

UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN  
INGEGNERIA BIOMEDICA

**Mano artificiale: sviluppo e  
programmazione in applicazione  
biomedicale**

*Relatore:*

PROF. ROBERTO LOT

*Laureando:*

FRANCESCO VARNIER

2010258

Anno Accademico 2022/2023



## **Abstract**

Le mani umane sono strumenti sofisticati, importanti per interagire con la realtà esterna ma ancora poco conosciute dal punto di vista ingegneristico. Per comprendere al meglio il loro funzionamento e per realizzare una loro replica, si è deciso di analizzare alcuni modelli presenti all'interno della comunità scientifica per catturarne le caratteristiche salienti. Infine, si è costruito un prototipo funzionante di mano robotica stampato in 3D che è stato sottoposto a test per verificarne le potenzialità.



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Diffusione delle mani artificiali . . . . .	2
1.2	Requisiti qualitativi . . . . .	3
1.3	Problema dell'efficienza energetica e soluzioni proposte . . . . .	6
1.4	Soft Materials . . . . .	8
<b>2</b>	<b>Stato dell'arte</b>	<b>11</b>
2.1	Analisi dei modelli e stato dell'arte attuale . . . . .	11
2.1.1	Mano robotica - DRL . . . . .	12
2.1.2	Mano robotica - InMoov . . . . .	13
2.1.3	Mano robotica - SoftHand Pro-H . . . . .	15
2.1.4	Mano robotica - Biomimetic robotic hand . . . . .	17
2.2	Tecnologie sperimentali in fase di sviluppo . . . . .	18
2.2.1	Protesi a controllo mentale . . . . .	18
2.3	Risultati dell'analisi dei modelli di stato dell'arte . . . . .	19
<b>3</b>	<b>Considerazioni preliminari e scelte stilistiche</b>	<b>21</b>
3.1	Materiale e stampaggio . . . . .	22
3.2	Servo Motore . . . . .	23
3.2.1	Integrazione con Arduino . . . . .	23
3.2.2	Caratteristiche di bloccaggio . . . . .	23
3.3	Componenti non inclusi . . . . .	24
<b>4</b>	<b>Progettazione della protesi</b>	<b>27</b>
4.1	Progettazione del pollice . . . . .	27
4.2	Posizionamento motori . . . . .	28
4.3	Collegamento con filo . . . . .	30
4.4	Meccanica del dito . . . . .	31
4.5	Meccanica del polso . . . . .	34

---

4.6	Collegamento ad Arduino . . . . .	36
<b>5</b>	<b>Programmazione dei movimenti</b>	<b>39</b>
5.1	Le funzioni di movimento . . . . .	40
5.1.1	Setup . . . . .	40
5.1.2	Movimento Base . . . . .	41
5.1.3	Angolo di sfasamento - theta . . . . .	42
5.1.4	moveToOrigin . . . . .	43
5.2	I movimenti predefiniti . . . . .	44
5.2.1	moveStandard . . . . .	44
5.2.2	Movimento sincrono . . . . .	45
5.2.3	Afferrare . . . . .	47
<b>6</b>	<b>Test e risultati</b>	<b>49</b>
6.1	Presa pallina da Padel . . . . .	51
6.2	Presa bicchiere in vetro . . . . .	52
6.3	Presa telecomando . . . . .	53
6.4	Conclusioni . . . . .	54
<b>7</b>	<b>Appendice</b>	<b>57</b>
7.1	Schema elettrico - Arduino . . . . .	58
7.2	Scheda tecnica - Servo Motore . . . . .	59
7.3	Meccanismo motori . . . . .	60
7.3.1	Visione d'insieme laterale - blocco motore . . . . .	60
7.3.2	Visione d'insieme superiore - blocco motore . . . . .	61
7.4	Codice MatLab - moveStandard . . . . .	61
7.5	Codice MatLab - moveHand . . . . .	63
7.6	Codice MatLab - grab . . . . .	65
	<b>Bibliografia</b>	<b>69</b>

# Capitolo 1

## Introduzione

Nell'attuale panorama e contesto sociale si può notare la tendenza all'incremento della diffusione dei robot e all'intensificarsi delle loro interazioni con l'ambiente. In particolare questo fenomeno può essere inquadrato in un processo di trasformazione più ampio in cui questi assumono un ruolo sempre più importante nelle diverse aree della società. Inoltre significativo è anche il miglioramento delle capacità dei robot di interagire con l'ambiente circostante. Infatti, la loro affermazione all'interno delle industrie è accompagnata da un incremento del loro utilizzo anche in altri contesti, fuori dagli ambienti controllati, come nelle operazioni di soccorso, nelle esplorazioni oceaniche oppure in spazi casalinghi e aperti al pubblico. In tutti questi casi, perciò, è di fondamentale importanza garantire la sicurezza dell'interazione tra uomo e macchina, prestando particolare attenzione all'*end-effector*, la parte del meccanismo che sostanzialmente interagisce con l'ambiente. Tra i vari robot, quelli che spesso svolgono l'interazione fisica sul mondo sono le pinze o le mani robotiche, in grado soprattutto di afferrare, ruotare o spostare un oggetto.

In questo progetto, è stato preso in considerazione il fattore economico, in termini di materiali, consumi, riusabilità e dispendio energetico. In secondo luogo, sono stati valutati fattori quali la facilità di integrazione con sensori, le possibilità di possibili sviluppi futuri e la realizzabilità mediante le risorse a disposizione. Infine, è stata posta un'ultima considerazione all'aspetto estetico del design.

Presi in considerazione tali fattori, segue la motorizzazione della mano e in seguito il controllo di essa, mediante codice. Nella fase di ricerca, si è posta la focalizzazione sulla parte di sviluppo del movimento e programmazione software, piuttosto che sulla progettazione meccanica della mano, nonostante siano stati necessari degli accorgimenti obbligatori per ottenere il prodotto desiderato.

## 1.1 Diffusione delle mani artificiali

La capacità dell'uomo di usare le mani per interagire con il mondo circostante è la caratteristica peculiare che ne ha permesso l'evoluzione e il dominio sulle altre specie animali sulla terra. In particolare lo sviluppo del pollice opponibile e la conseguente capacità di afferrare e utilizzare oggetti come utensili o come armi permisero ad esempio di scagliare pietre con precisione contro un bersaglio o di scalfire due rocce, l'una contro l'altra, in modo da poter creare nuovi strumenti, come il Chopper o l'Amigdala. La grande importanza e la versatilità delle mani umane spinsero scienziati e inventori a studiarne l'anatomia e il funzionamento proprio per cercare di replicarle artificialmente nella versione, appunto, di "mano robotiche". Tuttavia, sebbene all'apparenza le mani umane possano sembrare semplici, la loro replica e la loro implementazione nell'ambiente esterno sono complicate e rappresentano una sfida tutt'oggi ancora aperta, volta a far sì che queste interagiscano efficacemente con un mondo progettato e disegnato a "misura d'uomo".

L'enorme varietà di azioni che possono svolgere le mani umane, permettono quindi di realizzare delle tecnologie che operino in vari ambiti; in particolare i principali settori in cui quest'ultime trovano impiego sono:

1. Esplorazione e manutenzione spaziale
2. Esplorazione ed interazione delle profondità oceaniche
3. Soccorso nei luoghi di calamità naturali
4. Supporto all'interno dell'industria e del commercio
5. Supporto e sostituzione in ambito protesico e biomedicale
6. Servizio e supporto in ambiente domestico e pubblico

Tra queste, le mani robotiche più sviluppate e consolidate sono quelle impiegate come protesi in ambito biomedicale e possono essere classificate in tre tipologie:

a. **Cosmetiche.**

Svolgono un ruolo puramente estetico e sono prive di funzionalità e fisse, non dinamiche.



