipologia. Si sottolinea inoltre che i software di supporto utilizzati sono stati SolidWorks per la realizzazione del Parasolid da importare nel modello realizzato e MATLAB per il calcolo dei dati da introdurre all'interno del modello e per la fase di post-processing dei dati di output delle simulazioni effettuate.

L'attività di tesi è costituita da tre step principali che sono i seguenti:

- La realizzazione del modello dove si è effettuata la modellazione del sistema meccanico all'interno dell'ambiente ADAMS, importando i Parasolid necessari, creando le sfere all'interno del percorso elicoidale e del percorso di ricircolo, generando le opportune forze necessarie (come forze di contatto e carico assiale) ed imponendo i vincoli minimi indispensabili per eliminare i gradi di libertà delle parti che potrebbero rendere irrealistico il modello;
- 2. Il setting dei parametri di simulazione dove si sono imposti i parametri riguardanti l'analisi dinamica, l'analisi statica, le forze di contatto e l'output di calcolo;
- 3. Il post-processing dove si sono osservate le simulazioni effettuate e si sono analizzati i risultati ottenuti tramite grafici significativi.

# **2** ARCHITETTURA EMA

La gestione dell'assetto di un velivolo viene effettuata tramite l'applicazione di forze e momenti sull'intera struttura e le seguenti azioni dinamiche possono essere generate o tramite la spinta dei motori oppure tramite la variazione della geometria del velivolo sfruttando le superfici mobili presenti, le quali vengono gestite dai comandi di volo che permettono al pilota di eseguire le manovre che desidera. Esistono due tipologie di comandi di volo:

Comandi di volo primari, presenti in Figura 2.1, che corrispondono a quelle superfici tramite le quali è possibile variare l'assetto e la traiettoria del velivolo gestendo i tre gradi di libertà rotazionali principali del sistema che sono rollio, beccheggio ed imbardata



Figura 2.1 – Comandi primari di volo [2]

Comandi di volo secondari, presenti in Figura 2.2, che corrispondono a tutte quelle superfici che hanno il compito di gestire ed ottimizzare le manovre dei comandi primari, migliorando l'aereodinamica del velivolo a seconda delle condizioni di volo



Figura 2.2 – Comandi secondari di volo [2]

Generalmente per la gestione della posizione delle superfici mobili, si utilizzano dei servocomandi, in particolare il più tradizionale è quello idraulico. Un esempio classico di questa tipologia di attuazione è presente nella Figura 2.3:



Figura 2.3 – Schema di un'attuazione di tipo idraulico [2]

Il servocomando idraulico presente in Figura 2.3 è un tradizionale sistema retroazionato in posizione dove l'input viene fornito dal pilota attraverso o un comando elettrico o un comando meccanico e l'output è la posizione del cilindro che comporta l'orientazione della superficie mobile. Nel seguente schema di attuazione, il comando meccanico impone la posizione del cassetto della valvola che permette di indirizzare l'olio dell'impianto in una determinata camera del cilindro, generando una differenza di pressione che crea una forza di spinta sul pistone e fa muovere la superficie di comando. È una soluzione molto efficiente per il suo ridotto rapporto peso-potenza e la bassa probabilità di grippaggio che comporta una buona affidabilità, ma allo stesso tempo presenta degli svantaggi legati al peso dell'intero circuito idraulico e alle perdite idrauliche presenti nel sistema. Una soluzione elettromeccanica rispetto alla tradizionale idraulica presenta dei vantaggi che sono un minore peso del servosistema da cui ne deriva un consumo minore di carburante e quindi un minor impatto ambientale, una maggiore facilità e flessibilità a livello di progettazione del velivolo e dei costi minori. Ovviamente l'utilizzo di EMA comporta anche degli svantaggi come un rapporto peso/potenza minore, una difficoltà maggiore nel raffreddare i componenti del servosistema e una minore affidabilità legata alla maggiore probabilità di grippaggio.

Nel seguente Capitolo 2 si analizza come è composto l'attuazione elettromeccanica d'interesse e si osserva la collocazione della vite a ricircolo di sfere nel sistema. L'architettura EMA, di cui una possibile soluzione può essere quella presente nella Figura 2.4, è composta dai seguenti elementi:

- Un motore brushless DC (BLDC)
- Un riduttore epicicloidale ad ingranaggi
- Una vite a ricircolo di sfere
- Un Motor Drive Electronics (MDE)
- Un Electronic Control Unit (ECU)
- Un Resolver
- Un Linear Variable Displacement Transducer (LVDT)
- Una frizione
- Dei component secondari quali cuscinetti, guide lineari, freni



Figura 2.4 – Schema di un'attuazione elettromeccanica [3]

Ogni EMA presenta un'unità di controllo elettronica (ECU) che rappresenta il fulcro di tutto il sistema di attuazione: esso elabora i dati in input che corrispondono al set di posizione che viene imposto dal pilota, elabora i feedback di posizione e di velocità che arrivano dai trasduttori presenti ed elabora il segnale di corrente da fornire al motore brushless DC; inoltre fornisce tutte le informazioni di stato dell'EMA al computer di controllo di volo. Normalmente il loop di posizione e velocità ha un controller digitale mentre il loop di corrente è completamente analogico. Si sottolinea che l'LVDT è collocato nella madrevite della vite a ricircolo di sfere mentre il resolver è collocato nella parte ad alta velocità del riduttore. Il comando di corrente generato dall'ECU viene mandato in input nel Motor Drive Electronic (MDE) il quale aziona il circuito di controllo della corrente e genera il comando da fornire al motore brushless a corrente continua (BLDC). La coppia d'uscita del motore viene fornita tramite un riduttore epicicloidale alla vite a ricircolo di sfere che converte il moto rotatorio in moto lineare. L'utilizzo del riduttore serve per generare delle coppie elevate rispetto a quelle del motore e permette di soddisfare le richieste di posizione del sistema. La

forza esterna presente sulla madrevite è la risultante di tutte le componenti di forze che agiscono sulla superficie di controllo come carichi aereodinamici, fenomeni di attrito, azioni inerziali e carichi esercitati sulla stessa superficie da altri attuatori. Considerando che sulla superficie di controllo agiscono più attuatori, possono verificarsi situazioni indesiderate come condizioni di contrasto tra le varie forze esercitate dai vari elementi che potrebbe portare ad uno stallo del sistema mantenendo l'ultima posizione imposta; per ovviare il seguente problema sono state generate diverse soluzioni. Generalmente alcune ragioni che comportano fenomeni di inceppamento sono l'usura, contraccolpi, fenomeni vibratori, aumenti indesiderati di attriti. La frizione presente è un elemento di sicurezza che permette di disconnettere il motore dagli elementi meccanici presenti in caso di guasto.

# 2.1 Motore Brushless a corrente continua (BLDC)

Questa tipologia di motore, a differenza dei classici motori a corrente continua, non presenta spazzole ma bensì presentano un commutatore elettrico, offrendo ottime prestazioni in quanto, tramite questa impostazione del sistema, non sono presenti fonti di usura e di perdita di potenza che sono normalmente presenti nei classici motori a corrente continua a causa per l'appunto delle spazzole. Inoltre, questa tipologia permette di ottenere ulteriori vantaggi come per esempio delle migliori caratteristiche di velocità-coppia, una risposta dinamica migliore e delle condizioni di funzionamento meno rumorose. Il BLDC si presta molto bene nelle applicazioni aereonautiche in quanto questi motori sono molto compatti e leggeri e permettono di ottenere potenze elevate e ciò è fondamentale in questo campo. Un sistema del genere però, rispetto ai tradizionali sistemi DC, presenta delle complessità di realizzazione che sono causate dalla presenza di una centralina elettronica che gestisca il corretto funzionamento del motore e questo porta a costi più elevati; inoltre il motore brushless a corrente continua necessita di un microcontrollore che sfrutti i segnali di posizione del rotore forniti dai sensori ad effetto Hall per eccitare in maniera corretta le varie bobine presenti nel sistema.

A differenza dei motori classici a corrente continua che presentano lo statore costituito da magneti permanenti e il rotore composto da avvolgimenti, i motori brushless presentano una struttura completamente opposta. Lo statore è composto da lamierini d'acciaio fessurati assialmente che fungono da sede per gli avvolgimenti di rame che percorrono l'intero perimetro interno. Invece il rotore è caratterizzato dalla presenza di magneti permanenti con un numero di coppie polari N-S (Nord-Sud) che va da 2 a 8. Aumentare il numero di coppie di magneti permette di ottenere coppie

maggiori e di ridurre il "ripple di coppia" rendendo la coppia di output più stabile a fronte però di una maggiore complessità del controllo, costi maggiori e riduzione della velocità massima.

# 2.2 Riduttore epicicloidale

Questa tipologia di rotismo è caratterizzata dalla presenza di ingranaggi ad assi mobili definiti come satelliti e da altri ingranaggi ad assi fissi definiti come solare e corona. Gli assi dei satelliti sono collegati direttamente al portatreno, il quale li porta in rotazione, quindi le ruote satellitari ruotano intorno al loro asse e ruotano intorno all'asse del portatreno, che coincide con l'asse del solare. Rispetto ai rotismi ordinari, è possibile ottenere rapporti di trasmissione molto elevati e un sistema molto compatto. Ovviamente più si cerca di ottenere un rapporto di trasmissione elevato, più le prestazioni sono meno performanti in quanto il rendimento decresce. L'utilizzo del riduttore in questo sistema permette di utilizzare a valle motori più veloci e con coppie più basse, in maniera da ridurre costi, pesi e ingombri sul BLDC.

#### 2.3 Frizione

Le frizioni generalmente vengono utilizzate per trasmettere una coppia tra due alberi coassiali. Il principio di funzionamento è quello di interfacciare una o più superfici tra di loro dove alcune sono solidali con l'albero di input, altre con quello di output. Nel tradizionale funzionamento, quando avviene l'innesto della frizione, le due regioni a contatto inizialmente strisciano tra di loro, generando perdite di energia meccanica trasformata in calore a causa dell'attrito, successivamente avviene la fase di aderenza tra le due regioni a contatto e i due alberi ruotano alla stessa velocità.

Nel sistema di attuazione preso in esame, la frizione non viene utilizzata per trasmettere il moto ma bensì per disconnettere i due alberi. Ciò viene fatto per ragioni di sicurezza per scollegare il motore dal sistema in caso di guasto e questa operazione viene gestita e imposta da comandi elettrici generati dalla logica di controllo. La presenza della frizione migliora l'affidabilità del sistema.

#### 2.4 Sistema di controllo

Il sistema di controllo è composto dall'ECU e dal MDE, tramite i quali si effettuano tre loop annidati di controllo della posizione, della velocità e della corrente. I primi due anelli elencati sono effettuati dall'ECU quindi sono controlli digitali, pertanto è necessario che siano presenti dei moduli di conversione analogico-digitale che permetta all'unità relativa di elaborare i segnali analogici provenienti dai trasduttori presenti. L'anello di corrente è interamente gestito dall'MDE ed è di tipo analogico. I feedback analogici presenti sono quello della posizione fornito dall'LVDT che è collocato sulla madrevite mentre quello di velocità è generato dal resolver che è collocato a monte del riduttore in maniera da rilevare anche le velocità più basse del motore, dove l'LVDT purtroppo non rileva nulla in quanto corrisponderebbe ad uno spostamento lineare troppo piccolo.

#### 2.5 Vite a ricircolo di sfere

Le viti a ricircolo di sfere sono una particolare tipologia di trasmissione, capace di trasformare il moto rotatorio in un moto lineare o viceversa, con la differenza rispetto ad una classica vite trapezoidale di perdite meccaniche minori grazie alla presenza delle sfere all'interno della pista che rotolano. La presenza dell'attrito volvente è il grande vantaggio di questo sistema, che garantisce ottime prestazioni, sia per l'elevata efficienza del sistema sia per la buona precisione nel posizionamento. La seguente trasmissione è composta dai seguenti elementi:

- Un albero filettato esternamente, che corrisponde alla vite, che presenta un grado di libertà di rotazione intorno al suo asse
- Una madrevite filettata internamente, denominata chiocciola che presenta un grado di libertà di traslazione lungo l'asse della vite
- Le sfere che si interpongono tra vite e madrevite e percorrono il percorso elicoidale e quello di ricircolo; esse sono il fulcro del sistema, in quanto permettono l'effettiva trasformazione del moto, da rotazionale a lineare o viceversa

Un deviatore che permette alle sfere giunte alla fine del tratto elicoidale di entrare nel percorso di ricircolo in maniera da poter ritornare al punto di partenza del circuito



Figura 2.5 – Schema di una vite a ricircolo di sfere

Le viti a ricircolo di sfere si possono distinguere in 2 famiglie, che caratterizzano il tipo di applicazione:

- Le viti di posizionamento che sono quei sistemi caratterizzati da un'elevata rigidezza, da un ottimale precisione di posizionamento e da un elevato livello di ripetibilità di posizionamento
- Le viti di trasporto che sono quella tipologia che permette la movimentazione di un determinato carico senza una particolare richiesta di precisione nel posizionamento.

Nei sistemi EMA si utilizzano le viti di posizionamento, in quanto è richiesta una buona precisione per poter effettuare in maniera corretta le operazioni di controllo all'interno del servosistema.

#### 2.5.1 Geometria del profilo

La geometria del profilo del filetto influisce molto sulle prestazioni della trasmissione, in quanto influisce sulle condizioni di contatto e sullo strisciamento tra sfera e pista. Il profilo dei filetti può essere di due tipologie:

Profilo circolare che è caratterizzato dal fatto che i due fianchi della pista fanno parte dello stesso arco e i due archi presentano lo stesso centro di curvatura; con questa tipologia di profilo si manifestano complesse condizioni di pressione hertziana ed inoltre sono presenti fenomeni di strisciamento importanti tra sfera e pista che comportano fenomeni di usura, generazione di perdite di calore e conseguentemente una diminuzione del rendimento e della durata complessiva del sistema; sistemi vite-madrevite che utilizzano questa tipologia di profilo

vengono utilizzati in campi applicativi dove è richiesto carico elevato e velocità lineari molto basse

Profilo gotico che è caratterizzato invece dal fatto che i due fianchi della pista sono due archi distinti e presentano centri di curvatura differenti e una discontinuità nel punto d'intersezione; questa tipologia di profilo permette di ottimizzare le pressioni di contatto e di ridurre lo strisciamento tra sfera e pista, il che permette di ottenere minor usura, minor generazione di perdite di calore, un miglior rendimento del sistema e durata globale del sistema



Figura 2.6 – Rappresentazione delle due tipologie di profili per le piste vi vite e madrevite: a sinistra è presente il profilo circolare mentre a destra è presente il profilo gotico

#### 2.5.2 Ricircolo delle sfere

Il ricircolo delle sfere è un elemento essenziale del sistema che permette di "riutilizzare" le sfere che hanno già percorso tutto il tratto elicoidale della pista. Il ricircolo viene realizzato generalmente tramite due possibili tipologie di dispositivi:

- Ricircolo interno alla madrevite tramite l'utilizzo di un liner radiale che è comunemente utilizzato per filetti con passo minimo di 3 mm e non superiore a 20 mm ed è inserito radialmente nel corpo della madrevite limitando il percorso delle sfere ad ogni singolo giro; questa soluzione è molto utile quando sono presenti elevate velocità angolari
- Ricircolo interno mediante un deflettore frontale che è generalmente utilizzato per filetti con passo maggiore di 10 mm e per i filetti a più principi; questa tipologia è caratterizzata da un foro lungo la direzionale assiale della madrevite che congiunge i due deflettori che riportano

le sfere dal foro al filetto e viceversa; questa soluzione consente di avere una buona flessibilità nella scelta del passo ma, oltre al maggiore ingombro della madrevite, l'asimmetria generata dal percorso di ricircolo limita la velocità angolare massima ammessa

Ricircolo esterno dove le sfere percorrono la pista di rotolamento nel suo intero sviluppo e vengono fatte ricircolare tramite un percorso esterno alla pista stessa caratterizzato da un tubetto deviatore in acciaio inox



Figura 2.7 – Rappresentazione delle due principali tipologie di ricircolo delle sfere: a sinistra è presente il ricircolo interno mediante il liner radiale mentre a destra è presente il ricircolo interno mediante deflettore frontale

#### 2.5.3 Vite a ricircolo di sfere scelta per il modello

Per generare il modello in ambiente ADAMS, è stata scelta una vite a ricircolo di sfere con profilo gotico e con percorso di ricircolo interno con deflettori frontali. L'utilizzo della seguente vite è solo un riferimento per poter creare l'intero modello nel software desiderato. Se eventualmente l'utente volesse cambiare tipologia di sistema, sarà necessario effettuare nuovamente alcuni calcoli legati principalmente alla posizione delle sfere all'interno della pista.

Le caratteristiche geometriche principali del sistema vite-madrevite analizzato sono le seguenti:

- >  $D_{vite} = 40 \ mm$  che corrisponde al diametro nominale della vite stessa
- >  $D_{madrevite} = 63 mm$  che corrisponde al diametro esterno della madrevite
- >  $D_m = 41,07 mm$  che corrisponde al diametro dell'elica nominale
- >  $P_h = 20 mm$  che corrisponde al passo dell'elica
- $\sim \alpha_{elica} = 8,81 \, deg$  che corrisponde all'angolo dell'elica in corrispondenza del diametro  $D_m$
- >  $L_{vite} = 1000 mm$  che corrisponde alla lunghezza della vite
- >  $r_{b0} = 3,175 \ mm$  che corrisponde al raggio delle sfere

- >  $r_{s0} = 3,556 mm$  che corrisponde al raggio di curvatura della pista nel piano normale al filetto presente sia sulla vite che sulla madrevite
- >  $n_{avv} = 2,75$  che corrisponde al numero di avvolgimenti dell'elica presenti all'interno della madrevite
- > L = H = 0,27 mm che corrispondono agli offset lungo le due direzioni nel piano normale al filetto dei centri di curvatura degli archi rispetto al centro di un arco di un profilo circolare della pista; questi parametri sono presenti in quanto si sta analizzando un sistema ad arco gotico

Nel Capitolo 3 vengono illustrati passo a passo i parametri utilizzati per i vari calcoli effettuati in maniera da rendere più comprensibile l'iter di calcolo, richiamando anche quelli qui sopra citati contestualizzandoli e definendo quelli secondari.

#### **3** DEFINIZIONE DEI PARAMETRI PRINCIPALI DEL MODELLO

Al fine di realizzare il modello, è necessario avere i parametri principali per definire e verificare il sistema vite-madrevite in maniera completa a livello dinamico. I parametri necessari sono la posizione delle sfere nello spazio, le rigidezze di contatto tra vite-sfera, madrevite sfera e sfera-sfera e l'angolo di contatto: il primo serve per collocare correttamente le sfere all'interno della pista, il secondo per definire le forze di contatto per vincolare la madrevite alla vite e il terzo per verificare che il valore dell'angolo di contatto reale sia circa il valore ideale. Si sottolinea che i seguenti elementi sono stati calcolati in ambiente MATLAB.

