



UNIVERSITÀ POLITECNICA DELLE MARCHE  
FACOLTÀ DI INGEGNERIA

---

Corso di Laurea magistrale in Ingegneria Meccanica

**MODELLAZIONE DELLA RIGIDEZZA DI UN MECCANISMO COMPLIANT TRAMITE RETI  
NEURALI**

**MODELING THE STIFFNESS OF A COMPLIANT MECHANISM USING NEURAL  
NETWORKS**

Relatore:

**Prof. Matteo Claudio Palpacelli**

Tesi di Laurea di:

**Sebastiano Santoni**

Correlatore:

**Dott. Paolo Sernani**

**A.A. 2021 / 2022**



## **Abstract**

All'interno di questa trattazione vengono approfonditi alcuni concetti riguardanti una particolare categoria di elementi costruttivi che sta prendendo sempre più piede nelle applicazioni in ambito meccanico, i giunti flessibili. L'utilizzo di meccanismi compliant comporta notevoli vantaggi dal punto di vista economico e realizzativo con la possibilità di realizzare vere e proprie catene cinematiche sfruttando cedevolezza localizzate all'interno di un'unica struttura; ciò è reso possibile dall'associazione di processi di produzione altamente specifici, come l'additive manufacturing, alle ottimizzazioni topologiche realizzabili per mezzo di software di analisi agli elementi finiti. Le relazioni che intercorrono tra le forze esterne applicate al giunto e gli spostamenti e rotazioni di quest'ultimo sono esplicabili tramite opportune matrici di rigidità, tuttavia, nel caso di geometrie più complesse, non è sempre semplice percorrere la via analitica. L'obiettivo che ci si è posto è dunque quello di realizzare un modello, facendo uso delle potenzialità delle reti neurali, che sia in grado di approssimare tali relazioni per un giunto ad intagli circolari: attraverso l'apprendimento su una notevole quantità di campioni sarebbe difatti possibile generalizzare l'analisi per le più svariate tipologie di compliant mechanism.

# Indice

<b>1 Introduzione</b> .....	6
<b>2 Giunti flessibili</b> .....	11
2.1 Relazioni costitutive.....	11
2.2 Giunto lineare.....	15
2.3 Giunto ad intagli.....	20
2.4 Giunto a fasce incrociate.....	25
2.5 Giunto rotoidale.....	28
2.6 Giunto cruciforme.....	30
<b>3 Sistemi di produzione</b> .....	33
3.1 Lavorazione alle macchine utensili.....	33
3.2 Macchine abrasive.....	36
3.4 Elettroformatura.....	39
3.6 Assemblaggio.....	46
<b>4 Reti neurali</b> .....	51
4.1 Caratteristiche generali e perceptrone.....	51
4.2 Overfitting.....	55
4.3 Multi-layer perceptron.....	59
<b>5 Reti neurali applicate a sistemi meccanici</b> .....	63
5.1 Quadrilatero articolato.....	63
5.2 Shallow neural networks in Matlab.....	65
5.3 Reti neurali in Python.....	69
5.5 Support Vector Regression.....	77
<b>6 Reti neurali applicate a meccanismi compliant</b> .....	80
6.1 Analisi FEM tramite Ansys.....	80
6.2 Generazione dei dati.....	84
6.3 Analisi dei risultati.....	86
<b>7 Conclusioni</b> .....	93
<b>Bibliografia</b> .....	95

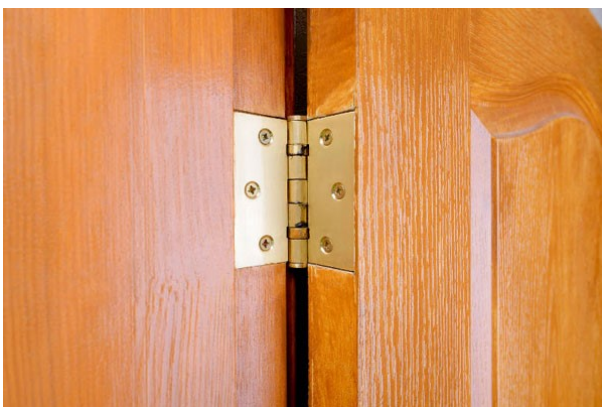


# 1 Introduzione

La parola “Macchina”, nel corso degli anni, si è vista utilizzare nei più disparati ambiti della vita di tutti i giorni andando ad indicare indifferentemente autoveicoli, mezzi d’opera (che riguardino attività di trasporto, edilizia, escavazione o agricoli) e macchinari di produzione industriale. Per non parlare poi dell’accezione del diminutivo “Macchinetta”, che tende a identificare una quantità di elementi costruttivi sempre più varia e che spazia (da regione a regione) tra gli ambiti culinari, fotografici, di distribuzione automatica e via discorrendo. In realtà con “Macchina” Franz Reuleaux intese “Una combinazione di corpi resistenti disposti in modo tale che attraverso di loro le forze della natura siano costrette a produrre lavoro accompagnato da certi determinati movimenti”, difatti è proprio grazie al tipo di interazione che sussiste tra questi componenti rigidi che si possono ottenere catene cinematiche in grado di espletare le movimentazioni desiderate. Le interazioni in questione sono rappresentate dai “vincoli”, che non sono altro che coppie cinematiche destinate a limitare la mobilità relativa tra i componenti presi in considerazione, col fine ultimo di ridurre il numero di gradi di libertà dell'intero sistema. In prima approssimazione questi vincoli vengono considerati ideali tralasciando, quindi, tutte le problematiche relative alla loro effettiva realizzazione fisica: l’utilizzo, ad esempio, di coppie rotoidali con i relativi centri di istantanea rotazione e di coppie prismatiche agenti lungo una linea retta è fondamentale per il corretto studio delle relazioni cinematiche e dinamiche che intercorrono all’interno del sistema meccanico, ma sono tanti i parametri di cui si deve tener conto in seconda battuta.

I principali aspetti da considerare in una reale coppia cinematica possono essere raggruppati in due macroaree ovvero “realizzazione” ed “efficacia”. Per quanto riguarda la realizzazione è importante sottolineare come i giunti, in grado di interfacciare più componenti rigidi, sono spesso composti da una grande quantità di elementi costruttivi di elevata precisione. In quanto tali, sono

richiesti sforzi economici di produzione (maggiore componentistica implica maggior costo) e di assemblaggio che possono andare a gravare sulle spese e sulle tempistiche nella fase di realizzazione del sistema meccanico. Inoltre, andando a ragionare sul ciclo di vita del prodotto che si intende ottenere, è lecito aspettarsi che una quantità più ampia di pezzi implichi necessariamente una manutenzione superiore per un'alta varietà di componenti soggetti a possibili rotture. Relativamente all'efficacia, coerentemente con l'interfacciarsi di numerose unità costruttive, insorgono tutte quelle problematiche relative alla lubrificazione delle superfici a contatto e al conseguente attrito che intercorre: l'adeguato apporto di fluidi antifrizione è necessario al corretto svolgimento delle funzioni da parte della coppia cinematica e ciò comporta, naturalmente, maggiori costi di manutenzione e particolari attenzioni costruttive in fase di progettazione. L'accoppiamento tra superfici geometriche causa poi l'inevitabile nascita di attriti che vanno ad intaccare la precisione del giunto: gli sforzi richiesti per i medesimi movimenti aumentano mentre l'accuratezza di quest'ultimi diminuisce a causa dell'usura. Negli esempi sottostanti sono raffigurati giunti comunemente utilizzati nella vita di tutti i giorni.



*Figura 1 – a) Esempio di giunto rotoidale nella cerniera di una porta; b) Esempio di giunto sferico in una lampada orientabile.*

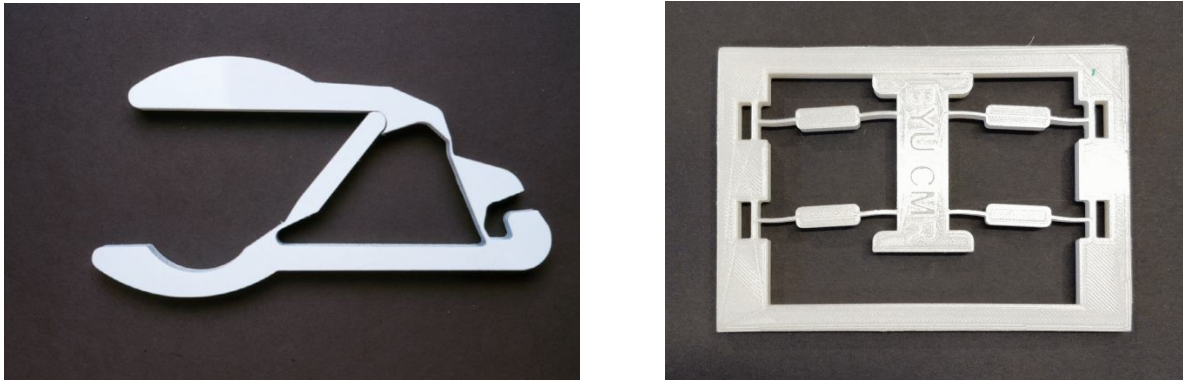
Aldilà dei lati negativi citati precedentemente, è comunque evidente come la rigidità di tali vincoli meccanicamente realizzati sia in grado di garantire la stabilità dei vari centri di rotazione e delle rette d'azione, evitando dunque gli inconvenienti dovuti alle deformazioni elastiche. In questa trattazione vedremo come proprio questi effetti di flessibilità e deformazione elastica (negativi per i giunti comunemente realizzati) siano invece i protagonisti di un'altra classe di elementi chiamati per l'appunto "Giunti Flessibili".

Noti in letteratura anglosassone con il nome di "Flexure Joints", questi giunti permettono di coniugare uno sforzo proveniente dall'esterno con uno spostamento ben definito grazie alla cedevolezza stessa del giunto, trasformando l'energia fornita in input in una deformazione di carattere elastico in output. L'utilizzo di questa tipologia di coppie cinematiche origina quindi dei veri e propri meccanismi flessibili conosciuti come "Compliant Mechanism" o, in altre parole, catene cinematiche ricavate da un unico materiale di partenza che viene accuratamente sagomato in punti ben definiti, in modo tale da conferire le caratteristiche di flessibilità desiderate.

In figura 2a possiamo osservare una pinza realizzata interamente a partire da un unico pezzo di materiale plastico; qui, grazie agli intagli ben visibili realizzati nelle opportune posizioni del meccanismo, è stato possibile ottenere un moltiplicatore della forza in input applicata all'estremità sinistra che sfocia nella parte terminale destra con pressioni di notevole intensità. È bene notare come non siano le dimensioni, quanto le proporzioni tra le varie parti, a garantire l'efficacia del sistema che può quindi andare ad operare su scale estremamente ridotte che avrebbero richiesto grandi sforzi di realizzazione e montaggio. Dall'altra parte invece (fig 2b) è rappresentato un vero e proprio interruttore: il compliant in questo caso è definito bistabile dato che riesce a garantire due posizioni di equilibrio, OFF quando il componente centrale non è in contatto con la cornice esterna e ON quando con l'applicazione di forza, lo scorrimento



longitudinale pone in relazione le due superfici (sulle quali possono essere interfacciati, ad esempio, le componentistiche elettroniche tipiche degli switch).



*Figura 2 – a) Meccanismo compliant con funzione di pinza; b) Meccanismo compliant con funzione di interruttore tra due differenti configurazioni.*

I giunti flessibili sono in grado di ovviare alla quasi totalità dei problemi di realizzazione ed efficacia delle comuni coppie cinematiche, infatti l'abbondanza di componentistica è presto risolta dalla natura intrinseca dei flexure che vengono originati da un unico "grezzo" originario mentre il problema dell'assemblaggio non si pone. Altri vantaggi risiedono poi nelle procedure di ottenimento del prodotto finito: la realizzazione dei compliant può avvenire per mezzo di diversi procedimenti meccanici, lasciando grande libertà al processo di manifattura e dunque alla catena di produzione; un'altra peculiarità è poi quella delle dimensioni, dato che non vi sono né componenti né montaggi che potrebbero risultare complessi per grandezze estremamente contenute, i giunti flessibili sono la soluzione ideale nelle applicazioni microscopiche e di precisione. Il guadagno, inoltre, nella minor massa di questa classe di meccanismi può giocare un ruolo fondamentale nel discorso di riduzione delle potenze in gioco necessarie. Le questioni sollevate, invece, per quanto concerne l'efficacia dei normali giunti meccanici, non trovano riscontri con quelli flessibili: non essendoci parti con moto relativo a contatto decadono tutti i discorsi relativi all'attrito e alla seguente lubrificazione necessaria.

Naturalmente, anche in questo caso, ci sono degli svantaggi riguardanti i flexure che è bene considerare. La corretta stima, ad esempio, delle caratteristiche elastiche dell'elemento e della relativa matrice di rigidezza è un processo che richiede estrema attenzione, al pari della realizzazione stessa del giunto, dove la tolleranza nelle misure (visto l'utilizzo in microscala) è particolarmente rigida. Un altro fattore di cui tener conto è l'entità delle forze in gioco e degli spostamenti ad essi associati: elevati sforzi non sono tollerabili dalle strutture flessibili a causa dei fenomeni di isteresi e di deformazione plastica che andrebbero a ridurre significativamente il limite di resistenza a fatica; allo stesso modo è pericoloso entrare nel campo delle "large deflections" principalmente per motivi di non linearità delle equazioni costitutive che implicano maggiori complessità di calcolo.

L'obiettivo di questa trattazione è proprio quello di svincolare il modello fisico dalla complessa parte analitica che regola le deformazioni elastiche proprie del giunto flessibile. Per fare ciò verranno utilizzate delle reti neurali con il compito di originare output di spostamento a partire dai parametri costruttivi del giunto in questione, come input. Per "allenare" la rete neurale verranno utilizzati i dati ottenuti tramite una serie di simulazioni agli elementi finiti per mezzo del software di calcolo Ansys.

## 2 Giunti flessibili

### 2.1 Relazioni costitutive

Nei giunti protagonisti di questa trattazione, l'aggettivo "flessibili" sta ad indicare la fase elastica nella quale sono soliti lavorare queste coppie cinematiche. La fase elastica di un materiale è quella associata ad un intervallo di tensioni e deformazioni che sono tra loro linearmente legate: fu Robert Hooke (1635-1703) il primo a formulare una relazione di proporzionalità tra tensione normale  $\sigma$  (data dal rapporto tra una forza ed una superficie, [Pa]) e deformazione  $\varepsilon$  (data dal rapporto tra l'allungamento del campione di riferimento e la sua dimensione iniziale, adimensionale) data dalla costante E chiamata "Modulo di Young". È bene far notare come le aree delle sezioni possano variare istante per istante durante la deformazione e, analogamente, è più corretto considerare l'integrale sugli incrementi istantanei di allungamento; così vengono di seguito riportate le differenze tra i valori ingegneristici e reali.

$$\sigma_{eng} = \frac{F}{A_0} \qquad \sigma_{true} = \frac{F}{A} = \frac{F}{A_0} \frac{A_0}{A} = \sigma_{eng}(1 + \varepsilon_{eng}) \qquad (2.1)$$

$$\varepsilon_{eng} = \frac{\Delta L}{L_0} \qquad \varepsilon_{true} = \int \frac{\delta L}{L} = \ln \frac{L_0 + \Delta L}{L_0} = \ln(1 + \varepsilon_{eng}) \qquad (2.2)$$

In questo caso, come spiegato in precedenza, il valore dei carichi e degli spostamenti non può essere troppo elevato e non sarà dunque necessario prestare particolare attenzione alle distinzioni tra le due tipologie di formule. Facendo poi un piccolo passo in avanti, è intuitivo pensare come a sforzi lungo un'asse siano associate deformazioni lungo lo stesso ma in realtà, queste, non sono l'unico effetto: nelle altre direzioni si osserva un allungamento di segno opposto valutabile per mezzo di un secondo parametro  $\nu$  chiamato "Coefficiente di Poisson" in onore di S.D. Poisson (1781-1840). Lo stato tensionale in un punto è l'insieme di tutti i vettori tensione in quel punto: pensando ad un elemento generico di volume infinitesimo  $dV$ , si possono distinguere le

sollecitazioni  $\sigma$ , perpendicolari alla superficie in questione, dalle tensioni  $\tau$  chiamate “tangenziali” e distribuite quindi sul piano tangente alla superficie in quel punto. Le tensioni tangenziali agenti, ad esempio su un elemento piano di area  $dS$ , sono responsabili degli scorrimenti angolari indicati con la lettera  $\gamma$  che rappresentano delle distorsioni, come evidente in figura 3, a cui non sono però associate variazioni d’area.

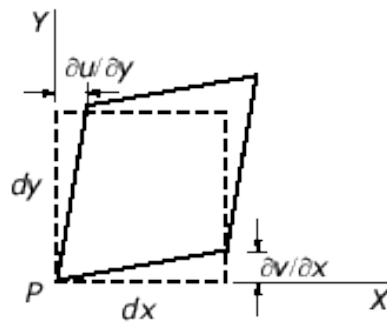


Figura 3 – Distorsioni dell’elemento di materiale preso in considerazione.

$$\varepsilon_x = \frac{\partial u}{\partial x} \qquad \varepsilon_y = \frac{\partial v}{\partial y} \qquad \varepsilon_{xy} = \frac{\gamma_{xy}}{2} = \frac{1}{2} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \qquad (2.3)$$

Attraverso lo sviluppo in serie di Taylor e, limitandosi ai termini del primo ordine, è possibile considerare deformazioni  $\varepsilon$  e scorrimenti  $\gamma$  in termini differenziali con questi ultimi che, analogamente a quanto detto in precedenza per le  $\sigma$ , sono legati linearmente agli sforzi tangenziali dal modulo di resistenza a taglio  $G$ . I valori dei coefficienti  $\nu$ ,  $G$  ed  $E$  sono caratteristici per ogni tipo di materiale e possono a loro volta variare in base alle direzioni lungo le quali vengono valutati (come ad esempio nei compositi dove si hanno diversi comportamenti direzionali in base alla disposizione delle fibre di vetro o carbonio). L’esistenza di un tratto iniziale elastico con relazione biunivoca lineare consente la generalizzazione della legge di Hooke al caso tridimensionale. Per fare ciò si ricorre a vettori di tensione e deformazione (associati all’elemento

dS) che si possono facilmente ricondurre a forze/momenti e spostamenti/rotazioni (associati all'intero elemento considerato), per mezzo di integrazioni sulle aree e sui differenziali nelle relazioni precedentemente riportate. I vettori 6X1 così ottenuti sono linearmente dipendenti per mezzo di una matrice 6X6 chiamata di "Rigidezza" (o della sua inversa, di "Cedevolezza") che permette di delineare l'intero complesso delle relazioni costitutive che intercorrono tra forze e spostamenti.

$$\begin{bmatrix} f_{x1} \\ f_{y1} \\ m_{z1} \\ f_{x2} \\ f_{y2} \\ m_{z2} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \\ k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} \\ k_{41} & k_{42} & k_{43} & k_{44} & k_{45} & k_{46} \\ k_{51} & k_{52} & k_{53} & k_{54} & k_{55} & k_{56} \\ k_{61} & k_{62} & k_{63} & k_{64} & k_{65} & k_{66} \end{bmatrix} \begin{bmatrix} u_{x1} \\ u_{y1} \\ \theta_{z1} \\ u_{x2} \\ u_{y2} \\ \theta_{z2} \end{bmatrix}$$

Figura 4 – Matrice di rigidezza 6X6.

Ogni elemento della matrice è caratterizzato generalmente da moduli di Young e di resistenza a taglio, da coefficienti di Poisson e momenti di inerzia I (nelle relazioni con i momenti meccanici), tuttavia in componenti più complessi vengono spesso utilizzati coefficienti propri del sistema considerato, in modo da facilitarne i calcoli e rendere la scrittura più compatta. Per "parametri" o "moduli indipendenti" verranno comunque ora considerati i valori di  $\nu$ , G ed E. Alcune semplificazioni su questi parametri possono essere adottate a seconda delle caratteristiche del materiale preso in considerazione: i materiali conservativi (o di Green) sono quelli correlabili ad una funzione chiamata potenziale elastico f che genera una corrispondenza tra i termini simmetrici della matrice, quest'ultima presenta quindi 21 moduli indipendenti; ai materiali ortotropi appartengono, invece, i compositi citati in precedenza dove esistono tre assi di simmetria materiale (ovvero tre direzioni privilegiate) che riducono il numero di parametri necessari a 12 (9 in caso di conservatività); vi sono infine i materiali isotropi, che hanno il medesimo

comportamento in tutte le direzioni e su cui concentreremo le nostre attenzioni, per cui sono sufficienti due soli parametri.

Una volta chiariti i concetti cardine per l'analisi del comportamento elastico, ci si può focalizzare sulla loro applicazione nel campo dei flexure. I giunti in questione sono classificabili per mezzo dell'asse singolo, doppio o multiplo attorno al quale manifestano la propria cedevolezza, a questi sono poi chiaramente associati uno, due o più centri di istantanea rotazione chiamati anche "pivot". Il punto di forza di questi elementi risiede proprio nel coniugare cedevolezza elevate nelle direzioni di pivot e rigidità tali da non permettere movimenti, nelle altre direzioni. Per applicazioni con movimenti nel piano vengono utilizzati i flexure ad asse singolo mentre le altre due tipologie, più complesse, sono pensate per applicazioni nelle tre dimensioni.

Vengono ora presentate alcuni esempi di giunti flessibili, per ognuno di essi si intende evidenziare le caratteristiche di rigidità nel piano in presenza di sforzi esterni combinati e in relazione ai corpi rigidi a cui sono connessi.

## 2.2 Giunto lineare

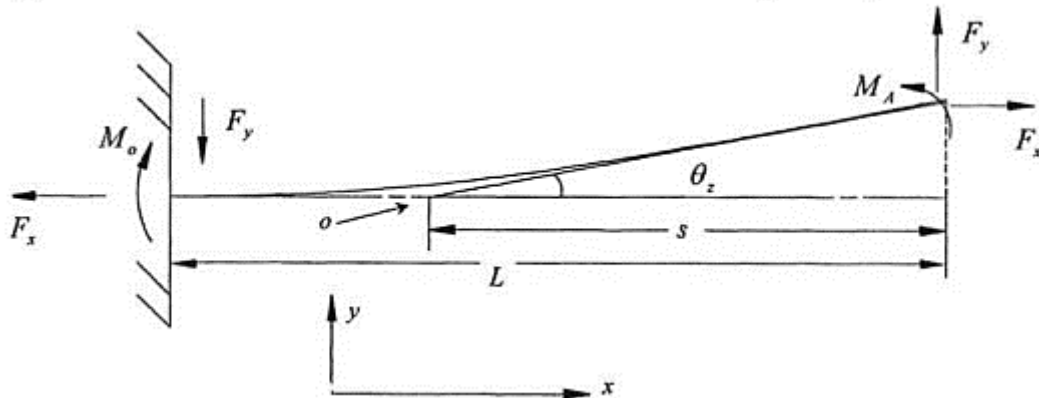


Figura 5 – Rappresentazione schematica di un giunto flexure lineare con i relativi carichi applicati. [11]

Il giunto lineare è il più semplice tra quelli realizzabili nel piano dato che, di fatto, non è altro che un'applicazione nei compliance mechanism dell'elemento trave (figura 5), protagonista di tutti i trattati riguardanti la scienza delle costruzioni. Ciò che contraddistingue questo giunto da tutti gli altri è la mancanza di intagli, dato che in questo caso, la flessibilità è correlata proprio alla lunghezza dell'elemento in questione. Il centro di rotazione è generalmente definito come il punto di intersezione tra la retta tangente all'estremo libero e l'asse indeformato, come tale tenderà dunque a variare la sua posizione con l'aumentare della deformazione: nel campo dei piccoli spostamenti si avranno quindi piccole variazioni del c.i.r. date da

$$s = \frac{\delta_y}{\tan \theta_z} \approx \frac{2}{3}L \quad (2.4)$$

In tal caso l'analisi viene inizialmente eseguita a partire dalla linea elastica di una trave incastrata

$$EI \frac{d^2y}{dx^2} = F_y(L - x) \quad EI \frac{dy}{dx} = F_y \left( Lx - \frac{x^2}{2} \right) + C_1 \quad EIy = F_y \left( L \frac{x^2}{2} - \frac{x^3}{6} \right) + C_1x + C_2 \quad (2.5)$$

A cui vengono applicate poi le corrette condizioni al contorno (spostamento ed angolo di rotazione nullo all'estremo vincolato) per ricavare le relazioni desiderate all'estremo nullo e, successivamente, le relazioni di rigidità angolare e lineare

$$\theta_{x=L} = \frac{F_y L^2}{2EI} \qquad \delta_{x=L} = \frac{F_y L^3}{3EI} \qquad (2.6)$$

$$k_{\theta_z F_y} = \frac{F_y}{\theta_z} = \frac{2EI}{L^2} \qquad k_{\delta_y F_y} = \frac{F_y}{\delta_y} = \frac{3EI}{L^3} \qquad (2.7)$$

Per rendere però più vicino alla realtà il modello ottenuto, è corretto considerare che gli sforzi nei giunti flessibili sono solitamente applicati ai corpi rigidi ad essi connessi. È sufficiente quindi aggiungere all'estremo libero il componente indeformabile.