**b. A energia corporea**

Funzionano mediante la forza del paziente; collegando un sistema di cavi e di molle ad una parte del corpo, consente il movimento del meccanismo in risposta ad una contrazione muscolare.

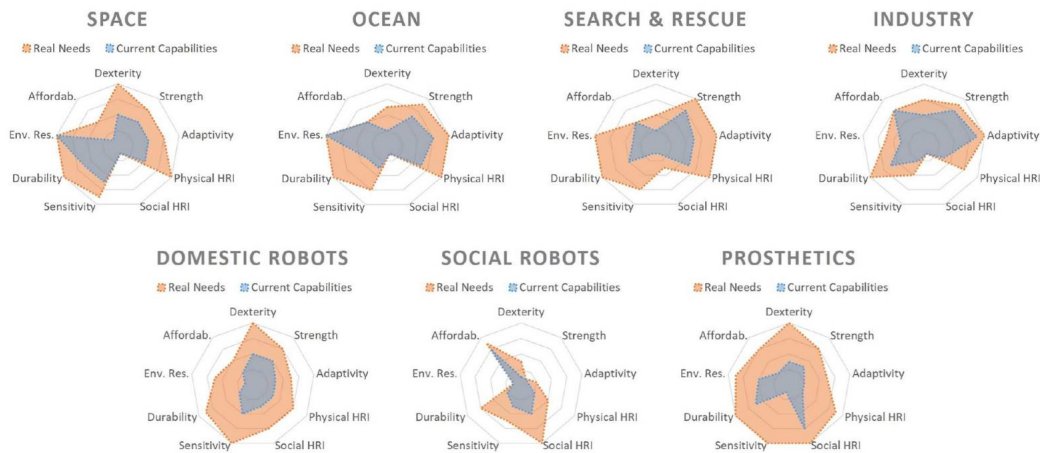
**c. Mioelettriche**

Funzionanti mediante l'utilizzo di un sensore mioelettrico, il quale accoglie lo stimolo muscolare, lo analizza e lo trasforma in moto da trasmettere alla protesi.

Tuttavia, spesso le mani robotiche implementate con questo metodo dispongono di un solo grado di libertà, cosa che le rende di fatto delle pinze piuttosto che delle vere e proprie mani. Aggiungere libertà di movimento, però, significa sia complicare la meccanica della protesi e la comprensione del funzionamento della macchina da parte del paziente, sia incrementare il costo complessivo. Inoltre, spesso rendere più sofisticati questi dispositivi, significa anche renderli più fragili e anche più pesanti, caratteristiche che allontanano dall'ottimale interazione tra uomo e macchina ideale. La mano robotica diventa di fatto ancora insostituibile alla sua controparte umana. Ad aumentare questo divario contribuiscono il design non camuffato delle protesi, lontano da qualsiasi parvenza di aspetto umano, la durabilità e le funzionalità limitate e il costo poco accessibile.

## 1.2 Requisiti qualitativi

L'evoluzione delle mani robotiche rappresenta una sfida in corso nell'ambito di numerosi settori e presenta complessità sia dal punto di vista tecnologico che da quello pratico e sostenibile. Questo processo di sviluppo è caratterizzato da diverse sfide che richiedono attenzione e risoluzione verso le principali caratteristiche, quali libertà di movimento, sensibilità, forza e precisione, adattabilità, affidabilità, durabilità, resistenza all'ambiente, accessibilità economica e interazione sociale. La situazione attuale è riassunta in figura [1.1], dove in base all'opinione personale di vari ricercatori, è stato assegnato un punteggio da 1 a 5 sullo stato attuale della tecnologia dei vari settori di interesse, tenendo in considerazione le varie mani robotiche attualmente ben consolidate all'interno dell'industria e nel commercio. Successivamente è stato assegnato un punteggio anche per le caratteristiche che dovrebbero assumere le tecnologie per svolgere al meglio il proprio compito nei vari ambiti [5].



**Figura 1.1:** Rappresentazione dello sviluppo tecnologico nei diversi settori di interesse. In blu, lo stato attuale della tecnologia; in arancione, le qualità necessarie idealmente richieste.

Verranno ora analizzate le necessità che una mano robotica deve soddisfare, le sue caratteristiche e i fattori da tenere in conto in base al compito da svolgere e il contesto nel quale sono collocate.

### Movimento diversificato

La mano, innanzitutto, deve essere in grado di eseguire una varietà di movimenti e gesti per adattarsi a diverse situazioni. Maggiori sono i movimenti disponibili, maggiore è la precisione con la quale un certo gesto o movimento può essere compiuto, assumendo un maggior numero di configurazioni per un maggior numero di situazioni.

### Sensibilità tattile

La mano deve essere poi dotata di sensori tattili per rilevare la forma, la consistenza e la temperatura degli oggetti. Un maggior numero di sensori implica un maggior controllo su oggetti di dimensioni minute, così da permettere di operare su dimensioni sempre più ridotte. Parte della sensibilità deriva anche dai movimenti e dalla progettazione meccanica adeguata, che deve consentire spostamenti minimi, in risposta a input di segnale minimi, all'interno del *range* di interesse.

### Forza e precisione

La mano deve essere in grado di applicare una forza coerente ed adeguata per afferrare e manipolare oggetti di diverse dimensioni e forme. I movimenti devono essere in grado di poter fornire una coppia minima per oggetti fragili e leggeri, ma

allo stesso tempo devono poter manipolare in sicurezza anche oggetti più pesanti, applicando coppie maggiori, e riuscire a garantire una presa efficace per tutta la durata della manipolazione, cosa che risulta fondamentale per la sicurezza. Questo ruolo può essere ricoperto dagli attuatori stessi, ma è più conveniente adoperare meccanismi di sicurezza esterni, che impediscano il moto retrogrado sollecitato da fonti esterne [3]. Si rimanda alla sezione [1.3] per una visione più ampia di questi meccanismi.

### **Adattabilità**

La mano deve essere in grado di adattarsi alle diverse forme e dimensioni degli oggetti, nonché alle variazioni ambientali. L'adattabilità rappresenta la capacità di una mano robotica di poter operare in situazioni differenti, con oggetti differenti e talvolta in ambienti differenti, per riuscire quindi a svolgere una vasta gamma di movimenti in base a ciò con cui interagisce. Si rimanda alla sezione [1.4] per ulteriori spiegazioni.

### **Affidabilità**

Un'altra caratteristica da considerare durante la prototipazione di una mano artificiale, è la capacità di essere affidabile e resistente all'usura, sia in termini di materiale, sia dal punto di vista meccanico. L'impiego di materiali in grado di sopportare non solo grossi sforzi meccanici e ripetuti nel tempo, ma anche abrasioni e urti esterni, è fondamentale per garantire sicurezza all'utilizzatore. Altrettanto importanti sono i dispositivi di bloccaggio per impedire moto retrogrado attuato dall'esterno, in grado di bloccare il movimento della mano anche in condizione di motori spenti.

### **Controllo intuitivo**

L'interfaccia utente deve essere facile da usare e intuitiva, consentendo agli utenti di controllare e imparare ad operare in tempi ragionevolmente brevi la mano robotica.