### 3.1 Centri delle sfere

Al fine di definire correttamente la posizione delle sfere all'interno della pista, è risultato necessario calcolare la collocazione nello spazio di ogni singola sfera, in maniera da poterle inserire all'interno del modello in ADAMS. Per farlo, si sono calcolati i vari tratti del percorso che le sfere devono compiere all'interno della madrevite; osservando la Figura 3.1, la traiettoria è composta dai seguenti tratti:

- un tratto elicoidale (1) che corrisponde al percorso in cui le sfere sono in presa e trasmettono il moto alla madrevite
- due tratti rettilinei tangenti all'elica (2) (6) che servono uno per far deviare le sfere dal percorso elicoidale e farle entrare all'interno del sistema di ricircolo, l'altro per inserire le sfere all'interno del percorso elicoidale
- due raccordi (3) (5) che servono per veicolare le sfere verso il percorso rettilineo lungo la direzionale assiale della madrevite
- un tratto rettilineo (6) che permette l'effettivo ricircolo delle sfere, chiudendo così la traiettoria spaziale delle sfere



Figura 3.1 – Traiettoria con i relativi tratti elencati

Inizialmente, si definiscono i seguenti sistemi di riferimento:

- Il sistema oxyz fisso nello spazio con l'asse y coincidente con l'asse della vite che corrisponde al sistema di riferimento principale presente in ADAMS
- > Il sistema ox'y'z' fisso nello spazio con l'asse z coincidente con l'asse della vite
- Il sistema otnb mobile e solidale con il centro della sfera, il quale ha l'asse t tangente alla traiettoria elicoidale, l'asse n che è lungo la direzione che interseca l'asse della vite e l'asse b che è il prodotto vettoriale dei due versori precedentemente descritti; esso viene definito come sistema di Frenet-Serret

La matrice  $T_0$  che permette il passaggio dal sistema di riferimento ox'y'z' a quello oxyz è la seguente:

$$\begin{cases} x \\ y \\ z \end{cases} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{cases} x' \\ y' \\ z' \end{cases} = T_0 \begin{cases} x' \\ y' \\ z' \end{cases}$$
(3.1)

La matrice  $T_5$  che permette il passaggio dal sistema di riferimento *otnb* a quello ox'y'z' è la seguente:

$$\begin{cases} x'\\y'\\z' \end{cases} = \begin{bmatrix} -\cos\alpha_{elica}\sin\theta & -\cos\theta & \sin\alpha_{elica}\sin\theta\\ \cos\alpha_{elica}\cos\theta & -\sin\theta & -\sin\alpha_{elica}\cos\theta\\ \sin\alpha_{elica} & 0 & \cos\alpha_{elica} \end{bmatrix} \begin{cases} t\\n\\b \end{cases} = T_5 \begin{cases} t\\n\\b \end{cases}$$
(3.2)

Nella Figura 3.2 sono rappresentati i 3 sistemi di riferimento:



Figura 3.2 – Sistemi di riferimento

I parametri utilizzati nel calcolo della traiettoria sono i seguenti:

- ▶  $R_m = \frac{D_m}{2} = 20,53 \text{ mm}$  che corrisponde al raggio dell'elica nominale
- >  $\alpha_{elica} = 8,81 \ deg$  che corrisponde all'angolo dell'elica in corrispondenza del diametro  $D_m$
- $R = 27,94 \, mm$  che corrisponde al modulo del vettore che identifica l'asse del foro assiale della madrevite nel piano xz nel sistema di riferimento oxyz
- >  $\alpha = 7,3 \ deg \ e \ \beta = 2,7 \ deg$  che sono gli angoli che permettono di identificare la posizione del foro assiale della madrevite nel piano *xz* nel sistema di riferimento *oxyz*
- >  $\theta$  che è l'angolo di azimut misurato nel sistema ox'y'z', che varia a seconda del punto considerato nel tratto elicoidale

- >  $n_{avv} = 2,75$  che corrisponde al numero di avvolgimenti dell'elica presenti all'interno della madrevite
- >  $r_F = 5 mm$  che rappresenta il raggio dei raccordi
- >  $y_A = 64,59 mm$  che rappresenta l'offset del punto A nel sistema di riferimento *oxyz* lungo la direzione y (assunto in base alla costruzione del modello CAD)



Figura 3.3 – Vista della traiettoria sul piano xz del sistema di riferimento di ADAMS

Considerando la Figura 3.4, per avere tutti i dati per definire la traiettoria delle sfere, è necessario definire i seguenti parametri:



Figura 3.4 – Schema utile al calcolo di parametri richiesti per la traiettoria

$$R_1 = \sqrt{R^2 - R_m^2}$$
(3.3)

$$L_{rett} = \frac{R_1 - r_F(1 + \sin \alpha_{elica})}{\cos \alpha_{elica}}$$
(3.4)

Da qui è possibile definire la traiettoria delle sfere all'interno della madrevite effettuando dei calcoli principalmente geometrici. I tratti sono così definiti:

> 1) Il tratto elicoidale viene definito tramite l'angolo di azimut  $\theta$  che varia nell'intervallo  $[-\alpha, 2\pi n_{avv} + \beta]$ ; le coordinate nello spazio secondo nel sistema di riferimento *oxyz* sono le seguenti:

$$\begin{cases} x = R_m \cos \theta \\ y = R_m (\theta + \alpha) \tan \alpha_{elica} + y_A \\ z = -R_m \sin \theta \end{cases}$$
(3.5)

Queste equazioni corrispondono alle coordinate di un'elica con asse parallelo alla direzione y del sistema di riferimento *oxyz* 

> 2) Il tratto rettilineo tangente in B, definito come  $(x_B, y_B, z_B) = (x(\theta), y(\theta), z(\theta))$  dove  $\theta = 2\pi n_{avv} + \beta$ ; per farlo è necessario definire il vettore tangente in B nel sistema di riferimento *otnb* che è il seguente

$$\underline{t_B} = [L_{rett}, 0, 0]$$

Successivamente è possibile calcolare le coordinate del tratto rettilineo nel sistema di riferimento *oxyz* tramite l'utilizzo della matrice di rotazione nel seguente modo:

$$\begin{cases} x \\ y \\ z \end{cases} = T_0 T_5 \begin{cases} t \\ n \\ b \end{cases}$$
 (3.6)

3) Il tratto di raccordo tangente al tratto 2) si definisce identificando innanzitutto il centro di curvatura rispetto al sistema di riferimento *otnb* collocato nel punto B che è il seguente:

$$O_3 = [L_{rett}, 0, -r_F]$$

Da qui si definisce la curva di raccordo nelle coordinate di Frenet-Serret, suddividendola in 2 tratti:

1° tratto 
$$\begin{cases} t_{31} = [L_{rett}, L_{rett} + r_F] \\ b_{31} = b_{0_3} - \sqrt{r_F^2 - (t_{31} - t_{0_3})^2} \\ n_{31} = 0 \end{cases}$$
(3.7)

2° tratto 
$$\begin{cases} t_{32} = [L_{rett} + r_F \cos \alpha_{elica}, L_{rett} + r_F] \\ b_{32} = b_{0_3} - \sqrt{r_F^2 - (t_{32} - t_{0_3})^2} \\ n_{32} = 0 \end{cases}$$
(3.8)



Figura 3.5 – Schema del raccordo presente nel tratto 3)

Successivamente per calcolare le coordinate nel sistema oxyz si utilizza il sistema di equazioni (3.6).

- 4) Il tratto parallelo all'asse y si calcola utilizzando il punto C che fa parte del tratto 3 e il punto C' che fa parte del tratto 5 che verrà descritto successivamente; per calcolarlo basta mantenere costanti le coordinate x e z e far variare la y dalla coordinata di C fino al punto C'
- 5) Il tratto di raccordo tangente al tratto 6) si definisce identificando innanzitutto il centro di curvatura rispetto al sistema di riferimento *otnb* collocato nel punto A che è il seguente:

$$O_5 = \left[-L_{rett}, 0, r_F\right]$$

Da qui si definisce la curva di raccordo nelle coordinate di Frenet-Serret, suddividendola in 2 tratti:

1° tratto 
$$\begin{cases} t_{51} = [-(L_{rett} + r_F), -L_{rett}] \\ b_{51} = b_{0_5} - \sqrt{r_F^2 - (t_{51} - t_{0_5})^2} \\ n_{51} = 0 \end{cases}$$
(3.9)

2° tratto 
$$\begin{cases} t_{52} = \left[ -(L_{rett} + r_F \cos \alpha_{elica}), -(L_{rett} + r_F) \right] \\ b_{52} = b_{0_5} + \sqrt{r_F^2 - (t_{52} - t_{0_5})^2} \\ n_{52} = 0 \end{cases}$$
(3.10)



Figura 3.6 - Schema del raccordo presente nel tratto 5)

Successivamente per calcolare le coordinate nel sistema *oxyz* si utilizza il sistema di equazioni (3.6).

▶ 6) Il tratto rettilineo tangente in A, definito come  $(x_A, y_A, z_A) = (x(\theta), y(\theta), z(\theta))$  dove  $\theta = -\alpha$ ; per farlo è necessario definire il vettore tangente in A nel sistema di riferimento *otnb* che è il seguente

$$\underline{t_A} = [-L_{rett}, 0, 0]$$

Successivamente per calcolare le coordinate nel sistema *oxyz* si utilizza il sistema di equazioni (3.6).

Tramite questa definizione dei tratti, è possibile definire l'intero percorso che le sfere intraprendono durante il funzionamento della vite a ricircolo di sfere. Successivamente, all'interno dell'ambiente MATLAB, attraverso un ciclo for dove si impone che i centri delle sfere debbano essere distanziati nello spazio di una quantità pari al diametro delle sfere e che i centri appartengano al percorso, si è calcolata la collocazione delle sfere all'interno della pista, come è possibile vedere in Figura 3.7.



Figura 3.7 – Posizione delle sfere nello spazio all'interno della pista

# 3.2 Rigidezza di contatto

Per descrivere in maniera completa la dinamica del sistema vite a ricircolo di sfere, risulta necessario definire le forze di contatto che si instaurano all'interno della trasmissione in esame. Esse si generano ogni qual volta che si presenta una compenetrazione tra corpi, pertanto sono di tre tipologie ossia quella vite-sfera, madrevite-sfera e sfera-sfera. Il parametro fondamentale che permetterà nel Capitolo 4 di definire i contatti tra i corpi in ADAMS è la rigidezza di contatto, la quale è stata calcolata tramite la teoria di Hertz.

La teoria del contatto di Hertz prevede le seguenti ipotesi:

- > Le superfici dei due corpi a contatto sono descrivibili tramite polinomi di secondo grado
- > Le aree di contatto sono idealmente lisce e quindi non sono presenti irregolarità delle superfici
- Il materiale è isotropo, omogeneo ed elastico e le tensioni e le deformazioni sono all'interno del campo elastico e soddisfano la legge di Hooke
- ▶ Le tensioni sono nulle ad elevate distanze dalla zona di contatto
- Il carico è perpendicolare al piano tangente alle due superfici dei corpi, pertanto gli sforzi di taglio sono nulli sia dentro che fuori dalla zona di taglio
- > Il carico presente nella zona di contatto corrisponde alla forza di spinta che preme i due corpi
- L'area di contatto presenta dimensioni molto piccole rispetto alle ai raggi di curvature dei corpi a contatto sotto carico
- > La superficie di contatto degenera in un punto quando è assente la forza esterna

Nella Figura 3.8 sono rappresentate due superfici generiche a contatto rappresentate con due sistemi di riferimento che sono per il corpo I  $ox_I y_I z_I$  mentre per il corpo II  $ox_{II} y_{II} z_{II}$ ; come si osserva nella Figura 3.8, gli assi  $z_I$  e  $z_{II}$  non sono rappresentati, ma è rappresentato un unico asse z in quanto i due assi sono collineari e quindi l'unica differenza dei due è legata al segno tramite il quale si legge la compenetrazione tra i due corpi. L'angolo  $\omega$  rappresenta la differenza di orientazione tra il piano  $(0, x_I, z)$  e il piano  $(0, x_{II}, z)$ .

I parametri utilizzati nel calcolo delle rigidezze di contatto sono i seguenti:

- ▶  $R_m = \frac{D_m}{2} = 20,53 \text{ mm}$  che corrisponde al raggio dell'elica nominale
- >  $r_{b0} = 3,175 mm$  che corrisponde al raggio delle sfere
- >  $r_{s0} = 3,556 mm$  che corrisponde al raggio di curvatura della pista nel piano normale al filetto presente sia sulla vite che sulla madrevite
- > E = 208 GPa che corrisponde al modulo di Young dell'acciaio
- > v = 0,29 che corrisponde al coefficiente di Poisson dell'acciaio
- >  $\beta_{c,i} = 45 \ deg$  che corrisponde all'angolo di contatto nominale, ossia all'angolo ideale di contatto che si è assunto per calcolare le rigidezze di contatto

Le superfici a contatto sono definite tramite le loro curvature principali lungo le direzioni perpendicolari x e y definite secondo il sistema di riferimento del corpo considerato. Si sottolinea che le curvature sono definite come  $\rho = 1/r$  e che esse assumono un valore negativo nel caso in cui il centro di curvatura si trovi esternamente al corpo.



Figura 3.8 – Rappresentazione dei corpi a contatto [4]

Per le tre tipologie di forze di contatto, le curvature sono state definite nella seguente maniera:

Contatto sfera-vite:

$$\rho_{Ix} = \frac{1}{r_{b0}} \\
\rho_{Iy} = \frac{1}{r_{b0}} \\
\rho_{IIx} = -\frac{1}{r_{s0}} \\
\rho_{IIy} = \frac{\cos \beta_{c,i}}{R_m - r_{b0} \cos \beta_{c,i}}$$
(3.11)

Contatto sfera-madrevite

$$\begin{cases}
\rho_{Ix} = \frac{1}{r_{b0}} \\
\rho_{Iy} = \frac{1}{r_{b0}} \\
\rho_{IIx} = -\frac{1}{r_{s0}} \\
\rho_{IIy} = -\frac{\cos \beta_{c,i}}{R_m + r_{b0} \cos \beta_{c,i}}
\end{cases}$$
(3.12)

Contatto sfera-sfera

$$\begin{cases}
\rho_{Ix} = \frac{1}{r_{b0}} \\
\rho_{Iy} = \frac{1}{r_{b0}} \\
\rho_{IIx} = \frac{1}{r_{b0}} \\
\rho_{IIy} = \frac{1}{r_{b0}}
\end{cases}$$
(3.13)

Si sottolinea che le successive equazioni che verranno illustrate verranno applicate per ogni tipologia di contatto, sfruttando le curvature calcolate con le equazioni (3.11), (3.12) e (3.13). Si evidenzia inoltre che nelle formule (3.11) e (3.12) per il calcolo di  $\rho_{IIy}$ , è stata utilizzata la formula descritta in [5].

Facendo riferimento agli studi svolti in precedenza sul contatto hertziano [4], la geometria di contatto è definita tramite la differenza e la somma tra le varie curvature nel seguente modo:

$$\sum \rho = \rho_{Ix} + \rho_{Iy} + \rho_{IIx} + \rho_{IIy}$$
(3.14)

$$F_{\rho} = -\frac{\sqrt{(\rho_{Ix} - \rho_{Iy})^{2} + (\rho_{IIx} - \rho_{IIy})^{2} + 2(\rho_{Ix} - \rho_{Iy})(\rho_{IIx} - \rho_{IIy})\cos 2\omega}}{\Sigma \rho}$$
(3.15)

Le curvature relative sono definite nel seguente modo:

$$A = \frac{\sum \rho}{4} \left( 1 + F_{\rho} \right) \tag{3.16}$$

$$B = \frac{\sum \rho}{4} \left( 1 - F_{\rho} \right) \tag{3.17}$$

L'area di contatto si suppone sia un'ellisse dove a e b corrispondono ai semiassi; generalmente è difficile capire quale dei due sia il semiasse maggiore. Un parametro che risulta fondamentale per il calcolo della rigidezza di contatto è il parametro di ellitticità che è definito come  $\kappa = a/b$ . Il seguente elemento è necessario per il calcolo degli integrali ellittici di prima specie  $\mathcal{F}(\kappa)$  e di seconda specie  $\mathcal{E}(\kappa)$ , definiti nella seguente maniera:

$$\mathcal{F}(\kappa) = \int_0^{\frac{\pi}{2}} \left[ 1 - \left( 1 - \frac{1}{\kappa^2} \right) \sin^2 \phi \right]^{-\frac{1}{2}} d\phi$$
(3.18)

$$\mathcal{E}(\kappa) = \int_{0}^{\frac{\pi}{2}} \left[ 1 - \left( 1 - \frac{1}{\kappa^2} \right) \sin^2 \phi \right]^{\frac{1}{2}} d\phi$$
(3.19)

Per il calcolo di questi integrali, si ricorre ad una soluzione numerica che fornisce una approssimazione con delle espressioni esplicite che sono le seguenti:

$$\mathcal{F}(\kappa) = (\alpha_0 + \alpha_1 m + \alpha_2 m^2) - (\alpha_3 + \alpha_4 m + \alpha_5 m^2) \ln m$$
(3.20)

$$\mathcal{E}(\kappa) = (\beta_0 + \beta_1 m + \beta_2 m^2) - (\beta_3 m + \beta_4 m^2) \ln m$$
(3.21)

| $lpha_0$   | 1,3862944 | $lpha_4$   | 0,1213478 | $\beta_2$ | 0,1077812  |
|------------|-----------|------------|-----------|-----------|------------|
| $\alpha_1$ | 0,1119723 | $\alpha_5$ | 0,0288729 | $\beta_3$ | 0,2452727  |
| α2         | 0,0725296 | $eta_0$    | 1         | $eta_4$   | 0,04124496 |
| α3         | 0,5       | $eta_1$    | 0,4630151 |           |            |

Tabella 3.1 – Coefficienti della approssimazione polinomiale degli integrali completi di prima e seconda specie [4]

Dove il parametro m è definito come  $m = 1/\kappa^2$  e assume un valore compreso tra 0 e 1 in quanto  $\kappa \ge$ 1. Da qui si può osservare come sia determinare il valore del parametro di ellitticità in maniera da avere tutti i dati necessari per calcolare la rigidezza di contatto; esso è definito tramite la seguente espressione:

$$\kappa = (B/A)^{\gamma} \tag{3.22}$$

Per poter calcolare  $\kappa$ , si sfrutta una espressione numerica dell'esponente  $\gamma$ , utilizzando il parametro  $X = \log_{10}(B/A)$  nel seguente modo:

$$\gamma = \frac{2}{3} \left( \frac{1 + \mu_1 X^2 + \mu_2 X^4 + \mu_3 X^6 + \mu_4 X^8}{1 + \mu_5 X^2 + \mu_6 X^4 + \mu_7 X^6 + \mu_8 X^8} \right)$$
(3.23)

| $\mu_1$ | 0,40227436                   | $\mu_5$ | 0,42678878                |
|---------|------------------------------|---------|---------------------------|
| $\mu_2$ | $3,7491752 \cdot 10^{-2}$    | $\mu_6$ | $4,2605401 \cdot 10^{-2}$ |
| $\mu_3$ | 7,4855761 · 10 <sup>-4</sup> | $\mu_7$ | $9,0786922 \cdot 10^{-4}$ |
| $\mu_4$ | $2,1667028 \cdot 10^{-6}$    | $\mu_8$ | $2,7868927 \cdot 10^{-6}$ |

Tabella 3.2 – Coefficienti per il calcolo del parametro di ellitticità [4]

A questo punto è possibile definire la rigidezza di contatto sfruttando il parametro di ellitticità e gli integrali ellittici di prima e seconda specie, calcolando innanzitutto il parametro  $\delta^*$  funzione di  $\kappa$  e il modulo di Young equivalente  $E^*$ , utili per il calcolo:

$$\delta^* = \frac{2\mathcal{F}(\kappa)}{\pi} \left(\frac{\pi}{2\kappa^2 \mathcal{E}(\kappa)}\right)^{1/3} \tag{3.24}$$

$$\frac{1}{E^*} = \frac{1 - \nu_I^2}{E_I} + \frac{1 - \nu_{II}^2}{E_{II}}$$
(3.25)

Si sottolinea che  $E_I$  e  $v_I$  corrispondono alle caratteristiche del materiale della superficie I mentre  $E_{II}$  e  $v_{II}$  sono quelli invece della superficie II. Successivamente è possibile definire la rigidezza di contatto nel seguente modo:

$$k_H = \frac{2^{\frac{5}{2}}}{3} \frac{E^*}{(\delta^*)^{3/2} (A+B)^{1/2}}$$
(3.26)

La forza di contatto che si genera quando è presente una penetrazione dei due corpi, è esprimibile nel seguente modo:

$$F_{contatto} = k_H \delta^{3/2} \tag{3.27}$$

Dove  $\delta$  rappresenta la compenetrazione tra i due corpi. Da qui si può osservare che la rigidezza di contatto ha come unità di misura  $[N/mm^{3/2}]$ . Come detto precedentemente, è stata definita una rigidezza  $k_H$  per ogni tipologia di forza di contatto ed i risultati ottenuti sono i seguenti:

| Tipologia di contatto | $k_H \left[ N/mm^{3/2} \right]$ |
|-----------------------|---------------------------------|
| Sfera – Vite          | $1,91 \cdot 10^{5}$             |
| Sfera – Madrevite     | 5,29 · 10 <sup>5</sup>          |
| Sfera – Sfera         | 5,10 · 10 <sup>5</sup>          |

Tabella 3.3 – Rigidezze di contatto utilizzate nel modello ADAMS

#### 3.3 Angolo di contatto

Come definito precedentemente nel Capitolo 3.2, l'angolo  $\beta$  di contatto idealmente assume un valore di 45°, in realtà esso è nell'intorno di questo valore e dipende da dove avviene il contatto quindi dove sulla vite e dove sulla madrevite e in che quadrante avviene nel sistema *otnb* nel piano nb, come rappresentato in Figura 3.9 ma anche soprattutto dal carico esterno e dalla deformazione dei due corpi che sono in contatto. Di conseguenza risulta necessario il calcolo dell'angolo di contatto sfera-pista. Questa procedura che verrà illustrata è stata eseguita nella fase di post-processing del modello. I sistemi di riferimento utilizzati per la definizione dell'angolo di contatto sono i seguenti:

- > Il sistema di riferimento  $ox_V y_V z_V$  che corrisponde al sistema mobile e solidale alla vite, con  $z_V$  che è collineare con l'asse della vite,  $x_V$  che interseca il punto di inizio elica della vite e  $O_V$  che è il punto presente all'estremità della vite e giace sull'asse  $z_V$
- > Il sistema di riferimento  $ox_{MV}y_{MV}z_{MV}$  che corrisponde al sistema mobile e solidale alla madrevite, con  $z_{MV}$  che è collineare con l'asse della vite,  $x_{MV}$  che interseca il punto di inizio elica della madrevite e  $O_{MV}$  che è il punto presente all'estremità della madrevite e giace sull'asse  $z_{MV}$
- > Il sistema mobile di Frenet-Serret otnb, precedentemente descritto nel Capitolo 3.1

I parametri utilizzati nel calcolo sono i seguenti:

- →  $R_m = \frac{D_m}{2} = 20,53 \text{ mm}$  che corrisponde al raggio dell'elica nominale
- >  $P_h = 20 mm$  che corrisponde al passo dell'elica
- >  $\alpha_{elica} = 8,81 \ deg$  che corrisponde all'angolo dell'elica
- >  $\theta$  che è l'angolo di azimut misurato sia nel sistema  $ox_V y_V z_V$  sia nel sistema  $ox_{MV} y_{MV} z_{MV}$ , che varia a seconda del punto considerato nel tratto elicoidale
- > L = H = 0,27 mm che corrisponde all'offset del centro di curvatura degli archi rispetto al centro di un arco di un profilo circolare della pista, presente sia lungo l'asse b e lungo l'asse n nel sistema di riferimento *otnb*

I successivi passaggi che saranno illustrati verranno applicati sia sulla vite utilizzando sia il sistema di riferimento  $ox_W y_W z_W$ , sia il sistema di riferimento  $ox_{MV} y_{MV} z_{MV}$ . All'interno della pista, i centri delle sfere assumono idealmente una posizione definibile tramite le equazioni di un'elica nel seguente modo:

$$\underline{R}_{B/O}^{xyz} = R_m \begin{bmatrix} \cos\theta \\ \sin\theta \\ \theta \cdot \tan\alpha_{elica} \end{bmatrix}$$
(3.28)

Si sottolinea che questa posizione definita con le equazioni (3.28) rappresenta anche i centri del sistema di riferimento *otnb*. In realtà la posizione reale è differente e ciò viene misurata e valutata all'interno di ADAMS ed è definita come  $\underline{R}_{F/0}^{xyz}$ . Conoscendo le 2 posizioni, è possibile definire il vettore differenza nel seguente modo:

$$\underline{R}_{F/B}^{xyz} = \underline{R}_{F/O}^{xyz} - \underline{R}_{B/O}^{xyz}$$
(3.29)

Ottenuto il vettore che rappresenta la distanza reale del centro sfera dal sistema di riferimento *otnb*, è possibile definirlo rispetto al sistema *otnb* sfruttando la matrice T5 definita nella (3.2):

$$\underline{R}_{F/B}^{tnb} = T_5(\theta)^{-1} \underline{R}_{F/B}^{xyz}$$
(3.30)

Dove l'angolo azimutale viene definito come:

$$\theta = \operatorname{atan2}(y, x) + \operatorname{round}\left(\frac{2\pi z}{P_h}\right)$$
(3.31)

Dove round corrisponde alla funzione presente in ambiente MATLAB che permette di arrotondare per eccesso se il numero decimale supera 0,5 mentre per difetto se il numero decimale è al di sotto di 0,5.