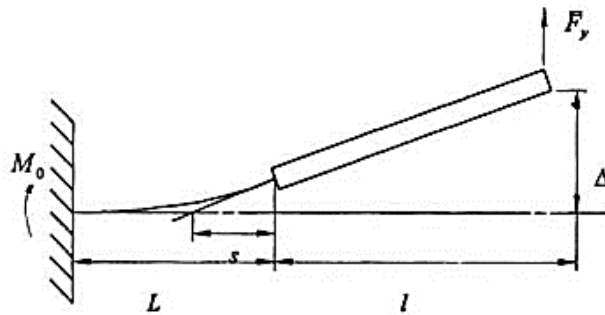


Figura 6 – Giunto lineare con forza verticale applicata al corpo rigido ad esso connesso. [11]

Grazie alle analisi di Thorpe, 1953, fu formulato il modello in figura 6 dove viene, in prima approssimazione, considerata la sola forza trasversale  $F_y$ ; i valori di freccia verticale all'estremità e di distanza dell'estremo, prima considerato libero, dal centro di istantanea rotazione sono date da

$$\Delta_y = F_y \frac{L(l^2 + Ll + L^2 / 3)}{EI} \qquad s = \frac{lL + 2L^2 / 3}{L + 2l} \qquad (2.8)$$

Mentre le equazioni riguardanti spostamento verticale e rotazione sono rappresentate nella 2.9, si può osservare come annullando la lunghezza del corpo rigido collegato all'estremità del giunto ( $l=0$ ) si ricada nelle condizioni di forza trasversale applicata all'estremo libero di una trave (2.6).

$$y = \frac{F_y}{3EI} (L + l)^3 \qquad \frac{dy}{dx} = \frac{F_y}{2EI} (L + l)^2 \qquad (2.9)$$



Il modello di Thorpe viene infine completato dalla compresenza di carichi trasversali e longitudinali. Per evitare di appesantire troppo la trattazione, sono di seguito rappresentate in maniera speculare le equazioni di interesse con un confronto tra le casistiche di forze assiali di trazione (a sinistra) e compressione (a destra).

$$y = \Delta \left\{ \left( \frac{1+C_1}{L+l} \right) x - C_1 + C_1 \cosh\left(\omega \frac{x}{L}\right) - \frac{(1+C_1)}{\omega(1+l/L)} \sinh\left(\omega \frac{x}{L}\right) \right\} \quad y = \Delta \left\{ \left( \frac{1-C_2}{L+l} \right) x + C_2 - C_2 \cos\left(\omega \frac{x}{L}\right) - \frac{L(1-C_2)}{\omega(L-l)} \sin\left(\omega \frac{x}{L}\right) \right\}$$

$$C_1 = \frac{M_z}{F_x \Delta} = \frac{\omega \coth(\omega) + \frac{L}{l}}{\omega^2 \left(1 + \frac{l}{L}\right) + \frac{L}{l} (\omega \coth(\omega) - 1)} \quad C_2 = \frac{M_z}{F_x \Delta} = \frac{\omega \cot(\omega) + \frac{L}{l}}{\omega^2 \left(1 + \frac{l}{L}\right) + \frac{L}{l} (1 - \omega \cot(\omega))}$$

$$\Delta = \frac{F_y}{F_x} \frac{L+l}{1+C_1} \quad \Delta = \frac{F_y}{F_x} \frac{L+l}{1-C_2} \quad (2.10)$$

$$s = \frac{1}{C_1 \frac{\omega}{L} \sinh(\omega) + \left(\frac{1+C_1}{L+l}\right) (\cosh(\omega) - 1)} - l \quad s = \frac{1}{C_2 \frac{\omega}{L} \sin(\omega) + \left(\frac{1-C_2}{L+l}\right) (1 - \cos(\omega))} - l$$

$$\frac{dy}{dx} = \Delta \left\{ \frac{\omega}{L} C_1 \sinh\left(\omega \frac{x}{L}\right) - \frac{1+C_1}{L+l} \cosh\left(\omega \frac{x}{L}\right) + \left(\frac{1+C_1}{L+l}\right) \right\} \quad \frac{dy}{dx} = \Delta \left\{ \frac{\omega}{L} C_2 \sin\left(\omega \frac{x}{L}\right) - \frac{1-C_2}{L+l} \cos\left(\omega \frac{x}{L}\right) + \left(\frac{1-C_2}{L+l}\right) \right\}$$

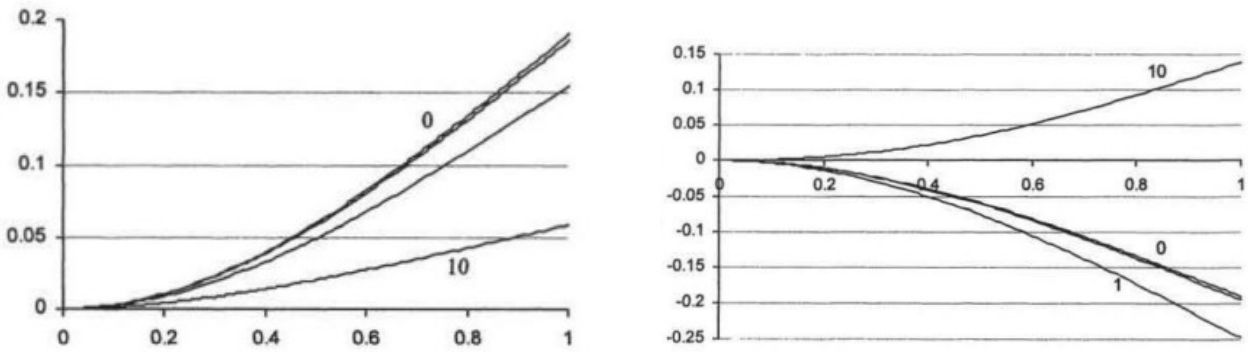


Figura 7 – Confronto tra le deformazioni (in ordinata) sullo stesso elemento di lunghezza unitaria (in ascissa) nei casi di forze esterne di: a) trazione; b) compressione. [11]

È necessario sottolineare un paio di aspetti che emergono dalle analisi dei due scenari a confronto. Nel grafico di sinistra è evidente che, all'aumentare del carico assiale di trazione, si generi una deformazione di entità minore a causa dei segni discordi dei momenti generati dalla forza trasversale e da quella longitudinale. Nella figura a destra, avendo invece invertito i segni delle

forze esterne, si nota come il valore di spostamento diventi addirittura positivo per valori di  $F_x$  superiori a 10N. Ciò è giustificato dal fatto che la trave è inizialmente deformata lungo i valori positivi delle ordinate, in tal caso al diminuire della forza assiale si avrà una deformazione verso valori negativi dovuta all'effetto della  $F_y$  verso il basso. Se la trave fosse deformata inizialmente verso il basso, gli effetti delle due forze si andrebbero invece a sommare a causa dei segni concordi dei relativi momenti generati. Emerge quindi la difficoltà relativa alla previsione degli spostamenti in caso di forze di compressione, poiché troppo influenzati dalle condizioni iniziali in cui si trova la trave considerata. Ad una forza assiale di compressione, inoltre, è sempre associata la verifica della condizione di "Buckling" di Eulero che renderebbe ancor più imprevedibile il comportamento del giunto: si dovrebbe difatti considerare la deformata associata al superamento del carico critico di Eulero esprimibile come

$$F_{cr} = \frac{\pi^2 EI}{4L^2} \quad (2.11)$$

Si può infine terminare l'analisi dei giunti lineari introducendo anche un momento flettente  $M$  per avere una panoramica dei comportamenti associati ad ogni tipo di sollecitazione esterna. Seguendo la teoria di Plainevaux, 1956, l'equazione differenziale

$$EI \frac{d^2 y}{dx^2} = -M_0 + F_x y - F_y x \quad (2.12)$$

Può essere riscritta facendo uso di alcune variabili come

$$X = \frac{x}{L} \quad 4\gamma^2 = \frac{F_x L^2}{EI} \quad \varphi = \frac{F_y L^2}{EI} \quad m_0 = \frac{M_0 L}{EI} \quad (2.13)$$

Si ottiene

$$\frac{d^2 y}{dX^2} \mp 4\gamma^2 y = -L(m_0 + \varphi X) \quad (2.14)$$

Risolvendo l'equazione (si osservi come le soluzioni nel caso di trazione sono funzioni iperboliche mentre in quello di compressione sono trigonometriche) e differenziando sulla variabile  $y$  come nei casi precedenti si ottengono, omettendo alcuni passaggi matematici, le equazioni caratteristiche per i casi di forze assiali di trazione (a sinistra) e compressione (a destra).

$$\frac{F_y}{\delta_y} = \frac{12EI}{L^3} \left( \frac{\gamma^3}{3(\gamma - \tanh(\gamma))} \right) \quad \frac{F_y}{\delta_y} = \frac{12EI}{L^3} \left( \frac{\gamma^3}{3(\tan(\gamma) - \gamma)} \right) \quad (2.15)$$

$$\frac{M_o}{\delta_y} = \frac{6EI}{L^3} \left( \frac{\gamma^2 \tanh(\gamma)}{3(\gamma - \tanh(\gamma))} \right) \quad \frac{M_o}{\delta_y} = \frac{6EI}{L^3} \left( \frac{\gamma^2 \tan(\gamma)}{3(\tan(\gamma) - \gamma)} \right) \quad (2.16)$$

$$\frac{\delta_x}{L} = \frac{3}{5} \left( \frac{\delta_y}{L} \right)^2 \left( \frac{5\gamma[\gamma(3 - \tanh^2 \gamma) - 3 \tanh \gamma]}{12(\gamma - \tanh \gamma)^2} \right) \quad \frac{\delta_x}{L} = \frac{3}{5} \left( \frac{\delta_y}{L} \right)^2 \left( \frac{5\gamma[\gamma(3 + \tan^2 \gamma) - 3 \tan \gamma]}{12(\tan \gamma - \gamma)^2} \right) \quad (2.17)$$

Considerando invece la sola componente relativa al momento flettente e, tralasciando ogni altro tipo di carico, si possono ricavare le formule tipiche della semplice teoria flessionale, queste ci saranno utili per qualche confronto con i giunti intagliati che affronteremo tra poco. La rigidezza angolare di una trave di lunghezza  $2a_x$  sottoposta a sola flessione nel piano è pari a

$$K_{\theta_z M_z} = \frac{M}{\theta} = \frac{EI}{2a_x} \quad (2.18)$$

Si ricava poi il valore di tensione corrispondente all'angolo massimo  $\theta$  raggiunto (tralasciando le concentrazioni di stress in corrispondenza dei bordi esterni) e, riordinando l'equazione, si ottiene un'espressione dello spessore

$$\sigma_y = \frac{Et}{2L} \theta_{max} = \frac{Et}{4a_x} \theta_{max} \Rightarrow t = \frac{4a_x}{E\theta_{max}} \sigma_y \quad (2.19)$$

Un'applicazione più pratica e di maggior interesse può essere quella relativa al quadrilatero articolato realizzato tramite questa tipologia di giunti e riportato in figura 8.

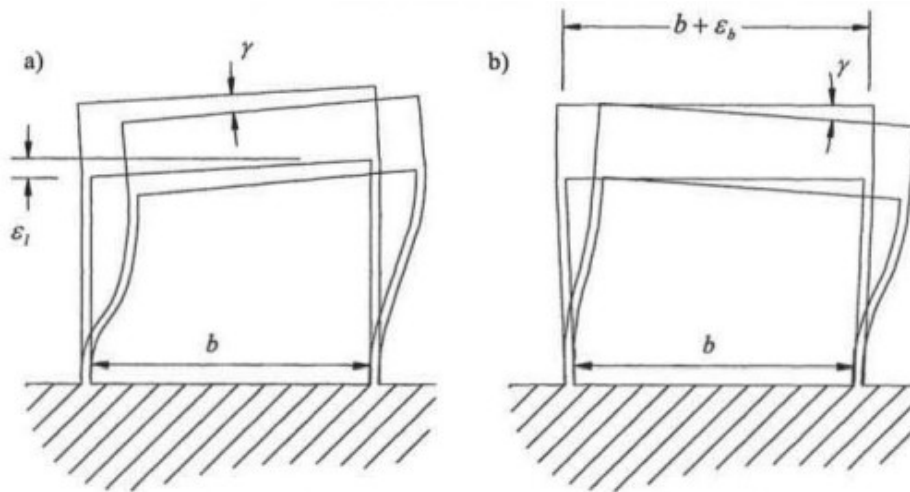


Figura 8 – Rappresentazione delle deformazioni, verticali ed orizzontali, di un quadrilatero realizzato come compliant mechanism tramite giunti lineari. [11]

Valutando le inesattezze che possono condizionare un compliant mechanism in fase di realizzazione, si può dare una stima delle imprecisioni associate ad una bassa accuratezza delle tolleranze sulla base di parametri come l'errore trasversale  $\varepsilon_l$  e longitudinale  $\varepsilon_b$

$$\gamma = \frac{\delta^2 \varepsilon_b}{2bL^2} \qquad \gamma = \frac{\delta \varepsilon_l}{2bL} \qquad (2.20)$$

## 2.3 Giunto ad intagli

La seconda categoria di giunti è forse la più famosa ed utilizzata, ovvero quella ad intagli. In questo caso la coppia cinematica è data da un unico componente grezzo che viene poi opportunamente sagomato e quindi privato di materiale in zone funzionali alla riduzione di rigidità lungo un asse. La struttura è generalmente formata da due estremità vincolate ai corpi rigidi (di cui si intende mantenere il moto relativo desiderato) collegate da un tratto di materiale di spessore inferiore ricavato da due scavi esterni di forma circolare, ellittica o rettangolare. Il parametro a cui fanno riferimento queste tipologie di forme è  $\varepsilon$  e sta ad indicare il rapporto tra asse maggiore e asse

minore dell'intaglio: esso è uguale a uno per sezioni circolari, maggiore di uno per quelle ellittiche e di valore teoricamente infinito per quelle rettangolari. A causa di questa strizione localizzata sarà possibile ottenere un centro di istantanea rotazione ben definito, a svantaggio però di tensioni interne al materiale di valore più elevato rispetto a quelle dei giunti lineari (in quel caso le tensioni vengono infatti distribuite lungo l'intero asse del componente) con conseguenti minori deformazioni ottenibili in campo elastico. Va sempre considerata poi, realisticamente parlando, la compresenza di sforzi assiali, di taglio e flettenti all'interno del giunto elastico che dovrà essere quindi progettato per agire in precise condizioni di carico, limitando al minimo le influenze di azioni "parassite" che verranno inevitabilmente a formarsi.

Per quanto concerne il flexure di geometria circolare nel piano, è di particolare interesse la cedevolezza associata al momento flettente applicato al giunto, di cui Paros e Weisbord fornirono una prima soluzione approssimata come

$$\frac{1}{k_{\theta_z M_z}} = \frac{3}{2Eb a_x^2} f(\beta) \quad (2.21)$$

Dove viene utilizzato il parametro adimensionale  $\beta$  per delineare la dipendenza dalla geometria utilizzata ed una sua funzione  $f(\beta)$  come fattore di cedevolezza, anch'esso adimensionale. Allontanandosi dalla soluzione "esatta" per piccoli valori di  $\beta$ , la sua funzione può essere semplificata ottenendo

$$K_{\theta_z M_z} \approx \frac{2Eb a_x^2}{3} \frac{(2\beta)^{5/2}}{3\pi} = \frac{2Ebt^{5/2}}{9\pi\sqrt{a_x}} \quad (2.22)$$

Il valore ottenuto dalla soluzione approssimata è minore di quella teoricamente esatta e la discrepanza dell'errore tende ad aumentare linearmente al crescere del parametro adimensionale, toccando l'8.25% per  $\beta=0,3$ . Successivamente furono attuate correzioni a questo modello matematico in quanto le tensioni ottenute venivano considerate erroneamente agenti solo lungo

l'asse longitudinale, fu quindi Ling, 1952, a formulare una soluzione ben rappresentativa degli sforzi dovuti a flessione. Tale teoria, che fu confermata sperimentalmente dalle analisi fotoelastiche eseguite da Frocht e Goodier, verte sull'utilizzo di un fattore di concentrazione delle tensioni K, in perfetta analogia con gli attuali metodi di calcolo degli sforzi in presenza di riduzioni di sezione. Lo stress  $\sigma$  può essere così calcolato

$$\sigma = K_t \frac{6M}{t^2 b} \quad (2.23)$$

Per valori di  $b$  compresi tra zero e 2,3 il valore di K può essere stimato e quindi rappresentato come

$$K_t = \left(\frac{2a_x}{D}\right)^{-9/20} = \left(1 + \frac{t}{2a_x}\right)^{9/20} = (1 + \beta)^{9/20} \quad (2.24)$$

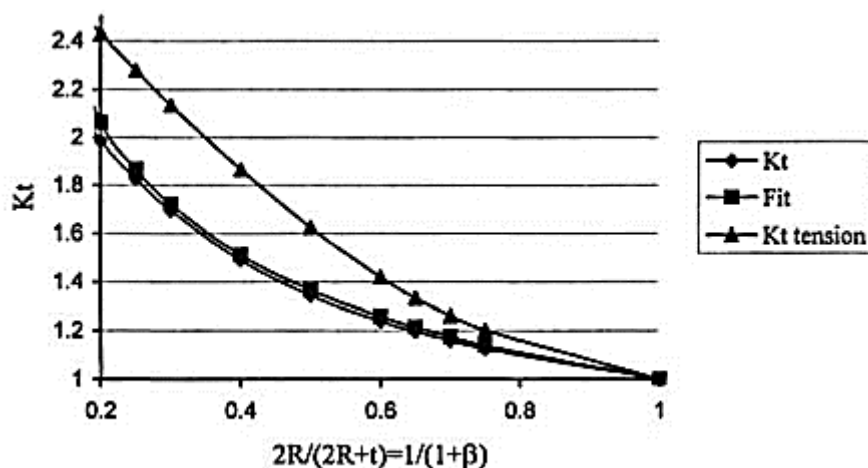


Figura 9 – Andamento del fattore di concentrazione delle tensioni  $K_t$ . [11]

Sostituendo poi i risultati appena ottenuti all'interno della relazione sulle tensioni  $\sigma$  è possibile ottenerne una stima in prossimità della riduzione di sezione (punto più sollecitato) a partire dallo spostamento angolare  $\theta$  che si intende ottenere. Viene poi riportata la relazione per l'ottenimento del massimo spessore della strizione, si ricordi come il coefficiente  $\beta$  sia legato alla geometria del giunto ed è quindi chiaro che  $t$  ne debba essere dipendente a sua volta.

$$\sigma = \frac{E(1 + \beta)^{9/20}}{\beta^2 f(\beta)} \theta \qquad t^2 = \frac{4Ea_x^2}{f(\beta)} (1 + \beta)^{9/20} \frac{\theta}{\sigma_y} \quad (2.25)$$

Seguendo sempre gli scritti di Paros e Weisbord viene trattato ora il giunto con intaglio a sezione ellittica. In questo caso, dividendo la sezione dell'intero giunto flessibile in segmenti verticali di valore  $dx$  ed integrando sul dominio si giunge, dopo alcune sostituzioni, al valore di  $\theta$

$$\theta_z = \frac{3M_z a_x}{2Eb a_y^3} \int_{-\pi/2}^{\pi/2} \frac{\cos \theta}{\left( \left( 1 + t/2a_y \right) - \cos \theta \right)^3} d\theta \quad (2.26)$$

Da qui è infine possibile ottenere l'equazione di cedevolezza dipendente dai parametri adimensionali legati alle caratteristiche geometriche del flexure. Successivamente è utile invertire la relazione, in modo da evidenziarne la rigidezza, ed esplicitare il parametro  $a_x$  (la semilunghezza è infatti un parametro costruttivo estremamente utile alla progettazione)

$$\frac{1}{k_{\theta_z M_z}} = \frac{3}{2Eb a_y^2} \varepsilon f(\beta_y) \Rightarrow K_{\theta_z M_z} = \frac{2Eb a_x^2}{3\varepsilon^3 f(\varepsilon \beta_x)} \quad (2.27)$$

$$\beta_y = \frac{t}{2a_y} \qquad \beta_x = \frac{t}{2a_x} \qquad \varepsilon = \frac{a_x}{a_y} = \frac{\beta_y}{\beta_x} \quad (2.28)$$

Plottare il parametro  $\varepsilon$  può essere utile a capire come i casi ellittici siano configurazioni intermedie tra il giunto lineare ( $\varepsilon = \infty$ ) e quello circolare ( $\varepsilon = 1$ ) sia dal punto di vista geometrico che da quello analitico. Nel grafico in figura 10 si osserva come all'aumentare del parametro  $\beta_x$ , a parità di spessore  $t$  tra i vari elementi, la rigidezza degli elementi con intaglio a curvatura non infinita cresca più rapidamente: ciò è giustificato dalla presenza di una maggior quantità di materiale resistente in prossimità della zona di massima strizione.

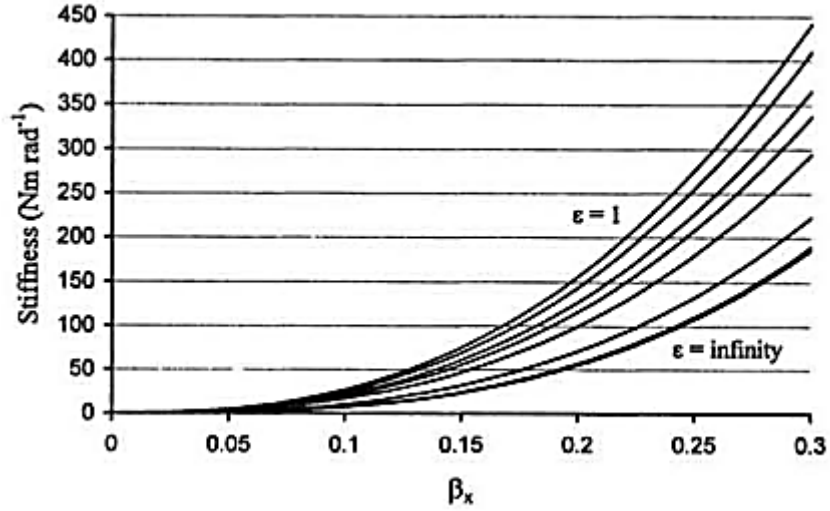


Figura 10 – Andamento della rigidezza del giunto flexure in funzione del parametro  $\beta_x$  e della geometria ( $\varepsilon$ ). [11]

Estendendo l'analisi alle tre dimensioni, la matrice di cedevolezza può essere arricchita dal termine di flessione attorno all'asse verticale e da quelli di compressione lungo tutte le direzioni. Per fare ciò vengono utilizzate funzioni, diverse dalle precedenti, dei parametri adimensionali  $\beta$  ed  $\varepsilon$  indicate con  $g$ .

$$\frac{1}{K_{\theta_y M_y}} = \frac{\theta_y}{M_y} = \frac{12\varepsilon}{Eb^3} g(\varepsilon\beta_x) \quad \frac{1}{K_{\delta_x F_x}} = \frac{\delta_x}{F_x} = \frac{\varepsilon}{Eb} g(\varepsilon\beta_x) \quad \frac{1}{K_{\delta_y F_y}} = \frac{1}{K_{\delta_z F_z}} = \frac{\varepsilon}{Gb} g(\varepsilon\beta_x) \quad (2.29)$$

Con 
$$g(\varepsilon\beta_x) = \frac{2(1+\beta_y)}{\sqrt{2\beta_y+\beta_y^2}} \tan^{-1} \sqrt{\frac{2+\beta_y}{\beta_y}} - \frac{\pi}{2} \quad (2.30)$$

Per  $\varepsilon = 1$ , i valori equivalenti di  $\beta_x = \beta_x = \beta$  e la condizione di circolarità  $a_x = a_x = R$ , possono essere utilizzate le equazioni semplificate di Paros e Weisbord per piccoli  $\beta$  ottenendo le rigidezze.

$$K_{\theta_z M_z} = \frac{M_z}{\theta_z} = \frac{2Ebt^{5/2}}{\sqrt{9\pi R}} \quad (2.31)$$

$$K_{\theta_z F_y} = \frac{F_y}{\theta_z} = \frac{2Ebt^{5/2}}{\sqrt{1-\beta^2} 9\pi R^{3/2}}$$



$$K_{\theta_y F_z} = k_{\delta_z M_y} = \frac{Eb^3\pi}{12R} \left[ \sqrt{\frac{1}{2\beta}} - \pi \sqrt{\frac{2}{3}} \right]^{-1}$$

## 2.4 Giunto a fasce incrociate

Per ovviare a necessità di spostamenti e rotazioni sempre maggiori nelle applicazioni meccaniche, non ottenibili per mezzo dei componenti precedentemente presentati, ci si è man mano orientati verso meccanismi strutturalmente più complesse con cedevolezze maggiori. Il componente rappresentato in figura 11 è un giunto a fasce incrociate, esso consiste essenzialmente in due basi (collegate ai corpi rigidi della catena cinematica in cui solitamente uno funge da telaio e l'altro da piattaforma con moto relativo) connesse tra loro da strisce piatte disposte simmetricamente.

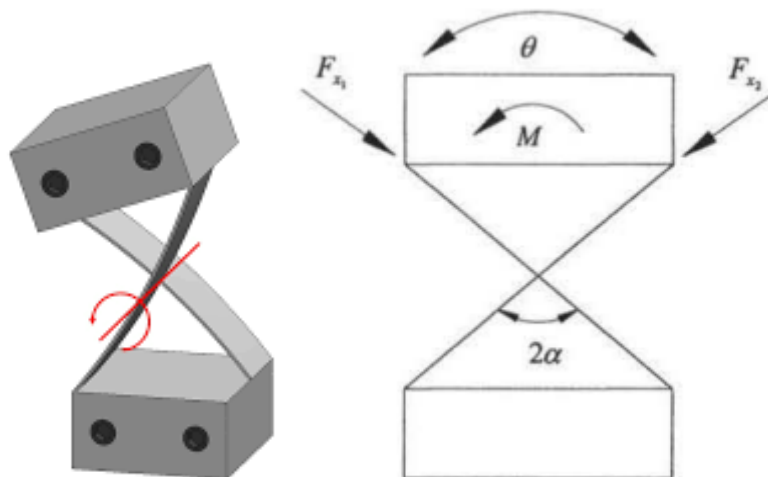


Figura 11 – Rappresentazione a) reale e b) schematica di un giunto a fasce incrociate con carichi esterni applicati ed angoli di riferimento caratteristici.