### **Compatibilità con l'ambiente circostante**

Importante è anche impedire alla mano robotica di infliggere danni involontari e indesiderati a cose, animali o persone con cui entra in contatto, e garantire, invece, che essa operi in modo sicuro nell'ambiente circostante.

## 1.3 Problema dell'efficienza energetica e soluzioni proposte

Come accennato nel paragrafo precedente, il fattore sicurezza, poter fare affidamento sulla mano robotica anche in situazioni straordinarie quali l'assenza di corrente elettrica, non deve costituire un problema o un rischio per il suo utilizzatore. Infatti, se in assenza di alimentazione i motori non riescono più a fornire coppia per mantenere l'equilibrio tra forze esterne e quelle interne, potrebbe avvenire il rilascio di oggetti fragili oppure pesanti con conseguente danneggiamento dell'oggetto stesso e, in caso peggiore, apportando un danno anche all'utente.

### Meccanismo diretto - Vite

Tra le varie possibili soluzioni sono presenti proposte energeticamente inefficaci, come le viti che accoppiate con la madre vite, rendono di fatto un movimento retrogrado impossibile a una forza esterna diretta. Dunque è necessario fornire una coppia esterna continua che imponga un moto di svitamento, opposto a quello imposto dal motore. Tuttavia, la presenza di una vite che trasmette il moto del dito non è efficiente, poiché il fenomeno su cui si basa il bloccaggio è basato sull'attrito [15] dissipando in continuo parte dell'energia meccanica del motore in calore.

### Meccanismo intrinseco - Riduttori

Nel caso di attuazione mediante motori, spesso è presente un meccanismo di riduzione che consente di aumentare la coppia erogata da piccoli ma veloci motori. Le dita, collegate in maniera diretta o indiretta a questi attuatori, sia vengono mosse dagli stimoli interni, ma anche trasmettono il moto proveniente da stimoli esterni. Grazie alla riduzione mediante ruote dentate, la coppia della sollecitazione esterna viene ridotta, di fatto dotando di una certa resistenza anche in condizioni passive la mano robotica. Tuttavia, è fondamentale che i pezzi componenti la mano risultino leggeri e compatti per permettere un'integrazione ottimale tra essi. Ciò significa che anche i motori che producono il moto saranno piccoli e quindi fragili, diminuendo di fatto la soglia massima di sollecitazione esterna alla quale la mano può essere sottoposta, rendendola quindi meno sicura in caso di mancanza di corrente [3]

### Non-Backdrivable Actuator Transmissions

Questa soluzione, proposta da Montagnani et al. [3], prevede l'uso di un meccanismo esterno accoppiato al motore, detto *non-backdrivable actuator transmissions* atto a bloccare l'azione indesiderata della forza esterna sul movente principale. Il meccanismo è composto sia da elementi rigidi, che elementi elastici, come molle, che permettono il moto in entrambe le direzioni, se attuato dal verso input e lo impediscono (sotto una certa soglia) se fornito dal verso output.

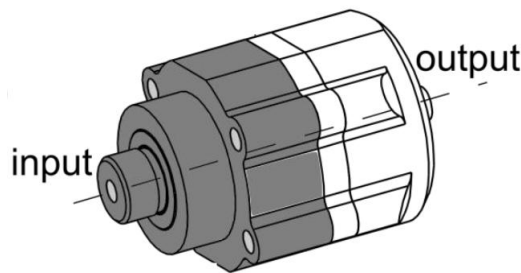
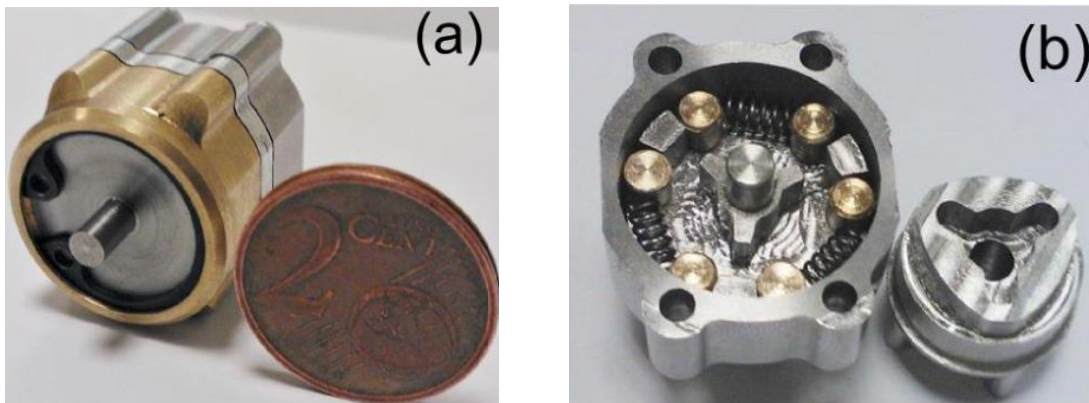


Figura 1.2: Schema di funzionamento - Montagnani et al., Non-Back-Drivable Mechanism [3]

Le dimensioni di questo dispositivo sono miniaturizzate, comparabili a una moneta da 2 cent, occupando nel complesso un volume di circa  $5500 \text{ mm}^3$  e pesando in totale  $32 \text{ g}$ , rendendolo ideale per un utilizzo in questo ambito dove lo spazio e la massa complessiva dell'oggetto finale sono di fondamentale importanza. La coppia esterna massima che il sistema può sopportare è di  $1.2 \text{ N} \cdot \text{m}$  (carico critico) con un'energia assorbita di  $1.5 \text{ J}$ , in grado di proteggere sia il motore, fragile e facilmente danneggiabile se soggetto a sollecitazioni eccessive, sia di garantire un range di sicurezza in caso di cedimenti interni del meccanismo principali o in assenza di corrente.

Di rilevanza sono le migliori riguardanti le prestazioni energetiche che questi dispositivi sono in grado di apportare. Infatti, non solo è possibile fare uso del *Non-backdrivable Mechanism* in caso di emergenza, ma al contrario, può essere usato anche per limitare l'energia utilizzata dagli attuatori per mantenere l'equilibrio, garantendo un risparmio di energia e una maggior durata del dispositivo, non solo dal punto di vista della batteria ma anche in termini di vita complessiva del motore. Infatti, durante i periodi di staticità, il motore si ritrova in posizione di riposo, evitando stati di stress ripetuti dovuti all'isometria ed essendo spento anche il consumo di energia risulta minimo.

In conclusione, seppur essendo un elemento esterno che comporta quindi un ingombro, anche se minimo con un conseguente aumento di peso e una diminuzione dello spazio disponibile per inserire all'interno della mano altri dispositivi di



**Figura 1.3:** Sulla sinistra figura (a), il meccanismo assemblato e paragonato alle dimensioni di un cent. Sulla destra figura (b), il meccanismo Non-Back-Drivable aperto - Montagnani et al. [3]

controllo, i vantaggi sono superiori agli svantaggi in quanto aumenta la durabilità totale della vita del motore, aumenta la sicurezza, garantendo una presa stretta in condizioni di assenza energetica.