Figura 3.9 – Definizione dell'angolo  $\beta$ : a sinistra si trova lo schema per il calcolo del  $\beta_V$  mentre a destra si torva lo schema per il calcolo del  $\beta_{MV}$ 

Considerando la Figura 3.9, il calcolo del vettore  $\underline{R}_{F/B}^{tnb}$  permette di definire l'angolo di contatto, che è stato classificato a seconda dei corpi a contatto, quindi o rispetto alla vite o rispetto alla madrevite, e del quadrante occupato dal centro di curvatura nel seguente modo:

$$\beta_{V,b_V+} = (b_V \ge 0) \cdot \operatorname{atan} \frac{L+b_V}{H+n_V}$$
(3.32)

$$\beta_{V,b_V-} = (b_V < 0) \cdot \operatorname{atan} \frac{L - b_V}{H + n_V}$$
(3.33)

$$\beta_{MV,b_V+} = (b_{MV} \ge 0) \cdot \operatorname{atan} \frac{L + b_{MV}}{H - n_{MV}}$$
(3.34)

$$\beta_{MV,b_{V}-} = (b_{MV} < 0) \cdot \operatorname{atan} \frac{L - b_{MV}}{H - n_{MV}}$$
(3.35)

Tramite questo metodo è possibile vedere, a seconda di come si colloca la sfera rispetto al sistema di riferimento *otnb*, la posizione del centro della sfera è intuire la regione di contatto.

#### 4 MODELLAZIONE IN MSC ADAMS

MSC ADAMS è un software che permette di effettuare analisi dinamiche multybody di sistemi meccanici. Tramite questo programma è possibile vedere ed analizzare come funziona un sistema meccanico, osservando la cinematica dei corpi, i carichi presenti e le possibili criticità del sistema. In questo ambiente è possibile creare i modelli in due modalità: o utilizzando l'interfaccia grafica oppure tramite programmazione scrivendo un file .cmd con supporto di macro. Nella seguente attività di tesi si è scelta la seconda modalità in maniera da poter generare script riutilizzabili per altri modelli e permettendo all'utente di personalizzare come vuole il modello. In questo Capitolo 4 si analizzeranno i vari passaggi effettuati per definire il modello, in particolare nel Capitolo 4.3 si osserveranno le varie macro ed i relativi files .cmd mentre negli altri Capitoli si farà sempre riferimento al file vite\_ricircolo\_sfere.cmd. Si sottolinea che per modellare l'intero sistema e per sfruttare in maniera opportuna i comandi di ADAMS si sono sfruttati i tasti F1 (che corrisponde all'ADAMS Help) e F3 (che corrisponde alla Command Window) del programma per vedere a quale comando corrisponde, a livello di script, ogni operazione nell'interfaccia grafica.

#### 4.1 Struttura dei files e comandi di default del modello

Prima di illustrare il modello e le macro relative, si vuole puntualizzare la struttura dell'intero modello e i comandi di default per definire con quale logica è stato strutturato l'intera vite a ricircolo di sfere.



Figura 4.1 – Struttura del modello a livello di directory

Da come si può osservare nella Figura 4.1, il main script è "vite\_ricircolo\_sfere.cmd", il quale raccoglie comandi e dati direttamente dalle varie directory nel seguente modo:

- "Calcolo\_parametri\_contatto" è la directory in cui sono stati svolti i calcoli esposti nel Capitolo 3.2 per definire le forze di contatto presenti nel modello, utilizzando le rigidezze di contatto che sono contenute all'interno del file "rigidezze contatto.mat"
- "Calcolo\_percorso&marker" è la directory in cui sono stati svolti i calcoli esposti nel Capitolo
   3.1 per definire la posizione delle sfere all'interno della pista, sfruttando il file
   "path\_sfere.mat"; inoltre è presente un altro file che è "dati\_sfere.mat" dove sono memorizzati
   alcuni dati relativi al calcolo effettuato che sono utili all'interno del modello di ADAMS
- "Geometrie" è la directory dove contiene il Parasolid della vite e della madrevite del sistema che vengono importate all'interno del modello
- "Macro" è la directory dove sono collocate tutte le macro utilizzate, che verranno illustrate e descritte nel Capitolo 4.2



Figura 4.2 – Mappa concettuale della struttura del modello

All'interno del main script, che verrà analizzato e descritto successivamente, sono definite le condizioni di default del modello. Questa parte viene illustrata precedentemente in quanto alcuni comandi sono utili per capire le operazioni successive. Qui è illustrato la parte di codice presente in "vite ricircolo sfere.cmd":

```
!---SET UNITS, CREATE AND SET GRAVITY
default units len=mm mass=kg force=newton
model create model="vite_ricircolo_sfere" fit_to_view=no
part attrib part_name=ground name_vis=off
part modify rigid_body mass_properties part_name=ground &
  material=.materials.steel
force create body gravitational gravity=gravity &
  x_comp=0 y_comp=-9806.65 z_comp=0
view manage modify render=shaded
```

Innanzitutto si sottolinea che il punto esclamativo all'interno dello script permette di commentare il testo che viene scritto successivamente sulla stessa riga. Con il comando default units si definiscono le unità di misura del modello che in questo caso sono [mm] per le lunghezze, [kg] per le masse e [N] per le forze; di default il tempo è misurato in [s] e l'angolo in [deg]. Successivamente si crea il modello che è una "directory" dove si raggruppa tutto il sistema quindi le parti, forze, vincoli, variabili etc...; per farlo si è utilizzato il comando model create, imponendo al programma di non effettuare il fit to view che corrisponde ad uno zoom appropriato del sistema che è scelto ed effettuato dal programma stesso. Un altro elemento importante è il ground, che rappresenta l'ambiente generale del sistema a cui corrisponde il sistema di riferimento fisso generico; i comandi ad esso associato sono solo relativi alla grafica e alla scelta del materiale ad esso associato. Nel modello si impone una forza di gravità lungo l'asse y del sistema di riferimento del ground con accelerazione pari a  $g = -9806,65 \text{ mm/s}^2$ ; per fare ciò si è utilizzato il comando force create body gravitational. L'ultima operazione è di natura unicamente grafica, per poter vedere le parti presenti con un aspetto solido. L'elemento molto importante che nel codice non è presente ma che di default viene definito da ADAMS, è l'orientamento dei sistemi di riferimento. Essi sono definiti tramite gli angoli di Eulero che sono dei parametri angolari i quali definiscono una sequenza di rotazioni del sistema di riferimento che si vuole orientare nello spazio (quella di default presente all'interno del programma è la ZXZ o detta anche 313); il calcolo è espresso nella seguente maniera:

$$R_{ZXZ}(\phi,\theta,\psi) = R(z,\phi)R(x,\theta)R(z,\psi)$$
(4.1)

Dove:

$$R(z,\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0\\ \sin\phi & \cos\phi & 0\\ 0 & 0 & 1 \end{bmatrix}$$
(4.2)

$$R(x,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$
(4.3)

$$R(z,\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0\\ \sin\psi & \cos\psi & 0\\ 0 & 0 & 1 \end{bmatrix}$$
(4.4)

Le rotazioni messe in post-moltiplicazione indicano che ogni rotazione viene applicata al sistema mobile, quindi per esempio se si effettua la prima rotazione intorno all'asse x, quella successiva verrà applicata al sistema appena ruotato. Il ground come orientamento presenta tutti parametri angolari pari a 0, generando una matrice d'identità, e rappresenta il sistema fisso globale del modello.

# 4.2 Struttura del modello in ADAMS

I vari elementi presenti all'interno del modello, vengono strutturati seguendo la Figura 4.3:



Figura 4.3 – Struttura degli elementi in ADAMS

Il modello è tutto raggruppato in una "directory" definita "modello" che in questo caso è chiamato ".vite\_ricircolo\_sfere". All'interno di esso, sono presenti le seguenti "sub-directory":

- "Bodies" dove sono presenti le parti, le quali raggruppano all'interno sia le superfici dei corpi sia i marker appartenenti ai corpi stessi
- "Connectors" dove sono presenti i vincoli che riducono i gradi di libertà di rotazione e di traslazione del sistema
- Motions" dove sono presenti i moti rotatori e traslatori che vengono imposti dall'utente sui vincoli, eliminando gradi di libertà
- ➤ "Forces" dove sono presenti forze e i momenti che vengono imposti al sistema
- "Elements" dove sono presenti vari elementi, come per esempio le splines (verranno illustrate nel Capitolo 4.3.1)
- "Measures" dove sono le misure che vengono calcolate durante la simulazione
- "Design Variables" dove si inseriscono tutte le variabili che possono migliorare la progettazione e renderla più flessibile
- "Simulations" dove sono presenti gli script che permettono di effettuare le simulazioni con i comandi voluti
- "Results" dove ci sono i file .res dove sono presenti i risultati delle simulazioni (verranno illustrate nel Capitolo 4.3.5)
- "All Other" dove sono presenti i materiali e le request (verranno illustrare nel Capitolo 4.3.5) per i file .res

## 4.3 Macro

Le macro sono degli script scritti nel linguaggio ADAMS che permettono di creare nuovi comandi all'interno del programma che sfrutta i comandi di default presenti in questo ambiente. L'utilizzo di questi elementi permette una maggiore facilità di modellazione del sistema in quanto consente di effettuare varie operazioni in maniera automatica, che sarebbe possibile effettuare liberamente in ADAMS ma richiederebbe un tempo eccessivo. All'interno del software è possibile generare due tipologie di macro:

- Macro non parametrizzate che semplicemente, una volta richiamate, eseguono le operazioni definite all'interno dello script
- Macro parametrizzate che permettono di definire dei parametri generali che vengono definiti direttamente dall'utente una volta lanciato il comando; questa tipologia consente di generalizzare le operazioni presenti all'interno dello script

Ogni macro necessita un nome, un nome per il comando da inserire all'interno del programma e le serie di operazioni che deve effettuare la macro una volta richiamata. Un esempio è generate\_spheres\_xyz.cmd dove si inseriscono i vari comandi all'interno della macro e poi all'interno del file principale vite\_ricircolo\_sfere.cmd viene importata e viene definita con un comando proprio (vedere Capitolo 4.3.1 e Capitolo 4.4). Questi comandi personalizzati è possibile realizzarli sia all'interno di ADAMS sia esternamente scrivendo dei file .cmd.

Le macro create per realizzare la vite a ricircolo di sfere sono le seguenti:

- "generate\_spheres\_xyz" che permette di generare le sfere all'interno della pista
- "generate\_contact\_force\_screwshaft\_sphere" che permette di generare le forze di contatto tra vite e sfera
- "generate\_contact\_force\_sphere\_nut" che permette di generare le forze di contatto tra madrevite e sfera
- "generate\_contact\_force\_sphere\_sphere" che permette di generare le forze di contatto tra sfera e sfera
- "generate\_request" che permette di generare le request di output per esportare i risultati dopo la simulazione

Nei prossimi Capitoli verranno illustrati passo a passo i vari script creati per questi comandi. Si sottolinea che ogni macro descritta rappresenta un file .cmd che viene richiamato poi direttamente nel file principale .vite\_ricircolo\_sfere.cmd per effettuare i comandi opportuni.

### 4.3.1 Generate\_spheres\_xyz

Nella prima parte si definiscono i parametri necessari per la creazione delle sfere all'interno della pista, che sono i seguenti:

- > Il modello che serve per dare il riferimento per creare le varie parti all'interno dello script
- Il file .mat che viene ricavato dal calcolo in MATLAB e dove sono presenti le posizioni dei centri delle sfere nello spazio calcolate secondo la procedura illustra nel Capitolo 3.1
- Il diametro delle sfere da imporre nella creazione delle parti all'interno del modello, il cui valore è stato limitato tra i valori compresi tra 1 mm e 4 mm; questa scelta è stata semplicemente una decisione del tesista che ha preferito limitare il valore del diametro delle sfere
- Un numero che fa da riferimento per identificare su quale madrevite si generano le sfere, nell'eventuale caso in cui si voglia inserire nel sistema una ulteriore madrevite

```
!$model:t=model:a
!$file_path:t=file(*.mat)
!$diameter:T=real:C=1:D=4
!$num_prefix:T=integer
!END OF PARAMETERS
```

Il primo passo è quello di importare le coordinate spaziali dei centri delle sfere tramite il comando file testdata read splines che genera delle splines all'interno di ADAMS

```
!---import data
file testdata read splines &
  model_name = $model &
  file_name = $file_path
!
```

Il file è strutturato come in Figura 4.4:

```
path_sfere - Blocco note
File Modifica Formato Visualizza ?
  2.0366965e+01
                  6.4590000e+01
                                  2.6090684e+00
  2.0203981e+01
                  6.5566487e+01 -3.6632843e+00
  1.8154443e+01 6.6542974e+01 -9.5935766e+00
  1.4409727e+01 6.7519461e+01 -1.4628065e+01
  9.3194971e+00 6.8495948e+01 -1.8296652e+01
  3.3559486e+00
                  6.9472924e+01 -2.0257298e+01
                  7.0449411e+01 -2.0324982e+01
 -2.9181562e+00
 -8.9197772e+00
                  7.1425898e+01 -1.8494813e+01
 -1.4090802e+01
                  7.2402873e+01
                                 -1.4935522e+01
 -1.7943256e+01
                  7.3379360e+01 -9.9829903e+00
 -2.0120251e+01
                  7.4355847e+01 -4.0982930e+00
                                  2.1690840e+00
 -2.0418511e+01
                  7.5332334e+01
    Figura 4.4 – Struttura del file "path sfere.mat"
```

Da come si può osservare, è un file formato da tante righe quante sono le sfere e da un numero di colonne pari al numero di coordinate: la prima corrisponde alla coordinata x, la seconda alla y e la terza alla z, secondo il sistema *oxyz* descritto nel Capitolo 3.1. Nell'importazione, si creano tre splines all'interno del programma; esse sono delle funzioni continue create avendo come riferimento i dati inseriti come input. la spline\_1 corrisponde alla prima colonna del file, la spline\_2 alla seconda e la spline 3 alla terza. A titolo d'esempio, si può osservare la Figura 4.5:

| Modify spline X |             |            |              |                     |  |  |  |
|-----------------|-------------|------------|--------------|---------------------|--|--|--|
| Na              | me SPLINE_1 |            |              |                     |  |  |  |
|                 | х           | Y          | Туре         | y=f(x) (2D) ▼       |  |  |  |
| 1               | 1.0         | 20.366965  | View as      | Tabular Data 🔹      |  |  |  |
| 2               | 2.0         | 20.203981  | X Units no   | _units              |  |  |  |
| 3               | 3.0         | 18.154443  | V Unite Inc  |                     |  |  |  |
| 4               | 4.0         | 14.409727  | i onico pric | _units              |  |  |  |
| 5               | 5.0         | 9.3194971  |              |                     |  |  |  |
| 6               | 6.0         | 3.3559486  | Linear e     | xtrapolation        |  |  |  |
| 7               | 7.0         | -2.9181562 |              |                     |  |  |  |
| 8               | 8.0         | -8.9197772 | Append r     | ow to X and Y data  |  |  |  |
| 9               | 9.0         | -14.090802 | Prepend      | row to X and Y data |  |  |  |
| 10              | 10.0        | -17.943256 |              |                     |  |  |  |
| 11              | 11.0        | -20.120251 | Insert R     | ow After            |  |  |  |
| 12              | 12.0        | -20.418511 | Remov        | ve Row              |  |  |  |
| 13              | 13.0        | -18.808921 |              |                     |  |  |  |

Figura 4.5 – Creazione della spline all'interno di ADAMS

Da come si può osservare in Figura 4.5, ogni spline presenta sulla colonna X una numerazione continua di numeri interi di default che, nella seguente applicazione, corrisponde agli indici delle sfere mentre sull'asse Y presenta i valori assegnati dal "path\_sfere.mat" che corrispondono alle diverse coordinate a seconda della spline considerata.

Successivamente si definisce un ciclo con for che itera sul numero di sfere, il quale viene dedotto contando il numero di righe presente in una delle tre splines tramite la funzione DB COUNT.

```
!---start cycle for create spheres
for var=$_self.ii start=1
end=(DB_COUNT("$model.spline_1","xs"))
```

All'interno del ciclo, per creare ogni sfera, si eseguono le seguenti operazioni:

Si definisce la variabile intera \$\_self.jj (in questo modo si definiscono le variabili locali temporanee che sono molto comode in fase di scrittura delle macro, in quanto si raggruppano più tipologie di variabili in un'unica variabile, la quale può essere agevolmente eliminata al termine dei comandi) che permette di sfruttare correttamente il ciclo per effettuare i vari comandi, tramite il comando var set

var set var=\$ self.jj int=(eval(INT(\$ self.ii)))

Si crea la parte sfera, imponendo delle condizioni di default, svincolando la parte dal ground, rendendola mobile (ground\_part=no) e classificando la sfera con il nome "Sfera\_(numero di prefisso)\_(valore di \$\_self.jj)", tramite il comando part create rigid\_body name and position

```
!---create sphere
part create rigid_body name_and_position &
part_name =("$model.Sfera_"//$num_prefix//"_"//
$_Self.jj) &
ground_part = no &
planar = no &
planar = no &
planar_axes = xy &
orientation = 0.0, 0.0, 0.0
```

Si assegna il materiale alla parte, che nel seguente caso è acciaio, tramite il comando part modify rigid\_body mass\_properties

```
part modify rigid_body mass_properties
part_name=("$model.Sfera_"//$num_prefix//"_"//
$_Self.jj)
material=.materials.steel
```

Si definiscono delle opzioni divise come per esempio il colore, tramite il comando part attributes

```
part attributes
part_name=("$model.Sfera_"//$num_prefix//"_"//
$_Self.jj)
color = YELLOW name_vis = off
```

Si crea un marker chiamato "MAIN\_SPHERE", che corrisponde ad un sistema di riferimento nella parte sfera, con origine nei centri delle sfere (qui si utilizzano le splines per definire la posizione precisa all'interno della pista) e con orientazione del marker pari a quella di default del marker del ground in quanto non è di principale interesse, tramite il comando marker create

```
marker create
marker_name=(eval("$model.Sfera_"//$num_prefix//"_"//
$_Self.jj//".MAIN_SPHERE")) &
location=(LOC_RELATIVE_TO({(eval($model.spline_1.ys
[$_self.jj])), (eval($model.spline_2.ys[$_self.jj])),
(eval($model.spline_3.ys[$_self.jj]))},$model.GROUND)) &
orientation = 0.0, 0.0, 0.0
```

Si genera l'ellissoide chiamato "Sphere" con tutti i raggi uguali nelle tre direzioni x, y e z con centro corrispondente all'origine del marker "MAIN\_SPHERE" tramite il comando geometry create shape\_ellipsoid

```
geometry create shape ellipsoid &
ellipsoid_name=("$model.Sfera_"//$num_prefix//"_"//
$_Self.jj//".Sphere_"//$num_prefix//"_"//$_Self.jj) &
x_scale_factor=($diameter) &
y_scale_factor=($diameter) &
z_scale_factor=($diameter) &
```

center\_marker=(eval("\$model.Sfera\_"//\$num\_prefix//
"\_"//\$\_Self.jj//".MAIN\_SPHERE"))

Alla fine della macro si cancellano le spline e le variabili temporanee \$\_self.ii e \$\_self.jj con i comandi var del e data\_element delete, in maniera da poter riutilizzare le stesse variabili nelle successive macro.

```
!---delete splines and variables
var del var=(EVAL(DB_CHILDREN($_self,"variable")))
data_element delete data_element=$model.spline_1
data_element delete data_element=$model.spline_2
data_element delete data_element=$model.spline_3
```

Si sottolinea che quando si richiama o si definisce qualche elemento come marker, parti o superfici, si sfrutta la logica di definizione spiegata nel Capitolo 4.2, per esempio per definire un marker il richiamo è:

#### .(nome modello).(nome parte).(nome marker)

Inoltre, quando si definisce un'opzione di un comando come per esempio nomi di elementi, valori di posizioni o altri parametri, quando essa è parametrizzabile come per esempio il nome di un elemento si può utilizzare l'estensione diretta come descritto qui sopra, mentre se non è parametrizzabile come per esempio il richiamo di un marker è necessario usare la funzione eval per far valutare da ADAMS l'estensione imposta e per realizzare l'opportuno collegamento con gli elementi. Queste nozioni appena descritte valgono ovviamente anche per le successive macro spiegate nei Capitoli successivi.