Il numero di fasce e la loro disposizione può variare a seconda delle necessità di rigidità torsionale che si intende mantenere, tuttavia le maggiori attenzioni vengono chiaramente rivolte all'intersezione dei piani inclinati lungo l'asse longitudinale del giunto: è proprio in quell'intorno spaziale che si trova, infatti, il centro di istantanea rotazione. Si è parlato di intorno perché, come detto, per i flexure non è sempre possibile mantenere fisso il pivot point; è stato comunque

dimostrato (anche se non verificato sperimentalmente) che il c.i.r. riesca a mantenere una maggiore stabilità sotto carico quando il punto d'incontro delle due fasce diagonali si trova all'87,3% della lunghezza di ciascuna striscia. In figura b) è rappresentato lo schema dei carichi a cui è soggetto solitamente il giunto, con gli sforzi assiali che agiscono lungo le fasce inclinate (esse si comportano di fatto come puntoni o tiranti) mentre il puro momento è associato alla rigidità flessionale tipica della trave ( $EI/L$ ). Per piccoli spostamenti, l'angolo di rotazione assunto dalla base superiore  $\theta$  è associato allo scorrimento verticale di quest'ultima  $\delta_y$  dalla formula

$$\theta = \frac{2\delta_y \sin \alpha}{L \sin \alpha} = \frac{2\delta_y}{L} \quad (2.32)$$

Per bassi valori di quest'angolo, il centro di rotazione coincide con l'intersezione delle fasce indipendentemente dall'angolo formato tra le due; si deve comunque tener conto della sensibilità della posizione del pivot all'aumentare della rotazione del giunto: il discorso si complica nell'ambito delle grandi deformazioni mentre le imprecisioni durante le fasi di realizzazione e assemblaggio possono inficiare pesantemente sul risultato finale. L'analisi del flexure verte sullo stesso modello di Plainevaux presentato precedentemente, il componente viene infatti associato ad una trave incastrata all'estremità della quale vengono applicate le componenti di forza assiale e trasversale insieme ad un momento meccanico. Utilizzando i medesimi parametri definiti per questioni di comodità in formula 2.13 si giunge all'equazione differenziale

$$\frac{d^2y}{dX^2} - 4\gamma^2 y = L(m_0 - \varphi X) \quad (2.33)$$

La cui soluzione è ottenuta imponendo le condizioni al contorno di spostamento verticale nullo ed angolo di rotazione pari a zero

$$y = \frac{Lm_0}{4\gamma^2} (\cosh(2\gamma x) - 1) - \frac{L\varphi}{8\gamma^3} (\sinh(2\gamma X) - 2\gamma X) \quad (2.34)$$

La comune rigidezza flessionale di una trave, direttamente proporzionale al prodotto tra il suo modulo di Young ed il momento di inerzia ed inversamente proporzionale alla sua lunghezza, può essere utilizzata per bassi carichi applicati commettendo errori praticamente trascurabili. Per sforzi esterni di maggior entità si può invece adoperare la rigidezza dipendente dal parametro  $\gamma$  così espressa

$$\frac{M_i}{\theta} \left( \frac{L}{EI} \right) = \frac{\gamma}{\tan(\gamma)} - \gamma^2 \quad \text{con} \quad \gamma = \sqrt{\frac{F_x L^2}{EI}} \quad (2.35)$$

È opportuno evidenziare un paio di valori caratteristici riguardanti la variabile di carico assiale appena introdotta: per elevate forze di compressione longitudinali all'asse si tende al valore limite di  $\gamma = \pi$  che corrisponde al carico critico di Eulero a buckling; la rigidezza a flessione diventa nulla in corrispondenza di  $F_x$  negative di intensità pari a  $\gamma = 0.866$ , mentre può essere considerata costante ed equivalente a quella di trave scarica assialmente nel range  $-1/4 < \gamma < 1/4$ . Tralasciando la concentrazione di stress in corrispondenza delle giunzioni tra le basi e le fasce, si può ottenere la tensione massima raggiunta all'estradosso del flexure ed il relativo angolo di rotazione limite come

$$\sigma = \frac{Et}{2L} \theta + \frac{F_x}{tb} \quad \theta_{max} = \frac{2L}{Et} \sigma_{max} - \frac{t}{24L} \quad (2.36)$$

L'ultima considerazione su questo tipo di giunti flessibili va fatta sulla stabilità del centro di istantanea rotazione durante le varie deformazioni. Si è visto che durante il moto della base superiore, le relazioni geometriche che intercorrono tra la rotazione  $\theta$  e lo spostamento del pivot point  $\delta_p$  verso le ordinate positive è dato, secondo Haringx, dalle seguenti formule (la seconda delle due fa riferimento al parametro  $2\alpha$  che rappresenta l'angolo compreso tra le due fasce incrociate ed è equivale solitamente a  $90^\circ$ )

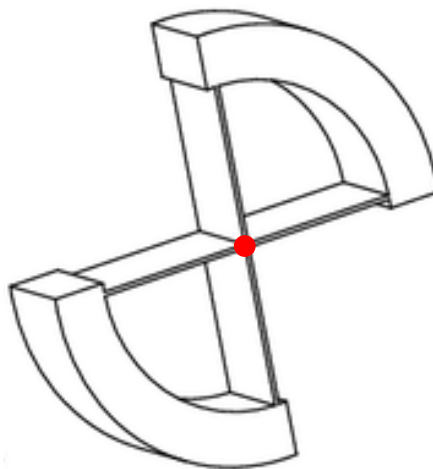
$$\frac{\delta_p}{L} = \left( \frac{2 \sin(\theta / 2)}{\theta} - \cos(\theta / 2) \right) \quad \frac{\delta_p}{L} = \frac{1}{\cos \alpha} \left( \frac{2 \sin(\theta / 2)}{\theta} - \cos(\theta / 2) \right) \quad (2.37)$$

Proprio con l'angolo retto compreso tra i due segmenti, i risultati ottenuti tramite sviluppo in serie di Taylor possono essere semplificati per ottenere

$$\frac{\delta_p}{L} = \frac{\sqrt{2}\theta^2}{12} \quad (2.38)$$

## 2.5 Giunto rotoidale

Un'evoluzione del giunto a strisce incrociate è quello rotoidale, anche qui la cedevolezza è dovuta a sottili sezioni di materiale appositamente rifinite ma che sono stavolta unite nel loro punto di intersezione. I vantaggi rispetto al flexure precedente risiedono soprattutto nel processo di realizzazione del giunto che, a differenza di quello incrociato, non necessita di assemblaggio e può essere ottenuto a partire da un unico grezzo di materiale.



*Figura 12 – Giunto rotoidale con indicato il centro di istantanea rotazione.*

Anche qui, i due segmenti circolari di maggior spessore presenti sulla diagonale del disegno in figura 12, fungono alternativamente da telaio e da base mobile collegata all'elemento rigido, di questa configurazione è però possibile sfruttarne la simmetria per semplificarne gli aspetti analitici. Viene quindi preso in considerazione solo uno dei due archi connessi al pivot (che si trova

chiaramente al centro del sistema) e, tale approssimazione, è possibile quando non vi sia la presenza di carichi radiali. Ogni variazione di angolo e posizione, sotto queste condizioni, sarà dunque simmetrica rispetto al centro (la rotazione complessiva, ad esempio, sarà quindi il doppio di quanto ottenuto dall'analisi del singolo arco). L'equazione parametrica di partenza è così formata

$$EI \frac{d^2 y}{dx^2} = F_y(R - x) - M_A \quad (2.39)$$

Seguendo il procedimento logico utilizzato più volte in precedenza, vengono applicate le condizioni al contorno (derivata prima pari a  $\theta/2$  e valore di  $y$  pari a  $R \sin(\theta/2)$  in corrispondenza di  $x = R$ ) alla variabile  $y$  ottenuta e alle sue derivate. Introducendo i parametri  $\varphi$  ed  $m$

$$\varphi = \frac{F_y R^2}{EI} = 12 \left[ \sin(\theta / 2) - \frac{\theta}{4} \right] \quad m = \frac{M_A R}{EI} = 6 \left[ \sin(\theta / 2) - \frac{\theta}{3} \right] \quad (2.40)$$

Si ottengono, per piccoli angoli di rotazione, le rigidezze

$$\frac{F_y}{\theta} \approx \frac{12EI}{L^2} \quad \frac{M_A}{\theta} \approx \frac{2EI}{L} \quad (2.40)$$

Si noti come per coppie applicate all'estremità del giunto, comprensive della componente data dal prodotto della forza  $F_y$  per il braccio  $R$ , il secondo termine della relazione appena espressa (esplicativo della rigidezza torsionale) diventi  $4EI/R$ . Mentre l'equazione differenziale di partenza si riduce alle soluzioni

$$y = \frac{\theta}{R} \left( x^2 - \frac{x^3}{2R} \right) \quad \frac{dy}{dx} = \frac{\theta}{R} \left( 2x - \frac{3x^2}{2R} \right) \quad \frac{d^2 y}{dx^2} = \frac{\theta}{R} \left( 2 - \frac{3x}{R} \right) \quad (2.41)$$

A seguito di manipolazioni matematiche che non vengono qui riportate poiché lunghe e verbose, si può ottenere inoltre il legame tra spostamento del c.i.r. ed angolo di rotazione

$$\frac{\delta_p}{L} = \frac{\sqrt{2}\theta^2}{60} \quad (2.42)$$

Confrontando questo risultato con il corrispondente ottenuto per il giunto a fasce incrociate, emerge un miglioramento della stabilità che è di ben cinque volte superiore per il giunto circolare. Particolare attenzione va rivolta al modulo di questo parametro dato che nel rotoidale, lo spostamento  $\delta_p$  è diretto verso il telaio fisso (al contrario del precedente dove è invece diretto verso la base mobile). Relativamente allo stress invece, il discorso è complicato dalla concentrazione di tensioni nei pressi del centro, dove sono collegate le due parti studiabili singolarmente. L'analisi va quindi eseguita con estrema cautela facendo uso di fattori di intaglio K accuratamente valutati sui raccordi, o per mezzo di analisi agli elementi finiti con mesh particolarmente affinata sulle superfici esterne nei pressi del pivot. Una prima stima per valutare la bontà degli eventuali dati ricavati può essere comunque data dall'equazione per la flessione

$$\sigma_{max} = \frac{M_{max}t}{2I} = \frac{Et}{R}\theta \quad (2.43)$$

## 2.6 Giunto cruciforme

L'ultimo giunto che verrà preso in considerazione è quello che più si distacca dal solo comportamento planare, il giunto cruciforme trova infatti maggiori applicazioni in catene cinematiche spaziali grazie alle diverse caratteristiche possedute lungo i suoi tre assi. Il flexure in questione è anch'esso realizzato a partire da un unico materiale originario dove vengono successivamente realizzate lavorazioni tali da ottenere sezioni a doppia T su ognuna delle facce appartenenti alla superficie laterale. In figura 13 ne è riportato un esempio, nella parte alta del disegno è rappresentata una freccia ad indicare il momento torcente a cui è sottoposto solitamente il componente trattato, la peculiarità di quest'ultimo è infatti quella di possedere un'elevata cedevolezza torsionale garantendo comunque una grande rigidità a flessione lungo le altre due direzioni.

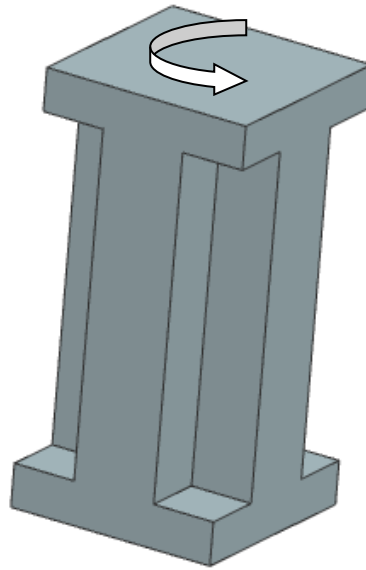


Figura 13 – Giunto cruciforme con momento torcente applicato.

Per quanto concerne il cruciforme, quella che viene fornita generalmente è una sottostima delle rigidezze ad esso correlate poiché le equazioni fanno riferimento ad un modello privo di vincoli: la rigidezza torsionale sarà quindi maggiore nelle condizioni di funzionamento reale mentre quella flessionale non vedrà particolari variazioni. Le relazioni appena citate possono essere quindi così ottenute (la seconda formula, per la flessione, è identica per gli assi x e y)

$$K_{\theta_z M_z} = \left(\frac{d}{t} - 0,373\right) \frac{2Gt^4}{3L} \qquad K_{\theta_x M_x} = \frac{Gt^4(1+\nu)\left(\frac{d}{t} + \frac{d^3}{t^3} - 1\right)}{6L} \qquad (2.44)$$

È utile, infine, definire il parametro adimensionale  $\chi$  (dato dal rapporto tra d e t definiti sulla sezione a croce) per valutare il rapporto di rigidezza tra flessionale e torsionale.

$$\frac{K_{\theta_x M_x}}{K_{\theta_z M_z}} = \frac{(1 + \nu)(\chi + \chi^3 - 1)}{4(\chi - 0,373)} \qquad (2.45)$$

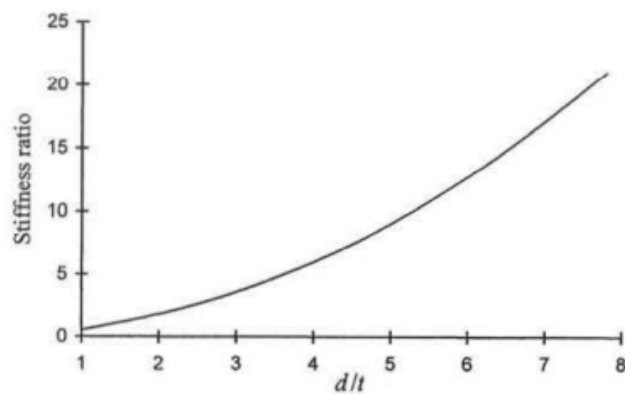


Figura 14 – Rigidezza del giunto cruciforme in funzione del parametro adimensionale  $\chi$ . [11]

Dal grafico in figura 14 appare evidente come la differenza tra i due arrivi all'ordine di grandezza una volta superato il valore di  $\chi$  di circa 6. Tale rapporto continuerà ad aumentare con la presenza di vincoli agli estremi del giunto a causa del diminuire di  $K_{\theta_z M_z}$  alla quasi costanza di  $K_{\theta_x M_x}$ .



## **3 Sistemi di produzione**

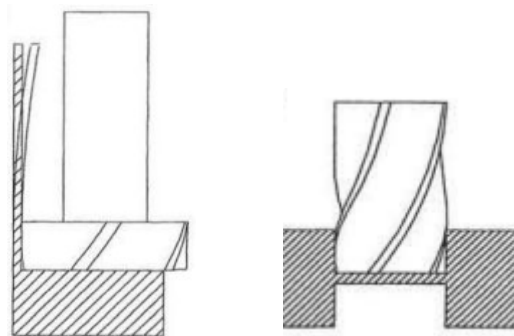
In questo capitolo verranno trattate tutte quelle che sono le procedure di ottenimento dei giunti flessibili evidenziando vantaggi e svantaggi per ciascun sistema di produzione. Si ricordi che di particolare interesse è l'utilizzo dei componenti, protagonisti della discussione, in applicazioni su microscala dove spesso quella dei flexure è l'unica strada percorribile per realizzare una coppia cinematica. Esempi di applicazione nella vita di tutti i giorni di componenti flessibili si trovano, ad esempio, nei sottili supporti in acciaio inossidabile delle testine magnetiche di lettura degli hard disk, negli strumenti scientifici di altissima precisione e persino negli accelerometri utilizzati nelle automobili come sensori di pericolo. I materiali utilizzati, a cui si è fatto riferimento come "grezzo" di partenza, variano anch'essi a seconda delle applicazioni (nelle equazioni del capitolo precedente i parametri geometrici sono sempre associati al modulo di Young) e con essi si modificheranno i dettami nella lavorazione. Un altro aspetto importante di cui non abbiamo tenuto conto precedentemente è quello dei corpi rigidi a cui deve essere garantito il moto relativo, questi infatti, andranno opportunamente vincolati all'estremità dell'elemento flessibile facendo attenzione a non intaccarne le funzionalità o le caratteristiche di linearità in campo elastico.

### **3.1 Lavorazione alle macchine utensili**



*Figura 15 – Macchine CNC per lavorazioni con asportazione di truciolo*

L'assoluta protagonista dei sistemi di produzione, nel mondo della meccanica, è la macchina utensile nella sua generalità. La sconfinata varietà di queste macchine le rende estremamente adattabili ad ogni tipo di lavorazione, su qualsivoglia materiale e soprattutto in fasi diverse del processo di realizzazione. In questo caso particolare, tuttavia, data la sensibilità del componente da lavorare, è importante andare ad agire alle macchine utensili con piccole forze di taglio, facendo attenzione alle direzioni di applicazione degli sforzi vista la cedevolezza per determinati assi dei flexure. L'utilizzo della fresatrice per sagomare correttamente il giunto, ad esempio, richiede alcuni accorgimenti durante la calibrazione del controllo numerico; prima di considerazioni relative al pezzo, è consigliabile utilizzare un utensile di taglio ben affilato e di piccole dimensioni che lavori ad alte velocità. La scelta dell'utensile e della corretta zona (e direzione di taglio) è cruciale ai fini di un buon lavoro mentre l'asportazione di truciolo crea forze assiali e radiali all'interno del materiale che generano tensioni; è quindi intuitivo pensare che la zona meno opportuna alle lavorazioni superficiali sia quella in prossimità delle strizioni del componente. Il taglio può essere solitamente ottenuto secondo due differenti direzioni di taglio, come riportato dal confronto in figura 16.



*Figura 16 – Due diverse metodologie e relativi taglienti per fresa a confronto. [11]*

Nel primo caso, il tagliente agisce nella zona più sottile per mezzo di interazioni radiali con un moto perpendicolare al piano dell'immagine mentre nel secondo è lo sforzo assiale a fare da

padrona, con sollecitazioni sulla strizione che sono comunque minori di quelle che si hanno per effetto della superficie laterale sulla base. La particolare conformazione dello strumento di taglio nell'esempio di sinistra è dovuta al problema delle vibrazioni: il moto rotatorio della fresa genera delle oscillazioni armoniche particolarmente evidenti nello strato più sottile del materiale, queste nella peggiore delle possibilità, possono corrispondere alla frequenza propria del componente causando il fenomeno della "risonanza". La superficie soggetta a vibrazioni non è quindi a contatto per la sua intera lunghezza con l'utensile (cosa che avverrebbe utilizzando la fresa nell'esempio di destra), e si osserva una riduzione dei rischi associati all'imprecisione della superficie e alla sua finitura. La fresa CNC (Computer Numerical Control) è senza dubbio la macchina utensile più utilizzata ed appropriata per giunti lineari, ellittici o di forma più complessa come quelli rotoidali. Una discreta semplificazione realizzativa si ha invece per i flexure ad intaglio circolare dove la rifinitura della zona di minor sezione può essere praticata per mezzo di normale foratura, qui la precisione della lavorazione dipende unicamente dalle tolleranze che è in grado di garantire la macchina utensile preposta, non richiedendo particolari correzioni durante le fasi successive del processo produttivo. La principale problematica per questo tipo di procedura si verifica nella realizzazione del secondo dei due fori, difatti mentre il primo va ad essere realizzato sul materiale di partenza nella sua completezza, il secondo si trova ora in una zona estremamente indebolita in corrispondenza dell'intaglio che non garantisce più la stabilità necessaria ad uno scavo così preciso. Per ovviare a questa situazione si è soliti conferire maggior rigidità in fase di seconda foratura per mezzo di sostegni che vanno ad occupare la zona di materiale mancante in modo da "restaurare" la condizione iniziale. La metodologia di sostegno più utilizzata attualmente è quella che sfrutta l'utilizzo di componenti liquide che vengono fatte solidificare in una sorta di formatura a cera persa. Numerosi sono gli studi sui vari tipi di cere che è possibile adoperare a seconda delle necessità, ma ciò che li accomuna è la ricerca di una soluzione ottimale per permettere la

successiva estrazione del sostegno dopo il secondo foro: i riscaldamenti fino alla temperatura di liquefazione della cera (si va dai 40 ai 140 °C a seconda della tipologia) o le dissoluzioni in un opportuno solvente (come ad esempio l'acetone), vanno entrambe accuratamente analizzate in modo da non intaccare l'integrità strutturale del giunto ottenuto.

### 3.2 Macchine abrasive

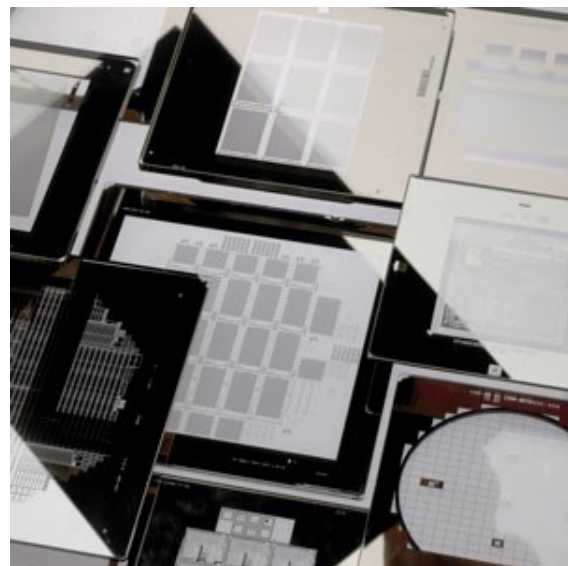


*Figura 17 – Mola da banco e cilindri diamantati per l'abrasione.*

Rimanendo vicini al mondo delle macchine utensili e, spesso utilizzate solamente in fase di finitura per componenti di grandi dimensioni, le smerigliatrici (più comunemente chiamate “mole”) sfruttano il moto traslante o rotatorio della superficie abrasiva ad essa montata per azioni di rettifica o sagomatura dei componenti. L'azione abrasiva è data dalla presenza di una grana composta da frammenti più o meno grandi di un materiale dalla durezza elevatissima (diamanti sintetici, carbo-nitruri e composti del boro per citarne alcuni) immersi in una matrice che funge da sostegno (in resina o materiale metallico come ottone e bronzo). Essendo la carta vetrata un tipo di carta abrasiva, con “grana” è indicata anche in questo caso il numero di frammenti medio su un segmento della lunghezza di un pollice: esso può variare da 40, per una sgrossatura grossolana, fino a valori superiori a 1000 per carteggiature finali e finiture. Questa categoria di macchine

utensili è adatta soprattutto per coppie cinematiche flessibili realizzate in materiale ceramico o comunque con caratteristiche di fragilità e durezza. Sebbene non sia così immediato, i materiali fragili sono in realtà ottimi nel campo delle piccole deformazioni grazie (oltre che alla loro capacità isolante) alla presenza della sola fase elastica prima del cedimento: i materiali duttili, infatti, manifestano un comportamento plastico prima di giungere a rottura, con conseguente nascita di deformazioni permanenti indesiderate. L'ultima considerazione da effettuare su questi materiali è sul sistema di ancoraggio del pezzo prima delle varie passate abrasive, le caratteristiche non magnetiche dei composti del silicio, ad esempio, non permettono l'utilizzo di magneti di presa mentre l'intrinseca delicatezza della struttura non consente l'utilizzo di sistemi di bloccaggio a pressione. Anche in questo caso si ricorre all'uso di cere o resine di fissaggio e, come in precedenza, lo svantaggio risiede nell'allungamento dei tempi di produzione per l'utilizzo di forni e trattamenti di pulitura successiva.

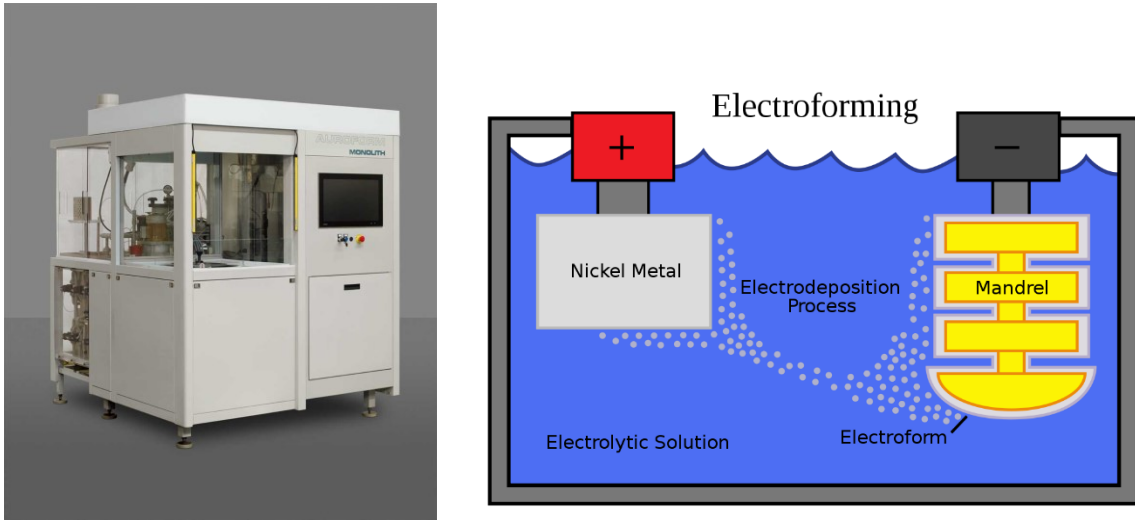
### 3.3 Litografia



*Figura 18 – a) Distributore centrifugo di materiale fotoresistente; b) Fotomaschere.*

La fotolitografia, nata come processo produttivo di maggior efficacia per i circuiti stampati, è l'assoluta protagonista nel campo dell'elettronica grazie alla capacità di realizzare (su grande scala) forme complesse a partire da sottili strati di rame. Il processo consiste nel partire da un substrato, chiamato "wafer" che altro non è che una matrice di silicio, che viene accuratamente pulito con una serie di trattamenti chimici e poi riscaldato. Viene poi applicata un'adeguata quantità di silicati organici per garantire l'adesione del materiale fotoresistente PR (e rendere il foglio ottenuto idrorepellente), quest'ultimo viene poi distribuito sul wafer e posto in forno a circa 100° per un minuto. Il tutto è poi coperto con una maschera intagliata con la forma da ottenere: il fascio di luce andrà a degradare le zone di PR rimaste scoperte, che verranno poi rimosse con l'utilizzo di una soluzione simile a quella utilizzata nella fase di "sviluppo" fotografico. Lo strato di ossido sottostante viene poi rimosso negli intagli "non coperti" con appositi agenti chimici, stessa cosa accade poi per il materiale fotoresistente ottenendo così il pezzo finito. La medesima procedura può quindi essere applicata nella realizzazione di giunti flessibili con buone caratteristiche di precisione e soprattutto con grandi volumi di produzione. La categoria di flexure così prodotta prende il nome di MEMS (Micro-Electro-Mechanical Devices) e sebbene fosse partita come un'attività di sola ricerca scientifica, la possibilità di realizzare componenti di queste dimensioni con fini tolleranze ne ha permesso il grande sviluppo all'interno del moderno mercato mondiale.