## 1.4 Soft Materials

La capacità di adattarsi è ottenuta mediante l'utilizzo di arti progettati per essere non solo fisicamente robusti e resistenti, ma anche flessibili e sensibili, così da poter interagire con il mondo circostante in modo simile a quello umano. L'utilizzo di materiali a rigidità variabile oppure materiali morbidi ("soft materials") permettono la realizzazione di dita adattabili, in grado di regolare la presa in base all'oggetto con cui devono interagire [11]. Inoltre, i materiali morbidi, rispetto ai materiali rigidi tradizionali che tuttavia sono affermati ed affidabili nell'industria e negli ambienti controllati, dimostrano proprietà simili a quelle dei tessuti viventi, poiché sono generalmente composti da polimeri organici che possono essere progettati per avere proprietà meccaniche specifiche come l'elasticità o la flessibilità. La loro struttura può presentarsi porosa o a celle aperte, dovuta in quanto sono presenti spazi vuoti all'interno del materiale, creati durante il processo di produzione, che conferiscono le proprietà di adattarsi a superfici irregolari e di assorbire efficacemente gli urti. [12]

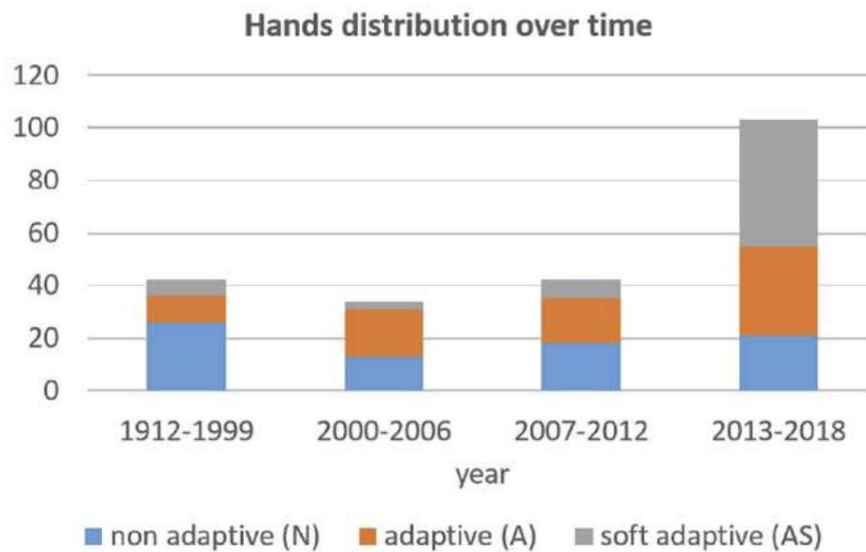
La realizzazione di "soft robotic parts" è possibile mediante l'impiego, nel contesto delle mani robotiche, di dita morbide con rigidità articolare regolabile. Utilizzando due tipi diversi di silicone e combinando attuazione a tendine e quella pneumatica, è possibile modificare sia la forma, che la rigidità articolare del dito, tramite la modifica della pressione dell'aria al suo interno [11].

L'utilizzo della tecnologia a materiali morbidi permette di creare geometrie complesse, non ottenibili mediante la robotica tradizionale così da ricreare arti peculiari, come lo sono i tentacoli. La piovra, essendo un invertebrato, non dispone di uno scheletro interno rigido, non facilmente ricreabile mediante l'uso di materiali rigidi. Un modello di tentacolo ispirato alla piovra è stato proposto dai ricercatori Laschi et al. nel 2012 [13], utilizzando attuatori a memoria di forma (SMA), i quali possono cambiare forma in risposta a stimoli esterni come calore o elettricità. In questo caso, gli SMA sono usati per contrarre la rete polimerica, in modo da cambiare la forma stessa del tentacolo.



**Figura 1.4:** Esempio di tentacolo realizzato con materiali morbidi

Il tentacolo fornisce sia capacità di presa, grazie alla sua rigidità variabile, sia capacità di movimento, data dai suoi innumerevoli gradi di libertà, conseguenza della sua struttura di corpo non-rigido. Si vuole notare che questi tipi di robot con estremità morbide, capaci di svolgere diverse funzioni, sono preferibili e sempre più utilizzati nel settore.



**Figura 1.5:** Andamento nel corso degli anni dell'utilizzo di materiali morbidi nello sviluppo di mani robotiche. (Dati estratti da Piazza et al. [14], grafico di Negrello et al. [5])

Come evidenziato nella pubblicazione [5], "Le tendenze recenti verso progettazioni morbide, multi-materiali e sensibili mostrano una crescente importanza quando i ricercatori lavorano su robot per compiti meno strutturati." Tuttavia, la complessità dello sviluppo di un robot tramite *soft-materials* impedisce la facile realizzazione in ambienti di lavoro non specializzati, se confrontata con la semplicità di realizzazione di un robot a materiali rigidi, i cui componenti possono essere reperiti con facilità.



# Capitolo 2

## Stato dell'arte

Nel contesto della ricerca e dello sviluppo delle mano robotiche, il presente capitolo si propone di esplorare lo stato dell'arte attuale di questa area di studio. L'importanza delle mano robotiche risiede nella loro capacità di emulare la complessità e la versatilità delle mani umane, aprendo nuovi orizzonti nell'ambito della robotica avanzata, dell'automazione industriale e delle applicazioni biomediche, ma anche in campi in via di sviluppo quali robotica domestica oppure l'utilizzo in ambienti come l'oceano, lo spazio, la ricerca e il salvataggio in luoghi ostili. Valutando i progressi compiuti finora, l'attenzione verrà riposta nella valutazione dei risultati ottenuti, gli obiettivi da raggiungere e nelle opportunità emergenti.

### 2.1 Analisi dei modelli e stato dell'arte attuale

La sezione inizia esaminando le innovazioni tecnologiche che hanno consentito l'avanzamento delle mano robotiche. Questo include lo studio dei materiali ad alte prestazioni, dei sistemi di attuazione avanzati, dei sensori di feedback e delle interfacce uomo-macchina. Successivamente si esploreranno le ultime scoperte e gli sviluppi promettenti che hanno contribuito a superare le limitazioni tecniche e a migliorare l'efficienza, la precisione e la sicurezza delle mani robotiche. Un aspetto fondamentale delle mano robotiche è la loro capacità di manipolazione, di afferrare oggetti con precisione e di eseguire movimenti complessi. Le mano robotiche si trovano poste di fronte a diverse sfide tecniche, come l'interazione con oggetti deformabili, l'adattamento a diverse forme e dimensioni, e la comprensione dei feedback sensoriali. Grazie a una panoramica dello stato dell'arte delle mano

robotiche, risulterà possibile valutare il livello di successo raggiunto e identificare le aree che richiedono ulteriori miglioramenti.

### 2.1.1 Mano robotica - DRL

Il primo esempio di stato dell'arte proviene dal "German Aerospace Research Center" (DLR) dell'Istituto di Robotica e Meccatronica, viene proposto una mano robotica a 13 DOF ("degree of freedom", gradi di libertà) controllata mediante feedback, da dei sensori [7].



Figura 2.1: Mano - DRL

La mano consiste di un design minimale, dove vengono implementate 3 dita e un pollice al fine di ridurre la complessità del meccanismo a livello implementativo e garantire un peso totale minore. Inoltre, il movimento del primo e del quarto dito è attuato da un solo motore a spazzola, utilizzando un ingranaggio per ripartire il movimento, il che significa che è necessario meno cablaggio per controllare i movimenti delle dita. Gli attuatori consistono in motori DC, ruote dentate e cinghie di trasmissione per portare il moto fino ai polpastrelli delle dita. La scelta di posizionare i motori direttamente all'interno del palmo della mano, deriva dal fatto che, con l'implementazione di un avambraccio, il prototipo nel suo complesso risulta più ingombrante; le dimensioni ridotte permettono un utilizzo accoppiato con un braccio industriale per aumentare ulteriormente l'area di lavoro, cosa più difficoltosa se invece si fosse sviluppata una mano con avambraccio dato il suo maggiore ingombro.