#### 4.3.2 Generate\_contact\_force\_screwshaft\_sphere

Innanzitutto è necessario definire come all'interno di ADAMS viene implementata la forza di contatto tra due corpi. Essa viene inserita all'interno del sistema tramite il comando contact, che permette di generare contatti tra corpi rigidi e flessibili, dotati eventualmente di attriti statici e dinamici. Si definisce il contatto nel seguente modo:

$$F = \begin{cases} K(x_1 - x)^{\epsilon} - STEP(x, x_1 - d, C_{MAX}, x_1, 0)\dot{x} & x < x_1 \\ 0 & x \ge x_1 \end{cases}$$
(4.5)

La prima parte, ossia  $K(x_1 - x)^{\epsilon}$  riguarda la parte elastica del contatto dove K rappresenta la rigidezza calcolata nel Capitolo 3.2 mentre  $\epsilon$  è l'esponente della forza assunto pari a 1,5 considerando la teoria di Hertz. La seconda parte, ossia  $STEP(x, x_1 - d, C_{MAX}, x_1, 0)$ , riguarda la parte viscosa del

contatto, che viene introdotta con lo scopo di incorporare un termine di smorzamento interno del materiale e per facilitare l'integrazione da parte del solver. Quest'ultima componente viene definita con la funzione STEP cubica che ha come variabile x e assume valore nullo quando  $x \ge x_1$  mentre assume il valore  $C_{MAX}$  quando  $x \le x_1 - d$  dove d corrisponde alla compenetrazione massima tra i due corpi; si osserva che la parte viscosa è funzione della velocità di compenetrazione tra i due corpi e si oppone alla velocità relativa di moto. Per capire meglio la variabile x e come si attiva la forza di contatto, si osserva la Figura 4.6:



Figura 4.6 – Schema del modello di contatto in ambiente ADAMS [1]

Tramite questa Figura 4.6, si osserva che quando i due corpi sono lontani e non a contatto, ossia quando  $x \ge x_1$ , si ha che la forza di contatto è nulla e questo è ragionevole in quanto non vi è alcuna compenetrazione; viceversa se i corpi sono a contatto e c'è compenetrazione, ossia  $x < x_1$ , si ha una forza di contatto, calcolata secondo la (4.5).

Tramite il contact presente in ADAMS, è possibile inserire nella forza di contatto anche la componente d'attrito che dipende dalla velocità di slittamento che a sua volta influenza il coefficiente d'attrito, il quale è definito come in Figura 4.7:



Velocità di slittamento

Figura 4.7 – Andamento del coefficiente d'attrito in funzione della velocità di slittamento [1]

Da come si evidenzia nella Figura 4.7, il coefficiente d'attrito assume un valore nullo quando la velocità di slittamento è nulla, successivamente cresce fino ad un valore  $\mu_s$  quando si ha velocità pari  $V_s$  ed infine decresce fino ad un valore  $\mu_d$  raggiunta la velocità  $V_d$  e si stabilizza su questo valore per velocità di slittamento maggiori.

A questo punto risulta molto chiaro quali siano i parametri necessari da fornire al programma per impostare ed inserire le forze di contatto volute, in particolare essi sono:

- ▶ K che corrisponde alla rigidezza di contatto, definita nel Capitolo 3.2
- $\triangleright$   $\epsilon$  che corrisponde all'esponente della forza elastica, definito nel Capitolo 3.2
- $\succ$   $C_{MAX}$  che corrisponde al valore massimo del coefficiente di smorzamento
- >  $d_{MAX}$  che corrisponde alla compenetrazione tra i due corpi a contatto alla quale è presente il valore massimo di smorzamento
- $\blacktriangleright$   $\mu_s$  che corrisponde al valore di attrito statico
- >  $\mu_d$  che corrisponde al valore di attrito dinamico
- $\succ$  V<sub>s</sub> che corrisponde al valore di velocità di slittamento a cui corrisponde  $\mu_s$
- >  $V_d$  che corrisponde al valore di velocità di slittamento a cui corrisponde  $\mu_d$

Ritornando alla macro, ora risultano ovvi i parametri richiesti in input che sono gli elementi qui sopra citati, inoltre si richiedono, come nel Capitolo 4.3.1, il file del path delle sfere per calcolare il numero di sfere, il modello per implementare le forze nel sistema desiderato e il prefisso per indicare su quale madrevite si vogliono creare i contatti voluti. A questo punto, iterando sul numero di sfere tramite un ciclo for, effettuando gli stessi passaggi come nel Capitolo 4.3.1, si creano contatti voluti seguendo il seguente iter:

Si definisce la variabile \$\_self.jj per avere la variabile d'appoggio per iterare come si vuole all'interno del ciclo tramite il comando var set

```
var set var=$ self.jj int=(eval(INT($ self.ii)))
```

Con il comando contact create si crea il contatto. Le geometrie tra cui si vuole il contatto sono inserite nell'i\_geometry\_name e nel j\_geometry\_name, che sono le superfici della vite e della sfera. La tipologia di contatto imposta è solid to solid in quanto, nonostante la sfera sia una geometria semplice, la pista è una geometria complessa. Ovviamente si inseriscono i parametri precedentemente elencati. i\_flip\_normal e j\_flip\_normal sono dei valori booleani che specificano se le normali devono essere capovolte ("yes") o meno ("no"); si è scelta per tutte e due le superfici "no"

```
!---create contact forces
contact create &
contact name =("$model.S"//$num prefix//" "//
$ Self.jj//" V") &
type = solid to solid &
i geometry name=(eval("$model.Vite.SOLID3"))
j geometry name=(eval("$model.Sfera "//$num prefix//
" "//$ Self.jj//".Sphere "//$num prefix//
" "//$ Self.jj)) &
i flip normal = "no"
                      &
j flip normal = "no"
                      &
stiffness = ($stiffness)
                          &
damping = ($damping)
                      δ
exponent = ($exponent) &
dmax = (\$dmax)
               &
coulomb friction = on &
mu static = ($mu static)
                          &
mu dynamic = ($mu dynamic)
                            &
stiction transition velocity =
($stiction transition velocity)
                                 &
friction transition velocity =
($friction transition velocity)
```

Alla fine del ciclo si eliminano le variabili utilizzate all'interno della macro per poterle riutilizzare successivamente.

```
!---delete splines and variables
var del var=number_of_spheres
var del var=(EVAL(DB_CHILDREN($_self,"variable")))
```

data\_element delete data\_element=\$model.spline\_1
data\_element delete data\_element=\$model.spline\_2
data element delete data element=\$model.spline 3

La stessa logica utilizzata all'interno di questa macro viene utilizzata nelle prossime macro che generano le forze di contatto, cambiando le superfici d'interesse ed eventualmente la tipologia di contatto, con delle differenze illustrate nei Capitolo 4.3.3 e Capitolo 4.3.4.

4.3.3 Generate\_contact\_force\_sphere\_nut

Il codice della macro utilizzato è il seguente:

```
!$model:t=model:a
!$file path:t=file(*.mat)
!$stiffness:T=real
!$damping:T=real
!$exponent:T=real
!$dmax:T=real
!$mu static:T=real
!$mu dynamic:T=real
!$stiction transition velocity=T:real
!$friction transition velocity=T:real
!$num prefix:T=integer
!END OF PARAMETERS
!---import data
file testdata read splines &
 model name = $model &
 file name = $file path
!
!---number of spheres
var set var=number of spheres
int=(eval(DB COUNT("$model.spline 1","xs")))
!
!---start cycle for contact forces sphere-nut from 1 to 71
for var=$ self.ii start=1
end=(eval(INT(number of spheres)))
  1
 var set var=$ self.jj int=(eval(INT($ self.ii)))
  !---create contact forces
  contact create &
```

```
contact name =
       ("$model.S"//$num prefix//" "//$_Self.jj//"_MV1")
                                                           &
       type = solid to solid &
       i geometry name=(eval("$model.Sfera "//$num prefix//
       " "//$ Self.jj//".Sphere "//$num prefix//
       " "//$ Self.jj))
                         &
       j geometry name=(eval("$model.Madrevite 1.
       SOLID1")) &
       i flip normal = "no"
       j flip normal = "no"
                             &
       stiffness = ($stiffness)
                                 &
       damping = (\$damping)
                            &
       exponent = ($exponent) &
       dmax = (\$dmax)
                      &
       coulomb friction = on
                              &
       mu_static = ($mu static)
                                 &
       mu dynamic = ($mu dynamic)
       stiction transition velocity =
       ($stiction transition velocity)
                                        &
       friction transition velocity =
       ($friction transition velocity)
 entity attributes
  entity name=("$model.S"//$num prefix//
  " "//$ Self.jj//" MV1") size=6
                                   &
       name vis=off
!---delete splines and variables
var del var=number of spheres
var del var=(EVAL(DB CHILDREN($ self, "variable")))
data element delete data element=$model.spline 1
data element delete data element=$model.spline 2
data element delete data element=$model.spline 3
```

! end !

L

La definizione e la scrittura del codice è identica a quella del Capitolo 4.3.2, con la differenza che in questo caso i corpi a contatto sono la sfera e la madrevite, pertanto la tipologia di contatto rimane identica in quanto la geometria della pista è complessa e quindi rimane solid to solid, mentre cambiano le i geometry name e j geometry name che in questo caso riguardano le superfici d'interesse.

Il codice della macro utilizzato è il seguente:

```
!$model:t=model:a
!$file path:t=file(*.mat)
!$stiffness:T=real
!$damping:T=real
!$exponent:T=real
!$dmax:T=real
!$mu static:T=real
!$mu dynamic:T=real
!$stiction transition velocity=T:real
!$friction_transition velocity=T:real
!$num prefix:T=integer
!END OF PARAMETERS
!---import data
file testdata read splines &
 model name = $model &
 file name = $file_path
!
!---number of spheres
var set var=number of spheres
int=(eval(DB COUNT("$model.spline 1","xs")))
!
!---start cycle for contact forces sphere-sphere from 1 to 70
for var=$ self.ii start=1 end=(eval(INT(number of spheres)-INT(1)))
  !
 var set var=$ self.jj int=(eval(INT($ self.ii)))
 var set var=$ self.kk int=(eval(INT($ self.ii)+INT(1)))
  !
  !---create contact forces
  contact create &
       contact name =
       ("$model.S"//$num prefix//" "//$ Self.jj//" S"
       //$num prefix//" "//$ Self.kk) &
       type = sphere to sphere &
       i_geometry_name=(eval("$model.Sfera "//$num prefix
       //"_"//$_Self.jj//".Sphere_"//$num_prefix
       //" "//$ Self.jj)) &
       j_geometry_name=(eval("$model.Sfera "//$num prefix
       //" "//$ Self.kk//".Sphere "//$num prefix
       //" "//$_Self.kk)) &
```

```
i flip normal = "no"
                             &
       j flip normal = "no"
                             &
       stiffness = ($stiffness) &
       damping = ($damping) &
       exponent = ($exponent) &
      dmax = (\$dmax)
                      &
       coulomb friction = on &
      mu static = ($mu static)
      mu dynamic = ($mu dynamic)
                                  &
       stiction transition velocity =
       ($stiction transition velocity)
                                       &
       friction transition velocity =
       ($friction transition velocity)
  entity attributes
 entity name=("$model.S"//$num prefix//" "//$ Self.jj
  //" S"//$num prefix//" "//$ Self.kk) size=6 &
      name vis=off
  !
end
1
!---contact force sphere 1-sphere 71
contact create &
  contact name =
  ("$model.S"//$num prefix//" "//number of spheres
  //" S"//$num prefix//" "//1) &
  type = sphere to_sphere &
  i geometry name=(eval("$model.Sfera "//$num prefix//" "
  //number of spheres//".Sphere "//$num prefix//" "
  //number of spheres))
                         &
  j geometry name=(eval("$model.Sfera "//$num prefix//" "
  //1//".Sphere "//$num prefix//" "//1)) &
  i flip normal = "no"
                        &
  j flip normal = "no"
  stiffness = ($stiffness)
                            &
 damping = ($damping)
                       &
  exponent = ($exponent) &
  dmax = (\$dmax)
                 &
  coulomb friction = on &
 mu static = ($mu static) &
 mu dynamic = ($mu dynamic)
  stiction transition velocity =
  ($stiction transition velocity)
                                   &
  friction transition velocity =
  ($friction transition velocity)
entity attributes
entity name=("$model.S"//$num prefix//" "//
```

```
number_of_spheres//"_S"//$num_prefix//"_"//1) size=6 &
name_vis=off
!
!
!---delete splines and variables
var del var=number_of_spheres
var del var=(EVAL(DB_CHILDREN($_self,"variable")))
data_element delete data_element=$model.spline_1
data_element delete data_element=$model.spline_2
data_element delete data_element=$model.spline_3
```

La definizione e la scrittura del codice è identica a quella del Capitolo 4.3.2, con le seguenti differenze:

- Si definisce all'interno del ciclo, un ulteriore variabile d'appoggio \$\_self.kk che corrisponde a \$\_self.jj+1 per creare le forze di contatto tra una sfera e la successiva
- Il ciclo itera sul numero di sfere meno 1, in quanto l'ultima forza di contatto, essendo tra la numero 71 e la 1 per chiudere il circuito, si è preferita crearla al di fuori del ciclo da come si evidenzia nel codice
- Le superfici d'interesse in questo caso sono quelle legate alle sfere, quindi ad ogni iterazione si considerano due sfere
- In questo caso la tipologia di contatto viene imposta sphere to sphere, in quanto avendo in questo caso due geometrie semplici e note al programma, conviene utilizzare questa modalità per rendere più performante il calcolo da parte della macchina

#### 4.3.5 Generate\_requests

Si riporta, a titolo d'esempio, parte del codice della macro d'interesse:

```
!$model:t=model:a
!$file_path:t=file(*.mat)
!$num_prefix:T=integer
!END_OF_PARAMETERS
!---import data
file testdata read splines &
   model_name = $model &
   file_name = $file_path
!
```

```
!---number of spheres
var set var=number of spheres
int=(eval(DB COUNT("$model.spline 1","xs")))
!
!CREATE REQUESTS FOR SPHERES ACCELERATION, VELOCITY, DISPLACEMENT
for var=$ self.ii start=1
end=(eval(INT(number of spheres)))
  !
 var set var=$ self.jj int=(eval(INT($ self.ii)))
  !
  output control create request
                                  &
     request name = ("D Sfera "//$num prefix//" "//
     $ Self.jj)
                 &
     component names = "S", "X", "Y", "Z" &
     results name = "displacement spheres"
                                             &
     fl = "" &
     f2 = (eval("DX(Sfera "//$num prefix//" "//$ Self.jj
     //".cm, 0, 0)"))
                       &
     f3 = (eval("DY(Sfera "//$num prefix//" "//$ Self.jj
     //".cm, 0, 0)"))
                      &
     f4 = (eval("DZ(Sfera "//$num prefix//" "//$ Self.jj
     //".cm, 0, 0)"))
                       &
     f5 = ""
              δ
     f6 = ""
              &
     f7 = ""
              &
     f8 = ""
```

Innanzitutto le requests sono delle "richieste" che si possono generare all'interno del programma che permettono di riportare all'interno del file complessivo di risultati, che ADAMS genera come file .res, i dati d'interesse descritti e pretesi dall'utente. Nel seguente caso sono stati definite tramite Fuction Builder, ossia tramite una funzionalità del programma che permette di generare delle funzioni come meglio si vuole. Il Dialog Box del Fuction Builder è rappresentato nella seguente Figura 4.8:

| Define a runtime function   | Function Builder          |     |                     |            |           |            | ×             |
|---|---------------------------|-----|---------------------|------------|-----------|------------|---------------|
| Math Functions <ul> <li>Assist</li> <li>A</li></ul> | Define a runtime function |     |                     | C Full nar | nes 🤉 Sho | rt names C | Adams ids     |
| Math Functions Assist   ABS   ACOS   AINT   ANINT   ANINT   ASIN   ATAN   ATAN2   Chebyshev Polynomial   COS   COS   COS   COS   COS   COS   COS   COS   DIM   EXP   Fourtier Cosine Series   |                           |     |                     |            |           |            |               |
| Math Functions       Assist       +         ABS       Acos       Ann         AINT       Annon       Load         ANINT       Arguments       Load         ATAN       Arguments       Cos         Cos       Getting Object Data       Getting Object Data         DELAY       DIM       Markers       Insert Object Name   |                           |     |                     |            |           |            |               |
| Math Functions       Assist       +         ABS       Assist       +         ABS       Acos       Ann         AINT       Ann       Arguments         ASIN       Arguments       Load         ATAN       Arguments       Arguments         Cos       Cos       Getting Object Data         DELAY       DIM       Markers         EXP       Fourtier Cosine Series       Insert Object Name   |                           |     |                     |            |           |            |               |
| Math Functions Assist   ABS Acos   AINT Name   ASIN Arguments   ATAN Arguments   Chebyshev Polynomial Getting Object Data   COS COSH   DIM EXP   Fourier Cosine Series Insert Object Name   |                           |     |                     |            |           |            |               |
| Math Functions       Assist       +         ABS       Assist       +         ALOS       AINT       Name       Load         AINT       Arguments       -       -         ATAN       ATAN       Arguments       -         Chebyshev Polynomial       Cos       -       -         COS       COSH       Getting Object Data       -         DELAY       DIM       Markers       -         EXP       Fourier Cosine Series       Insert Object Name  |                           |     |                     |            |           |            |               |
| ABS<br>ACOS<br>AINT<br>AINT<br>AINT<br>AINT<br>AINT<br>AINN<br>ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Fourier Cosine Series<br>Insert Object Name   | Math Functions - Ass      | ist |                     |            |           |            | +             |
| ACOS<br>AINT<br>ANINT<br>ANINT<br>ASIN<br>ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Fourier Cosine Series  | ABS                       | •   |                     |            |           |            |               |
| AINT<br>ANINT<br>ANINT<br>ASIN<br>ATAN<br>ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Exp<br>Exp<br>Enurier Cosine Series<br>Insert Object Name<br>LOad<br>Arguments<br>Cod<br>Cod<br>Cod<br>Cod<br>Cod<br>Cod<br>Cod<br>Cod   | ACOS                      | _   |                     |            |           | _          |               |
| ANINT<br>ASIN<br>ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Fourier Cosine Series   | AINT                      |     | Name                |            |           |            | Load          |
| ASIN<br>ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Exurier Cosine Series<br>Insert Object Name  | ANINT                     |     | Arguments           |            |           |            |               |
| ATAN<br>ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Fourier Cosine Series<br>Insert Object Name  | ASIN                      |     |                     |            |           |            |               |
| ATAN2<br>Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Exurier Cosine Series<br>Insert Object Name  | ATAN                      |     |                     |            |           |            |               |
| Chebyshev Polynomial<br>COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Exurier Cosine Series<br>Insert Object Name   | ATAN2                     |     |                     |            |           |            |               |
| COS<br>COSH<br>DELAY<br>DIM<br>EXP<br>Fourier Cosine Series<br>Insert Object Name   | Chebyshev Polynomial      |     |                     |            |           |            |               |
| COSH Getting Object Data DELAY DIM EXP Fourier Cosine Series Insert Object Name   | COS                       | _   |                     |            |           |            |               |
| DELAY<br>DIM Markers<br>EXP<br>Fourier Cosine Series  | COSH                      |     | Getting Object Data |            |           |            |               |
| DIM Markers Insert Object Name  | DELAY                     |     |                     |            |           |            |               |
| EXP Insert Object Name  | DIM                       |     | Markers 🔹           |            |           |            |               |
| Fourier Cosine Series   | EXP                       |     | ·                   |            |           | land (     | Nicest Manage |
|   | Fourier Cosine Series     |     |                     |            |           | Insert C   | Dbject Name   |
| Fourier Sine Series   | Fourier Sine Series       |     | Dist Dist Limits    |            |           |            | Marif         |
| Haversine Step  | Haversine Step            | -   | Plot Limits         |            |           |            | verify        |
| OK Apply Cancel   |                           |     |                     |            | OK        | Apply      | Cancel        |

Figura 4.8 – Dialog Box del Fuction Builder

Tramite questa finestra è possibile creare delle funzioni per molte applicazioni all'interno del programma come in questo caso requests, ma anche per measure, motion, marker ecc.... Ogni requests, per essere definita, necessita dei seguenti elementi:

- Il nome della request che permette semplicemente di creare l'elemento all'interno del programma
- I nomi delle componenti in maniera che nel post-processing sia possibile capire quale componente si sta osservando o analizzando
- Il nome del results in maniera da raggruppare alcune requests in un'unica request come per esempio le forze di contatto
- Le componenti che si vogliono avere come risultati: le componenti sono otto dove f1 e f5 corrispondono ai moduli di f2, f3, f4 e di f6, f7, f8 che normalmente in fase di simulazione vengono calcolate ma non vengono riportate nei risultati; generalmente le componenti f1 e f5 vengono lasciate vuote e si riempiono le altre, per esempio per le azioni dinamiche come i contatti in f2, f3 e f4 si riportano le forze lungo x, y e z secondo il sistema di riferimento del ground mentre in f6, f7 e f8 si riportano i momenti lungo x, y e z secondo il sistema di riferimento del ground

Come nel Capitolo 4.3.1, i parametri della macro richiesti sono il modello, il file path per eseguire correttamente le iterazioni nei cicli e il numero prefisso per richiamare e definire alcuni nomi di requests. Si sottolinea che anche in questo caso si definisce nei cicli la variabile d'appoggio \$\_self.jj

per effettuare le iterazioni correttamente. Tramite il comando output\_control create request si creano le requests volute e i dati richiesti per il modello sono i seguenti:

- Posizione, velocità lineari ed angolari e accelerazioni lineari ed angolari di tutte le sfere secondo il sistema di riferimento ground
- Posizione, velocità lineari ed angolari e accelerazioni lineari ed angolari della madrevite secondo il sistema di riferimento ground
- Posizione, velocità lineari ed angolari e accelerazioni lineari ed angolari della vite secondo il sistema di riferimento ground
- Posizione delle sfere secondo sia il sistema di riferimento mobile V\_MARKER\_THETA solidale alla vite sia il sistema di riferimento mobile M\_MARKER\_THETA solidale alla madrevite; essi sono dati molto importanti per i calcoli degli angoli di contatto in fase di postprocessing
- Le forze e i momenti di contatto tra sfera-vite, sfera-madrevite e sfera-sfera secondo il sistema di riferimento ground
- Le forze e i momenti sui vincoli, che sono un revolute e un perpendicular secondo il sistema di riferimento ground
- > Le forze e i momenti sul motion di tipo rotativo secondo il sistema di riferimento ground
- Le forze e i momenti della forza esterna che agisce assialmente sulla madrevite secondo il sistema di riferimento ground
- Il rendimento del sistema dedotto dalla measure creata appositamente

Si sottolinea che tra i dati elencati, ci sono alcuni elementi che non sono stati ancora spiegati, ma essi verranno analizzati successivamente nei prossimi Capitoli.