### 3.4 Elettroformatura

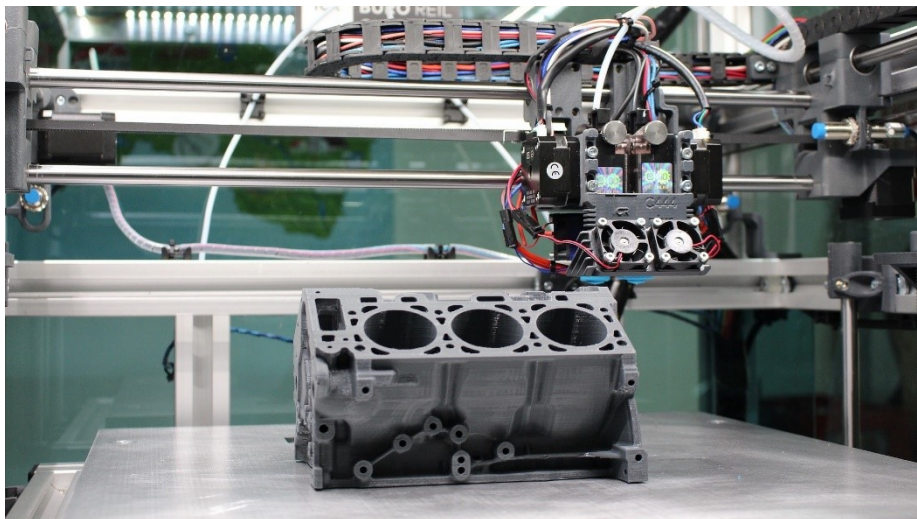


*Figura 19 – Macchina per l'elettroformatura e rappresentazione schematica del processo di elettrodeposizione.*

In precedenza si è parlato dell'utilizzo di resine come supporti per le lavorazioni alle macchine utensili intendendole come approssimazioni di procedure per "cera persa". Questa tipologia di formatura sfrutta l'utilizzo di un negativo realizzato per l'appunto in cera, da cui si ricava (dopo la colata di metallo fuso) il pezzo finito. Analogamente a questo sistema di produzione e sfruttando il fenomeno dell'elettrodeposizione si è arrivati ad utilizzare una nuova metodologia realizzativa chiamata "Elettroformatura". Il sistema si basa sull'utilizzo di un guscio di materiale conduttivo (la cui superficie esterna rappresenta il contorno interno del pezzo finale desiderato) che viene precedentemente passivato chimicamente per evitare la completa adesione del film di rivestimento e permettere lo smontaggio. La corrente viene fatta passare attraverso un bagno elettrolitico dove sono disciolti i sali del metallo da far aderire, una componente solida di quest'ultimo che è immersa nella soluzione funge da anodo mentre il catodo è il guscio solido di "attacco". L'elettroformatura termina una volta depositato lo strato necessario di metallo, successivamente viene rimosso il guscio tramite fusioni a temperature più elevate o dissoluzione

in agenti chimici. Il grande vantaggio dell'elettrodeposizione così applicata risiede nell'elevatissima precisione ottenibile con tolleranze nell'ordine dei nanometri, associata ad una larga varietà di geometrie ottenibili dato che ciò che va effettivamente lavorato nei dettagli è il guscio che funge da negativo. La finitura superficiale spesso non richiede lavorazioni successive ed il negativo del pezzo da ottenere può essere invece costituito anche da elementi plastici che sono più facilmente sagomabili. La pecca di questo procedimento è data dalle dilatate tempistiche di lavoro obbligando quindi a piccoli volumi di produzione.

### 3.5 Additive manufacturing



*Figura 20 – Realizzazione della testata di un motore V6 tramite stampa 3D.*

Con la presentazione dei differenti sistemi di realizzazione dei flexure joints, in questa discussione, si intendeva anche delineare un percorso lungo le innovazioni produttive che hanno caratterizzato l'industria meccanica nel corso degli anni. Questo viaggio non può che terminare con l'additive manufacturing, le stampanti 3D sono infatti diventate quasi di uso comune in alcune case, grazie anche all'abbordabilità economica dei modelli più semplici. Come evidente dal nome, la stampa 3D permette la realizzazione di solidi, l'input in ingresso non può chiaramente essere un file di



testo o un'immagine bensì un modello STL. Un file con questo formato (STereo Litography ovvero .stl) è un'evoluzione del comune .part o .STEP con cui si è soliti lavorare in ambienti di modellazione CAD perché contiene informazioni aggiuntive riguardanti l'orientamento del pezzo, la direzione di accrescimento, la tipologia e posizione dei supporti di fissaggio. Vengono inoltre definiti i parametri di "slicing" del modello per lo stampaggio a strati (la precisione dimensionale è inversamente proporzionale allo spessore di ogni layer che viene applicato) e le caratteristiche desiderate per il processo. L'apparecchio procede quindi alla stampa del pezzo che richiede comunque parecchio tempo, va inoltre considerato il post-trattamento necessario a migliorare le caratteristiche meccaniche ed estetiche per ottenere poi il prodotto finito. La vera rivoluzione attuata dall'additive manufacturing è stata quella di poter ottenere solidi con cavità e forme che non sono realizzabili attraverso nessun altro procedimento aprendo un vero e proprio mondo sull'ottimizzazione dei componenti meccanici. L'ottimizzazione topologica è la nuova frontiera degli elementi costruttivi poiché consente, grazie a software specifici, di trasformare geometrie comuni adibite a svolgere determinati compiti in precise condizioni di vincolo e carico, in forme tridimensionali che osservino i medesimi legami analitici ma con una notevole riduzione di massa, lasciando praticamente le sole parti "funzionali".



*Figura 21 – La successione delle fasi nel processo di ottimizzazione topologica di un componente meccanico.*

Ciò è ovviamente realizzabile solo attraverso manifattura additiva con un enorme guadagno sia sotto il punto di vista delle masse in gioco (e quindi delle prestazioni) sia per quanto concerne lo spreco di materiale. La sostenibilità di questo stampaggio è infatti una delle tematiche più dibattute all'interno di un mondo come quello meccanico che cerca sempre di migliorarsi per il bene comune, non vi è alcuno sfrido o residuo da smaltire a differenza delle lavorazioni alle macchine utensili visto che la solidificazione di materiale avviene solo nei punti previsti. Limitatamente all'additive di metalli, i costituenti di partenza sono polveri metalliche ottenute per atomizzazione (tramite gas o plasma) aventi composizione chimica omogenea, particelle sferiche e granulometrie tipiche, diverse a seconda del processo. I materiali utilizzati per questo tipo di procedimenti possono essere di diversa natura:

- **Ceramici:** elementi come zirconia, alumina e fosfato di calcio vengono utilizzati per creare componenti senza la presenza di fase plastica, ideali per i flexure joints. Sono state anche realizzate nuove classi di prodotti amorfi attraverso l'alternanza di strati adesivi e vetrosi che rendono il prodotto finale estremamente resistente;
- **Termoplastici:** senza dubbio i più utilizzati e con minor costo, i composti polimerici vengono spesso venduti in fili, polveri o trucioli. I più utilizzati sono l'acrilonitrile butadiene stirene (ABS), l'acido polilattico (PLA), il policarbonato (PC), il polivinilcloruro (PVC), il polipropilene (PP) ed il più comune polietilene (PE). Esiste inoltre un ramo di polimeri chiamato alcool polivinilico che è solubile in acqua ed è tipicamente usato come supporto temporaneo nelle strutture;
- **Biochimici:** per applicazioni biologiche di sostegno a strutture ossee e per impianti o protesi devono essere impiegati materiali che non vengano rigettati dall'organismo. Allo stesso tempo esse devono avere esattamente la forma del componente anatomico

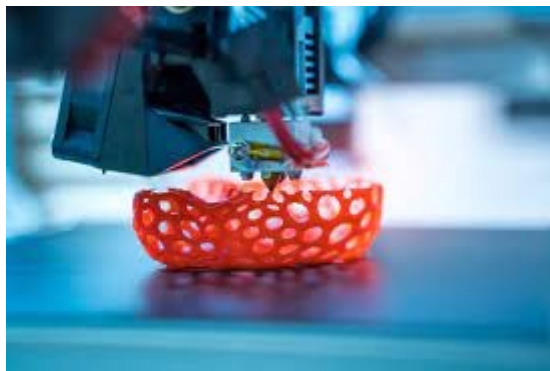
sostituito (legamenti, tendini, ossa) in modo tale da consentire un perfetto accoppiamento con la struttura biologica associata.

- **Metalli:** alcune limitazioni sorgono invece per questa classe dato che sono relativamente poche le leghe metalliche che è possibile stampare in 3D. Oltre ad una buona componente di acciai inossidabili e maraging, è possibile realizzare in additive composti di alluminio, titanio, bronzo e persino in Nitinol (una lega a memoria di forma capace di riassumere le condizioni originarie grazie all'apporto di calore dopo esser stato deformato). Su quasi 5500 leghe, tuttavia, solo un'esigua parte è utilizzabile a questi scopi, ciò è dovuto alle dinamiche di fusione e solidificazione che genererebbero microstrutture non tollerabili con grossi grani colonnari e cricche periodiche.

Ma come per ogni cosa, anche l'additive manufacturing presenta la sua parte di lati negativi. Da un punto di vista meramente produttivo, la stampa 3D è un processo altamente specifico e oneroso (economicamente e temporalmente parlando) che non è quindi destinato a componenti realizzabili con minor sforzo tramite i sistemi di produzione visti in precedenza. La lunghezza della procedura di stampaggio non la rende poi adatta a grandi volumi di produzione che richiederebbero tempistiche non accettabili. Per quanto riguarda invece il componente ottenuto per manifattura additiva, si dovrà sempre considerare un'analisi accurata di: rugosità superficiale e porosità (sono sempre necessarie ulteriori lavorazioni di finitura e solo alcune metodologie di additive prevedono l'utilizzo di un ambiente inerte e privo d'aria che penetri all'interno della struttura); stress residui (il materiale utilizzato si contrae durante la fase di solidificazione provocando tensioni all'interno che possono portare a cricche); microstruttura (la lenta solidificazione è causa di grani allungati e di grandi dimensioni che rendono fragile l'elemento minandone la resistenza meccanica); limiti geometrici (i componenti realizzabili non possono essere di grandi dimensioni a causa delle misure moderate del sistema di stampa). L'analisi nei

particolari degli aspetti legati alla stampa 3D richiederebbe una trattazione a parte quindi ci si limiterà ora a descrivere alcuni sistemi con cui tale procedura viene realizzata.

- **Material Extrusion:** il getto composto da metallo e materiale termoplastico viene depositato selettivamente per mezzo di un ugello nelle zone di interesse. La parte polimerica viene successivamente dissolta in un bagno liquido, garantendo la corretta distribuzione di forme e cavità mentre la componente metallica viene sinterizzata in fase di post-elaborazione in modo tale da incrementarne la densità e finitura. Il materiale utilizzato per strutture di sostegno e supporto è solitamente ceramico mentre è bene prestare attenzione ai ritiri che possono essere anche elevati.



*Figura 22 – Reticolo realizzato tramite estrusione di materiale plastico.*

- **Binder Jetting:** ovvero la deposizione a getto di un legante liquido su letto di polvere che viene mantenuto costante durante il processo per mezzo di un rullo che distribuisce lo strato senza soluzione di continuità. Il piatto su cui è ancorato il pezzo finale scorre man mano verso il basso sommerso dalle polveri non legate, la testina di stampa che si muove lungo un meccanismo planare rilascia da una serie di ugelli il legante liquido collegato ad un serbatoio. In alcune macchine può esser presente anche una sorta di lampada utilizzata per far evaporare eventuale legante in eccesso o permetterne la polimerizzazione. Il legame che consente l'ottenimento del pezzo finale

è dato da legante e polvere e non direttamente tra le particelle, dunque non si hanno tensioni residue significative mentre la velocità di costruzione è molto più alta rispetto ai metodi di fusione a letto di polvere basati su laser.



*Figura 23 – “Grappolo” di elementi realizzati in serie tramite binder jetting.*

- **Direct Energy Deposition:** il processo avviene con la fusione di un flusso di polvere o un filo metallico per mezzo di una fonte ad alta intensità energetica come laser, arco elettrico e fascio di elettroni. Rispetto alla tipologia a bagno di polvere, qui la disposizione di materiale è determinata da una testina dove convergono costituenti da fondere e concentrazione di energia. Le geometrie realizzate necessitano solitamente di ulteriori lavorazioni tramite controllo numerico a causa della poca morbidezza di linee ottenibile.



*Figura 24 – Fusione di flusso di polvere per mezzo di laser multipli.*

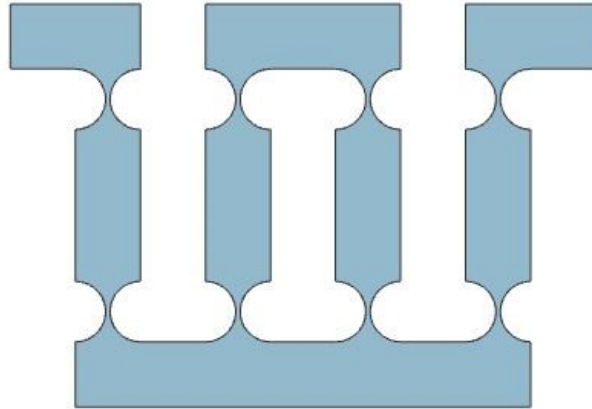
- **Powder Bed Fusion:** l'EBM (Electron Beam Melting) utilizza un fascio di elettroni come energia termica, ciò consente una maggiore velocità di produzione dato che il raggio può separarsi per riscaldare la polvere in più parti contemporaneamente e permette di non utilizzare supporti grazie al preriscaldamento della polvere prima della fusione. Il SLM (Selective Laser Melting) sfrutta invece più laser, ciascuno con un meccanismo di scansione a specchio indipendente dagli altri. Qui i supporti ricoprono un ruolo fondamentale al contrario del processo precedente, visto che devono permettere la dissipazione del calore generato alla pari del piatto di costruzione; il tutto è chiaramente svolto all'interno di un ambiente controllato con gas come argon ed azoto per ridurre il più possibile il grado di porosità del prodotto finito.



*Figura 25 – Fusione su letto di polvere tramite SLM.*

### **3.6 Assemblaggio**

I giunti flessibili, sebbene costituiti da un unico componente, sono naturalmente utilizzati all'interno di catene cinematiche anche complesse che necessitano quindi di un adeguato sistema di montaggio e assemblaggio. In alcune situazioni particolari, è l'intero compliant mechanism che rappresenta il prodotto finito delle precedenti lavorazioni meccaniche come, ad esempio, nel sistema di posizionamento traslazionale rappresentato in figura 26, esso funge da vincolo prismatico permettendo il solo moto orizzontale della base non vincolata posta al centro



*Figura 26 – Compliant mechanism composto da flexure ad intaglio circolare che funge da coppia prismatica.*

Nelle altre applicazioni, diventando costituenti di meccanismi a più componenti, i giunti flessibili devono essere fissati ai corpi rigidi secondo schemi opportuni assicurandone il corretto moto relativo; un altro aspetto da considerare è senz'altro quello relativo agli attuatori che governano le leggi di spostamento dell'assieme e che possono andare ad agire proprio in corrispondenza dei giunti. Il fissaggio può avvenire per mezzo di resine e sistemi di incollaggio solo in applicazioni dove i carichi esterni sono minimi, le condizioni dell'ambiente di lavoro non siano aggressive e la caratteristica riduzione di peso ed ingombri sia predominante a discapito di una manutenzione eventualmente maggiore (controlli più frequenti e accurata sensoristica). Per gli altri tipi di strutture si è soliti utilizzare i sistemi più classici di fissaggio (viti e bulloni fanno da padroni quando non c'è possibilità di eseguire saldature), coadiuvati da morsetti di chiusura che hanno compiti di: distribuzione su superfici maggiori degli sforzi di serraggio, simmetria di forze agenti sulle estremità del giunto, geometria ben progettata per ripartizione del momento flettente sviluppato e prevenzione di deformazioni plastiche. Questo discorso va generalizzato per ogni tipo di materiale costituente il giunto flessibile. In presenza di composti ceramici, ad esempio, è indispensabile ragionare sulle tensioni che vedono sovrapporsi gli effetti di compressione longitudinale e momento flettente in corrispondenza dei morsetti. Due sono gli aspetti di notevole

rilevanza per questi sistemi di grip, il primo riguarda lo stress in corrispondenza dei bordi delle superfici, dove le forme vengono arrotondate in modo tale da evitare dannose concentrazioni di  $\sigma$ . Il secondo verte su osservazioni sperimentali che suggeriscono come sia bene sfruttare materiali la cui resistenza meccanica è minore di quella del flexure: se  $\sigma_{s\_flex} < \sigma_{s\_mors}$  lo stress ai bordi del morsetto può generare plasticizzazioni che si ripercuoterebbero sul giunto portandolo a snervamento mentre, ragionando al contrario, se  $\sigma_{s\_flex} > \sigma_{s\_mors}$  le deformazioni nel cinematiso del compliant non saranno mai di natura plastica, quindi queste problematiche riguarderanno solo il sistema di grip. Un particolare da tenere sempre in considerazione nell'ambito della progettazione meccanica è inoltre quello delle dilatazioni termiche, vanno quindi realizzati appositi scassi nei supporti che tengano conto delle deformazioni associate alla temperatura, in modo tale da non generare sollecitazioni aggiuntive e prevenire la nascita di pericolose cricche. Eventuali correzioni alla struttura, una volta assemblata, possono riguardare l'aggiunta di componenti di sostegno alle parti funzionali dell'assieme, come ad esempio elementi a spicchio cilindrico per minor cedevolezza flessionale o rinforzi rigidi assiali per scongiurare pericoli di buckling.

Per quanto riguarda, invece, l'accoppiamento tra giunto ed attuatore che ne controlla il moto o le forze applicate, è possibile eseguire una stima degli spostamenti persi per mezzo di un'approssimazione delle rigidità in gioco secondo il modello statico ad un grado di libertà rappresentato in figura 27.



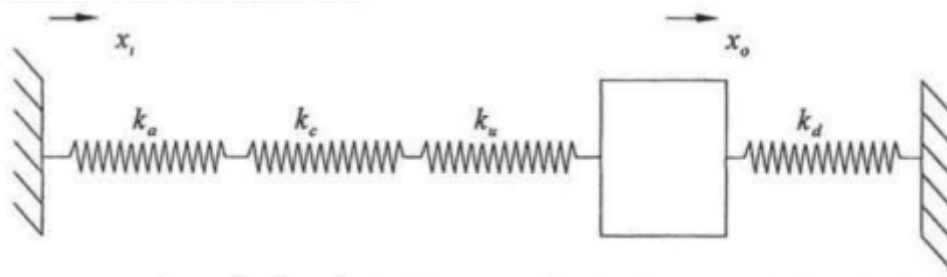


Figura 27 – Schematizzazione delle rigidità di giunto ed attuatore associate a quelle desiderate ed indesiderate. [11]

Indicando con  $x_i$  e con  $x_o$  la traslazione orizzontale rispettivamente dello strumento motore che controlla l'unico g.d.l. e del componente flessibile, è possibile definirne un rapporto tramite le rigidità equivalenti lungo le direzioni desiderate ed indesiderate  $k_d$  e  $k_u$ , insieme a quelle relative all'attuatore e al giunto  $k_a$  e  $k_c$  così espresso

$$\frac{x_o}{x_i} = \frac{k_a k_c k_u}{k_a k_c k_u + k_d (k_u k_a + k_c k_a + k_u k_c)} \quad (3.1)$$

Il sistema è idealmente perfetto per un rapporto pari a 1 e, come evidente nella formula, è preferibile avere valori di rigidità desiderata estremamente bassi in modo tale da avere una quasi uguaglianza tra numeratore e denominatore. Il problema risiede nel fatto che effetti desiderati ed indesiderati vadano di pari passo nel loro comportamento, quindi vi sarebbero associate grandi cedevolezza che causerebbero l'incontrollato propagarsi di effetti parassiti. Anche le dimensioni caratteristiche del modello su cui si sta lavorando, pregiudicano le scelte da effettuare in fase di progettazione; nelle microscale di nostro interesse sono preferiti controlli piezoelettrici estremamente accurati, a discapito di attuatori elettromagnetici che richiederebbero costi esorbitanti e forti dissipazioni di calore. Gli attuatori piezoelettrici sono capaci di associare grandi forze a spostamenti che arrivano fino a sensibilità nanometriche, mentre sono poco adatti per moti di maggiori dimensioni avendo bassa portata. Il loro funzionamento sfrutta un fenomeno caratteristico di materiali cristallini, come il quarzo, che vengono polarizzati (generando una

differenza di potenziale) quando soggetti ad una sollecitazione di tipo meccanico. Specularmente, se attraversati da corrente elettrica, generano una deformazione in campo elastico ad essa proporzionale; è sufficiente allora controllare il valore di tensione in ingresso all'attuatore per ottenere i cinematismi desiderati.

## 4 Reti neurali

### 4.1 Caratteristiche generali e perceptrone

Viene ora introdotta l'altra protagonista della nostra dissertazione, ovvero la rete neurale, di cui sono man mano evidenziati solo gli aspetti principali e quelli di nostro interesse. Il deep learning è senz'altro uno tra gli elementi più entusiasmanti del mondo informatico e anche uno dei più rivoluzionari; questo perché, a differenza dei comuni paradigmi della programmazione dove è l'utente ad affrontare il processo di problem solving tramite la scrittura di codici in linguaggio macchina, stavolta è il software ad individuare il corretto algoritmo di risoluzione sulla base dei dati forniti. In realtà, ciò che la rete è in grado di fornire non è il miglior algoritmo analiticamente possibile, bensì una struttura di calcolo capace di minimizzare l'errore tra la previsione in output ed il valore corretto di ciò che si intende ottenere, ma andiamo per gradi. Tra gli esempi di uso più comune di questi sistemi ci sono le classificazioni in tutte le loro sfaccettature: la possibilità di un calcolatore di riconoscere numeri o lettere scritte a mano, identificare melodie o parole, associare immagini, vertono tutte sull'utilizzo di schemi neurali. Le neural networks sono costruzioni logiche nate dalle osservazioni sul funzionamento dei sistemi biologici riguardanti il cervello e le attività neurali, da qui deriva infatti il nome di queste architetture che presentano elementi che si interfacciano continuamente tra loro proprio come nei neuroni. L'insieme di questi componenti, che sono regolati da precisi pesi e funzioni di attivazione, funziona di fatto come una scatola nera: essa riceve uno o più parametri, opportunamente trattati, come input in ingresso e fornisce la quantità di valori desiderata in uscita. Ciò che avviene all'interno di questa "black box" è un vero e proprio allenamento dei vari parametri della rete su un'ingente quantità di campioni (che presentano valori di ingresso e di uscita), aggiornandone iterativamente i valori caratteristici col fine ultimo di attuarne la più precisa ottimizzazione possibile. Il network così allenato viene poi

testato su un altro set di dati, chiamato di “test”, che ne verifica la bontà. Il rapporto che intercorre tra dati forniti ed ottenuti non viene mai ricavato come formulazione analitica dalla rete, ciò che viene piuttosto elaborato è il corretto peso da attribuire ad ogni dato e alla sua connessione attraverso la struttura con i risultati in uscita. L’unità funzionale è chiamata chiaramente “neurone” e ne è l’esempio più semplice il “perceptrone” raffigurato nell’immagine sottostante

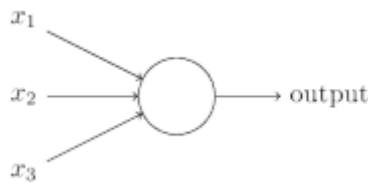


Figura 28 – Schematizzazione di un perceptrone. [19]

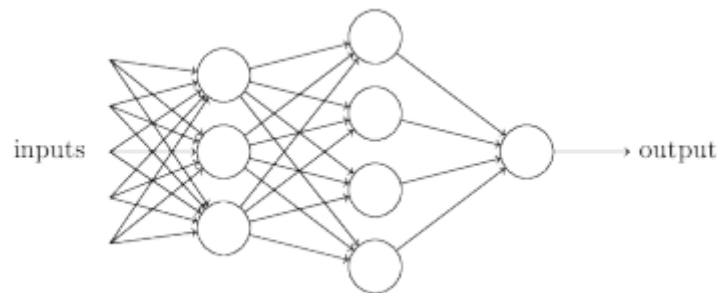
Come si può notare, l’elemento riceve (in questo caso) tre input  $x_1$ ,  $x_2$  e  $x_3$  con i relativi pesi  $w_1$ ,  $w_2$  e  $w_3$  e produce un output di valore 0 o 1 in base al superamento di una certa soglia chiamata comunemente “threshold”, da parte della sommatoria dei prodotti dei termini appena citati.