Per quanto riguarda il controllo, la mano è dotata dei sensori:

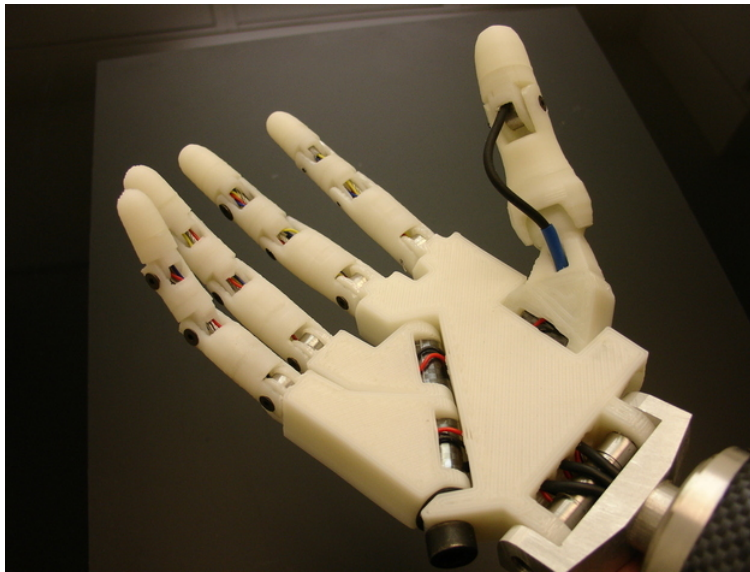
1. **Potenziometro**, basato su delle plastiche conduttive, fornisce l'angolo raggiunto dalle dita.
2. **Sensore tattile**, integrato sulle dita, permette di rilevare quando l'oggetto entra in contatto con la mano.
3. **Sensori di forza**, permettono di registrare le coppie e le forze applicate durante la presa e il contatto con un oggetto, usando i principi di un sensore piezoelettrico.



Figura 2.2: Dettaglio catena cinematica del dito, Mano - DRL

### 2.1.2 Mano robotica - InMoov

Nel campo dello sviluppo di dispositivi "artistici", è stato raggiunto un grado di avanzamento tecnologico grazie al progetto InMoov [9]. La mano robotica InMoov, caratterizzata da versatilità e realismo, si è affermata come una soluzione rilevante nell'ambito della robotica bionica, grazie all'approccio OpenSource: il design della mano robotica, infatti è reso liberamente disponibile al pubblico. Inoltre, risulta più accessibile dal punto di vista economico, poiché i costi di sviluppo e produzione possono essere condivisi tra una comunità più ampia, permettendo di accedere a tecnologie avanzate pur utilizzando risorse limitate. Pur nascendo come opera a fini artistici, molti sviluppatori hanno modificato InMoov, in modo da poter ampliare le sue funzioni ed essere utilizzato come punto di partenza di progetti più complessi, come i programmi di intelligenza artificiale o sviluppo di sensori e metodologie alternative di controllo della mano.



**Figura 2.3:** Mano - InMoov

Il design della mano ha premiato la fedeltà alla realtà, essendo dotata di 5 dita contrattili, che muovono grazie a dei motori esterni posizionati all'interno dell'avambraccio, a confronto della mano DRL che preferiva maggiore funzionalità a discapito della forma. Tuttavia, come già analizzato, la collocazione esterna degli attuatori influisce sul peso della mano isolata, che risulterà inferiore, a discapito della manovrabilità dell'insieme, che risulterà peggiore [7]. La mano è attuata mediante dei servo motori, puleggie, cinghie di trasmissione (fili) che permettono di trasmettere il movimento dall'avambraccio fino ai polpastrelli.

Per quanto riguarda l'automazione, esistono diversi metodi per consentire il controllo dei movimenti della mano, mediante sensori; tra questi:

1. **Sensore a ultrasuoni**, per misurare la distanza del braccio da un oggetto ed evitare collisioni o monitorare la velocità di approccio.
2. **Sensore di pressione**, per applicare delle forze differenti in contesti in cui è richiesta l'applicazione di pressioni differenti.
3. **Sensore mioelettrico**, per trasmettere l'input di comando, in questo caso la contrazione muscolare, al dispositivo di controllo degli attuatori.

Un esempio di sviluppo del modello InMoov si basa sulla progettazione di una mano antropomorfa a basso costo realizzata e stampata in 3D [10], strutturata specificatamente per la riproduzione della lingua dei segni. La necessità di compiere movimenti complessi, ha comportato l'aumento della capacità di movimento

della mano mediante l'introduzione di sei nuovi gradi di libertà: abduzione/adduzione per il pollice, l'indice e il medio, e una giuntura sferica parallela a tre gradi di libertà per il polso. Un miglioramento è stato effettuato alla meccanica complessiva della mano, come la sostituzione delle corde per la trasmissione del moto con dei fili di nylon per ridurre l'attrito e fornire maggiore flessibilità nella posizione degli attuatori.



**Figura 2.4:** Configurazioni assunte dalla mano - Linguaggio dei segni

### 2.1.3 Mano robotica - SoftHand Pro-H

Nel 2017 in occasione della competizione Cybathlon, l'Università di Pisa e l'istituto italiano della Tecnologia di Genova mettono a punto un modello di mano robotica che funziona combinando le tecnologie utilizzate nelle protesi alimentate a Potenza Corporea e quelle che usano sofisticati sensori mioelettrici, funzionanti con l'elettricità [17].

Il progetto prevede l'uso di un unico motore per controllare 19 gradi di libertà, risultando quindi fortemente sottoazionata. La funzionalità è garantita dall'utilizzo di *soft materials* in corrispondenza dei giunti e legamenti elastici che aggiungono alla mano proprietà e funzionalità difficilmente raggiungibili con il solo impiego di materiali rigidi. Di conseguenza, ne risulta migliorata la sicurezza, perché è garantito il ritorno in posizione originale delle dita in caso di dislocazione, ed è anche di fondamentale importanza per permettere una presa dinamica, in modo da poter maneggiare con un unico motore una varietà di oggetti di forma diversa.

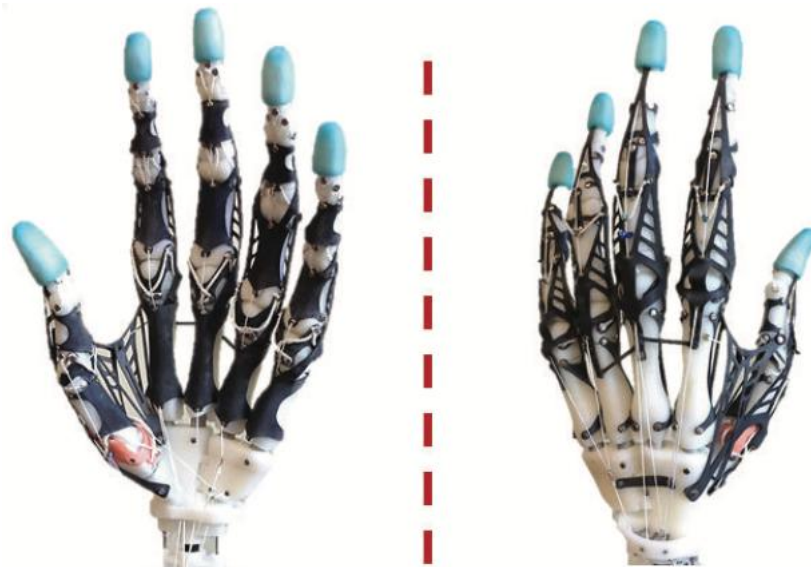


**Figura 2.5:** Configurazioni assunte e oggetti afferrabili dalla mano - immagine presa da [17]

L'attuazione a potenza corporea è possibile grazie a un cavo collegato alla spalla, il quale aziona una leva che causa la chiusura o apertura della mano a seconda che si contragga o si rilassi il muscolo; la posizione di riposo è ripristinata grazie ai tendini elastici. Quest'ultima modalità è intercambiabile con l'azione del motore mediante un meccanismo che, una volta percepito lo stimolo meccanico dovuto alla spalla, permette di passare a un controllo volontario, che subentra al controllo elettronico. I vantaggi del controllo alternato permettono sia di garantire la precisione e il controllo dovute ad una protesi *Body Powered*, ma al contempo di continuare a funzionare anche in assenza di stimoli muscolari, utilizzando un'alimentazione esterna. In questo modo l'operatore risulta meno affaticato se sottoposto a periodi prolungati di stress o a lavori stancanti di lunga durata.