## 4.4 Importazione e creazione delle parti

Nella prima fase di modellazione del sistema vite a ricircolo di sfere, è necessario importare il modello Parasolid di vite e madrevite tramite il comando file parasolid read nel seguente modo:

```
!---IMPORT PARASOLID FILE
file parasolid read &
  file_name = "./Geometrie/Geometria_ADAMS_v4_3_
  doppio_ricircolo_simplified_cut.x_t" &
  ref_markers = global &
  type = ascii &
  explode_assemblies = no &
```

```
model_name = .vite_ricircolo_sfere
!
   !---RENAME PARTS
entity modify entity=.vite_ricircolo_sfere.PART4
new=.vite_ricircolo_sfere.Vite
entity modify entity=.vite_ricircolo_sfere.PART2
new=.vite_ricircolo_sfere.Madrevite_1
!
   !---SET TRANSPARENCY FOR NUT
entity attributes entity_name=Madrevite_1 transparency=50
!
```

Durante la fase di importazione, sono necessari alcuni riferimenti che sono il modello dove si vogliono importare le parti, il sistema di riferimento rispetto al quale si vogliono orientare le parti, la tipologia di scrittura del file x\_t che in questo caso è ascii. Inoltre si impone l'assenza di incorporamento tra le parti, in maniera da mantenerle distinte. A questo punto sono presenti dei nomi di default per gli elementi importati dal Parasolid che sono scomodi a livello di script per richiamare le parti, pertanto si utilizza il comando entity modify per modificare i nomi e rinominarli. Infine con il comando entity attributes si modifica la trasparenza della madrevite con l'obiettivo di vedere graficamente sia in fase statica che in fase dinamica le sfere all'interno della pista.

Al fine di generalizzare le sfere e di crearle della dimensione opportuna, sono state scritte le seguenti linee di codice:

```
file testdata read splines &
  model_name = .vite_ricircolo_sfere &
  file_name = ".\Calcolo_percorso&markers\dati_sfere.mat"
!
    !---CREATE DESIGN VARIABLE FOR SPHERE RADIUS & SPHERE DIAMETER
var set var=RAGGIO_SFERA
real=(eval(.vite_ricircolo_sfere.spline_1.ys[1]))
var set var=DIAMETRO_SFERA
real=((.vite_ricircolo_sfere.RAGGIO_SFERA)*2)
data_element delete
data_element=.vite_ricircolo_sfere.spline_1
!
```

Dal calcolo del path delle sfere eseguito nell'ambiente MATLAB (Capitolo 3.1), si è estrapolato un file .mat (estratto dal file MATLAB in cui sono stati effettuati i calcoli del Capitolo 3.1) contenente il raggio delle sfere; esso viene letto come spline tramite il comando file testdata read splines e successivamente si definisce una design variable chiamata RAGGIO\_SFERA e un'altra di nome DIAMETRO SFERA collegata alla precedente tramite il comando var set var. A

questo punto è possibile creare le sfere sfruttando la macro descritta nel Capitolo 4.3.1 nel seguente modo:

```
!---IMPORT MACROS
macro read &
   macro name = generate spheres xyz
                                       8
   file name = "./Macro/generate spheres xyz.cmd"
                                                     &
   wrap in undo = yes
                        &
   create panel = yes
1
!CREATE SPHERES
generate spheres xyz
                       &
   model = .vite ricircolo sfere
                                   &
   file path = "./Calcolo percorso&markers/path sfere.mat"
                                                              &
   diameter = (.vite ricircolo sfere.DIAMETRO SFERA)
                                                        &
   num prefix = (1)
!
```

Per la creazione delle sfere bisogna far leggere da ADAMS la macro creata tramite il comando macro read. Tra le opzioni di lettura si è imposta la creazione del dialog box quando si lancia la macro dal Command Navigator (finestra presente nel programma che permette di vedere e utilizzare tutti i comandi di ADAMS) e la possibilità di annullare i comandi della macro durante il suo funzionamento. Usando il comando della macro, ossia richiamando il nome generate\_spheres\_xyz e imponendo i parametri desiderati, si ottengono le sfere all'interno del percorso. Si sottolinea che quando si impone il diametro delle sfere nella macro, non si utilizza il valore numerico ma si crea il collegamento diretto con la design variable DIAMETRO\_SFERA, così se eventualmente si vuole variare la dimensione dei corpi è possibile farlo cambiando semplicemente il valore della variabile.

### 4.5 Forze

Una volta definiti i corpi presenti nel sistema vite a ricircolo di sfere, si inseriscono le forze presenti nel sistema che corrispondono ai contatti presenti tra i vari corpi e alla forza esterna che corrisponde a tutte le azioni aereodinamiche che agiscono sulla superficie libera dell'EMA che si scaricano sulla madrevite; si esclude ovviamente la gravità già definita precedentemente nel Capitolo 4.1. Il codice relativo ai contatti è il seguente:

```
!---IMPORT MACROS
macro read &
    macro_name = generate_contact_force_sphere_sphere &
    file_name =
```

```
"./Macro/generate contact force sphere sphere.cmd" &
   wrap in undo = yes
                       æ
   create panel = yes
macro read &
   macro name = generate contact force sphere nut &
   file name =
   "./Macro/generate_contact force sphere nut.cmd" &
   wrap in undo = yes &
   create panel = yes
macro read &
  macro name = generate contact force screwshaft sphere &
   file_name = "./Macro/generate_contact force screwshaft
   sphere.cmd" &
  wrap in undo = yes &
  create panel = yes
1
!CREATE DESIGN VARIABLES FOR STIFFNESS
file testdata read splines &
 model name = .vite ricircolo sfere &
  file name =
  ".\Calcolo parametri contatto\rigidezze contatto.mat"
var set var=RIGIDEZZA Sfera Sfera CONTATTO
real=(eval(.vite ricircolo sfere.spline 1.ys[1]))
var set var=RIGIDEZZA Sfera Madrevite CONTATTO
real=(eval(.vite ricircolo sfere.spline 1.ys[2]))
var set var=RIGIDEZZA Vite Sfera CONTATTO
real=(eval(.vite ricircolo sfere.spline 1.ys[3]))
data element delete
data element=.vite ricircolo sfere.spline 1
!CREATE DESIGN VARIABLES FOR DAMPING
var set var=SMORZAMENTO Sfera Sfera CONTATTO real=1330
var set var=SMORZAMENTO Sfera Madrevite CONTATTO real=3700
var set var=SMORZAMENTO Vite Sfera CONTATTO real=3570
!CREATE DESIGN VARIABLE FOR EXPONENT CONTACT FORCE
var set var=ESPONENTE CONTATTO real=1.5
1
!CREATE DESIGN VARIABLE FOR DMAX
var set var=DMAX CONTATTO real=0.1
!CREATE DESIGN VARIABLE FOR COULOMB FRICTION
var set var=MU STATICO CONTATTO real=0.11
var set var=MU DINAMICO CONTATTO real=0.06
var set var=VELOCITA MU STATICO CONTATTO real=0.1
var set var=VELOCITA MU DINAMICO CONTATTO real=10
```

```
!
!CREATE CONTACT FORCES SPHERE-SPHERE
generate contact force sphere &
  model = .vite ricircolo sfere &
   file path =
   "./Calcolo percorso&markers/path sfere.mat" &
   stiffness = (.vite ricircolo sfere.
  RIGIDEZZA Sfera Sfera CONTATTO) &
  damping = (.vite ricircolo sfere.
   SMORZAMENTO Sfera Sfera CONTATTO)
                                      &
  exponent = (.vite ricircolo sfere.ESPONENTE CONTATTO) &
   dmax = (.vite ricircolo sfere.DMAX CONTATTO)
                                                 &
  mu static = (.vite ricircolo sfere.
  MU STATICO CONTATTO)
                        &
  mu dynamic = (.vite ricircolo sfere.
  MU DINAMICO CONTATTO)
                          &
   stiction transition velocity = (.vite ricircolo sfere.
  VELOCITA MU STATICO CONTATTO) &
   friction transition velocity = (.vite ricircolo sfere.
  VELOCITA MU DINAMICO CONTATTO)
                                  &
  num prefix = (1)
1
!CREATE CONTACT FORCES SPHERE-NUT
generate contact force sphere nut
                                  &
  model = .vite ricircolo sfere &
   file path =
   "./Calcolo percorso&markers/path sfere.mat" &
   stiffness = (.vite ricircolo sfere.
  RIGIDEZZA Sfera Madrevite CONTATTO)
                                        &
  damping = (.vite ricircolo sfere.
   SMORZAMENTO Sfera Madrevite CONTATTO)
                                         &
  exponent = (.vite ricircolo sfere.ESPONENTE CONTATTO)
                                                          &
   dmax = (.vite ricircolo sfere.DMAX CONTATTO)
                                                 &
  mu static = (.vite ricircolo sfere.
  MU STATICO CONTATTO) &
  mu dynamic = (.vite ricircolo sfere.
  MU DINAMICO CONTATTO)
                          &
   stiction transition velocity = (.vite ricircolo sfere.
  VELOCITA MU STATICO CONTATTO) &
   friction_transition_velocity = (.vite_ricircolo sfere.
  VELOCITA MU DINAMICO CONTATTO) &
  num prefix = (1)
!
!CREATE CONTACT FORCES SCREWSHAFT-SPHERE
generate contact force screwshaft sphere
                                          &
  model = .vite ricircolo sfere &
```

```
file path =
"./Calcolo percorso&markers/path sfere.mat"
                                              δ
stiffness = (.vite ricircolo sfere.
RIGIDEZZA Vite Sfera CONTATTO)
damping = (.vite ricircolo sfere.
SMORZAMENTO Vite Sfera CONTATTO)
                                   &
exponent = (.vite ricircolo sfere.ESPONENTE CONTATTO)
                                                        &
dmax = (.vite ricircolo sfere.DMAX CONTATTO)
                                               &
mu static = (.vite ricircolo sfere.
MU STATICO CONTATTO)
                      &
mu dynamic = (.vite ricircolo sfere.
MU DINAMICO CONTATTO)
                       &
stiction transition velocity = (.vite ricircolo sfere.
VELOCITA MU STATICO CONTATTO)
                               &
friction transition velocity = (.vite ricircolo sfere.
VELOCITA MU DINAMICO CONTATTO)
num prefix = (1)
```

!

Per creare i contatti all'interno del modello si applica la lettura delle varie macro tramite il comando macro read e successivamente si eseguono i comandi fornendo i parametri necessari, nella stessa modalità con cui si sono create le sfere nel Capitolo 4.4. Per i parametri di contatto sono state create delle design variables con il comando var set in maniera da poter gestire i contatti senza modificarli direttamente. Per quanto riguarda la parte elastica dei contatti, le rigidezze di contatto sono state importate tramite il comando file testdata read splines direttamente dal file rigidezze\_contatto.mat, calcolate in MATLAB eseguendo l'iter esposto nel Capitolo 3.2 mentre l'esponente è stato imposto pari a 1,5 come voluto dalla teoria di Hertz. Invece, per quanto riguarda la parte viscosa dei contatti, i coefficienti di smorzamento sono stati scelti considerando le esperienze di altri utenti su modelli simili a quello preso in esame ed effettuando simulazioni di calibratura tramite lo script DINAMICA\_faceting\_tolerance (vedere Capitolo 4.10). I valori dei coefficienti di smorzamento sono riportati qui sotto:

| Tipologia di contatto | $C_{MAX} [Ns/mm]$ |
|-----------------------|-------------------|
| Sfera – Vite          | 3570              |
| Sfera – Madrevite     | 3700              |
| Sfera – Sfera         | 1330              |

Tabella 4.1 – Valori dei coefficienti di smorzamento utilizzati nei contatti in ADAMS

Il valore della compenetrazione massima è stato scelto semplicemente utilizzando un valore consigliato dalla pratica. Per la componente d'attrito, i valori dei coefficienti d'attrito sono stati scelti dalla letteratura [3] considerando il caso di una vite a ricircolo di sfere lubrificata mentre i valori delle velocità  $V_s$  e  $V_d$  sono stati assunti in maniera da ottenere risultati realistici. La forza esterna è stata definita nello script nel seguente modo:

```
!---CREATE EXTERNAL FORCE
var set var=EXTERNAL FORCE MAG real=1000
marker create
marker=.vite ricircolo sfere.ground.
MARKER EXTERNAL FORCE &
    location = 0.0, 0.0, 0.0 &
    orientation = 0.0, -90.0, 0.0
entity attributes
entity name=.vite ricircolo sfere.ground.
MARKER EXTERNAL FORCE
                       &
    name vis=off
marker create
marker=.vite ricircolo sfere.Madrevite 1.
MARKER EXTERNAL FORCE
                      &
    location = 0.0, 0.0, 0.0
                              &
    orientation = 0.0, -90.0, 0.0
entity attributes
entity name=.vite ricircolo sfere.
Madrevite 1.MARKER EXTERNAL FORCE
                                   &
    name vis=off
force create direct single component force
                                            &
    single component force name=
    .vite ricircolo sfere.EXTERNAL FORCE
                                           &
    type of freedom=translational
                                   8
    action only = on &
    i marker name=.vite ricircolo sfere.
    Madrevite 1.MARKER EXTERNAL FORCE &
    j marker name=.vite ricircolo sfere.
    ground.MARKER EXTERNAL FORCE &
    function = "-(.vite ricircolo sfere.EXTERNAL FORCE MAG)
    *STEP5(time, 0, 0, 0.5, 1)"
entity attributes
entity name=.vite ricircolo sfere.EXTERNAL FORCE
                                                   &
    name vis=off
```

La seguente forza è lungo la direzione negativa dell'asse y del sistema di riferimento fisso ground ed ha un modulo che è stato variato a seconda della tipologia di simulazione che si è voluta fare. Innanzitutto è stata definita la variabile per il modulo della forza tramite il comando var set, successivamente per poter creare la forza è necessario avere due marker, creati tramite il comando marker create, uno solidale con il ground e uno solidale con la madrevite. Per definire la direzione corretta della forza, i due marker devono avere l'asse z lungo la direzione della forza voluta, questo spiega i corrispettivi angoli della sequenza di rotazioni 313 definiti nell'orientation. La posizione dei centri dei due marker corrisponde all'origine del sistema di riferimento fisso del ground in quanto la forza agisce lungo l'asse della madrevite, quindi non serve collocarli esattamente sulla chiocciola. Tramite il comando force create direct single\_component\_force, si può creare la forza, indicando che essa sia solo d'azione (in quanto è possibile creare azioni dinamiche di azione-reazione) e che sia di tipo traslatorio. La function scelta per la forza permette di non raggiungere subito il valore del modulo, ma tramite lo step si raggiunge dopo 0,5 s, inoltre è presente il segno negativo in quanto si oppone al moto della madrevite. Si sottolinea che il marker *i* rappresenta il corpo che è soggetto all'azione dinamica mentre il marker j rappresenta chi la esercita; un ragionamento molto simile si potrà riscontrare nella definizione dei vincoli e del motion nei prossimi Capitolo 4.6 e Capitolo 4.7.



Figura 4.9 – Esempio di andamento della forza esterna che agisce sulla madrevite nel tempo

### 4.6 Vincoli

Il modello necessita di alcuni vincoli cinematici per eliminare alcuni gradi di libertà di vite e madrevite, in maniera da rispecchiare la realtà fisica del sistema vite a ricircolo di sfere. Le sfere non necessitano di alcun vincolo cinematico in quanto devono essere libere di muoversi all'interno della pista e in realtà esse sono vincolate dinamicamente dalle forze di contatto. I vincoli che si impongono

sono quelli legati a vite e madrevite. Per quanto riguarda la vite si impone un revolute che corrisponde ad una cerniera, la quale elimina cinque gradi di libertà, tre di traslazione e due di rotazione, lasciando un solo grado di libertà di rotazione: tramite questa modalità la vite non trasla e ruota unicamente intorno al suo asse. Per quanto riguarda invece la madrevite, inizialmente si impongono un perpendicular e un cylindrical dove il primo elimina unicamente un grado di libertà di rotazione mentre il secondo elimina due gradi di libertà di traslazione e due di rotazione, in maniera tale che la madrevite trasli solo lungo la direzione dell'asse della vite; la somma dei due vincoli genera un translational, il quale elimina due gradi di libertà di traslazione e tre di rotazione ma non è stato inserito per il seguente motivo: il cylindrical, che è stato inserito all'interno del modello e poi successivamente eliminato, è stato utilizzato in una prima fase di progettazione solo per la calibrazione del modello, come verrà esplicitato nei successivi Capitoli; l'unico vincolo che si vuole realmente sulla madrevite è il perpendicular, in maniera da vincolare solo il grado di rotazione intorno all'asse della vite, lasciando tutte le irregolarità di posizione e di orientamento della chiocciola per rendere più realistiche le analisi che saranno fatte successivamente. Si analizza la costruzione di ogni singolo vincolo generato all'interno del modello:

Revolute Joint: per definire il seguente vincolo, servono due marker tramite il comando marker create, uno fisso appartenente al ground, l'altro mobile appartenente alla vite ed essi, collocati nell'origine del sistema globale del ground, devono avere l'asse z lungo l'asse della vite al quale corrisponde il grado di libertà presente nel vincolo; a questo punto, utilizzando il comando constraint create joint revolute, si crea dando come input come marker *i* quello mobile mentre come marker *j* quello fisso rispetto al quale si osserva la rotazione della vite.

```
!REVOLUTE JOINT SCREWSHAFT-GROUND
marker create
marker=.vite ricircolo sfere.ground.MARKER REVOLUTE
                                                      &
    location = 0.0, 0.0, 0.0
                             &
    orientation = 180.0, 90.0, 180.0
entity attributes
entity name=.vite ricircolo sfere.ground.
MARKER REVOLUTE &
    name vis=off
marker create
marker=.vite ricircolo sfere.Vite.MARKER REVOLUTE
                                                    &
    location = 0.0, 0.0, 0.0 &
    orientation = 180.0, 90.0, 180.0
entity attributes
entity name=.vite ricircolo sfere.Vite.
```

```
MARKER REVOLUTE &
    name vis=off
constraint create joint Revolute &
    joint name=.vite ricircolo_sfere.REVOLUTE_JOINT
                                                      &
    i marker name=.vite ricircolo sfere.
    Vite.MARKER REVOLUTE
                         &
    j marker name=.vite ricircolo sfere.
    ground.MARKER REVOLUTE
constraint attribute
constraint name=.vite ricircolo sfere.REVOLUTE JOINT
name vis=off &
    color=MAGENTA
                   &
    size=40
```