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases} \quad (4.1)$$

Solitamente questa formulazione viene rimaneggiata in modo tale da esprimere l’attivazione (1) o meno (0) del neurone in questione attraverso l’utilizzo di vettori di input  $x$  e pesi  $w$ ; viene inoltre definito un “bias”, inteso come il negativo del threshold, che ricopre lo stesso ruolo di una costante  $C$  in una funzione  $f$ , ovvero scala i valori ed è misura di quanto sia facile per l’elemento dare come output 1.

$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases} \quad (4.2)$$

Come detto, la struttura che si intende realizzare presenta una serie di elementi in collegamento tra loro, essi sono raggruppabili per mezzo di strati, o “layers” che indicano la profondità di analisi dei valori in ingresso. Nel nostro caso verranno trattate solo tipologie “feed forward”, dove il segnale propaga solo verso elementi dello strato successivo, trascurando tutti quegli schemi in “feedback”, ovvero con collegamenti verso neuroni dello stesso o di precedenti strati. In figura 29 è rappresentata una classica architettura neurale composta da cinque parametri d’ingresso nell’input layer, due hidden layers composti rispettivamente da tre e quattro neuroni e un output layer che produce in uscita un singolo valore.



*Figura 29 – Rappresentazione grafica di una rete neurale di tipo MLP. [19]*

La rete ottenuta è quindi l’unione delle singole unità funzionali, dove ad ogni freccia di collegamento è associato un peso che verrà man mano modificato lungo le iterazioni col fine di giungere alla miglior convergenza possibile. Questo discorso, estremamente semplice per il perceptrone, subisce delle modifiche quando si trattano invece neuroni dalle caratteristiche più avanzate. L’output in uscita da ogni elemento di calcolo, difatti, può assumere valori che sono diversi a seconda della funzione analitica che viene utilizzata; ciò è dovuto al fatto che il principale vincolo teorico da rispettare è quello che stabilisce che a piccoli cambiamenti nei pesi o nel bias, devono corrispondere piccole variazioni di output. Tale sorta di assioma non è verificabile per mezzo di due soli valori di output (0 e 1), dato che un lieve cambiamento di  $w_i$  potrebbe condizionare l’attivazione o meno del neurone. Per ovviare a questa problematica sono state

introdotte delle “funzioni di attivazione” che permettono l’ottenimento di valori intermedi tra i classici ON e OFF e garantiscono una migliore propagazione del dato per basse variazioni di  $w_i$ . Ne è un esempio la sigmoide, che per valori estremali approssima bene il comportamento del perceptrone, ma generalmente definisce l’output come una funzione  $\sigma(wx+b)$  così formulata e rappresentata

$$\sigma(z) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + \exp(-\sum w_j x_j - b)} \quad (4.3)$$

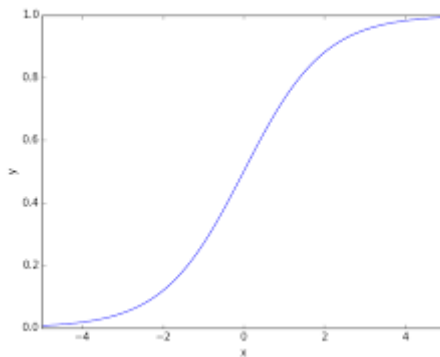


Figura 30 – Funzione di attivazione Sigmoide.

Se ipotizziamo che questo sistema rappresenti un neurone del primo strato nascosto dello schema in figura 29, si può pensare che il valore generato in uscita venga poi moltiplicato per il relativo peso di collegamento al secondo hidden layer e qui si ripeta il processo (simultaneamente agli altri neuroni) fino a giungere ad un valore di output. Il risultato in uscita differirà da quello previsto di una certa quantità che definisce la funzione di errore  $E(w)$ . L’Error Back Propagation (EBP) minimizza la somma degli errori quadratici (uscita desiderata – uscita effettiva) agendo sul gradiente di  $E(w)$  ed andando ad eseguire una variazione sui valori dei pesi  $\Delta w_i$  in funzione di un parametro  $\eta$  chiamato “learning rate” compreso tra 0 e 1.

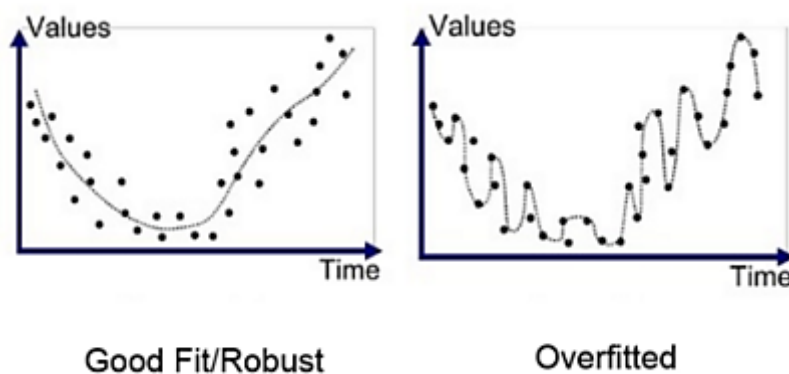
$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i} \quad (4.4)$$

La convergenza in tal caso non è comunque garantita poiché si può incappare in situazioni di minimo relativo a seconda della randomizzazione della matrice dei pesi di partenza. L'aggiornamento viene realizzato su ogni peso dell'intero dataset di allenamento attraverso delle epoche di apprendimento, ovvero cicli effettuati sul 100% dei dati in ingresso (che possono essere raggruppati in batch per facilitarne l'analisi). È possibile definire, inoltre, un comando di "early stopping" nel caso in cui alcuni parametri di controllo (come accuratezza o errore quadratico medio) non subiscano miglioramenti nel corso dell'allenamento, ciò permette di terminare l'ottimizzazione risparmiando tempo e calcoli dato che si è giunti al miglior risultato verosimilmente ottenibile. L'insieme di tutte queste iterazioni analitiche è chiamato "apprendimento", ovvero il processo per mezzo del quale la rete aggiorna la propria struttura al fine di approssimare una qualche funzione (che è quella che intercorre tra dati in ingresso e uscita). Nella classificazione di immagini, ad esempio, la figura può essere interpretata come un vettore monodimensionale contenente la totalità dei pixel: nel caso in cui il soggetto fosse rappresentato in scala di grigi, ognuna delle "celle" di colore possederà un valore compreso tra 0 e 255 che, opportunamente normalizzato, può essere valutato per addestrare una rete neurale al riconoscimento di un preciso pattern.

## 4.2 Overfitting

Le tipologie di reti neurali realizzabili sono innumerevoli, ognuna ottimale a seconda dell'utilizzo che se ne intende fare, ma un aspetto negativo che ne accomuna la maggior parte è il fenomeno dell'"overfitting". Un modello che lavora su un gran numero di campioni può sicuramente "abituarsi" a dare risultati che siano in grado di descrivere una nutrita varietà di situazioni, eppure non è lecito pensare che una maggior quantità di dati in ingresso renda migliore la rete. Questo perché l'architettura rischia di adattarsi troppo agli input in ingresso, imparando quasi "a

memoria” gli output associati: ciò che si ottiene in tal caso è una neural network che lavora bene sugli ingressi di allenamento, ma non è in grado di cogliere le relazioni che regolano un fenomeno non rendendo il modello generalizzabile a nuove situazioni. Una spiegazione grafica del fenomeno è rappresentata nei due grafici sottostanti.



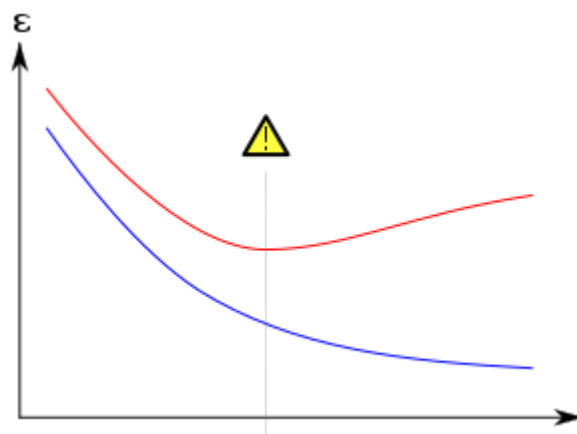
*Figura 31 – Visualizzazione grafica del fenomeno di overfitting.*

Per riuscire a cogliere il fenomeno dell’overfitting, e non invalidare dunque il modello, si possono utilizzare alcuni accorgimenti in fase di progettazione della rete neurale, prestando particolare attenzione ad alcuni parametri di interesse. La prima regola che è bene seguire è quella di dividere i dati in tre insiemi: l’insieme di training, è quello più corposo dei tre e corrisponde circa al 70% dei dati totali a disposizione (le percentuali di composizione fanno parte degli iperparametri della rete, che ne determinano l’apprendimento), esso costituisce di fatto l’algoritmo del sistema dato che la rete ne valuta iterativamente la corrispondenza con i cosiddetti “target” e aggiusta il proprio comportamento per diminuirne l’errore con gli “obtained outputs”; l’insieme di validazione (attorno al 20%) con nuovi dati in ingresso mai valutati in precedenza, essi forniscono al programmatore un’analisi della bontà del modello realizzato che ne permette di migliorarne gli iperparametri; l’insieme di test con i rimanenti dati, diversi da tutti i precedenti, verifica che effettivamente la rete sia ben allenata e funzioni su dataset estremamente generalizzabili. Un’altra



metodologia che può essere applicata successivamente è la “cross-validation” che cerca di diminuire la dipendenza diretta dei risultati dalla scelta più o meno corretta del dataset, essa considera di fatto l’unione degli insiemi di train e validation e ne attua una permutazione. Utilizzando un esempio pratico, si può pensare di utilizzare un dataset di mille elementi e dividerli in dieci gruppi, in tal caso la validazione incrociata prenderà iterativamente come esempio di test uno degli insiemi col 10% dei dati performando un allenamento sugli altri novecento elementi che andranno a costituire il train. Questo processo avviene per tutti e dieci i gruppi e se ne valuta lo “score” (un’accuratezza dei risultati) in modo tale da verificare che il modello sia svincolato dalla attribuzione dei dati all’insieme di train o test. Una volta effettuata una partizione degli input, e creata la rete neurale desiderata, si possono eseguire le varie epoche di apprendimento dove è sempre conveniente tenere d’occhio alcune variabili. Il parametro generalmente più utilizzato, indipendentemente dalla tipologia di NN realizzata, è la “loss”, ovvero una funzione che è in grado di comparare i valori di output desiderati ed ottenuti. Per i normali metodi di regressione, la loss è implementata come una funzione della media delle differenze tra target e previsioni, elevate al quadrato. Per gli algoritmi di classificazione, invece, è solitamente utilizzata la “cross-entropy”: viene valutato l’insieme dei parametri che hanno caratterizzato la previsione in output e si esegue un’analisi di tipo logaritmico sulle differenze con i target. Ad esempio, se abbiamo una corrispondenza di 0.9 su un neurone di output pari a “4” nella classificazione dei numeri scritti a mano, e la cifra rappresentata nell’immagine è effettivamente un quattro, avremo un basso valore di loss in uscita in relazione ad una differenza di 0.1. Un’altra variabile caratteristica dei problemi di classificazione è invece la “accuracy”, che non è altro che la percentuale di previsione effettuate correttamente (se su un insieme di mille immagini, la NN riesce a stimare correttamente la classe di appartenenza di 975 di esse, l’accuracy sarà di 0.975 con un’accuratezza percentuale del 97.5%). Tramite il controllo, soprattutto per via grafica di queste caratteristiche di rete è possibile capire se

si sta incappando in un overfitting alla struttura dati. Il primo caso è quello raffigurato nel grafico in figura 32, dove in blu è rappresentata la loss nell'insieme di training mentre in rosso quella nell'insieme di validazione, l'eccessiva abitudine della rete neurale qui è evidente soprattutto da un punto di vista concettuale: mentre l'architettura lavora sempre meglio sui dati per l'allenamento all'aumentare delle epoche, essa va in difficoltà con il dataset di validazione che è completamente diverso generando un aumento della loss ed una divergenza della soluzione.



*Figura 32 – Raffigurazione della discrepanza della loss tra gli insiemi di training e quelli di validazione.*

Un caso più attinente alla realtà di utilizzo delle neural network è invece quello in figura 33, qui vengono riportati i risultati di loss e accuracy in un problema di classificazione di immagini, alcuni degli iperparametri utilizzati sono la funzione di cross-entropy con batch di analisi di dieci campioni e learning rate  $\eta = 0.5$  per la back propagation.

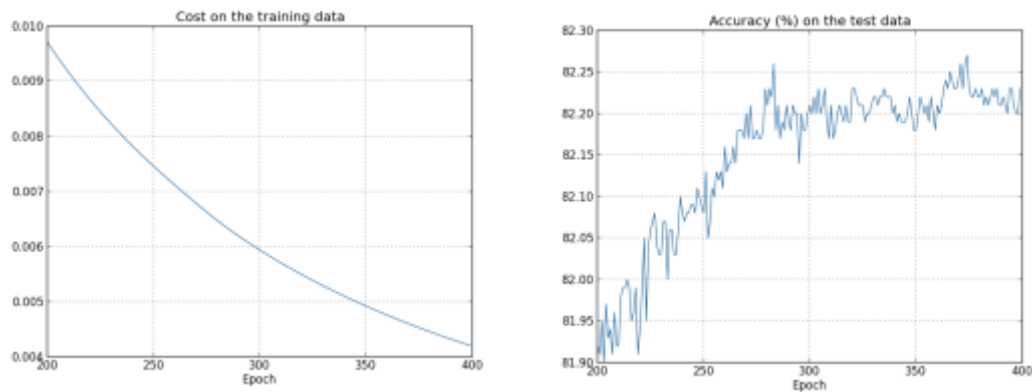


Figura 33 – Loss intesa come costo sull’errore ed accuratezza di una rete neurale utilizzata per un problema di classificazione di immagini. [19]

Le epoche di apprendimento utilizzate sono quattrocento e sono riportate in ascissa, nel grafico di sinistra è rappresentato l’andamento della loss che come si può ben vedere ha un ottimo comportamento monotono decrescente, ciò potrebbe far pensare ad un modello effettivamente ben congegnato, ma quanto riportato dalla curva dell’accuracy a fianco solleva qualche perplessità. Essa sale fino a raggiungere un valore massimo dell’82% circa, per poi avere delle fluttuazioni attorno a quel valore concludendo di fatto il processo di apprendimento. Ciò che è possibile dedurre da questo confronto è che attorno all’epoca numero 280 non si ha più alcun miglioramento nell’accuratezza della rete neurale, mentre la loss fa credere che stia diventando man mano più performante; possiamo dire, in conclusione, che non ha senso procedere nel processo di apprendimento oltre la 280<sup>a</sup> epoca dato che la struttura non è più in grado di generalizzare il suo comportamento e andrebbe in overfitting. Ciò è modulabile attraverso il comando di early stopping citato in precedenza.

### 4.3 Multi-layer perceptron

L’obiettivo che ci si è posti in questa discussione è quello di compiere un primo passo verso la caratterizzazione di un sistema meccanico piuttosto complesso da un punto di vista analitico,

utilizzando una rete neurale. Sulla base di una quantità più o meno grande di campioni, si cercherà di approssimare le leggi che governano il comportamento elastico dei flexure joints. Per fare ciò, è di fondamentale importanza capire quale tra le varie architetture di reti neurali più si presta alle nostre esigenze: le immagini non sono coinvolte in questo studio e ciò che intendiamo ottenere in output è un valore ben preciso, e non una corrispondenza ad una data classe di appartenenza, dunque non vengono prese in considerazione tutte quelle strutture utilizzate per la classificazione, come le “convolutional neural network” (CNN). Allo stesso modo, il tempo non gioca un ruolo fondamentale nelle previsioni trovandoci a lavorare in condizioni statiche, viene dunque scartata l’ipotesi di una “long short-term memory network” (LSTM) che si intende comunque menzionare per eventuali usi successivi di questo studio, dato che potrebbe consentire uno studio dinamico di previsione temporale anche nell’ottica della robotica collaborativa. Ottenere output, sulla base degli esempi di corrispondenza tra ingressi ed uscite nei campioni di allenamento, può essere inteso di fatto come un’analisi di regressione: si cercherà infatti di stimare un’eventuale relazione funzionale tra una o più variabili dipendenti (che da un punto di vista meccanico possono riguardare spostamenti, rotazioni, tensioni o deformazioni) e più variabili indipendenti (come ad esempio parametri geometrici di lunghezza, spessore e raggi di curvatura o dinamici come forze e momenti applicati).

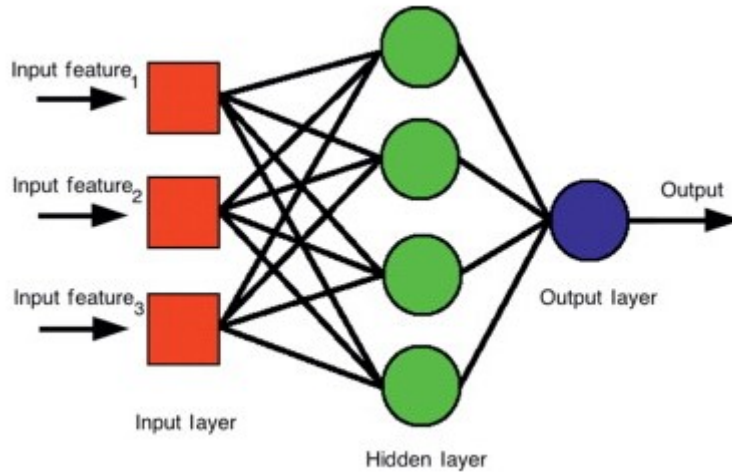


Figura 34 – Schematizzazione di un Multi-Layer Perceptron.

Una volta eseguite queste considerazioni, si è scelto di utilizzare una struttura chiamata “Multi-Layer Perceptron” (MLP) ovvero una rete neurale di tipo feedforward dove gli elementi costituenti (il nome è ambiguo, dato che vengono in realtà utilizzati neuroni e non perceptron nella maggior parte dei casi) sono completamente interconnessi tra loro, sfruttano i meccanismi di backpropagation già menzionati e consentono di approssimare funzioni continue garantendo la risoluzione di problemi non linearmente separabili. L’architettura è quella praticamente già descritta, con la presenza di almeno tre strati di cui uno di input, uno di output ( $o(x)$ ) ed uno nascosto ( $h(x)$ ), mentre per ogni neurone la computazione che viene eseguita è sempre

$$o(x) = G(b(2) + W(2)h(x)) \quad h(x) = s(b(1) + W(1)x) \quad (4.5)$$

Con  $G$  ed  $s$  che sono le funzioni di attivazione, che possono essere diverse da strato a strato, e la cui scelta risulta decisiva nell’ottenimento di una rete neurale che si comporti in maniera ottimale. Il numero di campioni e di neuroni deve essere scelto con particolare attenzione: un’esagerata quantità dei primi produce overfitting mentre l’opposto comporterebbe una bassa capacità di generalizzazione e applicazione della rete; un incremento dei secondi invece permette una più alta accuratezza delle previsioni a discapito di un aumento di tempo per le computazioni (è comune

pratica ridurne il numero una volta giunti ad un buon risultato, dato che spesso le variazioni nella soluzione sono poco sensibili mentre la struttura diventa visibilmente più snella). Per completezza ed utilità nelle fasi successive di implementazione della NN, vengono di seguito riportate le principali funzioni di attivazioni utilizzabili.






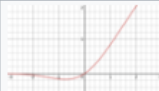


Name	Plot	Function, $g(x)$	Derivative of $g$ , $g'(x)$
Identity		$x$	1
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$\begin{cases} 0 & \text{if } x \neq 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$
Logistic, sigmoid, or soft step		$\sigma(x) \doteq \frac{1}{1 + e^{-x}}$	$g(x)(1 - g(x))$
Hyperbolic tangent (tanh)		$\tanh(x) \doteq \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$1 - g(x)^2$
Rectified linear unit (ReLU) <sup>[8]</sup>		$(x)^+ \doteq \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ $= \max(0, x) = x \mathbf{1}_{x>0}$	$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$
Gaussian Error Linear Unit (GELU) <sup>[5]</sup>		$\frac{1}{2}x \left( 1 + \operatorname{erf} \left( \frac{x}{\sqrt{2}} \right) \right)$ $= x\Phi(x)$	$\Phi(x) + x\phi(x)$
Softplus <sup>[9]</sup>		$\ln(1 + e^x)$	$\frac{1}{1 + e^{-x}}$
Exponential linear unit (ELU) <sup>[10]</sup>		$\begin{cases} \alpha (e^x - 1) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ with parameter $\alpha$	$\begin{cases} \alpha e^x & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ 1 & \text{if } x = 0 \text{ and } \alpha = 1 \end{cases}$

Figura 35 – Classificazione delle principali funzioni di attivazione utilizzabili.

## 5 Reti neurali applicate a sistemi meccanici

### 5.1 Quadrilatero articolato

Ciò che resta da fare ora è capire come coniugare l'architettura neurale scelta con quanti e quali parametri di interesse, cercando di ottenere risultati che ben approssimano un fenomeno in particolare. Per prendere confidenza con le reti neurali e per ricercarne l'effettiva adattabilità ai sistemi meccanici, si è deciso di partire da un classico quadrilatero articolato (fig. 36), che presenta un solo grado di libertà, e di indagarne le possibili relazioni sfruttando un MLP. L'idea è quella di costruire un modello che sia possibilmente generalizzabile per le analisi di tipo statico: questo consentirebbe di realizzare una solida base di partenza su cui poter andare a lavorare, per adattare successivamente a diverse tipologie di sistemi meccanici, le caratteristiche più adeguate e gli iperparametri più corretti.

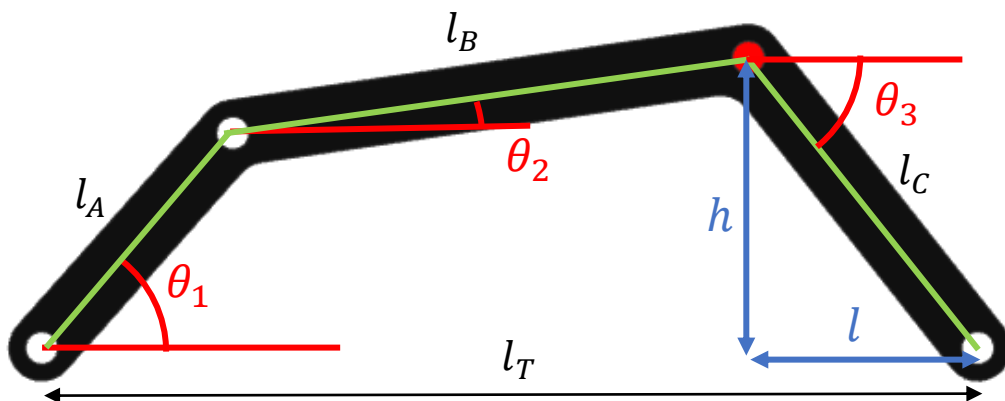


Figura 36 – Schematizzazione del quadrilatero articolato con i relativi parametri utilizzati nell'analisi.

Per prima cosa è stato necessario realizzare un dataset che venisse incontro alle nostre esigenze di estrema variabilità del dato, in modo tale che la rete potesse allenarsi ed essere testata su campioni diversi tra loro. Una scelta eseguita in fase di caratterizzazione è stata quella di partire da

un'architettura che fosse in grado di gestire una grande quantità di input e output, quasi a voler dare un grado di complicazione maggiore di quello effettivamente necessario per i giunti elastici. Tale decisione è stata presa per una questione di flessibilità della rete, in via teorica sarebbe quindi possibile valutare contemporaneamente una maggior quantità di aspetti in fase di realizzazione, lasciando maggior libertà al progettista. Si è così passati a definire gli input geometrici propri del sistema, dove nelle prime cinque colonne sono presenti diecimila valori, espressi in millimetri, riguardanti le lunghezze di ognuna delle tre aste sospese ( $l_A$ ,  $l_B$ ,  $l_C$ ), del telaio ( $l_T$ ) a terra e dell'angolo caratteristico ( $\theta_1$ ) che funge da movente dell'intero sistema. Per ottenere una sufficiente randomizzazione dei dati si è utilizzata la funzione "CASUALE" di Excel a partire da un valore di partenza per le aste ( $v_0 + \text{CASUALE}() \cdot 10$ ), mentre per l'angolo si è deciso di limitarsi all'analisi degli angoli compresi tra 0 e 180 gradi ( $\text{CASUALE.TRA}(0;180)$ ), il tutto nel rispetto della condizione di Grashof:  $M + m \leq i_1 + i_2$  (con M e m asta più lunga e più corta mentre  $i_1$  e  $i_2$  di lunghezza intermedia), in modo tale da avere almeno una manovella e nessun valore che fosse NaN.