### 2.1.4 Mano robotica - Biomimetic robotic hand

Nel progetto di cui si parla nell'articolo [16] si presenta una mano robotica biomimetica progettata per imitare la biomeccanica della mano umana e implementarne le principali caratteristiche. Nella prima parte, viene effettuato uno studio sulle componenti meccaniche di una mano umana; rappresentano un sistema complesso che le permettono di essere agile, precisa.



**Figura 2.6:** Visione frontale e posteriore della mano Biomimetica - immagine presa da [16]

Nel prototipo in questione, sono stati replicati gli elementi meccanici di una mano biologica:

- **Ossa**, scansionate prima mediante la risonanza magnetica e successivamente stampate in 3D, per ottenere una versione dettagliata della superficie come le giunture terminali oppure gli inserimenti dei tendini.
- **Articolazioni**, che rappresentano i collegamenti tra un osso e un altro, determinano i gradi di libertà delle dita.
- **Legamenti**, bande fibrose posizionate al lato delle dita, limitano il raggio di movimento (ROM, range of motion) complessivo del dito, servono a stabilizzare il dito e a impedire movimenti innaturali. Nel prototipo sono implementati mediante l'uso di gomme, tagliate a laser.
- **Tendini e muscoli**, generano e ripartiscono il movimento lungo il dito. I muscoli vengono paragonati ai servo motori mentre i tendini ai fili che

vengono usati per distribuire il movimento. Quest'ultimi si differenziano in estensori e flessori, che distendono e piegano le dita. Siccome i muscoli responsabili del movimento della mano sono situati nell'avambraccio, da qui la scelta di collocare i motori non direttamente sul palmo, come prevedeva la mano DRL, ma bensì dentro l'avambraccio.

- **Estensori**, sono una struttura fibrosa, in collagene, che avvolge le dita e funge da sistema di frenata, regolando costantemente i momenti torcenti posti sulle articolazioni delle dita durante la fase di contrazione. Costituisce un efficiente sistema elastico di tensione che sostituisce l'azione di eventuali molle esterne.

Secondo l'articolo [16], le rappresentazioni meccaniche tradizionali non sono paragonabili ai modelli che usano materiali morbidi, poichè seppur permettendo un'approssimazione delle controparti umane, comprendono in modo grezzo la loro cinematica ma vanno inevitabilmente incontro a delle discrepanze. Infatti, i componenti biomeccanici all'interno delle mani, come ossa, articolazioni, tendini e muscoli rappresentano un sistema fisico complesso che presenta proprietà uniche che, durante il processo di meccanizzazione, vengono inevitabilmente scartate, qualità che vengono riprodotte più fedelmente da materiali non convenzionali come plastiche elastiche e soft materials.

## 2.2 Tecnologie sperimentali in fase di sviluppo

Interessanti studi e progetti riguardano metodi innovativi ma difficilmente implementabili in larga scala, allo stato attuale della tecnologia. Nei paragrafi successivi vengono analizzate alcune soluzioni rivoluzionarie, non ancora implementabili ma che in un futuro potrebbero trovare spazio all'interno dell'industria biomedica.

### 2.2.1 Protesi a controllo mentale

Un progetto condotto presso l'Università del Michigan [18], prevede il collegamento chirurgico dei nervi periferici al computer, tramite elettrodi. La principale limitazione delle protesi moderne, infatti, risiede nella difficoltà di controllo da parte dell'operatore, che spesso rinuncia al loro impiego poiché risulterebbe più complicato rispetto all'utilizzo dell'arto amputato libero [19]. La soluzione proposta è quindi di controllare direttamente la mano con gli impulsi nervosi, tramite una tecnica rigenerativa dell'interfaccia dei nervi periferici (RPNI) che prevede



di stabilire in primo luogo un collegamento, e in seguito addestrare il programma che assocerà gli impulsi elettrici che scaturiscono dal movimento che l'operatore desidera imporre. I risultati del test, eseguito su un campione di persone soggette ad amputazione, sono promettenti, dimostrando risultati positivi in termini di coordinazione, controllo e manipolazione di oggetti o esecuzione di movimenti, come per esempio giocare a carta-sasso-forbice. Tuttavia, questa soluzione è ancora limitata all'ambito della ricerca, a casi di studio particolari, mentre le principali soluzioni utilizzate negli scenari più tipici sono costituite da tecniche non invasive, come le protesi mioelettriche e a potenza corporea [17]

## 2.3 Risultati dell'analisi dei modelli di stato dell'arte

Dall'analisi di alcuni dei modelli più rilevanti all'interno della comunità scientifica, sono stati presi in considerazione gli aspetti più importanti e innovativi al fine di replicare una mano robotica tanto più performante, nei limiti possibili dell'ambiente di lavoro. In particolare, sono state presi in considerazione la *underactuation*, una caratteristica della catena cinematica della mano robotica che permette di adattare la presa alla forma del dito, come verrà in seguito mostrato nel paragrafo (4.4), usato in modo preponderante nella mano SoftHand Pro-H.

La scelta di collocare i motori non direttamente sulla mano ma posizionandoli all'esterno, trasferendo il movimento mediante dei tendini, tiene in conto le considerazioni fatte per la mano DRL; Si è preferito adottare la linea di pensiero come quella adottata nella mano Biomimetica, per rispettare l'anatomia e la meccanica naturale della mano, includendo nel prototipo anche un avambraccio .

Di interessante impiego ma proibitivi per quanto riguarda l'implementazione nel progetto sono l'utilizzo di tecnologie più complesse, come soft materials, che permettono di simulare più coerentemente con il modello anatomico umano il comportamento meccanico delle dita, introducendo degli elementi elastici, come per esempio nella mano Biomimetica. I meccanismi di sicurezza, come *NBDM* 1.3, che introdurrebbero nuove interessanti dinamiche, con un potenziale aumento della durata energetica della protesi. Infine, i sensori, come il sensore *EMG* per rilevare la contrazione muscolare, oppure sensori di pressione, proibitivi sia a livello di costi che a livello di implementazione, permetterebbero un controllo puntuale della protesi mediante l'uso di meccanismi di feedback.