Perpendicular Joint: anche qui servono due marker, uno fisso al ground, l'altro mobile appartenente alla madrevite, i quali, collocati nell'origine del sistema globale del ground, devono avere gli assi z che giacciono su un piano perpendicolare all'asse della vite e che siano perpendicolari tra di loro; a questo punto utilizzando il comando constraint create primitive\_joint perpendicular si crea il vincolo fornendo come marker *i* il sistema di riferimento mobile sulla madrevite mentre come marker *j* il sistema di riferimento fisso sul ground

```
!PERPENDICULAR JOINT 1
marker create marker=.vite ricircolo sfere.
Madrevite 1.MARKER PERPENDICULAR 1 &
    location = 0.0, 0.0, 0.0
    orientation = 0.0, 0.0, 180.0
entity attributes entity name=.vite ricircolo sfere.
Madrevite 1.MARKER PERPENDICULAR 1
    name vis=off
marker create marker=.vite ricircolo sfere.
ground.MARKER PERPENDICULAR 1 &
    location = 0.0, 0.0, 0.0 &
    orientation = 90.0, 90.0, 0.0
                           entity name=.vite ricircolo sfere.
entity
           attributes
ground.MARKER PERPENDICULAR 1
                               &
    name vis=off
constraint create primitive joint Perpendicular
                                                 &
    jprim name=.vite ricircolo sfere.
    PERPENDICULAR 1
                    &
    i marker name=.vite ricircolo sfere.
    Madrevite 1.MARKER PERPENDICULAR 1
                                       &
    j marker name=.vite ricircolo sfere.
```

```
ground.MARKER_PERPENDICULAR_1
constraint attribute
constraint_name=.vite_ricircolo_sfere.
PERPENDICULAR_1 &
    name_vis=off &
    color=ORANGE &
    size=40
```

Cylindrical joint: in questo caso servono due markers, uno mobile con la madrevite e uno mobile con la vite ed essi devono avere l'asse z coincidente con l'asse della vite a cui corrisponderanno il grado di libertà di traslazione e quello di rotazione lungo quella direzione; tramite il comando constraint create joint cylindrical si genera il vincolo, dando come input come marker *i* il sistema di riferimento mobile della madrevite mentre come marker *j* il sistema di riferimento mobile della vite.

```
!CYLINDRICAL JOINT 1
marker create
marker=.vite ricircolo_sfere.Vite.
MARKER CYLINDRICAL 1 &
    location = 0.0, 0.0, 0.0
                             æ
    orientation = 0.0, -90.0, 0.0
entity attributes
entity name=.vite ricircolo sfere.
Vite.MARKER CYLINDRICAL 1 &
    name vis=off
marker create marker=.vite ricircolo sfere.
Madrevite 1.MARKER CYLINDRICAL 1 &
    location = 0.0, 0.0, 0.0
    orientation = 0.0, -90.0, 0.0
entity attributes entity name=.vite ricircolo sfere.
Madrevite 1.MARKER CYLINDRICAL 1 &
    name vis=off
constraint create joint Cylindrical
    joint name=.vite ricircolo sfere.
    CYLINDRICAL JOINT 1 &
    i marker name=.vite ricircolo sfere.
    Madrevite 1.MARKER CYLINDRICAL_1 &
    j marker name=.vite ricircolo sfere.
    Vite.MARKER CYLINDRICAL 1
constraint attribute
constraint name=.vite ricircolo sfere.
CYLINDRICAL JOINT 1 &
    name vis=off &
    color=GREEN &
```

```
size=40
constraint delete
constraint name=.vite ricircolo sfere.CYLINDRICAL JOINT 1
```

# 4.7 Motion

Il motion è un moto che si può fornire ad un vincolo presente nel modello che può essere rotatorio o traslatorio e può rappresentare una posizione, una velocità o un'accelerazione. Esso può essere definito tramite delle function create nel Fuction Builder. L'utilizzo di motion elimina un grado di libertà in quanto la rotazione o la traslazione che si impone esplicita totalmente la variabile. Nel seguente modello, si impone un motion sul revolute joint, imponendo una velocità e simulando un collegamento al motore. Il codice utilizzato è il seguente:

```
!---CREATE MOTION
var set var=ROTATIONAL_VELOCITY_RPM real=100
constraint create motion
motion_name=.vite_ricircolo_sfere.MOTORE &
   joint=.vite_ricircolo_sfere.REVOLUTE_JOINT &
   type_of_freedom=rotational &
   function = "-ROTATIONAL_VELOCITY_RPM*((2*PI)/60)
   *STEP5(time, 0, 0, 0.2, 1)" &
   time_derivative="velocity"
constraint attributes
constraint_name=.vite_ricircolo_sfere.MOTORE name_vis=off
!
```

Innanzitutto si crea la variabile per il modulo della velocità che, da come si evidenzia dal nome della design variable stessa, è valutata in rpm. Successivamente tramite il comando constraint create motion si crea il moto desiderato, imponendo il vincolo in cui si vuole definire il grado di libertà, la tipologia della variabile presa in esame, il tipo di moto che si vuole generare e la funzione che lo definisce. La funzione permette di non raggiungere subito il valore del modulo, ma tramite lo step si raggiunge dopo 0,2 s, inoltre è presente un segno – in quanto si vuole una rotazione opposta rispetto al sistema di riferimento del revolute joint. Si sottolinea che nella definizione del motion è presente una costante che permette la conversione da rpm a rad/s.



Figura 4.10 – Esempio di andamento della velocità angolare della vite nel tempo

# 4.8 Marker per l'angolo di contatto

Per il calcolo in fase di post-processing dell'angolo di contatto, è necessario avere due ulteriori marker, uno solidale alla vite e uno solidale alla madrevite, che abbiano l'asse z coincidente all'asse della vite e l'asse x che intersechi l'inizio dell'elica della pista, in maniera da poter valutare successivamente l'angolo di azimut in maniera ottimale. Il codice utilizzato è il seguente:

```
!---CREATE MARKER FOR SCREW BETA
marker create
marker=.vite ricircolo sfere.Vite.V MARKER THETA
                                                   &
    location = 0.0, 0.0, 0.0
                              &
    orientation = 0.0, -90.0, -90.0
entity attributes
entity name=.vite ricircolo sfere.Vite.V MARKER THETA
                                                        &
    name vis=off
!
!CREATE DESIGN VARIABLE FOR M MARKER THETA
file testdata read splines
                            &
  model name = .vite ricircolo sfere
                                       &
  file name = ".\Calcolo percorso&markers
  \dati angolo contatto.mat"
var set var=THETA OFFSET M
real=(eval(.vite ricircolo sfere.spline 1.ys[5])) units=angle
!
    !---CREATE MARKER FOR NUT BETA
```

```
marker create
marker=.vite_ricircolo_sfere.Madrevite_1.M_MARKER_THETA &
    location = 0.0, (.vite_ricircolo_sfere.
    Y_OFFSET_M), 0.0 &
    orientation = 0.0, -90.0, (-(.vite_ricircolo_sfere.
    THETA_OFFSET_M+90.0))
entity attributes entity_name=.vite_ricircolo_sfere.
Madrevite_1.M_MARKER_THETA &
    name_vis=off
!
```

Oltre a definire come design variable l'angolo di offset con il comando var set che è di natura unicamente geometrica ed è dedotta dal CAD della vite/madrevite, si definiscono i due marker tramite il comando marker create.

### 4.9 Measure per il rendimento

La measure è una misura che si può creare all'interno di ADAMS per valutare le varie grandezze fisiche già definite all'interno del modello oppure delle funzioni create tramite Function Builder, durante la simulazione. In questo caso la measure è stata utilizzata per valutare il rendimento meccanico del sistema, che in questo caso è stato definito nel seguente modo:

$$\eta_M = \frac{F_{ext,MV} \cdot v_{y,MV}}{C_{y,V} \cdot \omega_{y,V}}$$
(4.6)

Dove  $F_{ext,MV}$  corrisponde alla forza esterna che agisce sulla madrevite,  $v_{y,MV}$  è la velocità lineare della madrevite,  $C_{y,V}$  corrisponde alla coppia presente sulla vite e  $\omega_{y,V}$  è la velocità angolare della vite. Si sottolinea che la y a cui fa riferimento l'equazione (4.6), corrisponde alla y del sistema di riferimento globale del ground. Il codice relativo è il seguente:

```
!---CREATE MEASURES FOR EFFICIENCY
measure cre function
measure=.vite_ricircolo_sfere.RENDIMENTO
function="ABS((SFORCE(EXTERNAL_FORCE, 0, 3, 0)
*VY(Madrevite_1.cm, 0, 0, 0))/(MOTION(MOTORE, 1, 7, 0)
*WY(Vite.cm, 0, 0)))" &
    units = no_units &
    create_measure_display = no
    ! set measure plotting attributes
measure attributes measure=.vite_ricircolo_sfere.RENDIMENTO
!
```

Semplicemente tramite il comando measure cre function e fornendo come funzione l'equazione (4.6), si crea la measure. Il display della misura che si sta effettuando è stato eliminato in maniera da togliere l'implementazione grafica graduale della measure durante la fase di simulazione del modello, in maniera da non appesantire a livello computazionale la macchina.

### 4.10 Script di simulazione

Lo script di simulazione è un insieme di comandi che impone al programma di effettuare determinate simulazioni, di tipo dinamico, cinematico e statico. Nel seguente modello sono state definite due tipologie: la prima è stata utilizzata unicamente per calibrare il modello, la seconda per ottenere i risultati voluti. Il codice relativo è il seguente:

```
!---CREATE SIMULATION
simulation script create &
    sim script name =
    .vite ricircolo sfere.DINAMICA faceting tolerance
                                                        &
    initial static = no &
    type = dynamic &
    step size = 1E-06
                       &
    end time = 5E-03
!
simulation script create &
    solver commands = "SIMULATE/STATIC", "SIMULATE/DYNAMIC,
    END=1.0, DTOUT=1.0E-04"
                            &
    sim script name = .vite ricircolo sfere.DINAMICA 1s
!
```

Tramite il comando simulation script create si possono generare i seguenti script ed essi possono essere o quelli predefiniti per le simulazioni dinamiche, oppure quelli in cui si possono scegliere i comandi da fornire al programma. La seconda tipologia è molto interessante se l'utente vuole personalizzarsi la simulazione e vuole creare qualche particolarità come per esempio attivare o disattivare vincoli, motion o altri elementi oppure cambiare valori di design variable nel bel mezzo della simulazione.

# 4.11 Requests

Le requests sono degli elementi che permettono di raggruppare dei risultati a seconda delle richieste dell'utente. Esse sono molto importante se si vuole ridurre il peso in memoria dei risultati, in maniera da poterli gestire liberamente senza complicazione. Richiamando la macro descritta nel Capitolo 4.3.5, si legge la macro con il comando macro read e si utilizza il nuovo comando per creare tutte insieme e contemporaneamente le requests necessarie, fornendo opportunamente i parametri necessari. Il codice è il seguente:

```
macro read
           &
  macro_name = generate_requests
                                   &
   file name = "./Macro/generate requests.cmd"
                                                 &
   wrap in undo = yes
                      &
   create panel = yes
!
generate requests
                   &
   model = .vite ricircolo sfere &
   file path = "./Calcolo percorso&markers/
   path sfere.mat" &
   num prefix = (1)
!
```

### **5** SETTINGS DEL MODELLO

All'interno del modello, nello script vite\_ricircolo\_sfere.cmd, è presente inizialmente una fase di definizione dei vari parametri relativi alle impostazioni della simulazione, in cui si impongono le specifiche volute per l'integrazione delle equazioni, per le implementazioni grafiche e per il file .res risultati. Il codice relativo è il seguente:

```
!---SET PARAMETERS FOR SIMULATION
I.
!DYNAMICS
executive set numerical model=.vite ricircolo sfere
integrator=HHT
!
!CONTACTS
!executive set preferences model=.vite ricircolo sfere
contact geometry library=parasolids
var set var=FACETING TOLERANCE real=18000
executive set preferences model=.vite ricircolo sfere
contact faceting tolerance=(.vite ricircolo sfere.
FACETING TOLERANCE)
!
!EXECUTABLE
simulation set solver preference=external
var set var=THREAD COUNT real=8
executive control set preferences
thread count=(.vite ricircolo sfere.THREAD COUNT)
!
!EOUILIBRIUM
var set var=MAXIT real=1500
var set var=ERROR real=5.0E-03
var set var=IMBALANCE real=5.0E-02
var set var=TLIMIT real=1.0E-03
var set var=ALIMIT real=4.0E-04
var set var=STABILITY real=1.0E+03
executive set equilibrium model = .vite ricircolo sfere &
  maxit = (.vite ricircolo sfere.MAXIT)
                                           &
  error = (.vite ricircolo sfere.ERROR)
                                           &
   imbalance = (.vite ricircolo sfere.IMBALANCE)
                                                   &
  tlimit = (.vite ricircolo sfere.TLIMIT)
                                             &
  alimit = (.vite_ricircolo_sfere.ALIMIT)
                                             &
   stability = (.vite ricircolo sfere.STABILITY)
!
!OUTPUT
output set results model = .vite ricircolo sfere
                                                   æ
```

```
displacements = on &
   velocities = off
                     &
   accelerations = off
                        8
   applied forces = off
                         &
   reaction forces = off
                          &
   friction summary = off
                           &
   contact incidents = off
                            &
   data structures = off
                          &
   floating markers = off
                            &
   linear = off
                 &
   system elements = off
                          &
   node incidents = off &
   request = on &
   formatted = on
                   &
   xrf = on &
   decimal places = 7
!
!DISPLAY
simulation set show all_messages=yes
simulation single set update=none
simulation single set monitor=none
T
```

Nei successivi Capitoli si analizzeranno i vari campi dei settings, spiegando le opzioni presenti e le scelte effettuate nel relativo caso.

# 5.1 Dynamics

In questo setting si definisce quale integrator scegliere nel solver, che corrisponde ad un controllo interno dell'integrazione numerica delle equazioni differenziali non lineari del moto nelle simulazioni dinamiche. I vari integrator possono essere di due tipologie: nella forma originale dove ad ogni iterazione numerica l'errore di tutte le soluzioni delle variabili deve essere minore rispetto alla tolleranza imposta mentre nella forma modificata non è necessario che la tolleranza richiesta sia rispetta per tutte le soluzioni. In alcuni casi la forma modificata può essere utile per velocizzare la simulazione riducendo i tempi di calcolo I vari integratori possono essere classificati nel seguente modo:

| Integrator                                    | Original Corrector  | Modified Corrector  |
|---|---|---|
| GSTIFF 13,<br>WSTIFF 13                       | "Si effettua un controllo sugli<br>spostamenti, sui modi di flettere<br>dei corpi flessibili, sulle forze<br>applicate e sulle variabili di stato<br>delle equazioni differenziali; in<br>pratica tutte le variabili ad<br>eccezione delle forze di contatto e<br>dei moltiplicatori lagrangiani"   | "Si effettua un controllo sugli<br>spostamenti, sui modi di flettere<br>dei corpi flessibili e sulle variabili<br>di stato delle equazioni<br>differenziali"  |
| GSTIFF SI2,<br>WSTIFF SI2,<br>HASTIFF SI1/SI2 | "Si effettua un controllo sugli<br>spostamenti e sulle velocità dei<br>corpi, sui modi e sulle velocità dei<br>corpi flessibili, sulle forze<br>applicate e sulle variabili di stato<br>delle equazioni differenziali; in<br>pratica tutte le variabili ad<br>eccezione delle forze di contatto,<br>dei moltiplicatori lagrangiani e dei<br>moltiplicatori lagrangiani delle<br>velocità" | "Si effettua un controllo sugli<br>spostamenti e sulle velocità dei<br>corpi, sui modi e sulle velocità dei<br>corpi flessibili e sulle velocità<br>relative e sulle variabili di stato<br>delle equazioni differenziali" |
| HHT,<br>NEWMARK                               | "Si effettua un controllo sulle<br>accelerazioni, sulle accelerazioni<br>dei corpi flessibili e sulle variabili<br>di stato delle equazioni<br>differenziali"   | "Sono ignorate le fluttuazioni dei<br>moltiplicatori lagrangiani"   |

Tabella 5.1 – Tipologie di correttori e descrizioni relative [11]

Nel seguente modello, considerando che per ogni sfera ci sono tre forze di contatto, si evidenzia un numero notevole di contatti i quali aumentano di molto il tempo di calcolo, pertanto si predilige l'integrator HHT, in quanto si effettua un controllo completo del modello, comprese soprattutto le forze di contatto. Si è utilizzata la forma originale dell'integrator, in maniera da avere i risultati il più affidabili e completi possibili, anche se ciò non è completamente possibile e ciò si evidenzierà nei prossimi Capitolo 5.2.

### 5.2 Contacts

Le operazioni tridimensionali del contatto, che corrispondono all'algoritmo con il quale si inserisce il contatto e si calcola all'interno delle simulazioni, possono essere definite con due diverse "Geometry Library" che sarebbero due modi per identificare le superfici a contatto:

- Parasolids che è la modalità in cui viene valutato esattamente il profilo geometrico della superficie a contatto, il che rende l'algoritmo molto preciso ma allo stesso tempo molto lento
- Default\_Library che è la modalità basata sul RAPID, algoritmo che ricostruisce la superficie tramite una mesh ti tipo poligonale, il che rende il calcolo molto più rapido ma meno preciso e ciò dipende dal *faceting tolerance*; questo parametro indica quanto viene discretizzata la superficie tramite la mesh di triangoli

Di default viene inserito il RAPID ma ovviamente si può scegliere quale tipologia utilizzare. Nel seguente caso, si è osservato che il Parasolid era troppo lento nel calcolo, pertanto si è preferito il RAPID e si è studiato il parametro principale, ossia il *faceting tolerance*. Per definirlo, si sono effettuate delle simulazioni con lo script "DINAMICA\_faceting\_tolerance", impostando il sistema con due sole sfere, senza alcun moto sulla vite e senza la forza di gravità, variando il parametro fino al raggiungimento del valore ottimale. Come riferimento di partenza si è scelta la forza di contatto tra Sfera\_1\_1 e madrevite per capire l'entità del *faceting tolerance* e ciò si può osservare nella Figura 5.1:



Figura 5.1 – Andamento della forza di contatto tra Sfera\_1\_1 e madrevite
Da come si evidenzia in Figura 5.1, a bassi valori di *faceting tolerance*, sono già presenti delle notevoli compenetrazioni tra corpi e questo non ha senso in quanto, per come è stato costruito il modello e considerando il fatto che non è presente gravità, non c'è contatto. Di conseguenza si è aumentato il valore del *faceting tolerance* per discretizzare meglio le superfici presenti, raggiungendo un valore ottimale del seguente parametro che è pari a 14000. Successivamente si è considerata, sempre nelle stesse tipologie di simulazioni, la forza di contatto tra Sfera\_1\_2 e madrevite, da come si osserva nella Figura 5.2:



*Figura 5.2 – Andamento della forza di contatto tra Sfera\_1\_2 e madrevite* 

A questo punto si può osservare che il valore ottimale è 17000, in quanto la forza è nulla e così dovrebbe essere per le condizioni imposte nel modello. Facendo una simulazione con motion pari a 0 sulla vite e senza la gravità ma con tutte le sfere, si è evidenziato che con questo valore del parametro, alcune forze non sono completamente nulle, pertanto il valore scelto e deciso per le simulazioni è 18000.

L'utilizzo di faceting tolerance molto elevati, in particolare superiori a 1000, generalmente è sconsigliato perché richiedono più tempo nel calcolo, ma in questo caso, osservando le simulazioni fatte e i tempi impiegati, si è notato qualcosa di diverso e ciò è visibile nella Figura 5.3:



Figura 5.3 – Andamento del tempo di calcolo al variare del faceting tolerance

Quando il valore del faceting tolerance è molto basso, la discretizzazione non è delle migliori quindi avviene subito la compenetrazione tra la sfera e la pista, facendo subito intervenire la forza di contatto, generando forze elevate e irreali e rallentando il calcolo. Aumentando il valore del faceting tolerance, si osserva che la forza di contatto è minore in quanto la compenetrazione è sempre più piccola e di conseguenza il tempo di calcolo si riduce. Andare verso valori troppo elevati di questo parametro, porterebbe ad una compenetrazione sempre più piccola o addirittura nulla tra i corpi ma allo stesso tempo una discretizzazione troppo fitta che è più precisa ma complica il calcolo e ne aumenta il tempo.

### 5.3 Executable

In questa sezione si sceglie l'opzione sulla preferenza di come effettuare la simulazione del modello, ossia se utilizzando sia ADAMS Solver (che è la parte del programma che effettua il calcolo delle equazioni del moto) che ADAMS View (corrisponde all'interfaccia grafica del programma) che corrisponde all'opzione Internal, effettuando l'upgrade graduale dell'animazione durante il calcolo, oppure se utilizzare unicamente ADAMS Solver che corrisponde all'opzione External. Nel seguente modello si è utilizzata la seconda opzione in quanto l'implementazione ad ogni iterazione dell'animazione è molto pesante a livello computazionale e allunga i tempi di calcolo della simulazione. Inoltre si è imposto il thread count ossia quanti thread devono lavorare in parallelo nella fase di calcolo; in questo caso si è imposto 8, in quanto il processore utilizzato è un quad core con 8 thread.