Asta A	Asta B	Asta C	Telaio	Teta1	Teta2'	Teta3'	h'	l'	Teta2''	Teta3''	h''	l''	A	B	C
29,61875	73,09913	53,85786	70,81161	1,134464	-1,22788	0,894375	-41,9993	-33,7154	0,364794	-1,75746	52,92233	9,994821	-8522,51	3924,502	6561,61
23,53109	68,10775	49,53456	78,37448	1,553343	-0,94517	0,693923	-31,6802	-38,0793	0,358997	-1,28009	47,45619	-14,198	-10619,9	3204,811	8816,892
26,89335	68,3193	52,77252	71,40919	2,75762	-0,65676	0,6429	-31,6381	-42,2371	0,448382	-0,85128	39,69138	-34,7784	-13164,3	1376,555	11266,3
20,07442	72,26652	47,71294	74,10059	2,600541	-0,65298	0,780342	-33,5671	-33,9083	0,427478	-1,00585	40,29913	-25,5442	-13197	1494,34	11389,92
25,53673	69,57072	45,42616	73,92048	1,832596	-0,86661	0,674138	-28,3561	-35,489	0,272154	-1,2686	43,36764	-13,5197	-11205	3432,144	9870,05
21,52694	68,52486	48,35615	75,02165	1,256637	-0,99572	0,872228	-37,0294	-31,0989	0,413814	-1,45413	48,02746	-5,62852	-9370,02	2805,865	7450,881
26,88053	69,9459	47,59278	70,29571	0,750492	-1,09439	1,170195	-43,8247	-18,5598	0,399664	-1,86492	45,54902	13,79707	-7083,63	2564,563	5527,494
21,80942	67,51317	54,0883	76,29044	2,338741	-0,7838	0,632487	-31,9744	-43,6254	0,443973	-0,97232	44,6874	-30,4726	-12346,9	2118,346	10239,98
20,91703	74,42716	47,86914	79,18287	2,024582	-0,76752	0,757131	-32,8784	-34,7917	0,348199	-1,17645	44,19506	-18,3916	-13151,6	2798,477	11407,52
28,5002	65,34793	45,7274	75,58093	0,750492	-1,09147	1,002769	-38,5465	-24,6	0,409052	-1,68519	45,42854	5,219411	-7153,93	2540,347	5553,319
28,04845	72,33291	47,42555	72,72981	1,884956	-0,90331	0,688497	-30,1331	-36,6221	0,269925	-1,32188	45,96389	-11,6835	-11775,4	3859,057	10319,97
20,82287	68,15688	52,58891	75,69609	0,349066	-0,97779	1,220666	-49,3982	-18,0391	0,725372	-1,47308	52,33806	-5,13042	-7651,15	970,805	5080,95
24,26854	74,12397	48,31196	78,41907	1,518436	-0,93256	0,819213	-35,2971	-32,9872	0,323808	-1,42796	47,81997	-6,87716	-11437,2	3592,83	9699,628
21,46368	67,43086	47,27754	71,50024	1,291544	-1,01362	0,885365	-36,5997	-29,927	0,404038	-1,49494	47,1416	-3,58264	-8844,78	2782,496	7038,711

Figura 37 –Architettura della struttura dati utilizzata per la definizione della rete neurale del quadrilatero.



Le colonne successive sono invece occupate dagli output che intendiamo generare in uscita. Risolvendo le equazioni di chiusura del meccanismo è possibile ottenere le espressioni analitiche degli altri due angoli ( $\theta_2, \theta_3$ ) e le posizioni finali di un end effector posizionato sull'ultimo giunto rotoidale del quadrilatero ( $h, l$ ) in un sistema di riferimento posto sulla cerniera di collegamento col telaio a destra.

$$\begin{aligned}\theta_2 &= 2 \arctg \left( -\frac{B \pm \sqrt{A^2 + B^2 - C^2}}{C - A} \right) \text{ se } A \neq C \\ \theta_2 &= \left( \pi, -2 \arctg \frac{A}{B} \right) \text{ se } A = C \\ \theta_2 &= \left( -\frac{\pi}{2}, 2 \arctg \frac{A + B}{A - B} \right) \text{ se } B = C \\ \theta_3 &= \arctg 2 \left( -(l_1 \sin \theta_1 + l_2 \sin \theta_2), -(l_1 \cos \theta_1 - l_T + l_2 \cos \theta_2) \right)\end{aligned}\tag{5.1}$$

Per fare ciò è necessario definire delle variabili “di appoggio” così espresse

$$A = 2l_2(l_1 \cos \theta_1 - l_T) \quad B = 2l_1l_2 \sin \theta_1 \quad C = (l_1 \cos \theta_1 - l_T)^2 + (l_1 \sin \theta_1)^2 + l_2^2 - l_3^2 \tag{5.2}$$

In tutte le considerazioni, data la molteplicità della soluzione analitica, ci concentreremo unicamente sui risultati che descrivono un'ordinata positiva dell'end effector (ovvero la colonna gialla) in modo tale da conferire univocità agli output della rete neurale.

## 5.2 Shallow neural networks in Matlab

Un aiuto nella creazione di reti neurali può essere dato, nel toolbox di Matlab, dalla funzione `deepNetworkDesigner` che consente di utilizzare modelli già formati e pre-allenati a seconda delle necessità. Questo editor di architetture permette inoltre di sfruttare una lista di elementi costitutivi che possono essere utilizzati tramite drag and drop per formare rapidamente uno schema funzionale.

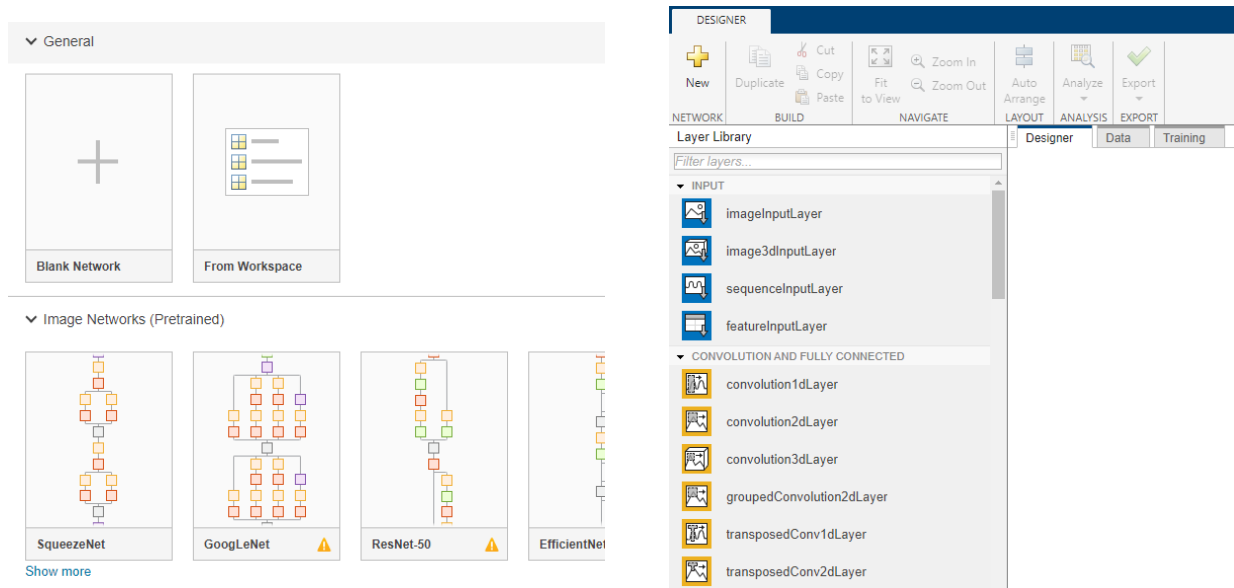


Figura 38 – Esempi di schemi implementativi presenti all'interno del deep learning toolbox di Matlab.

Una volta creato l'insieme dei campioni sui quali si intende far agire la rete neurale, viene utilizzato il Deep Learning Toolbox per avere un primo approccio con la struttura neurale, si andrà dunque a realizzare un semplicissimo script per un'analisi abbastanza superficiale del problema (Shallow Neural Network).

```

1  clear
2  clc
3  close all
4
5  load Quad_input.mat
6  load Quad_output.mat
7
8  input_data = Quad_input';
9  target_data = Quad_output';
10
11 net = feedforwardnet(20,'trainrp');
12 net = configure(net, input_data, target_data);
13
14 [rete,tr] = train(net, input_data, target_data);
15 view(net)
16 plotperf(tr)
17
18 output_data = net(input_data);
19 trOut = output_data(tr.trainInd);
20 vOut = output_data(tr.valInd);
21 tsOut = output_data(tr.testInd);
22 trTarg = target_data(tr.trainInd);
23 vTarg = target_data(tr.valInd);
24 tsTarg = target_data(tr.testInd);
25 plotregression(trTarg, trOut, 'Train', vTarg, vOut, 'Validation', tsTarg, tsOut, 'Testing')

```

Figura 39 – Codice per la realizzazione di una shallow neural network.

Il dataset (composto in questo caso da soli mille dei diecimila elementi generati) viene diviso in due matrici distinte di input e output, tramite il comando di “import”, che sono successivamente salvate nella stessa cartella di lavoro in modo da poter essere caricate nel file dei comandi snellendo la procedura attraverso il load. Le matrici dei campioni opportunamente divise vanno poi trasposte poiché la struttura che dovrà essere allenata richiede gli esempi sulle colonne ed i parametri sulle righe. Tramite il comando `feedforwardnet` viene implementata l’architettura del modello, il primo parametro esprime il numero di neuroni presenti all’interno dell’unico hidden layer presente mentre il secondo, riportato tra virgolette delinea la funzione di training che verrà utilizzata, in questo caso di tratta di Resilient Backpropagation. Successivamente la rete è preparata ad individuare dati di input e target grazie al `configure` che riceve il nome della NN ed i valori di ingresso e uscita per l’allenamento. Con le medesime opzioni espresse tra parentesi si procede poi ad allenare la neural network tramite il comando di `train`, in uscita è utile ottenere la variabile “tr” per tener traccia del processo e valutarne l’efficacia. Vengono infine ottenute le previsioni sui valori di input semplicemente applicando loro la rete stessa (`net`). Le righe di codice successive sono alquanto ripetitive ma utili per due motivi. Il primo è quello di assicurare che l’analisi sia svolta in maniera corretta da un punto di vista teorico: sebbene non sia stata fatta alcuna esplicitazione dei dati di train, validation o test in fase di configurazione, ciò è avvenuto lo stesso in maniera automatica, come evidente dalle funzioni proprie applicate a target ed output data (`trainInd`, `valInd`, `testInd`). La seconda è invece riguardante l’ottenimento dei risultati, grazie al confronto attuato sui due insiemi dal `plotregression`. Di seguito vengono raffigurate le uscite prodotte rispettivamente dai comandi di `view` e `plotperf` (fig. 40); a sinistra possiamo osservare la rete neurale nel suo complesso con evidenziate le principali caratteristiche: accanto alla casella di input è indicato il numero di quest’ultimi, da essa parte il collegamento del valore in uscita che viene moltiplicato per la matrice dei pesi  $W$  e a cui viene sommato il bias, il

venti indica chiaramente il numero di neuroni nello strato nascosto mentre all'uscita di "hidden" è rappresentata la funzione di attivazione utilizzata (sigmoide), il discorso è il medesimo per lo strato di output dove però la funzione è  $o(x) = x$  (identità). La figura di destra permette invece di avere un continuo aggiornamento sul processo di apprendimento ed è colma di informazioni sull'andamento del training. In questo caso il processo è terminato alla 548° epoca (su un valore limite di mille) dopo circa tre minuti e mezzo a causa del raggiungimento dei validation checks impostati di default, ovvero l'errore nell'insieme di validazione è aumentato per sei iterazioni consecutive e sono stati riaggiornati i valori di pesi e bias a quelli precedenti. Infine sono riportati i valori di performance e gradiente, che sono visibilmente maggiori di quelli di arresto, con i relativi algoritmi di errore ai minimi quadrati e randomizzazione di pesi e dati.



Figura 40 – a) Schema MLP realizzato; b) Monitor delle caratteristiche e dell'avanzamento del training della rete implementata.

Un'altra prova degli automatismi della rete nella scelta e nell'analisi dei dataset di allenamento, validazione e test emerge dai risultati di regressione ottenuti e rappresentati in figura 41. Per ogni insieme di dati sono rappresentati i singoli valori di target e la retta di interpolazione che meglio ne approssima il comportamento, la valutazione della bontà della regressione è affidata al Coefficiente di correlazione di Pearson R.

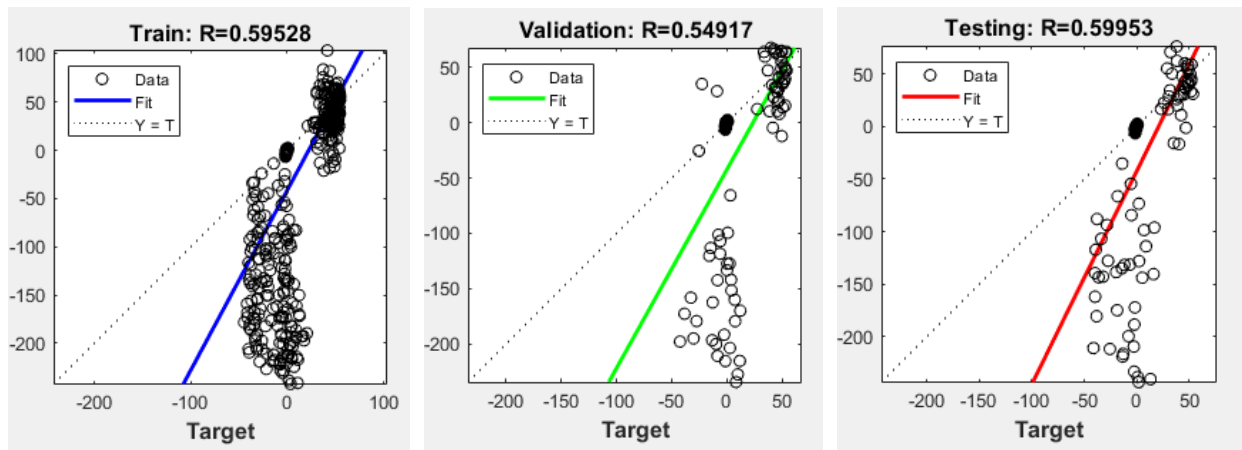


Figura 41 – Risultati dell'interpolazione dei dati nell'insieme di allenamento, validazione e test.

Questo parametro testimonia, sulla base del rapporto tra la covarianza di due variabili e il prodotto tra le loro deviazioni standard, la correlazione lineare tra i valori: i parametri registrati sono abbastanza lontani dalla condizione di perfetta corrispondenza (valore unitario) ma possono comunque fornire un punto di partenza per l'implementazione di una rete più performante.

### 5.3 Reti neurali in Python

La generalizzazione che si intende eseguire deve però poter comprendere stime di non linearità associate alle grandi deformazioni e necessita di appropriate funzioni di attivazione. Lo script preso in considerazione in Matlab è difatti estremamente superficiale, non si è in alcun modo definita una classe di iperparametri che potesse esser supervisionata e corretta iterativamente e la quasi totalità dei parametri di training sono rimasti quelli di default. Per poter ovviare a queste

problematiche si è deciso di sfruttare un ambiente più proprio delle reti neurali e che rendesse la struttura facilmente customizzabile, ci si è così spostati su Python e più nello specifico sulle librerie di Scikit-learn, fondamentale nell'apprendimento automatico e nelle procedure di regressione e macchine vettoriali (in simbiosi con altre librerie di natura prettamente matematica e di trattamento dati come NumPy e Pandas), e di Keras, un'Application Programming Interface importabile tramite Tensorflow, tra i più utilizzati nel machine learning grazie alla semplicità di realizzazione e gestione di reti neurali.

La prima cosa da fare è quindi quella di installare i pacchetti sopracitati in Python, tramite il comando `pip`, in modo tale da poterli importare correttamente nelle primissime righe di codice; l'editor di codice utilizzato in questo esempio è Visual Studio Code mentre per software come Colab non è necessario scaricare manualmente alcuna libreria poiché già contenuta all'interno. Si procede quindi ad importare i pacchetti necessari: Tensorflow e quindi Keras insieme a Sklearn per la gestione delle strutture neurali (per semplicità di programmazione vengono qui esplicitate subito le funzioni che si utilizzeranno), Pandas e NumPy per l'organizzazione dati e Matplotlib per la visualizzazione dei risultati.

```
1  import tensorflow as tf
2  from tensorflow import keras
3  from keras import initializers
4  from keras.models import Sequential
5  from keras.layers import Dense
6  from keras.callbacks import EarlyStopping
7
8  import pandas as pd
9  import numpy as np
10 import sklearn
11 from sklearn.model_selection import train_test_split
12
13 import matplotlib
14 import matplotlib.pyplot as plt
```

Il passo successivo è quello di caricare il file Excel realizzato in precedenza, la funzione `read_excel()` di Pandas consente la lettura del file il cui percorso viene inserito come input,

mentre la `to_numpy()` trasforma i dati ottenuti in un array NumPy facilmente gestibile. Il dataset che ne emerge viene quindi partizionato nelle unità di input e output per l'allenamento e per il test prestando attenzione a scegliere l'adeguato numero di campioni. Non è stata effettuata alcuna ulteriore diversificazione tra train e validation dato che a breve utilizzeremo un comando realizzato appositamente per assolvere a questa funzione.

```
16 df = pd.read_excel (r'C:Quadrilatero parametrizzato plus.xlsx')
17 dati = df.to_numpy()
18 input_train_raw = dati[0:3000,0:5]
19 target_train = dati[0:3000,9:12]
20 x_test_raw = dati[9700:10003,0:5]
21 t_test = dati[9700:10003,9:12]
```

Un passaggio che è bene effettuare in fase di preparazione dei dati è quello di normalizzare i parametri di input, questa accortezza permette di non perdere informazioni con differenze, anche grandi, tra i valori in ingresso generando una scalatura comune che permette di incrementare notevolmente l'accuratezza del modello (un eventuale numero troppo più grande degli altri vincolerebbero l'ottimizzazione iterativa del modello solo lungo quel parametro).

```
23 input_train = input_train_raw/input_train_raw.max(axis=0)
24 x_test = x_test_raw/x_test_raw.max(axis=0)
```

La sezione di codice successiva è in realtà estremamente legata alle funzioni di ottimizzazione che caratterizzano la rete neurale in ogni passaggio, questo perché non tutti gli algoritmi analitici utilizzabili sono in grado di generare in uscita dei risultati che siano indistintamente positivi e negativi. Per svincolarsi da queste restrizioni sui codomini sono stati definiti dei valori di soglia che fossero in grado di "traslare" correttamente le soluzioni su valori unicamente maggiori di zero. Una volta completato il processo, in fase di post processing, è sufficiente sottrarre il valore di soglia sempre positivo ottenuto tramite l'`absolute()` del valore minimo (`min()`)

```

26  soglia1_train = np.absolute(target_train[:,1].min(axis=0))
27  soglia1_test = np.absolute(t_test[:,1].min(axis=0))
28  target_train[:,1] = target_train[:,1] + soglia1_train
29  true_values = np.copy(t_test)
30  t_test[:,1] = t_test[:,1] + soglia1_test

```

Si passa ora al cuore pulsante della neural network tramite la definizione di una funzione chiamata `get_mlp()` che ha il compito di caratterizzare l'architettura. La struttura viene immediatamente inizializzata con la creazione di un modello tramite la funzione `Sequential()`, ad essa viene poi aggiunto un hidden layer tramite l'`add()` e la definizione di un `Dense()` che nel primo caso riceve come parametri il numero di neuroni (`units`), l'opportuna funzione di attivazione (`activation`) e la dimensione della matrice di input in ingresso (`input_shape`). Si noti come sia stata definita anche un'inizializzazione dei valori presenti all'interno della matrice dei pesi, in modo tale da limitare la varianza dei risultati tra una simulazione e l'altra e scongiurare la possibilità di incappare in minimi relativi della funzione da ottimizzare. La procedura viene poi ripetuta per uno strato addizionale di neuroni nascosti con numero di unità minore ed infine per la definizione dell'output layer dove tra le unità figura la quantità di uscite desiderate. Vengono successivamente impostati i parametri di fondamentale importanza riguardanti l'ottimizzatore analitico sfruttato (ciascuno adatto a particolari esigenze come l'SGD e l'RMSprop), dove viene scelto l'Adam, il sistema di controllo della funzione di loss realizzato tramite l'errore logaritmico sullo scarto quadratico medio e la relativa metrica di controllo.

```

32  def get_mlp():
33      model = Sequential()
34      model.add(Dense(units=30, kernel_initializer=initializers.RandomNormal(mean=0.0, stddev=0.05), bias_initializer=initializers.Ones(), activation='relu'))
35      model.add(Dense(units=15, activation='relu'))
36      model.add(Dense(units=3, activation='relu'))
37      model.compile(optimizer='Adam', loss='MeanSquaredLogarithmicError', metrics=['MeanSquaredError'])
38      return model

```

La NN trova quindi realizzazione semplicemente richiamando la funzione appena creata ed attribuendola alla nostra variabile rappresentante il multi-layer perceptron (`mlp`). Di quest'ultima è richiesta una rappresentazione nel terminale tramite il `summary()` che restituisce l'uscita



sottostante. Qui è visibile l'architettura della rete con il valore dei relativi parametri che è possibile allenare, dato della somma tra numero di bias e prodotto tra numero di neuroni e numero di input.

```
Model: "sequential_1"
-----
Layer (type)                Output Shape         Param #
-----
dense_3 (Dense)             (None, 30)          180
dense_4 (Dense)             (None, 15)          465
dense_5 (Dense)             (None, 3)           48
-----
Total params: 693
Trainable params: 693
Non-trainable params: 0
```

Figura 42 – Output del terminale per la descrizione dell'architettura della neural network.

Si giunge successivamente alla fase di definizione degli iperparametri e delle metodologie di allenamento della rete. Si procede quindi alla creazione della `run_experiment()` che riceve come variabili la funzione di rete precedentemente definita e i dataset di input e output per gli insiemi di allenamento e test. La divisione tra allenamento e validazione viene eseguita tramite il `train_test_split()` della libreria `model_selection` all'interno di `Sklearn` che la attua in modo randomico sul dieci per cento, in questo caso, del test size. Per definire l'`EarlyStopping()` del pacchetto `callbacks` di `Keras` è necessario decidere il parametro da monitorare per intervenire, la `loss`, ed il numero di epoche entro cui non deve variare per non palesare miglioramenti ed arrestare l'apprendimento, cinque; la `verboosità` impostata sul valore unitario permette di visualizzare l'andamento del `train` prima di fermarsi e una volta verificatosi l'es vengono rigenerati i pesi precedenti. Nella variabile `history`, poi, si immagazzinano i veri e propri iperparametri di `train` e `validazione` con il numero massimo di epoche, le dimensioni del `batch` ed il `callback` all'arresto preventivo tramite il comando di `fit()`. Con l'`evaluate()` si considera

l'accuratezza del modello e si memorizzano scarti quadratici medi ed errori di valutazione nelle variabili `loss` e `mse` mentre con il `predict()` vengono generate le previsioni della rete ed archiviate in `predictions`. Si realizza infine il processo di taratura di cui si è parlato in precedenza per l'ottenimento di risultati che consentano numeri minori di zero e si chiude la funzione con le variabili che si desidera restituire in uscita.

```
43 def run_experiment(get_model, input_train, target_train, x_test, t_test):
44     model = get_model()
45     model.summary()
46
47     x_train, x_val, t_train, t_val = train_test_split(input_train, target_train, test_size=0.1, random_state=42)
48
49
50     es = EarlyStopping(monitor='loss', patience=5, mode='min', verbose=1, restore_best_weights=True)
51     history = model.fit(x=x_train, y=t_train, validation_data=(x_val, t_val), epochs=300, batch_size=32, callbacks=[es], verbose=1)
52     loss, mse = model.evaluate(x_test, t_test, batch_size=32, verbose=1)
53     predictions = model.predict(x_test, batch_size=32, verbose=1)
54     predictions[:,1] = predictions[:,1] - soglia1_test
55
56     print("Testing loss: ", loss)
57     print("Testing mse: ", mse)
58
59     return model, history, predictions
```

Per avere una stima dell'accuratezza delle previsioni ottenute, si è scelto inoltre di riportare nel terminale la media degli errori percentuali commessi su ogni tipologia di output. Il codice si conclude con la rappresentazione degli andamenti di `loss` e scarto quadratico medio andando a valutare le due componenti nel parametro `history` presente all'interno della variabile `history_data` in uscita da `run_experiment()`. In ascissa sono riportate le epoche di apprendimento e le curve riguardanti insieme di `train` e `validation` sono plottate in colori diversi.

```
66 err_perc = np.mean(np.absolute(((true_values - predictions_mlp)/(true_values))*100), axis=0)
67 print("L'errore percentuale è di:", err_perc)
68
69 def plot_loss(history_data):
70     plt.figure(figsize=(10,10))
71     plt.plot(history_data.history['loss'], label='loss')
72     plt.plot(history_data.history['val_loss'], label='val_loss')
73     plt.ylabel('loss')
74     plt.xlabel('epoch')
75     plt.grid(True)
76     plt.legend(loc='upper left')
77     plt.show()
78
79 def plot_mse(history_data):
80     plt.figure(figsize=(10,10))
81     plt.plot(history_data.history['mean_squared_error'], label='mean_squared_error')
82     plt.plot(history_data.history['val_mean_squared_error'], label='val_mean_squared_error')
83     plt.ylabel('mse')
84     plt.xlabel('epoch')
85     plt.grid(True)
86     plt.legend(loc='upper left')
87     plt.show()
88
89 plot_loss(history_mlp)
90 plot_mse(history_mlp)
```

## 5.4 Analisi dei risultati

È di particolare interesse, per i fini della trattazione, osservare come la variazione delle caratteristiche della rete influenzi l'accuratezza del risultato, ciò consente di stabilire le condizioni ottimali nelle quali la NN deve lavorare. Eseguire una grande quantità di test sul codice appena descritto è la chiave per la valutazione del rapporto tra precisione e costo computazionale, non è infatti detto che la corrispondenza sia lineare e che ad una maggiore "pesantezza" di calcolo siano associati output migliori. In questo caso le variabili che contraddistinguono il funzionamento dell'architettura creata sono tre: il primo è il numero di campioni di riferimento presi in considerazione dei diecimila che sono stati inizialmente calcolati, il secondo è la quantità di neuroni che costituiscono ciascun hidden layer ed il terzo è il massimo numero di epoche lungo il quale viene eseguito l'addestramento (senza considerare l'eventuale arresto per early stopping). Al termine di un lungo processo di ottenimento dei risultati, si è valutato l'errore percentuale commesso su ciascuno dei tre output e se ne è eseguita la media per poter rappresentare in un grafico 3D le relazioni tra i tre parametri che si è scelto di far variare. È emerso che il peggior risultato, dal punto di vista della vicinanza delle previsioni al risultato atteso, è quello ottenuto con mille campioni di input che vengono iterativamente analizzati su due strati nascosti composti rispettivamente da 30 e 10 neuroni per un periodo di addestramento di 50 epoche. In questo caso l'errore percentuale medio commesso sugli output è del 21% con uno scarto quadratico medio di 26.458.

```
Epoch 50/50
29/29 [=====] - 0s 12ms/step - loss: 0.0294 - mean_squared_error: 28.9074 - val_loss: 0.0253 - val_mean_squared_error: 27.8984
4/4 [=====] - 0s 3ms/step - loss: 0.0356 - mean_squared_error: 26.4581
4/4 [=====] - 0s 2ms/step
Testing loss: 0.0355503186583519
Testing mse: 26.4581241607666
L'errore percentuale è di: [23.04806378 23.48823864 17.45229235]
```

Nelle immagini sottostanti vengono poi rappresentati gli andamenti di loss e MSE al progredire delle epoche di addestramento

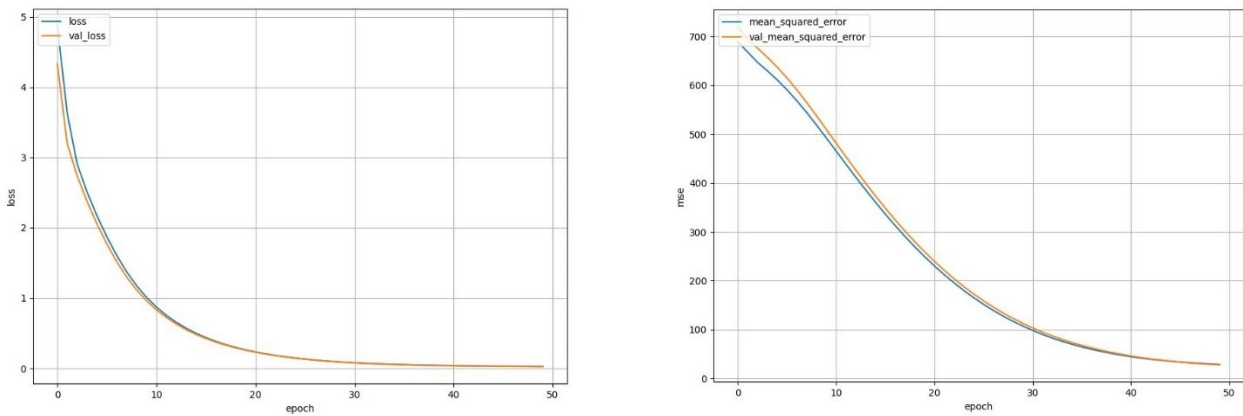


Figura 43 – Andamenti di loss e scarto quadratico medio all’aumentare delle epoche di addestramento.