## Capitolo 3

# Considerazioni preliminari e scelte stilistiche

Le sezioni precedenti hanno gettato le fondamenta per poter effettuare delle scelte di progettazione che possano produrre un elaborato finale quanto più efficace, nei limiti delle risorse, tecnologie e dell'ambiente di lavoro. Nonostante queste limitazioni, comunque sono stati presi come riferimento degli standard riguardevoli per un primo modello di mano robotica che soddisfi un'ampia gamma di funzioni. Le caratteristiche principali considerate per lo sviluppo della suddetta mano sono state in primo luogo l'accessibilità; infatti lo scopo principale era quello di progettare una mano artificiale a budget ridotto, replicabile, che non si incentrasse principalmente sullo suo sviluppo meccanico tanto quanto il suo uso come manichino per lo sviluppo di funzioni software e, in un ipotetico futuro, anche come strumento di appoggio per il test e la calibrazione di sensori, algoritmi di *machine learning* e tutto ciò che concerne il *post-processing*, una volta finita e stabilizzata la fase fisica-meccanica. Queste considerazioni hanno portato ad appoggiarsi al modello meccanico open source InMoov (sezione 2.1.2). Il design è fedele a quello di una mano reale, fattore rilevante sia per mettere a suo agio l'operatore (uno dei motivi che incentiva il successivo abbandono della protesi [19]), sia per avere un modello anatomico realistico in fase di collaudo di software e algoritmi di controllo della mano stessa. Il modello OpenSource permette inoltre il libero accesso allo schema meccanico della mano, permettendo una notevole velocizzazione della fase di progettazione, ponendo l'attenzione sullo sviluppo dei metodi citati in precedenza.

## 3.1 Materiale e stampaggio

Oltre al costo, altri fattori principali per lo sviluppo di una mano robotica efficace sono la resistenza e il peso del modello. La scelta di usare un polimero, l'acido polilattico (PLA), soddisfa le precedenti funzioni in quanto si tratta di un polimero estremamente comune, economico (il costo di quello utilizzato per il progetto è di 0.00219 € al grammo) ma al contempo resistente. Esso rappresenta in ambito biomedico delle proprietà aggiuntive; l'acido lattico è un composto naturale normalmente presente nell'organismo e quindi ben tollerato e facilmente metabolizzabile. In caso di interazione accidentate con l'organismo, l'acido polilattico è quindi facilmente smaltibile e non cancerogeno per l'organismo e può entrare in contatto senza presentare un rischio da parte dell'operatore. Nella vita comune è possibile che in seguito a un ripetuto utilizzo ci sia usura del dispositivo con conseguente rilascio nell'ambiente di piccole dosi di particelle. In seguito ad attività dove si entra a contatto con l'organismo interno, come i pasti o l'utilizzo di oggetti a uso igienico, le particelle di materiale inevitabilmente si trasmettono nell'organismo; nel caso di materiali come il PLA, questa trasmissione non comporta un rischio poiché viene smaltito facilmente.

### Stampa 3D

Il progetto InMoov è stato pensato per poter essere replicato in plastica tramite l'uso della stampa 3D e i modelli aperti al pubblico sono ottimizzati per essere riprodotti con questa tecnica, che risulta più accessibile e veloce rispetto a una tradizionale modellazione di materiali ferrosi come la fresatura di acciai o alluminio o taglio con sistemi computerizzati (CNC). L'applicazione di modifiche o la calibrazione in corso d'opera risulta più pulita e semplice rispetto ai materiali tradizionali che necessitano di un luogo di lavoro specializzato e di strumenti tecnici, se paragonata alla modellazione o la rifinitura di plastiche, che possono essere eseguite con oggetti di uso più comune senza l'impiego di strumenti altamente specifici.

Il PLA rappresenta uno dei materiali più comuni per quanto riguarda la stampa 3D grazie alla sua bassa temperatura di fusione ( $\sim 170^{\circ}C$ ) e le sue ottime prestazioni meccaniche. Ha un peso specifico di  $1.3 \text{ g/cm}^3$ , il che lo rende adatto alla stampa di oggetti leggeri: grazie alla costruzione a struttura alveolare, lasciando i volumi interni parzialmente vuoti, si conferisce alla struttura un'elevata resistenza, senza però aumentare eccessivamente il peso complessivo.

Tutto ciò si traduce in un facilitato controllo degli elementi da parte dei motori e maneggevolezza del prototipo finale.

## 3.2 Servo Motore

La motorizzazione del prototipo è svolta dal servo motore, un dispositivo meccanico ed elettrico che genera una coppia in risposta a un segnale di controllo. È costituito da un sistema di ingranaggi che costituisce un rapporto di riduzione, garantendo la forza e la precisione necessari per trasmettere il movimento dall'albero del motore all'asse di uscita del servomotore. Il circuito riceve un segnale di controllo, che determina la posizione desiderata; confrontandolo con la posizione reale dell'asse, regola il motore per raggiungere e mantenere la nuova configurazione. Le sue peculiarità lo rendono ideale per l'uso in ambito robotico, per assegnare a ogni dito una posizione ben precisa che varia solo in base all'angolo che si vuole associare (verrà assegnato 0 gradi la posizione completamente aperta, 180 gradi la posizione completamente chiusa). Per le specifiche del motore si veda l'appendice (7.2)

### 3.2.1 Integrazione con Arduino

La possibilità di "scrivere" la posizione desiderata tramite un semplice impulso elettrico, rende il dispositivo ideale per essere pilotato da una board elettronica di comando, come per esempio lo è Arduino. L'ampiezza del movimento, come detto in precedenza, dipende dall'impulso elettrico fornito, in particolare dal suo Duty cycle.

Dalla scheda tecnica del prodotto [4], si ricavano dei valori di max e min di circa 1 ms e 2 ms. Sperimentalmente, si verifica che si ha una rotazione minima per valori di un impulso di durata pari a  $5.44 \cdot 10^{-4}$  secondi e rotazione massima per valori di un impulso di durata pari a  $2.40 \cdot 10^{-3}$  secondi. Questi parametri saranno utili in futuro per la programmazione dell'impulso da inviare al servo motore.

### 3.2.2 Caratteristiche di bloccaggio

Dalla scheda tecnica del componente risulta che è in grado di sviluppare una coppia di  $245.2 \text{ N} \cdot \text{mm}$ . Come specificato nella sezione (1.3), è di fondamentale importanza che il servo motore opponga una resistenza passiva anche in assenza

di corrente, per garantire un risparmio di energia e sicurezza anche in condizioni di motori spenti. La coppia passiva che il motore sviluppa è sufficiente affinché le dita della mano non "rammolliscano" non appena il motore si spegne, ma invece venga mantenuta la posizione impostata. Come verrà specificato all'interno della sezione (6), la coppia passiva è sufficiente a tenere in mano oggetti di piccolo peso, risultando inefficace per oggetti pesanti. Tuttavia, una possibile miglioria delle prestazioni passive del servo motore si potrebbe ottenere implementando un sistema di bloccaggio esterno, applicato tra i motori e il sistema di trasmissione del movimento. Una soluzione proposta è quella riportata da Controzzi et al. [6], che prevede di utilizzare un meccanismo delle dimensioni comparabili a una moneta da 2 cent, come già accennato nella sezione (1.3). Nonostante le dimensioni siano un punto a favore dell'implementazione facilitata all'interno del prototipo, la sua replica è invece estremamente complicata, se non impossibile tramite la tecnologia attuale di stampa 3D ed estremamente costosa la sua realizzazione mediante l'utilizzo di tecniche CNC per la lavorazione di metalli.

### 3.3 Componenti non inclusi

Per completezza di informazione, viene incluso una parziale lista di possibili soluzioni che, per la mancanza di spazio usabile nel prototipo o tempo a disposizione, non sono state incluse nel progetto. Tuttavia, potrebbero essere implementate in progetti simili a quello qui svolto, in versioni alternative o rivisitate altrettanto valide o più sofisticate.