# 5.4 Equilibrium

L'equilibrio statico serve per eliminare tutti transitori prima di avviare l'analisi dinamica del modello, ciò permette a madrevite e alle sfere di assestarsi sulla vite, così tutte le grandezze cinematiche o dinamiche partono da una situazione di regime. Il calcolo della posizione di equilibrio statico è di tipo iterativo e necessita dei seguenti parametri esposti in [11] e nell'ADAMS Help (comando F1):

- Alimit che corrisponde al massimo incremento angolare permesso ad ogni iterazione durante la simulazione statica
- Tlimit che corrisponde al massimo incremento lineare permesso ad ogni iterazione durante la simulazione statica
- > Maxit che corrisponde al numero massimo di iterazioni per trovare l'equilibrio statico
- Error che corrisponde alla soglia di convergenza della correzione relativa; il processo iterativo effettuato durante l'analisi di equilibrio non può convergere prima che tutte le correzioni relative rilevanti siano inferiori a questo valore
- Stability che corrisponde ad una sorta di coefficiente di smorzamento che permette di stabilizzare il processo di iterazione ed evitare che le iterazioni divergano
- Imbalance che corrisponde alla soglia di convergenza dello squilibrio delle equazioni durante la simulazione statica ed equivale dimensionalmente ad una forza; il processo iterativo effettuato durante l'analisi di equilibrio non può convergere prima che ogni squilibrio di equazione sia inferiore a questo valore

La simulazione statica, utilizzando i parametri di default, non funzionava in questo modello, pertanto si sono eseguite numerose prove fino a quando non si sono trovati i valori ottimali per definire la posizione di equilibrio; la procedura è stata la seguente, partendo come primo step con i parametri di default imposti da ADAMS:

- > 1) Aumentare il parametro di Maxit, ottenendo il valore di 1500
- ➢ 2) Aumentare il parametro di Error, ottenendo il valore di 5E-03
- > 3) Aumentare il parametro di Imbalance, ottenendo il valore di 5E-02
- 4) Ridurre i parametri di Tlimit e di Alimit, ottenendo i valori per il primo di 1E-03 mentre per il secondo di 4E-04
- > 5) Aumentare il parametro di Stability, ottenendo il valore di 1000

#### 5.5 Output

In questa parte legata ai settings, si definisce come immagazzinare e cosa salvare i risultati ottenuti durante le simulazioni effettuate all'interno di ADAMS. Innanzitutto si sceglie quali grandezze salvare nel file .res dei risultati; le opzioni che si possono scegliere sono molteplici ma nel nostro caso si sono scelte unicamente le request che sono gli elementi a cui si è interessati all'interno del sistema vite a ricircolo di sfere e le posizioni nello spazio dei corpi in maniera da poter riprodurre le animazioni delle simulazioni effettuate. Alcune opzioni sono state eliminate perché alcune sono già comprese nelle request, altre perché non presenti nel modello. L'opzione più importante da eliminare è il node\_incidents che sono degli step aggiuntivi che sono creati in maniera del tutto autonoma da ADAMS per identificare tutti gli istanti temporali in cui avviene il contatto; questi appesantiscono di molto i risultati, pertanto risulta necessario eliminare il flag. Inoltre è possibile scegliere il formato del file risultato .res, che può essere scelto tra ASCII, Binary e XML; nel seguente caso, per ridurre ulteriormente il peso dei files, si è utilizzata la modalità XML, in quanto, essendo formattata, si possono imporre alcuni vincoli di scrittura, in questo caso è stato scelta una limitazione dei numeri decimali, pari a 7.

## 5.6 Display

Tramite queste opzioni si può scegliere cosa far vedere graficamente in ADAMS durante la fase di simulazione. Durante il calcolo bisogna evitare tutto ciò che pesa sulla macchina, pertanto si evitano gli updates delle toolbar e della grafica ad ogni step in maniera che tutto proceda il più velocemente. L'unica cosa che è settata a livello grafico sono i messaggi del programma che riguardano la simulazione statica e quella dinamica, in maniera da poter visionare i progressi della simulazione in fase d'opera.

# 6 POST-PROCESSING E RISULTATI



Figura 6.1 – Vista del modello in ADAMS

Una volta creato il modello e impostato i settings all'interno di ADAMS, è possibile effettuare le simulazioni desiderate: esse sono state realizzate variando principalmente due parametri, i quali sono la velocità angolare del motore e la forza esterna che agisce sulla madrevite che corrisponde alla risultante delle azioni dinamiche presenti sulla superficie libera. Dalle analisi effettuate si sono osservati ed elaborati vari dati, in modo da poter evidenziare alcuni aspetti specifici del modello. Le simulazioni principali effettuate sono divise in due categorie:

- Nove analisi in cui sono effettuate le combinazioni di tre parametri della forza esterna sulla madrevite che sono 1000, 2000 e 3000 N e tre parametri di velocità angolare della vite che sono 100, 200 e 300 rpm
- Due analisi in cui si mantengono fisse la forza esterna sulla madrevite e la velocità angolare della vite, che corrispondono a 21000 N e 1000 rpm, ma in una di esse si aumenta il diametro di una delle sfere dell'1% rispetto alle condizioni standard (che corrisponde in ADAMS alla Sfera\_1\_71)

Ogni file .res ottenuto nelle simulazioni, contiene le requests create all'interno di ADAMS e pertanto esse sono state esportate dal programma sottoforma di spreadsheet (.tab) che sono un file formattato simile al un file Excel. Tramite un lavoro di importazione e di elaborazione dati, è stato possibile osservare i risultati ottenuti nelle varie simulazioni in ambiente MATLAB, effettuando in alcuni casi dei calcoli per ottenere dei dati impliciti nei files.

## 6.1 Rendimento

Tramite principalmente le nove simulazioni effettuate variando forza e velocità come precedentemente descritto, utilizzando la measure definita nel Capitolo 4.9, si è calcolata per ogni simulazione il valore principale del rendimento utilizzando la funzione median di MATLAB, la quale permette di calcolare la mediana escludendo eventuali irregolarità dei dati presenti. Con i valori significativi per ogni simulazione, è stato possibile creare una mappatura dell'efficienza del sistema a seconda dei valori assegnati di forza esterna che agisce sulla madrevite e di velocità angolare della vite, presente in Figura 6.2:



Figura 6.2 – Mappatura del rendimento



Figura 6.3 – Andamento del rendimento in funzione della forza che agisce sulla madrevite



Figura 6.4 – Andamento del rendimento in funzione della velocità della vite

Da come si può evidenziare in Figura 6.2, per le velocità e per le forze scelte, la variazione del rendimento si aggira intorno al valore di 0,95 che corrisponde ad una efficienza ottima del sistema. Da come si può osservare nelle Figura 6.3 e Figura 6.4, sembra che aumentando la velocità angolare della vite e la forza esterna che agisce sulla madrevite, il rendimento aumenta leggermente. In realtà, considerando gli studi effettuati precedentemente [3], il rendimento varia nel seguente modo: se si aumenta la forza esterna, il rendimento aumenta fino al raggiungimento di un massimo, successivamente decresce mentre se invece si aumenta la velocità angolare della vite, il rendimento della vite, il rendimento di un massimo, successivamente decresce mentre se invece si aumenta la velocità angolare della vite, il rendimento

decresce. Il reale andamento che ci si aspetta non è visibile nelle Figure 6.2, 6.3 e 6.4, questo perché il range in cui si sono fatte variare forza e velocità sono una percentuale molto piccola del range totale di funzionamento della vite a ricircolo di sfere e quindi non si può osservare la reale curva di rendimento. Per poter vedere meglio l'effetto di questi due parametri si sono fatte altre quattro simulazioni aggiuntive che sono le seguenti:

- Due simulazioni dove si è mantenuta costante la forza esterna sulla madrevite pari a 3000 N e la velocità angolare della vite si è imposta pari a 800 e 1600 rpm
- Due simulazioni dove si è mantenuta costante la velocità angolare della vite pari a 300 rpm e la forza esterna sulla madrevite si è imposta pari a 18000N e 25000 N



Figura 6.5 – Andamento del rendimento in funzione della forza che agisce sulla madrevite con le simulazioni aggiuntive



Figura 6.6 – Andamento del rendimento in funzione della velocità della vite con le simulazioni aggiuntive

Da come si può osservare in Figura 6.5, aumentando la forza esterna che agisce sulla madrevite, il rendimento aumenta raggiungendo un massimo e poi successivamente decresce, proprio come ci si aspettava. Osservando invece la Figura 6.6, aumentando la velocità angolare della vite, il rendimento non varia molto e si mantiene circa costante: questo grafico sembra anomalo ma in realtà, non avendo inserito le dissipazioni per attrito volvente e per ventilazione, è plausibile che questo parametro non influisca sull'efficienza del sistema considerando la modellazione realizzata.

### 6.2 Angolo di contatto

Considerando le stesse simulazioni del Capitolo 6.1 e applicando le equazioni presenti nel Capitolo 3.3, è stato possibile calcolare in ambiente MATLAB gli angoli di contatto per ogni sfera presente nel modello e per ogni analisi effettuata all'interno di ADAMS. Da qui è possibile ottenere i vari andamenti degli angoli di contatto, un esempio è presente in Figura 6.7 considerando la simulazione in cui si sono imposti 3000 N per la forza esterna mentre 300 rpm per la velocità angolare della vite:



Figura 6.7 – Andamento nel tempo dell'angolo di contatto della Sfera\_1\_14

Da come si può osservare nella Figura 6.7, l'andamento dell'angolo di contatto, a seconda del segno della coordinata b del centro della sfera nel sistema di riferimento *otnb* e di dove avviene il contatto (o sulla vite o sulla madrevite), assume un valore che varia tra 0 e circa 45° quando è nel percorso elicoidale (ciò dipende dalle equazioni (3.32), (3.33), (3.34), (3.35)) mentre quando è nella fase di ricircolo assume dei valori casuali e non desiderati, in quanto le formule utilizzate valgono unicamente nel tratto elicoidale. Ciò permette di capire quando una sfera entra od esce dalla fase di ricircolo. A questo punto, si è visto come varia il valore dell'angolo di contatto a seconda della velocità angolare della vite e della forza esterna, considerando un determinato istante (0,06 s) e osservando la Figura 6.8 e Figura 6.9:



Figura 6.8 – Andamento dell'angolo di contatto in funzione della velocità angolare della vite, Sfera\_1\_50



Figura 6.9 – Andamento dell'angolo di contatto in funzione della forza esterna, Sfera\_1\_50

Da come si evidenzia nelle Figura 6.8 e Figura 6.9 e tenendo conto del Capitolo 3.3, all'aumentare della forza esterna e della velocità angolare della vite, aumenta leggermente il  $\beta_V$  mentre diminuisce leggermente nel caso del  $\beta_M$  e ciò è in accordo con gli studi fatti in letteratura [9].

#### 6.3 Velocità di rivoluzione e di rotazione

Considerando gli studi effettuati precedentemente [9] ed utilizzando le stesse simulazioni del Capitolo 6.1 e 6.2, si sono calcolate la velocità angolare di rivoluzione della sfera  $\omega_M$  attorno all'asse della vite e quella di rotazione della sfera  $\omega_R$  intorno al proprio centro. Per quanto riguarda la  $\omega_M$ , essa è definita come la derivata temporale dell'angolo di azimut  $\theta$  (parametro definito e usato nel Capitolo 3), quindi si calcola nel seguente modo:

$$\omega_M = \frac{d\theta}{dt} \tag{6.1}$$

Da come si osserva nell'equazione (6.1), per definire la velocità angolare di rivoluzione della sfera si utilizza una semplice derivata del primo ordine. Si sottolinea che l'angolo  $\theta$  è stato definito nel sistema di riferimento ox'y'z' (vedere Capitolo 3.1). Considerando i sistemi di riferimento discussi nel Capitolo 3, per quanto riguarda la  $\omega_R$ , servono le componenti di velocità angolari della sfera nel sistema di riferimento *otnb* per effettuare il calcolo; esse sono presenti nei file risultati .res ma sono nel sistema di riferimento *oxyz* del ground. Si definiscono le seguenti ulteriori matrici:

$$\begin{cases} \omega_x \\ \omega_y \\ \omega_z \end{cases} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{cases} \omega_{x'} \\ \omega_{y'} \\ \omega_{z'} \end{cases} = T_1 \begin{cases} \omega_{x'} \\ \omega_{y'} \\ \omega_{z'} \end{cases}$$
(6.2)

$$\begin{cases} \omega_{x'} \\ \omega_{y'} \\ \omega_{z'} \end{cases} = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{cases} \omega_{x_V} \\ \omega_{y_V} \\ \omega_{z_V} \end{cases} = T_V \begin{cases} \omega_{x_V} \\ \omega_{y_V} \\ \omega_{z_V} \end{cases}$$
(6.3)

Dove nella matrice presente nella (6.3), l'angolo  $\gamma$  corrisponde alla rotazione che effettua la vite durante il funzionamento ed è definito all'interno di MATLAB tramite la funzione cumtrapz con la quale si integra la velocità angolare della vite in funzione del tempo. Si sottolinea che l'angolo  $\theta$  è stato definito nel sistema di riferimento  $ox_V y_V z_V$  (vedere Capitolo 3.3). A questo punto è possibile calcolare le varie componenti nel sistema di riferimento *otnb* nel seguente modo:

$$\begin{cases} \omega_x \\ \omega_y \\ \omega_z \end{cases} = T_5 T_V T_1 \begin{cases} \omega_t \\ \omega_n \\ \omega_b \end{cases}$$
 (6.4)

In questa maniera è possibile calcolare la velocità  $\omega_R$ :

$$\omega_R = \sqrt{\omega_t^2 + \omega_n^2 + \omega_b^2} \tag{6.5}$$

Nella Figura 6.10 è possibile osservare le tre componenti di velocità angolare nel sistema di riferimento *otnb* nella simulazione con forza esterna sulla madrevite pari a 3000 N mentre la velocità angolare della vite pari a 300 rpm:



Figura 6.10 – Andamento nel tempo delle velocità angolari nel sistema di riferimento otnb

Da come si può evidenziare dalla Figura 6.10, la componente lungo la direzione t è trascurabile mentre quelle principali sono quelle sul piano normale al filetto ossia lungo gli assi n e b: ciò si può intuire considerando che il contatto avviene nel piano nb, pertanto la componente lungo t a livello teorico dev'essere piccola. Inoltre si osserva che il valore delle componenti di velocità lungo n e lungo b sono circa uguali e ciò si può dedurre dall'angolo di contatto che è circa 45°, il quale impone la direzione principale di rotazione della velocità  $\omega_R$ , la quale rappresenta il modulo delle tre componenti. L'andamento delle componenti  $\omega_n \in \omega_b$  è circa simile a quello della velocità angolare della vite, infatti si stabilizza al valore temporale di 0,2 s. Queste componenti sono molto interessanti in quanto generalmente è difficile capire cosa succede all'interno della vite a ricircolo di sfere.

A questo punto sono state analizzate le  $\omega_R$  e le  $\omega_M$  in funzione della forza esterna che agisce sulla madrevite e della velocità angolare della vite:



Figura 6.11 – Andamento nel tempo  $\omega_M$  in funzione della forza esterna che agisce sulla madrevite, Sfera\_1\_50



Figura 6.12 - Andamento nel tempo di  $\omega_M$  in funzione della velocità angolare della vite, Sfera\_1\_50



Figura 6.13 – Andamento nel tempo  $\omega_R$  in funzione della forza esterna che agisce sulla madrevite, Sfera\_1\_50



Figura 6.14 – Andamento nel tempo di  $\omega_R$  in funzione della velocità angolare della vite, Sfera\_1\_50

Da come si evidenzia dalle Figure 6.11, 6.12, 6.13 e 6.14 e dagli studi effettuati precedentemente [9], la velocità angolare di rivoluzione e quella di rotazione della sfera non sono influenzate dal carico assiale che agisce sulla madrevite ma bensì dalla velocità angolare della vite, difatti più questa aumenta, più  $\omega_M$  e  $\omega_R$  crescono in valore. Si riporta l'andamento di  $\omega_M$  e  $\omega_R$  al variare della velocità angolare della vite  $\omega_V$ :



Figura 6.15 – Andamento della  $\omega_R$  e della  $\omega_M$  al variare della velocità angolare della vite

Come si evidenzia nella Figura 6.15, la velocità  $\omega_R$  assume valori che sono il triplo rispetto a quelli della  $\omega_M$  e ciò viene confermato anche dalla letteratura [9].

Per avere un ulteriore conferma della validità del modello, si è calcolata la velocità angolare di rivoluzione  $\omega_M$  con la formula presente in letteratura [9] che è la seguente:

$$\omega_{M,th} = \omega_V \frac{1}{1 + \frac{\left(1 + \frac{r_{b0}}{R_m} \cos \beta_{MV}\right) (\cos \beta_V + tan\gamma \sin \beta_V)}{\left(1 - \frac{r_{b0}}{R_m} \cos \beta_V\right) (\cos \beta_{MV} + tan\gamma \sin \beta_{MV})}}$$
(6.6)

Dove  $\gamma$  è l'angolo giroscopico che viene assunto pari 45°. Considerando la simulazione in cui si è posta forza esterna che agisce sulla madrevite pari a 1000 N e velocità angolare della vite pari a 100 rpm, si è effettuato il confronto, ottenendo i seguenti andamenti:



Figura 6.16 – Confronto della velocità di rivoluzione tra il valore ottenuto nel modello e quello teorico

Dalla Figura 6.16 si evidenzia che i due andamenti sono molto simili, pertanto ciò che si è ottenuto nel modello rispecchia i risultati presenti in letteratura [9].

## 6.4 Velocità di strisciamento

Tramite le equazioni (6.1), (6.2), (6.3), (6.4), (6.5) e tenendo conto delle simulazioni utilizzate nei Capitoli 6.1, 6.2, e 6.3, è possibile calcolare la velocità lineare di strisciamento tra sfera- vite e tra sfera-madrevite. Per farlo si definisce innanzitutto *d* che corrisponde al modulo delle componenti che identificano la posizione del centro della sfera rispetto al sistema di riferimento mobile della vite  $ox_V y_V z_V$  (definito nel Capitolo 3.3) che ruota alla velocità angolare  $\omega$  nel seguente modo:

$$d = \sqrt{x_V^2 + y_V^2 + z_V^2} \tag{6.7}$$

A questo punto si definisce la velocità di strisciamento tra sfera e vite attraverso la seguente formula:

$$\left|\overline{V_{S,V}}\right| = \left|\overline{V_{Bbl}} - \overline{V_{BSW}}\right| \tag{6.8}$$

Dove:

$$\overrightarrow{V_{Bbl}} = \begin{cases} d(\omega_M - \omega) + \omega R_m \cos \alpha_{elica} + r_{b0}(\omega_b \cos \beta_V - \omega_n \sin \beta_V) \cos \alpha_{elica} \\ 0 \\ -\omega R_m \sin \alpha_{elica} \end{cases} \begin{cases} t \\ b \end{cases}$$
(6.9)

$$\overrightarrow{V_{BSW}} = \begin{cases} (R_m - r_{b0} \cos \beta_V) \cos \alpha_{elica} \\ -r_{b0} \sin \beta_V \sin \alpha_{elica} \\ -(R_m - r_{b0} \cos \beta_V) \sin \alpha_{elica} \end{cases} \begin{cases} t \\ n \\ b \end{cases}$$
(6.10)

Successivamente si definisce la velocità di strisciamento tra sfera e vite nel seguente modo:

$$\left|\overrightarrow{V_{S,MV}}\right| = \left|\overrightarrow{V_{Abl}} - \overrightarrow{V_{Ant}}\right| \tag{6.11}$$

Dove:

$$\overrightarrow{V_{Abl}} = \begin{cases} d(\omega_M - \omega) + \omega R_m \cos \alpha_{elica} + r_{b0}(\omega_b \cos \beta_{MV} - \omega_n \sin \beta_{MV}) \\ 0 \\ -\omega R_m \sin \alpha_{elica} \end{cases} \begin{cases} t \\ b \end{cases}$$
(6.12)

$$\overrightarrow{V_{Ant}} = \begin{cases} -\omega R_m \sin \alpha_{elica} \tan \alpha_{elica} \\ 0 \\ -\omega R_m \sin \alpha_{elica} \end{cases} \begin{cases} t \\ n \\ b \end{cases}$$
(6.13)

A questo punto è possibile calcolare le varie velocità d'interesse:



Figura 6.17 – Andamento nel tempo della velocità di strisciamento tra sfera e madrevite al variare della velocità angolare della vite



Figura 6.18 – Andamento nel tempo della velocità di strisciamento tra sfera e madrevite al variare della forza esterna che agisce sulla madrevite



Figura 6.19 – Andamento nel tempo della velocità di strisciamento tra sfera e vite al variare della velocità angolare della vite



Figura 6.20 – Andamento nel tempo della velocità di strisciamento tra sfera e vite al variare della forza esterna che agisce sulla madrevite

Da come si può osservare nelle Figure 6.17, 6.18, 6.19, 6.20, le velocità lineari di strisciamento sia tra sfera-vite e tra sfera-madrevite non sono influenzate dalla forza esterna che agisce sulla madrevite ma bensì dalla velocità angolare della vite, infatti più essa aumenta, più  $|\overrightarrow{V_{S,V}}| \in |\overrightarrow{V_{S,MV}}|$  aumentano. Inoltre si evidenzia che la velocità di strisciamento tra sfera e madrevite è minore rispetto a quella tra sfera e vite.