Effettuando man mano le variazioni necessarie si è invece giunti al miglior risultato, sotto riportato, ottenuto con il triplo dei campioni rispetto a quello precedentemente descritto, ovvero tremila, con un numero di neuroni leggermente superiore negli hidden pari a 75 e 25, per un periodo di trecento epoche. Tale risultato presenta, come evidente, una loss estremamente bassa associata ad uno scarto quadratico medio di 0.745 che consente di ottenere risultati con un errore medio del 2,42%.

```
81/85 [=====>...] - ETA: 0s - loss: 4.6381e-04 - mean_squared_error: 0.5509Restoring model weights from the end of the best epoch: 246.
85/85 [=====>] - 1s 15ms/step - loss: 4.5608e-04 - mean_squared_error: 0.5430 - val_loss: 0.0011 - val_mean_squared_error: 0.6915
Epoch 251: early stopping
10/10 [=====>] - 1s 70ms/step - loss: 5.1945e-04 - mean_squared_error: 0.7447
10/10 [=====>] - 1s 6ms/step
Testing loss: 0.000519447959959507
Testing mse: 0.7446565628051758
L'errore percentuale è di: [3.34753489 1.4804123 2.51601425]
```

Tenendo uno storico degli errori medi sugli output associati ai parametri che sono stati variati di volta in volta è stato possibile realizzare i grafici in figura 44 che rappresentano l’andamento dell’errore nelle tre dimensioni

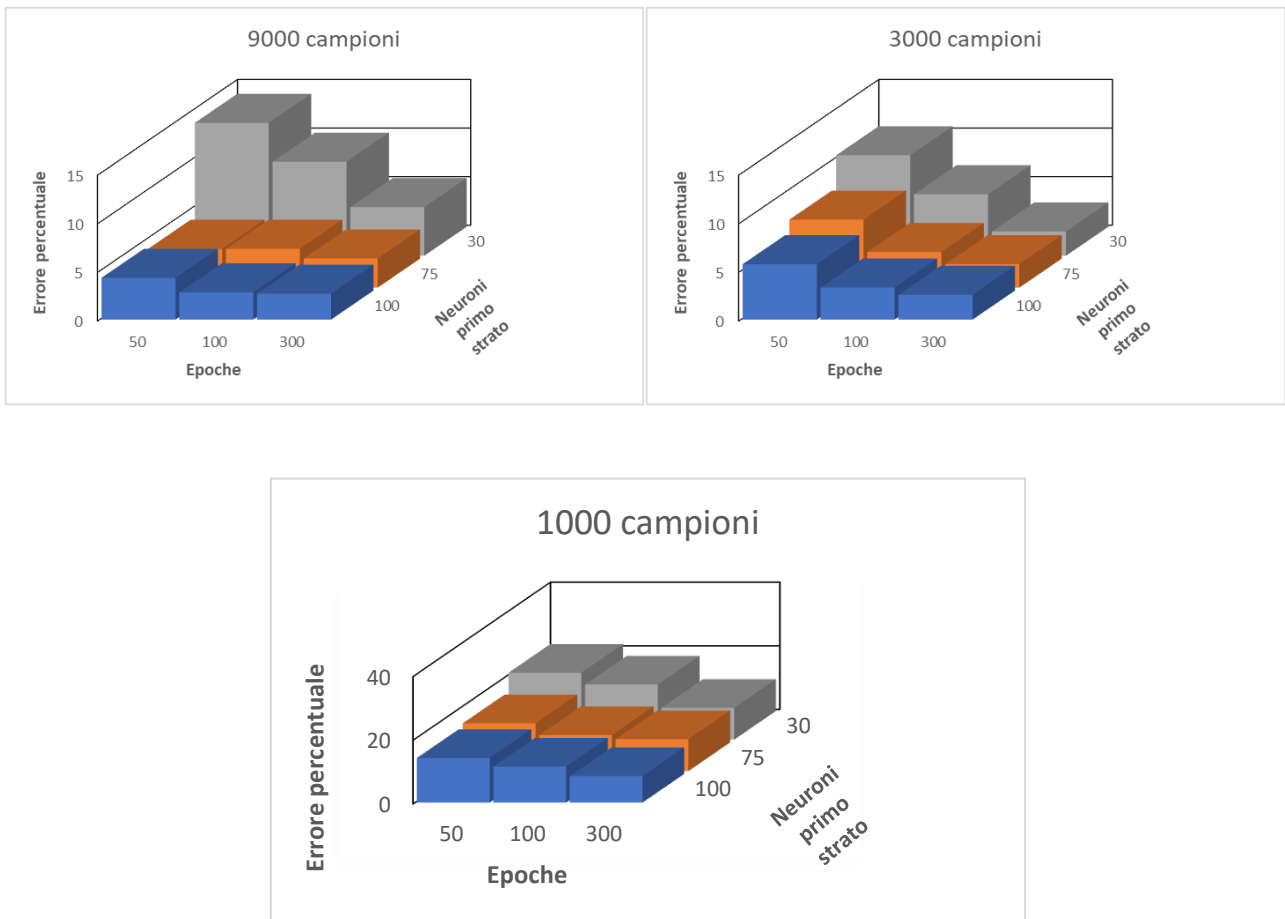


Figura 44 – Andamento degli errori percentuali commessi in funzione del numero di campioni, epoche e neuroni utilizzati.

Come si può notare, l'andamento è chiaramente discendente all'aumentare degli elementi di calcolo della neural network ma ciò che colpisce è che l'imprecisione sugli output raggiunga un minimo per poi risalire leggermente aumentando numero di campioni, neuroni ed epoche. Ciò implica che superato il valore ottimale, il gioco tenda a non valere più la candela: non si hanno miglioramenti nell'approssimazione dei risultati e, al contrario, si rischia di avvicinarsi alle condizioni di overfitting perdendo generalità nella rete.

## 5.5 Support Vector Regression

Un metodo alternativo che viene invece utilizzato per il quarto parametro, ed espande le possibilità di studio sulle reti neurali a molteplici soluzioni, è quello della Support Vector

Regression. Questo tipo di tecnica viaggia parallelamente a quella presentata nell'addestramento delle reti neurali, come essa è infatti in grado di rappresentare funzioni non lineari anche complesse, sfruttando però una multidimensionalità spaziale del problema di regressione. Ciò che fa l'SVR è di fatto un'estensione dei classici problemi di classificazione con insiemi linearmente separabili ad un iperpiano (o un insieme di iperpiani) in uno spazio a più dimensioni, questo consente di approssimare relazioni che possono essere non lineari in un piano tramite funzioni lineari in un'altra direzione. Per questo tipo di analisi è sufficiente importare il pacchetto di Support Vector Machine (SVM) da Sklearn ed utilizzarne la relativa funzione di SVR() facendo attenzione a caratterizzarla con gli opportuni parametri. Tramite il comando di fit() visto in precedenza si ha l'apprendimento del corretto algoritmo con cui effettuare la regressione, mentre la valutazione della bontà dell'analisi è affidata allo score() che fa le veci del coefficiente di determinazione  $R^2$ .

```
27 vett = svm.SVR(kernel='rbf', C=1, epsilon=0.05).fit(input_train, target_train)
28 print("Lo score è:", vett.score(input_train, target_train))
29 predictions = vett.predict(x_test)
30 err_perc = np.mean(np.absolute(((t_test - predictions)/(t_test))*100), axis=0)
31 print("L'errore percentuale è di:", err_perc)
```

L'ascissa con cui viene caratterizzato l'end effector fa riferimento ad un sistema di coordinate posto all'estremo destro del telaio, i valori saranno dunque indistintamente positivi e negativi per ogni tipo di configurazione. In questo caso non è necessario utilizzare alcuna soglia con cui "traslare" i risultati, viene invece fissato il parametro epsilon, per mezzo del quale è possibile valutare una sorta di "raggio d'azione" della funzione di loss da minimizzare, ad un valore di 0.05. Successivamente vengono fatti variare il numero di campioni presi in considerazione ed il coefficiente di regolarizzazione C e si valuta l'andamento dell'errore percentuale rappresentato tridimensionalmente nella figura sottostante.

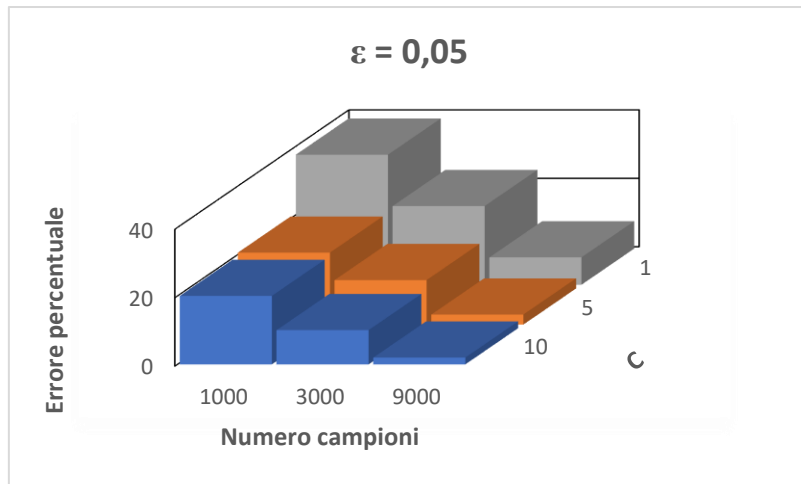


Figura 45 – Andamento dell'errore percentuale commesso nella SVR con  $\epsilon$  costante in funzione del parametro  $C$  e del numero di campioni utilizzati.

I risultati ottenuti con il vettore di supporto sono estremamente più sensibili al numero di campioni utilizzato e trovano quindi maggiori applicazioni per casistiche che presentino un'elevata quantità di dati a disposizione. Si noti come al progressivo triplicare dei dati per l'allenamento, l'errore percentuale scenda drasticamente (ad esempio per  $C=10$ ) dal 20% al 10%, per giungere infine all'1.98%. Ciò potrebbe sollevare problemi di overfitting nel caso di una piccola quantità di parametri di input, ma è comunque bene sottolineare come i Support Vector Machines rimangano una valida alternativa nel caso di reti con un singolo output desiderato: il vero vantaggio risiede infatti nell'enorme guadagno da un punto di vista temporale rispetto alle NN, con attese di calcolo decimate a parità di campioni presi in considerazione.

## 6 Reti neurali applicate a meccanismi compliant

### 6.1 Analisi FEM tramite Ansys

L'analisi prosegue spostandosi sul software Ansys che viene utilizzato per le analisi agli elementi finiti: ciò fungerà da anello di collegamento tra i sistemi di reti neurali implementati per il riconoscimento di un pattern ed i giunti flessibili. Sarà infatti sufficiente realizzare un modello in due dimensioni, dato che per gli elementi di interesse si ha sempre una conformazione planare del giunto con una misura notevolmente minore delle altre due, dove è possibile eseguire parametrizzazioni di ogni tipo. Riprendendo per un attimo il discorso delle neural network, va sottolineato come queste siano particolarmente adatte per analisi che comportino una grande quantità di dati in input (come il numero di pixel in un'immagine con i relativi valori di RGB); ciò consentirebbe di generalizzare estremamente lo studio FEM creando dipendenze anche con le variabili che caratterizzano la geometria stessa del giunto. Prima di entrare nella descrizione delle scelte effettuate in fase di preprocessing, vale la pena considerare la totalità delle dipendenze che si possono attribuire al giunto ad intaglio circolare in figura 46, protagonista dello studio.

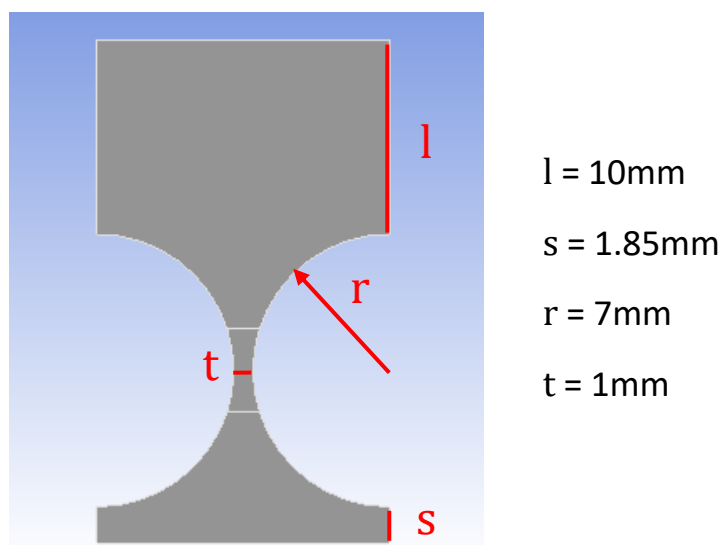


Figura 46 – Giunto flessibile 2D ad intagli circolari con dimensioni associate.



La geometria del flexure consentirebbe di definire input caratteristici riguardanti: lo spessore minimo della strizione  $t$ , il raggio di curvatura  $r$  degli intagli realizzati, la lunghezza  $l$  a cui si trova la superficie di attuazione dei carichi esterni ed infine la distanza  $s$  dalla superficie vincolata. Altre grandezze caratteristiche che andrebbero ad influire sul comportamento del sistema sono quelle relative al numero ed alla qualità degli elementi finiti che approssimano l'interezza della forma parametrizzabili tramite dimensioni di mesh, per non parlare della tipologia di materiale impiegato e le relative varianze su modulo di Young e di Poisson. Lo scorcio degli ingressi nel parameter set dell'immagine 47 rappresenta solo nove delle innumerevoli complicazioni parametriche che è possibile effettuare durante il processo di progettazione, il tutto naturalmente a scapito di simulazioni notevolmente più dispendiose da un punto di vista numerico e conseguentemente temporale.

Table of Outline A12: Design Points of Design of Experiments									
	B	C	D	E	F	G	H	I	J
1	P1 - t (mm)	P2 - r (mm)	P3 - l (mm)	P4 - s (mm)	P5 - dist (mm)	P6 - Surface Body Thickness (mm)	P9 - Force X Component (N)	P10 - Force Y Component (N)	P11 - Moment Magnitude (N.m)
71	0,90117	7,4651	10,07	1,7674	4,5988	1,7326	0,95349	1,0465	0,052326
72	1,5988	7,4651	10,07	1,7674	4,5988	1,7326	0,95349	0,95349	0,052326
73	0,90117	6,5349	11,93	1,7674	4,5988	1,7326	0,95349	1,0465	0,052326
74	1,5988	6,5349	11,93	1,7674	4,5988	1,7326	0,95349	0,95349	0,052326
75	0,90117	7,4651	11,93	1,7674	4,5988	1,7326	0,95349	0,95349	0,052326
76	1,5988	7,4651	11,93	1,7674	4,5988	1,7326	0,95349	1,0465	0,052326
77	0,90117	6,5349	10,07	2,2326	4,5988	1,7326	0,95349	1,0465	0,047674
78	1,5988	6,5349	10,07	2,2326	4,5988	1,7326	0,95349	0,95349	0,047674
79	0,90117	7,4651	10,07	2,2326	4,5988	1,7326	0,95349	0,95349	0,047674

Figura 47 – Esempio di dataset parametrico ottenibile tramite Design of Experiments su Ansys.

Lo stesso discorso vale per le uscite che Ansys è in grado di fornire, in questo caso però le reti neurali non si comportano bene nel definire con sufficiente accuratezza una grande quantità di output, va quindi prestata attenzione alle variabili che si intende ottenere. Si è scelto allora di concentrare l'attenzione su un componente meccanico che fosse già realizzato, con misure e tolleranze ben definite, e ci si è prefissati l'obiettivo di associarvi una rete neurale che fosse in grado di descriverne il comportamento elastico con discreta precisione. In altre parole, si vuole ricavare una costruzione analitica che associ forze e momenti agenti sull'elemento alle tensioni,

spostamenti e rotazioni di quest'ultimo, svincolandosi completamente dalla definizione di una matrice di rigidità e vertendo invece su un numero di simulazioni adeguato all'apprendimento delle NN.

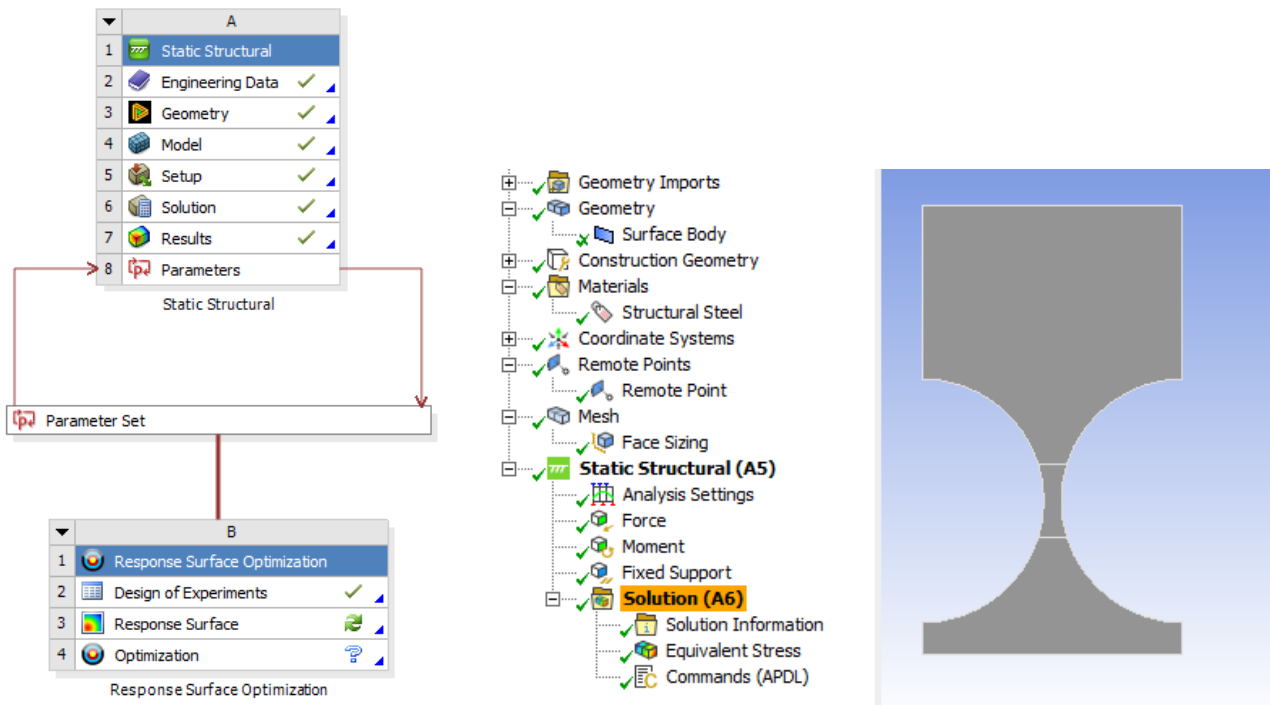


Figura 48 – Struttura utilizzata per l'analisi ed albero dei parametri definiti in Static Structural.

Il flexure joint è stato realizzato sul programma di modellazione di Ansys chiamato Design Modeler, dove sono state inserite le quote necessarie e si è scelto di semplificare la tipologia di analisi in due dimensioni per facilitarne i calcoli inserendo una Surface Body Thickness di un millimetro. Il preprocessing si sposta poi all'interno del modulo Static Structural dove viene impostato il classico acciaio strutturale come materiale e ne vengono considerati gli effetti non lineari associati alle grandi deformazioni. Le rotazioni vengono calcolate a partire da un "remote point" posto in corrispondenza della superficie libera, sull'esatta verticale al pivot point (che funge anche da centro del sistema di riferimento usato). Per quanto concerne la scelta della mesh si è deciso di utilizzare elementi di ordine quadratico, con una size di 0.4 millimetri, che consentissero

un adeguato numero di nodi senza appesantire troppo la computazione. Data la natura iterativa delle reti neurali e l'ingente numero di dati per l'apprendimento, infatti, le tempistiche ricoprono un ruolo fondamentale all'interno del processo di progettazione e si rischiano spese temporali davvero troppo dilatate. Un affinamento viene comunque realizzato tramite un sizing superficiale di due decimi di millimetro in corrispondenza della strizione, dove vi è una concentrazione degli stress e delle deformazioni ed è dunque richiesta una maggiore accuratezza nei risultati. Dopo aver verificato che le metriche di controllo della qualità degli elementi fossero abbastanza elevate, si sono impostati gli step di controllo e l'utilizzo di grandi spostamenti all'interno degli analysis settings. La forza esterna è definita per mezzo delle sue componenti lungo gli assi X e Y (che costituiscono i primi due parametri di rete) e viene applicata anch'essa al centro della superficie libera; nella medesima posizione viene successivamente posizionato il momento flettente agente lungo la direzione Z, terzo ed ultimo parametro. I vincoli del modello vengono completati imponendo che sia fissa al telaio la base inferiore, dopodiché si procede a definire i parametri di output desiderati: in questo caso è di particolare interesse la tensione equivalente, calcolata tramite Von Mises, che si sviluppa all'interno del pezzo (ci si deve assicurare di rimanere all'interno della fase elastica per evitare eventuali plasticizzazioni) gli spostamenti verticali e orizzontali ed infine l'angolo di rotazione attorno al pivot point. Queste considerazioni permettono, fissata la geometria, di determinare una serie di simulazioni da dare successivamente in pasto al multi layer perceptron con un totale di tre input e quattro output. Per fare ciò si sfrutta il modulo di Response Surface Optimization dove è presente una funzione di valutazione dei parametri: per mezzo del Design of Experiments è possibile generare una varietà di configurazioni nei quali vengono variati i parametri di ingresso ed eseguire un update dei dati per valutarne le uscite. Per mezzo di Excel vengono generati degli input randomici di forza che varia da -1 a 1 N, e di coppia che varia tra -

0.02 e +0.02 Nm; questi sono poi importati all'interno di Ansys come design points sottoforma di file CSV e l'insieme delle soluzioni viene lentamente ottenuto.

## 6.2 Generazione dei dati

Nella valutazione dei risultati si intende ancora una volta sottolineare l'importanza del fattore che più di tutti caratterizza le analisi presenti in questa trattazione: il tempo. A differenza del caso del quadrilatero, dove i dati venivano automaticamente ottenuti per via analitica su Excel e la maggior parte della spesa temporale era richiesta in fase di convergenza della rete neurale, stavolta si deve considerare l'enorme sforzo computazionale attribuito alle simulazioni FEM. La scelta di agire in condizioni statiche, non tenendo conto di eventuali accelerazioni legate al reale funzionamento del giunto (giustificate comunque da bassi valori di inerzia complessiva); di utilizzare mesh non estremamente accurate e che dunque richiedessero un elevato numero di nodi; ed infine quella di lavorare in situazioni di plane stress riducendo il problema a due sole dimensioni, sono state propedeutiche all'ottenimento di campioni in tempi verosimili. Con tali caratteristiche implementative, la valutazione di ciascuno dei design point generati per i nostri scopi richiede in media tre minuti e dieci secondi, un'inezia per un'analisi agli elementi finiti. Nell'ottica complessiva tuttavia, dove come visto in precedenza un migliaio di elementi di training portano ad una rete neurale tutto sommato poco accurata, è immediato calcolare che il tempo necessario per ricavare 1100 campioni comprensivi di training e test è di circa 58 ore. Tale risultato non tiene poi chiaramente conto delle tempistiche associate alle epoche di addestramento delle neural networks, altro fattore che è necessario considerare in fase di gestione organizzativa delle fasi di lavoro. Detto ciò, i campioni vengono ottenuti a batch di 250 elementi in modo tale da non vanificare intere analisi in caso di errori, e la variabilità ottenuta consente di abbracciare l'intero range dei valori di input, come evidente dai grafici in figura 49.