### 6.5 Forze di contatto

In questo Capitolo si tiene conto principalmente delle due simulazioni effettuate con forza esterna che agisce sulla madrevite pari a 21000 N e con velocità angolare della vite pari a 1000 rpm, la differenza sta nel fatto che in una le sfere sono tutte uguali, nell'altra ce n'è una con il diametro l'1% più grande, che corrisponde alla Sfera\_1\_71 in ADAMS. Come detto precedentemente, le forze di contatto presenti sono quelle tra sfera-sfera, vite-sfera e madrevite-sfera; le ultime due sono le più interessanti e per capirlo si osserva la Figura 6.21 e la Figura 6.22:



Figura 6.21 – Andamento nel tempo degli angoli di contatto e della forza di contatto tra vite-sfera per la Sfera\_1\_71



Figura 6.22 – Andamento nel tempo degli angoli di contatto e della forza di contatto tra madrevite-sfera per la Sfera\_1\_71

Da come si evidenzia nella Figura 6.21 e Figura 6.22, anche le forze di contatto in esame, come gli angoli di contatto, evidenziano la fase di ricircolo: esse assumono valori o trascurabili a contatto con la madrevite oppure nulli a contatto con la vite quando sono nel tratto di ricircolo mentre hanno valori diversi da 0 quando sono nel tratto elicoidale in quanto le sfere sono sottoposte al carico dinamico esterno che agisce sulla madrevite. Questo è un altro modo per poter evidenziare dai dati ricavati dalle varie simulazioni quando una sfera entra o esce dal tratto di ricircolo.

A questo punto si sono osservate le forze di contatto tra vite-sfera e madrevite-sfera nelle due simulazioni d'interesse per vedere cosa succede nei risultati ottenuti in ADAMS quando una sfera presente all'interno della pista è più grande rispetto alle condizioni standard:



Figura 6.23 – Andamento nel tempo degli angoli di contatto e della forza di contatto tra vite-sfera per la Sfera\_1\_71 nelle due simulazioni d'interesse



Figura 6.24 – Andamento nel tempo degli angoli di contatto e della forza di contatto tra vite-sfera per la Sfera\_1\_71

Teoricamente se la sfera è più grande rispetto alle condizioni standard, si dovrebbe bloccare all'interno della pista e dovrebbe andare a contatto con la pista in quattro punti. Osservando le animazioni dei risultati ottenuti, si vede che la madrevite avanza linearmente senza alcun problema particolare e che la sfera compenetra la pista in quattro punti: questo è plausibile in quanto non c'è nessun vincolo particolare che impedisca questo fatto all'interno del programma. Inoltre si osserva nelle Figure 6.23 e 6.24 che l'angolo di contatto non cambia molto rispetto alle condizioni standard, questo perché il calcolo tiene conto di come si sposta la sfera e non di dove effettivamente avviene il contatto tra superfici. Per capire l'anomalia all'interno del modello, se si osservano le forze di contatto tra vite-sfera e madrevite-sfera nella Sfera\_1\_71, si evidenziano valori importanti rispetto alle condizioni standard: questo perché la compenetrazione tra i vari corpi in gioco è elevata e considerando come sono definiti i contatti all'interno di ADAMS, genera un modulo delle forze molto alto. A questo punto si osserva la distribuzione del carico assiale in un certo istante (0,7 s) sulle varie sfere presenti nella vite a ricircolo di sfere:



Figura 6.25 – Distribuzione del carico assiale sulle sfere – Contatto Sfera-Madrevite



Distribuzione del carico assiale sulle sfere - Contatto Sfera-Vite

Figura 6.26 - Distribuzione del carico assiale sulle sfere – Contatto Sfera-Vite

Da come si evidenzia nelle Figure 6.25 e 6.26, nella distribuzione del carico sulle sfere nella condizione standard, il carico non è costante su tutte le sfere perché la madrevite non è assialsimmetrica a causa della presenza del foro del canale di ricircolo e quindi tende ad inclinarsi a causa del suo stesso peso. Nell'altra condizione si osserva che buona parte della forza esterna che agisce sulla madrevite si scarica sulla Sfera\_1\_71 rendendo la distribuzione del carico assiale non omogenea, infatti i contatti tra sfera-vite e sfera-madrevite assumono valori molto superiori rispetto agli altri contatti delle sfere. Sugli altri contatti si vede invece che alcuni hanno valori maggiori rispetto alle condizioni standard, altri viceversa, questo perché la presenza della sfera più grande all'interno della pista destabilizza il sistema e crea squilibrio dinamico e ciò sarà più chiaro nel Capitolo 6.6. Si osserva inoltre che le sfere immediatamente vicine alla sfera più grande hanno valori di forza decisamente più piccoli perché, nonostante la madrevite si inclini, comunque non entrano granchè in contatto perché sono molto vicini alla sfera grossa.

A questo punto si è calcolata la pressione massima nell'area di contatto: innanzitutto si sottolinea che considerando la teoria di Hertz dalla letteratura [4], l'area di contatto è un'ellisse caratterizzata da due semiassi definiti nel seguente modo (considerando i parametri del Capitolo 3.2):

$$a = a^* \left(\frac{3F_{contatto}}{2(A+B)E^*}\right)^{1/3}$$
(6.14)

$$b = b^* \left(\frac{3F_{contatto}}{2(A+B)E^*}\right)^{1/3}$$
(6.15)

Dove:

$$a^* = \left(\frac{2\kappa^2 \mathcal{E}(\kappa)}{\pi}\right)^{1/3} \tag{6.16}$$

$$b^* = \left(\frac{2\mathcal{E}(\kappa)}{\pi\kappa}\right)^{1/3} \tag{6.17}$$

La pressione massima nell'ellisse di contatto si calcola tramite la seguente formula:

$$p = \frac{3}{2} \frac{F_{contatto}}{\pi ab} \tag{6.18}$$



Figura 6.27 – Pressione hertziana nel tempo nelle due condizioni considerate

Generalmente la vite a ricircolo di sfere è paragonabile ad un cuscinetto a sfere, quindi considerando la teoria di Hertz e il contatto sfera-sfera, si può definire la tensione massima come  $\sigma_{MAX} =$  $0,62p_{contatto}$ . Un materiale per questa tipologia di applicazioni può essere il 100Cr6, che presenta tensioni di snervamento che variano tra i 1500 e i 2200 MPa, di conseguenza, tenendo conto dell'ipotesi precedente, pressioni di contatto che variano tra i 2420 e i 3550 Mpa. Osservando la Figura 6.27, nelle condizioni standard si hanno pressioni di contatto accettabili considerando il ragionamento fatto mentre quando c'è la sfera più grossa si hanno valori molto elevati che possono snervare la Sfera\_1\_71, ovalizzandola e incastrandola all'interno della pista. Ciò nel modello non è visibile nelle animazioni in quanto i contatti presenti sono perfettamente elastici e i corpi sono rigidi e non si deformano.

Per avere una visione più completa dei contatti, si sono osservate le varie forze di contatto su una sfera in funzione della velocità angolare della vite e della forza esterna che agisce sulla madrevite in maniera da visualizzare l'influenza di questi due parametri. Si sono considerate le nove simulazioni usate nei Capitoli 6.1, 6.2, 6.3 e 6.4 e si sono ottenuti i seguenti grafici:



Figura 6.28 – Andamento delle forze di contatto al variare della forza esterna che agisce sulla madrevite, Sfera\_1\_50



*Figura 6.29 – Andamento delle forze di contatto al variare della velocità angolare della vite, Sfera\_1\_50* Considerando la Figura 6.28, si può osservare come aumentando la forza esterna che agisce sulla madrevite, il modulo dei contatti assume valori più elevati, questo semplicemente perché le sfere sono più caricate dalle azioni dinamiche esterne. Considerando invece la Figura 6.29, si può evidenziare che le forze di contatto assumono circa lo stesso intervallo di valori, ma all'aumentare della velocità angolare della vite si osserva un andamento più oscillante e disturbato, ciò succede in quanto il contatto tra pista e sfera avviene con una maggiore frequenza.

# 6.6 Accelerazione della madrevite

Come detto precedentemente, considerando le simulazioni utilizzate nel Capitolo 6.5, la presenza di una sfera più grossa all'interno della pista porta ad uno squilibrio dinamico della chiocciola e ciò è visibile analizzando le varie componenti dell'accelerazione della madrevite nel sistema di riferimento del ground di ADAMS:



Figura 6.30 – Andamento nel tempo delle componenti dell'accelerazione della madrevite nel sistema di riferimento del ground

Da come si osserva in Figura 6.30, la presenza della sfera col diametro più grande genera delle accelerazioni molto elevate, in particolare sono molto importanti quelle lungo gli assi perpendicolari all'asse della vite e questo evidenzia lo squilibrio dinamico della chiocciola. Inoltre anche la componente lungo l'asse della vite è molto più elevata rispetto alle condizioni standard. Si osserva che, nella fase di ricircolo, osservando anche le Figure 6.23 e 6.24, tutte le componenti di accelerazione assumono un valore simile nelle due simulazioni e questo perché la Sfera\_1\_71 è entrata nella fase di ricircolo e quindi non deve sopportare il carico esterno. Si evidenzia che a seconda di dove si colloca la Sfera\_1\_71 all'interno del percorso elicoidale, si generano le accelerazioni lungo x e lungo z in maniera da opporsi ai contatti che sono presenti sulla sfera esaminata; ciò comporta l'andamento oscillante delle seguenti componenti di accelerazione, in particolare il numero delle oscillazioni corrisponde circa al numero di avvolgimenti dell'elica presente nella madrevite.

#### 7 CONCLUSIONI

Questa attività di tesi ha permesso di creare un modello che potesse rappresentare in maniera completa le azioni dinamiche e cinematiche che avvengono all'interno della vite a ricircolo di sfere, in maniera da riprodurre il funzionamento reale, compresa la fase di ricircolo la quale in altre modellazioni è difficile da realizzare. Il modello è parametrizzato all'interno del programma quindi, se si sceglie una nuova tipologia di sistema, basta semplicemente importare il CAD e il path delle sfere voluto. Oltre alle animazioni con le quali si può vedere come funziona il sistema, è possibile osservare i vari risultati o su ADAMS o su MATLAB per capire se il tutto funziona regolarmente. In realtà tramite i dati ottenuti si vede molto di più, infatti quando è si è inserita la sfera più grande all'interno della pista, le anomalie si sono evidenziate nei risultati ottenuti e non nell'animazione dove non si è notata alcuna criticità. La modellazione effettuata all'interno di ADAMS è molto completa e fornisce tutti gli elementi che permettono di capire cosa succede nel sistema, in particolare le sfere le quali, essendo racchiuse tra vite e madrevite, sono difficili da studiare nella loro cinematica e dinamica.

Il modello presenta un difetto importante che è il tempo di calcolo: difatti esso, a seconda delle simulazioni effettuate, impiega numerose ore se non giorni per ottenere i risultati desiderati, questo perché i contatti a livello computazionale pesano molto sulla macchina. D'altro canto c'è da dire che ADAMS non gestisce un numero di thread maggiore di 8 e in più è stata scelta una tipologia di sistema con molte sfere e quindi questo ha portato ad una inevitabile lunga attesa per il file .res.

Il modello è ancora migliorabile effettuando le dovute implementazioni, soprattutto per quanto riguarda il lancio di molteplici simulazioni, il salvataggio di ogni analisi effettuata e l'esportazione degli spreadsheets voluti per il post-processing in MATLAB, ma anche legate alla modellazione in sé come per esempio l'inserimento del bushing sul revolute joint che permette di rielaborare il vincolo rendendolo reale e non più ideale e quindi cedevole. Esso può essere reso ancora più generale ed efficiente, in maniera da poter soddisfare ogni esigenza ed interesse dell'utente.

# NOMENCLATURA

| $C_{\mathcal{Y},V}$    | Coppia motrice sulla vite lungo l'asse y nel sistema di riferimento oxyz  |
|------------------------|---|
| $D_m$                  | Diametro dell'elica nominale  |
| D <sub>madrevite</sub> | Diametro esterno della madrevite  |
| D <sub>vite</sub>      | Diametro nominale della vite  |
| Ε                      | Modulo di Young   |
| F <sub>contatto</sub>  | Forza di contatto tra i corpi I e II  |
| $F_{ext,MV}, F_{ext}$  | Forza esterna che agisce assialmente sulla madrevite  |
| Н                      | Offset lungo n nel piano normale al filetto dei centri di curvatura degli<br>archi rispetto al centro di un arco          |
| L                      | Offset lungo b nel piano normale al filetto dei centri di curvatura degli<br>archi rispetto al centro di un arco          |
| L <sub>rett</sub>      | Modulo del vettore BD nel piano tb  |
| L <sub>vite</sub>      | Lunghezza della vite  |
| $P_h$                  | Passo dell'elica nominale   |
| R                      | Modulo del vettore che identifica l'asse del foro assiale della madrevite<br>nel piano xz nel sistema di riferimento oxyz |
| <i>R</i> <sub>1</sub>  | Modulo del vettore BC nel piano tb  |
| $R_m$                  | Raggio dell'elica nominale  |
| V <sub>S,MV</sub>      | Velocità lineare di strisciamento tra sfera e madrevite   |
| $V_{S,V}$              | Velocità lineare di strisciamento tra sfera e vite  |
| $v_{y,MV}$             | Velocità lineare della madrevite lungo l'asse y nel sistema di riferimento<br>oxyz  |
| a, b                   | Semiassi dell'ellisse di contatto   |
| k <sub>H</sub>         | Rigidezza di contatto   |
| n <sub>avv</sub>       | Numero di avvolgimenti dell'elica presenti all'interno della madrevite  |

| p   | Pressione massima di contatto   |
|---|---|
| <i>r</i> <sub>b0</sub>                              | Raggio delle sfere  |
| $r_F$   | Raggio dei raccordi nel percorso di ricircolo   |
| $r_{s0}$  | Raggio di curvatura della pista   |
| $\mathcal{Y}_A$                                     | Offset del punto A nel sistema di riferimento oxyz lungo la direzione y   |
| α   | Angolo che permette di identificare la posizione del foro assiale della<br>madrevite nel piano xz nel sistema di riferimento oxyz |
| $\alpha_{elica}$                                    | Angolo dell'elica nominale in corrispondenza di $D_m$   |
| β   | Angolo che permette di identificare la posizione del foro assiale della<br>madrevite nel piano xz nel sistema di riferimento oxyz |
| $\beta_{c,i}$                                       | Angolo di contatto nominale ed ideale assunto per il calcolo delle<br>rigidezze   |
| $\beta_{V,b+}, \beta_{V,b-}, \beta_V$               | Angoli di contatto tra sfera e vite   |
| $\beta_{MV,b+}, \beta_{MV,b-}, \beta_M$             | Angoli di contatto tra sfera e madrevite  |
| δ   | Compenetrazione tra i corpi I e II  |
| $\eta_M$  | Rendimento meccanico della vite a ricircolo di sfere  |
| θ   | Angolo di azimut misurato nel sistema ox'y'z'   |
| ν   | Coefficiente di Poisson   |
| $ \rho_{Ix},  \rho_{Iy},  \rho_{IIx},  \rho_{IIy} $ | Curvature dei solidi a contatto nei sistemi di riferimento $ox_Iy_Iz_I$ e<br>$ox_{II}y_{II}z_{II}$                                |
| $\omega_M$  | Velocità angolare di rivoluzione della sfera attorno all'asse della vite  |
| $\omega_R$  | Velocità angolare di rotazione della sfera intorno al proprio centro  |
| $\omega_t, \omega_n, \omega_b$                      | Componenti di velocità angolare di $\omega_R$   |
| $\omega_{y,V}, \omega_V, \omega$                    | Velocità angolare della vite lungo l'asse y nel sistema di riferimento<br>oxyz  |

# **DESIGN VARIABLES**

| ALIMIT                               | $4 \cdot 10^{-4}$      | Massimo incremento angolare<br>permesso ad ogni iterazione durante<br>la simulazione statica   |
|--------------------------------------|------------------------|--|
| DIAMETRO SFERA                       | 6,35                   | Diametro della sfera   |
| DMAX_CONTATTO                        | 0,1                    | Compenetrazione tra i due corpi a<br>contatto alla quale è presente il<br>coefficiente di smorzamento massimo<br>Soglia di compensanza della |
| ERROR                                | $5 \cdot 10^{-3}$      | sogità al convergenza della<br>correzione relativa durante la<br>simulazione statica   |
| ESPONENTE_CONTATTO                   | 1,5                    | Esponente della parte elastica della<br>forza di contatto  |
| EXTERNAL_FORCE_MAG                   | Variabile              | Forza esterna che agisce sulla<br>madrevite  |
| FACETING_TOLERANCE                   | 1,8 · 10 <sup>4</sup>  | Valore che indica la discretizzazione<br>della superficie tramite la mesh di<br>triangoli  |
| IMBALANCE                            | $5 \cdot 10^{-2}$      | Soglia di convergenza dello<br>squilibrio delle equazioni durante la<br>simulazione statica  |
| MAXIT                                | 1,5 · 10 <sup>3</sup>  | Numero massimo di iterazioni per<br>trovare l'equilibrio statico   |
| MU_DINAMICO_CONTATTO                 | $6 \cdot 10^{-2}$      | Attrito statico del contatto   |
| MU_STATICO_CONTATTO                  | 0,11                   | Attrito dinamico del contatto  |
| RAGGIO_SFERA                         | 3,175                  | Raggio della sfera   |
| RIGIDEZZA_Sfera_Madrevite_CONTATTO   | 5,29 · 10 <sup>5</sup> | Rigidezza di contatto tra sfera e<br>madrevite   |
| RIGIDEZZA_Sfera_Sfera_CONTATTO       | 1,91 · 10 <sup>5</sup> | Rigidezza di contatto tra sfera e sfera  |
| RIGIDEZZA_Vite_Sfera_CONTATTO        | 5,1 · 10 <sup>5</sup>  | Rigidezza di contatto tra sfera e vite   |
| ROTATIONAL_VELOCITY_RPM              | Variabile              | Velocità angolare della vite   |
| SMORZAMENTO_Sfera_Madrevite_CONTATTO | 3,7 · 10 <sup>3</sup>  | Coefficiente di smorzamento massimo<br>della forza di contatto tra sfera e<br>madrevite  |
| SMORZAMENTO_Sfera_Sfera_CONTATTO     | 1,33 · 10 <sup>3</sup> | Coefficiente di smorzamento massimo<br>della forza di contatto tra sfera e<br>sfera  |
| SMORZAMENTO_Vite_Sfera_CONTATTO      | 3,57 · 10 <sup>3</sup> | Coefficiente di smorzamento massimo della forza di contatto tra sfera e vite   |

| STABILITY                     | $1 \cdot 10^{3}$  | "Coefficiente di smorzamento" che<br>permette di stabilizzare il processo di<br>iterazione durante la simulazione<br>statica |
|-------------------------------|-------------------|--|
| THREAD_COUNT                  | 8                 | Numero di thread che devono<br>lavorare in parallelo nella fase di<br>calcolo della simulazione sia<br>dinamica che statica  |
| TLIMIT                        | $1 \cdot 10^{-3}$ | Massimo incremento lineare<br>permesso ad ogni iterazione durante<br>la simulazione statica                                  |
| VELOCITA_MU_DINAMICO_CONTATTO | 10                | <i>Velocità di slittamento a cui corrisponde il valore di attrito statico di contatto</i>                                    |
| VELOCITA_MU_STATICO_CONTATTO  | 0,1               | Velocità di slittamento a cui<br>corrisponde il valore di attrito<br>dinamico di contatto                                    |

## **BIBLIOGRAFIA**

- Fabio Gallerani, «Analisi numerico sperimentale del rendimento nelle viti a ricircolo di sfere,» Tesi di Laurea Magistrale, Politecnico di Milano, Milano, 2012
- [2] Antonio Carlo Bertolino, «Modellazione di sistema vite/madrevite in attuatori elettromeccanici per comandi di volo,» Tesi di Laurea Magistrale, Politecnico di Torino, Torino, 2016
- [3] Antonio Carlo Bertolino, Massimo Sorli, Giovanni Jacazio, Stefano Mauro, «Lumped Parameters Modelling of the EMA's Ball Screw Drive with Special Consideration to Ball Grooves Interactions to Support Model-based Health Monitoring,», Mech. Mach. Theory, 137, pp. 188–210. Politecnico di Torino, 2019
- [4] J-F. Antoine, C.Visa e C. Sauvey, «Approximate analytical model for Hertzian elliptical contact problems,» *Transactions of the ASME*, Vol. 128, pp. 660-664, 2006
- [5] Normativa ISO 3408-4, 2006
- [6] Chin-Chung Wei, Ruei-Syuan Lai, «Kinematical analyses and transmission efficiency of a preloaded ball screw operating at high rotational speeds,» *Mechanism and Machine Theory* 46, pp. 880-898, 2011
- [7] Yongjiang Chen, Wencheng Tang, «Determination of contact stiffness in ball screws considering variable contact angles,» *J. Mechanical Engineering Science*, Vol. 228, pp. 2193-2203, 2013
- [8] Hongkui Jiang, Xianchun Song, Xiangrong Xu, Wencheng Tang, «Multibody dynamics simulation of Ball impact-contact mechanics in Ball Screw Mechanism,» University of Shandong Jianzhu, pp.1320-1323, 2010
- [9] Chin Chung Wei, Jen Fin Lin, «Kinematic Analysis of the Ball Screw Mechanism Considering Variable Contact Angles an Elastic Deformations,» *Journal of Mechanical Design*, Vol.125, pp. 717-733, 2003
- [10] Nannan Xu, Wencheng Tang, Yongjiang Chen, Dafei Bao, Yujie Guo, «Modeling analysis nd experimental study for the friction of a ball screw,» *Mechanism and Machine Theory* 87, pp. 57-69, 2015
- [11] MSC Software Learning Center Lessons, <u>https://mscsoftware.csod.com</u>