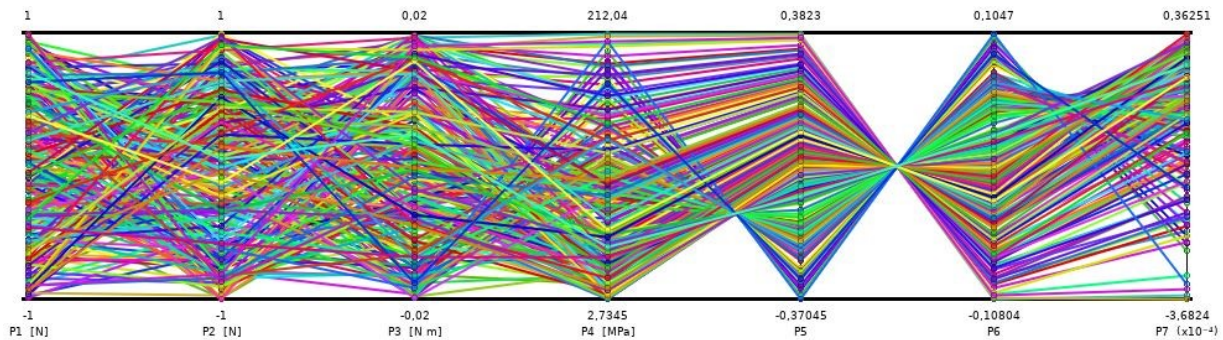


Figura 49 – Variabilità dei dati di input e output su una batch di 250 campioni.

Qui si può apprezzare, oltre alla grande quantità di design point raffigurati in differenti colori, come a rotazioni positive (in senso antiorario) siano associati spostamenti orizzontali negativi (verso sinistra) e dunque si abbia la classica forma “a clessidra” dell’insieme di curve; al contrario, le traslazioni verticali spaziano tra estremi di praticamente ugual entità ma con una maggiore concentrazione di campioni nel semiasse positivo. Per quanto riguarda l’influenza che ogni input ha invece sugli output di interesse, si può far riferimento all’immagine 50.

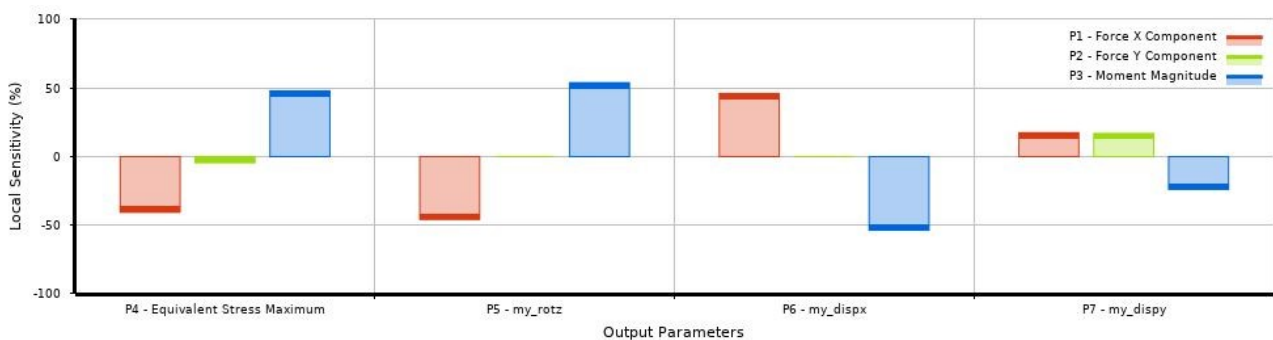


Figura 50 – Partecipazione dei tre valori di input in ciascuno degli output ottenuti.

È evidente come la componente di forza verticale risulti influente per le rotazioni, essendo questa applicata esattamente sulla verticale del pivot point e con braccio dunque nullo, stesso discorso vale per il moto lungo l’asse X dove non c’è partecipazione. Al contrario, la corrispettiva influenza dello sforzo orizzontale caratterizza lo spostamento verticale praticamente in egual misura rispetto alla  $F_y$ , tale asimmetria nel comportamento è dettata proprio dalla geometria del

giunto che risulta estremamente più cedevole lungo la direzione X. Questo grafico è infine estremamente utile per realizzare come in ogni output la maggior influenza sia comunque attribuibile al momento flettente agente, è ad esso che devono dunque essere rivolte le maggiori attenzioni in fase di progettazione.

Completato il ciclo di simulazioni che si pensano utili all'apprendimento della rete neurale, vengono importati all'interno di un file Excel tutti i dati di input e gli output ricavati ad essi associati, sarà questa la fonte alla quale andrà ad attingere la neural network secondo le medesime modalità riportate per il quadrilatero articolato. La difficoltà in questo caso è aumentata dalla notevole differenza negli ordini di grandezza dei parametri di output che si intende ottenere: agli stress che possono arrivare fino a centinaia di MPa sono infatti associate rotazioni di decimi di grado con spostamenti lungo X di centesimi di millimetro (o addirittura inferiori per quelle lungo l'asse Y). Verosimilmente sarà dunque necessario caratterizzare strutture neurali dedicate a ciascuno dei parametri in uscita.

### **6.3 Analisi dei risultati**

L'acciaio di riferimento per questo flexure joint è un AISI 316L al Silicio con bassa percentuale di carbonio (0.03%) e tensione di snervamento di 290 MPa, non ci si deve dunque stupire del basso valore degli angoli di rotazione dato che si sta lavorando con un modulo di Young di 200 GPa e con forze nell'ordine delle unità in dimensioni prossime al millimetro. A parità di sforzi e momenti esterni è chiaro che con componenti plastici si possano ottenere maggiori elongazioni rimanendo comunque in fase elastica (sviluppando però minori tensioni dato che bassi sarebbero gli stress sopportabili con queste tipologie di materiali), ma in questo caso si è preferito raggiungere valori più elevati di  $\sigma$  in modo tale da avere un range più ampio su cui far lavorare la rete neurale. Il

massimo valore di tensione equivalente di Von Mises registrato è quello rappresentato in figura 51, ottenuto con componenti unitarie positive di forza e momento flettente di  $-0.02\text{Nm}$ .

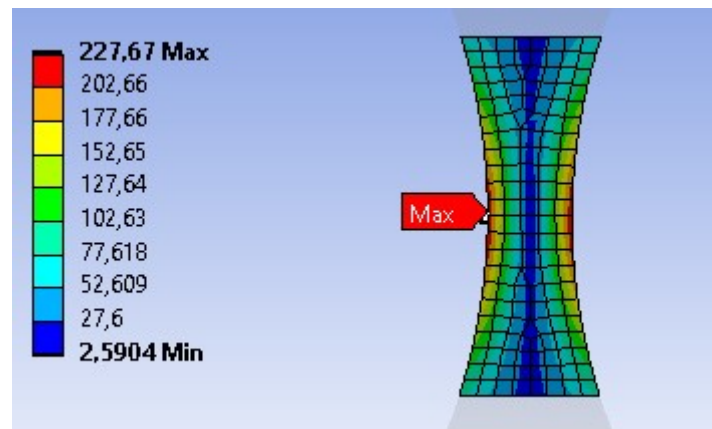


Figura 51 – Zoom della strizione dove si verifica il picco della tensione calcolata, in valore assoluto.

Come ipotizzato esso si trova in corrispondenza della strizione nei pressi dell'orizzontale al centro di istantanea rotazione, siamo molto lontani dalla plasticizzazione del materiale, il comportamento sarà quindi elastico e la relazione lineare. A seguito di questa considerazione si può quindi immaginare che la neural network, il cui codice è simile a quello presentato per il quadrilatero ad esclusione degli iperparametri e delle variabili di rete, si comporterà bene nella determinazione degli output nonostante il basso numero di campioni negli esempi di training.

Considerando il solo output di tensione, sono state eseguite varie prove al fine di determinare l'influenza di epoche, numero di campioni e numero di neuroni sulle previsioni in uscita dalla rete. Nell'iterare la procedura si è osservato come per una quantità più grande di campioni in input fossero necessarie meno epoche per giungere ad adeguati valori di loss, così per uniformare i risultati ottenuti si è scelto di impostare un massimale di trecento epoche di training e dove non fossero state necessarie si è lasciato intervenire l'early stopping. La funzione di attivazione utilizzata è stata anche in questo caso la Relu ed in tal modo si è riuscito a raggiungere un errore percentuale minimo dell'1,46% con venti neuroni nel primo strato e dieci nel secondo, per

ottenere ciò sono state sufficienti 154 epoche di addestramento. L'andamento dell'errore è quello rappresentato nel grafico in figura 52, come fatto in precedenza nelle analisi del quadrilatero.

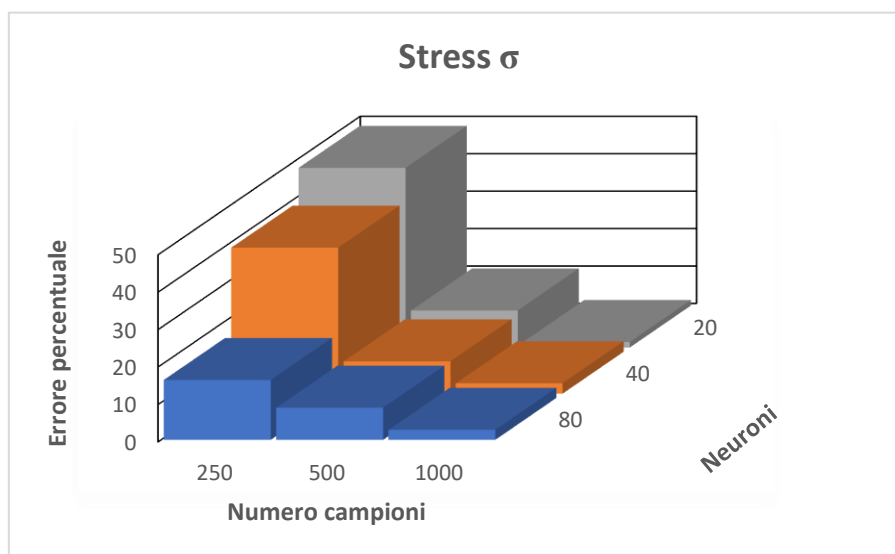


Figura 52 – Errore percentuale commesso dalla rete nella determinazione dello stress massimo in funzione del numero di campioni e di neuroni utilizzati.

In questo caso il lavoro di analisi è stato eseguito a ritroso rispetto alle volte precedenti, difatti ci si è trovati nelle migliori condizioni di lavoro della rete neurale in primissima battuta: il migliaio di correlazioni input-output si è rivelato il valore ottimo di training, al di sotto di esso si commettono errori abbastanza imponenti per la semplicità della funzione mentre al di sopra ci si troverà sicuramente in condizioni di overfitting. Infine va evidenziato come l'influenza del numero di neuroni sia estremamente marcata per pochi dati, mentre raggiunti i mille la rete riesce addirittura a lavorare meglio con meno unità computazionali.

Tutto un altro discorso è invece quello riguardante il secondo parametro in uscita, ovvero l'angolo di rotazione. In questo caso la complicazione risiede nell'indistinta positività o negatività del valore in uscita, ciò implica l'utilizzo di accortezze nell'uso della funzione ReLu (come l'utilizzo di appositi valori di soglia che portino la rete a lavorare su dati esclusivamente maggiori di zero) o l'eventuale percorrenza di strade lievemente differenti. La prima strada appare immediatamente difficile da



percorrere a causa di una divergenza dei risultati all'aumentare del numero di campioni di allenamento, non deve stupire che non sia in alcun modo garantita l'inversa proporzionalità tra numero di dati ed errore percentuale commesso. Per ovviare a tale situazione si è scelto di trascurare le soglie ed adoperare direttamente funzioni di attivazione che fossero in grado di generare output negativi, si è quindi utilizzata la "Softsign" per il primo strato di neuroni nascosti e l'arcotangente ("Tanh") per il secondo, la singola uscita viene poi generata riprendendo ancora una volta la prima funzione qui menzionata. La precisione delle previsioni rimane comunque abbastanza deludente con un errore medio dell'83%, così si è scelto di utilizzare altre 116 ore per la generazione di duemila (più altri duecento per l'insieme di test) campioni per verificare se con queste nuove caratteristiche di rete vi fossero correlazioni di proporzionalità.

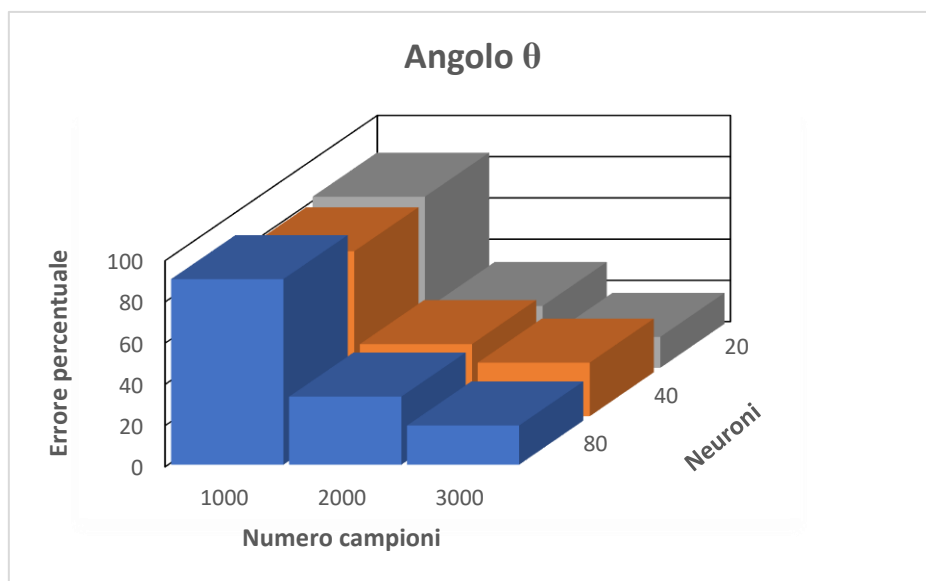
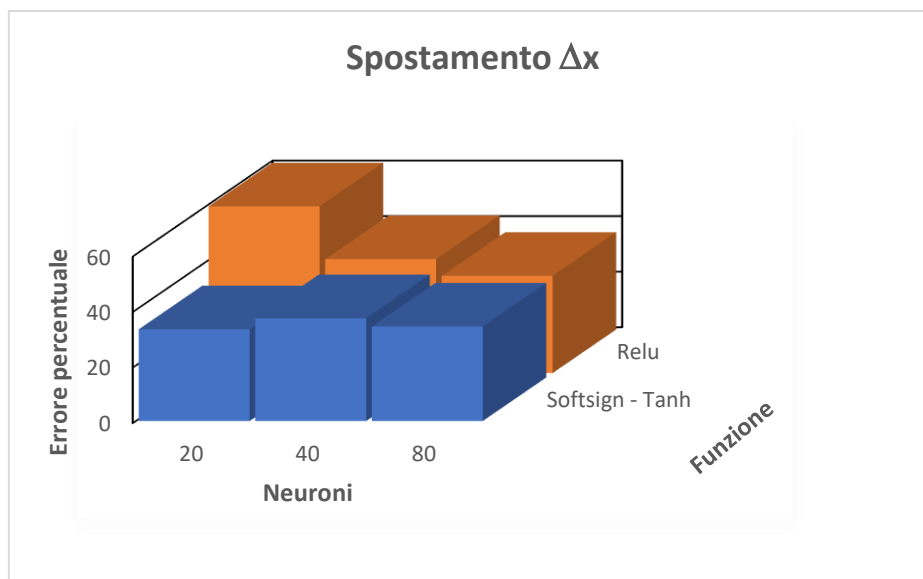


Figura 53 - Errore percentuale commesso dalla rete nella determinazione dell'angolo  $\vartheta$  in funzione del numero di campioni e di neuroni utilizzati.

I risultati ottenuti, mostrati nel grafico in figura 53, evidenziano come in questo caso l'allenamento su un maggior numero di esempi renda la rete più performante arrivando addirittura ad un errore del 15% con tremila campioni processati da venti neuroni di primo strato. Il miglioramento dell'accuratezza di rete con una minore quantità di unità di calcolo non è un'anomalia: basti

pensare al fatto che solitamente, nelle architetture neurali, gli input sono quantitativamente più grandi dei tre qui utilizzati quindi è come se la rete si trovasse ad affrontare complicazioni non necessarie, andando in overshooting. Si noti, inoltre, come lo sforzo per l'ottenimento degli insiemi di training sia di non poco conto da un punto di vista temporale e computazionale, ma ciò che è stato ottenuto dimostra che per questo parametro vale la pena di eseguire altri esempi e che quindi la relazione che intercorre tra forze applicate e rotazione, sebbene alquanto complicata, possa essere man mano ben approssimata.

La relazione che associa i tre input di carico esterno al terzo output di spostamento  $\Delta x$  sembra invece essere persino più complessa della precedente, presentando vantaggi e svantaggi in fase di implementazione della neural network che fosse in grado di stimarlo. Il vantaggio principale è quello della diretta proporzionalità tra numero di campioni e accuratezza delle previsioni, discorso che vale sia per funzioni di attivazione con output maggiori di zero (e successiva scalatura) che per quelle utilizzate per l'angolo  $\theta$ . Tuttavia, il limite evidenziato e riportato nelle colonne di figura 54 risiede nella lenta e difficoltosa convergenza ad un ottimo risultato.



*Figura 54 – Errore percentuale commesso (con 3000 campioni di training) dalla rete sullo spostamento  $\Delta x$  in funzione del numero di neuroni e della tipologia di funzione d'attivazione utilizzata.*

Per semplicità è riportata la sola casistica di riferimento ai tremila campioni, con un confronto tra le accuratezze ricavate utilizzando due diverse strategie implementative. I risultati finali sono quasi equivalenti con un errore percentuale minimo del 33% ricavato usufruendo di un primo hidden layer composto da 20 neuroni. Va comunque sottolineato come queste stime, simili per questo numero di dati in ingresso, siano significativamente diverse nelle loro progressioni: l'imprecisione media ottenuta con la funzione di attivazione ReLU scende da un massimo del 63% al 54% per terminare infine col 45% con l'utilizzo di mille, duemila e tremila campioni (errore minimo del 35% con ottanta neuroni di primo strato), mentre con l'uso di Softsign ed arcotangente si passa nel medesimo intervallo dall'81% al 56% ed infine al 38%. Volendoci ricollegare ai discorsi di gradiente di discesa negli ottimizzatori, viene dunque facile pensare che sul lungo periodo, inteso come maggiore quantitativo di campioni di training, quest'ultimo set di funzioni possa fornire buone approssimazioni della relazione ricercata.

Il quinto ed ultimo parametro, infine, rappresenta il tallone d'Achille della rete neurale. Lo spostamento  $\Delta y$  non consente infatti margine di manovra alla rete neurale a causa dell'ordine di

grandezza dell'output e della sua variabilità. Già dalla figura 49 era intuibile quanto complessa potesse essere la relazione che legasse la traslazione verticale ai carichi applicati vista la totale mancanza di simmetria e la non regolare distribuzione di campioni all'interno della semiretta negativa. Data l'entità dei valori in uscita, è stata effettuata una scalatura di quattro ordini di grandezza, in modo da avere un training che potesse esser chiamato tale con l'esclusivo utilizzo della funzione di attivazione "Softsign". Anche in questo caso si è notato un incremento dell'accuratezza all'aumentare degli elementi validi per l'allenamento, tuttavia il gradiente osservabile è estremamente orizzontale: si giunge ad un errore percentuale minimo del 71% su tremila dati elaborati con ottanta neuroni di hidden layer principale, a seguito di errori del 113% e 92% nei dataset con rispettivamente mille e duemila campioni.

## 7 Conclusioni

I meccanismi compliant stanno man mano espandendo il loro raggio d'azione ad una grande varietà di discipline tra loro differenti, grazie all'estrema duttilità realizzativa di questi componenti. Le stesse ottimizzazioni topologiche rappresentano una direzione sempre più percorsa nel mondo della progettazione meccanica, con guadagni tangibili sui fronti di economia e performance, dove è possibile implementare giunti flessibili che siano "cuciti su misura" in base all'applicazione necessaria, con forme e geometrie anche di notevole complessità. In queste condizioni, uno studio analitico delle relazioni che correlano materiale, dimensioni, forze trasmesse e spostamenti misurati non è sempre agevole; al contempo, le analisi agli elementi finiti rappresentano sicuramente una risposta attendibile con soluzioni che variano leggermente, però, da simulazione a simulazione e in base alla mesh utilizzata. In questo senso le reti neurali forniscono una valida alternativa alla determinazione di un pattern anche complesso, operando su una grande quantità di dati ricavati tramite FEM mediandone di fatto la variabilità. L'apprendimento dell'architettura neurale, tuttavia, può variare sensibilmente a seconda dell'output che si intende ottenere in uscita, non è dunque possibile sapere a priori il numero di campioni, la struttura funzionale della rete o il valore degli iperparametri che è più idoneo alle necessità del progetto. Da una parte, la mancanza di convergenza ad un risultato non può essere preventivata mentre dall'altra, è sempre bene affidarsi a chi è esperto di deep learning per evitare di incorrere nell'overfitting: la linea di demarcazione tra le due può comunque essere molto fine. In ogni caso, è il significativo tempo impiegato a fare da protagonista, da quello richiesto dalle numerose simulazioni per l'ottenimento dei dati a quello speso per l'apprendimento della rete: questo tempo non è prevedibile, meteorologi permettendo.



## **Bibliografia**

- 1) F. Pellicano, M. Callegari, P. Fanghella, *Meccanica applicata alle macchine*, CittàStudi edizioni, 2017.
- 2) S. Lenci, *Lezioni di meccanica strutturale*, Pitagora Editrice Bologna, 2015.
- 3) J. Paros, *How to design flexure hinges*, Mach Des, 1965.
- 4) Q. Liang, D. Zhang, Q. Song, Y. Ge, *Design and Optimization of Revolute Flexure Joints for Compliant Parallel Mechanisms*, *Sensors & Transducers Journal*, 2010.
- 5) M. Arredondo-Soto, E. Cuan-Urquizo, A. Gómez-Espinosa, *A Review on Tailoring Stiffness in Compliant Systems, via Removing Material: Cellular Materials and Topology Optimization*, *Applied Sciences*, 2021.
- 6) B. Deshmukh, S. Pardeshi, R. Mistry, S. Kandharkar, S. Wagh, *Development of a Four bar Compliant Mechanism using Pseudo Rigid Body Model (PRBM)*, *ScienceDirect*, 2014.
- 7) V. Megaro, J. Zehnder, M. Bacher, S. Coros, M. Gross, B. Thomaszewski, *A Computational Design Tool for Compliant Mechanisms*, *ACM Transactions on Graphics*, 2017.
- 8) A. E. Albanesi, V. D. Fachinotti, M. A. Pucheta, A. Cardona, *Synthesis of compliant mechanisms for segment-motion generation tasks*, *CIMEC*.
- 9) Y. Hu, B. Ju, W. Zhu, *Integrated Development of a Topology-Optimized Compliant Mechanism for Precise Positioning*, 2022.
- 10) N. Lobontiu, *Compliant Mechanisms: Design of Flexure Hinges*, CRC Press, 2002.
- 11) Stuart T. Smith, *Flexures Elements of Elastic Mechanisms*, CRC Press, 2000.
- 12) L. L. Howell, *Compliant Mechanisms*, John Wiley & Sons Inc, 2001.
- 13) G. Chen, J. Jia, Z. Li, *On Hybrid Flexure Hinges*, 2005

- 14) F. Parvari Rad, R. Vertechya, G. Berselli, V. Parenti-Castelli, Analytical compliance analysis and finite element verification of spherical flexure hinges for spatial compliant mechanisms, Elsevier, 2016.
- 15) Fred S. Eastman, The Design of Flexure Pivots, University of Washington, 1937
- 16) L. L. Howell, A. Midha, A Method for the Design of Compliant Mechanisms With Small-Length Flexural Pivots, ASME, 1994.
- 17) C. Ling, On Stress-Concentration Factor in a Notched Strip, Journal of Applied Mechanics, 1968.
- 18) S. X. Yanga, M. Meng, An efficient neural network approach to dynamic robot motion planning, Pergamon, 2000.
- 19) M. Nielsen, Neural Networks and Deep Learning, 2018.
- 20) Kevin Gurney, An introduction to neural networks, UCL Press, 1997.
- 21) S. Haykin, Neural Networks and Learning Machines, Pearson, 2008.
- 22) J. C. Principe, N. R. Euliano, W. C. Lefebvre, Neural and Adaptive Systems: Fundamentals Through Simulation: Chapter III, John Wiley & Sons, 1999.
- 23) H. Ramchoun, M. A. Janati Idrissi, Y. Ghanou, M. Ettaouil, Multilayer Perceptron: Architecture Optimization and Training, International Journal of Interactive Multimedia and Artificial Intelligence, Vol. 4, N°1.
- 24) A. Hyvarinen, E. Bingham, Connection between multilayer perceptrons and regression using independent component analysis, Elsevier, 2003.
- 25) O. O. Olatunji, S. Akinlabi, N. Madushele, P. A. Adedeji, I. Felix, Multilayer perceptron artificial neural network for the prediction of heating value of municipal solid waste, Aimspress, 2019.



*A Biba*