#### **Molle di ritorno**

Le molle a torsione possono essere impiegate per mantenere costante la tensione del filo lungo tutto il movimento e favorire un'ulteriore fluidità, ma anche per forzare le dita a mantenere una posizione tendenzialmente contratta, evitando configurazioni singolari. Nel progetto si sarebbero potute posizionare in corrispondenza dei giunti, all'interno dell'articolazione.

#### **Sistema di tensione**

Un sistema di tensione dedicato principalmente a mantenere i fili attivi, per esempio mediante l'uso di molle a trazione, da applicare a ogni singolo tendine. In questo caso, verrebbero posizionate all'interno dell'avambraccio e quindi non direttamente sulle dita, come invece suggeriva la soluzione precedente.

### **Sistema di motorizzazione in loco**

Un altro prototipo di mano robotica era stato preso in considerazione prima della realizzazione dell'attuale modello, che prevedeva il posizionamento dei motori direttamente sul palmo, adottando un sistema di trasmissione del movimento interamente rigido senza ricorrere all'utilizzo di tendini come nel progetto attuale. Tuttavia, per le considerazioni già effettuate nella sezione (2.3) non si è osservata nessuna sostanziale miglioria che potesse giustificare l'incremento della complessità totale del sistema di movimento.

### **Sensori**

All'interno del progetto non è presente un sistema sensoriale, tuttavia, è presente lo spazio fisico per implementare, per esempio, dei sensori di pressione in prossimità dei polpastrelli. In questo modo è possibile verificare quale forza è applicata istantaneamente sull'oggetto impugnato e, in caso, regolare i motori di conseguenza.

Un sensore EMG, che rileva il segnale elettrico proveniente dalle fibre muscolari, può essere usato per azionare la mano robotica e, con ausilio di algoritmi di elaborazione dei segnali, discriminare l'intensità dei segnali rilevati, anche per eventualmente eseguire diverse tipologie di movimento, in ingresso a varie intensità di contrazione. Sfortunatamente, il sensore è eccessivamente costoso se confrontato con costo totale sostenuto per il progetto e uno di buona qualità risulta difficilmente reperibile.

Un'interessante soluzione, per tracciare il movimento delle dita relativamente alla mano, è quella di posizionare su di esse dei sensori piezoelettrici, sensibili alla deformazione, oppure dei fili in fibra ottica che funzionano con il principio della riflessione della luce. Per un principio di reciprocità, sarebbe anche possibile inviare segnali di controllo ai motori, applicando i sensori su una mano reale, in modo da replicare fedelmente i movimenti effettuati da essa.





# Capitolo 4

## Progettazione della protesi

Lo sviluppo del design è stato velocizzato dalla presenza di un modello di mano e avambraccio già presente, dato che il modello a cui si fa riferimento è parte di un progetto Open Source, disponibile alla consultazione libera. Tuttavia, le sfide presentate dalla stampa, dall'assemblaggio e alla successiva messa in funzione della mano non sono venute a mancare, nonostante la fruizione della risorsa pubblica.

### 4.1 Progettazione del pollice

I modelli di riferimento analizzati nella sezione (2) fanno riferimento molto spesso a un modello di pollice sofisticato, dotato di più gradi di libertà rispetto alle altre dita della mano, siccome rappresenta il dito che svolge un ruolo più complesso rispetto alle altre dovendo essere in grado di impugnare oggetti. Gli esempi portati dalle mani della sezione (2) propongono modelli di pollice che oltre a contrarsi, riescono anche a ruotare attorno al palmo, in modo da poter combaciare con indice, medio, anulare e mignolo, pizzicando di fatto le superfici di contatto. La scelta di semplificare il movimento rappresenta sì una limitazione, ma allo stesso tempo risulta comunque in grado di afferrare diverse tipologie di oggetti.



**Figura 4.1:** Meccanismo della mano, posizione rilassata - progettazione meccanica

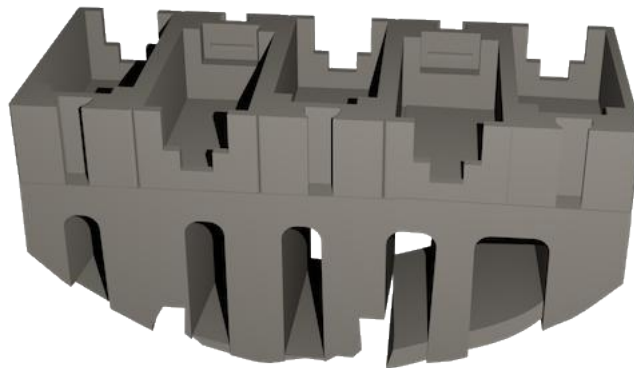
La meccanica approssimata del pollice risulta più semplice rispetto alla progettazione di un ulteriore grado di libertà motorizzato e quindi non ozioso, da controllare da un ulteriore servo motore. Esempi di questo tipo, che privilegiano la semplicità d'uso a discapito della complessità e sofisticatezza di movimento, si ritrovano descritti nel paragrafo (2.1.3), portando a pensare che non sempre maggior sofisticatezza significa miglioria nel prodotto finale, rivolto all'utente. Nell'immagine 4.1 è visibile il meccanismo di rotazione del pollice adoperato; in particolare, si osserva l'asse di rotazione parallela all'asse dell'avambraccio, passante per il dito indice.

## 4.2 Posizionamento motori

Tra gli svariati esempi di mano robotica, si è scelto di allocare i motori lontani dal meccanismo che dovranno pilotare e quindi collocarli non direttamente all'interno della mano, come suggerisce invece la soluzione adottata dalla DRL nella sezione (2.1.1) ma di usare uno spazio esterno aggiuntivo, in questo caso

l'avambraccio. Come già fatto osservare nella sezione (1) questa scelta permette di destinare ai motori un'area di manovra più ampia e di alleggerire il peso della mano, rendendola più agile. Tuttavia, il maggior ingombro che deriva dall'uso di un avambraccio, rende il prototipo maggiormente ingombrante, limitando gli spazi di manovra se in futuro vorrà essere implementato un braccio di spostamento tridimensionale.

Per gestire le tensioni generate dal movimento e garantire una posizione stabile sulla quale generare forza, è stato necessario progettare una base su misura che potesse alloggiare i 5 servo motori (figura 4.2).



**Figura 4.2:** Alloggio per i motori - progettazione meccanica

Come si vede in figura, i motori alloggiano su un supporto rialzato che appoggia solidalmente all'avambraccio e ne segue l'andamento della superficie, risultando particolarmente irregolare se osservato singolarmente ma molto regolare se viene fatto combaciare con l'appoggio sottostante (figura 4.3). I motori si incastrano negli appositi spazi visibili superiormente, in modo da poter essere rimossi agevolmente e al contempo, però, non uscire dalla sede dove sono stati posizionati.



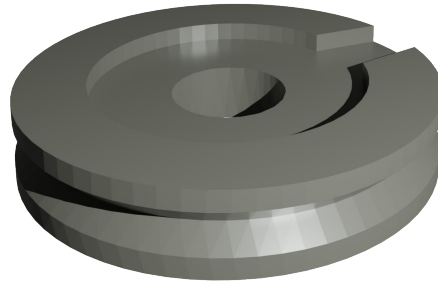
**Figura 4.3:** Alloggio per i motori integrato all'avambraccio- progettazione meccanica

Per una visione completa dell'inserimento dei motori all'interno degli alloggi, si consulti l'appendice 7.3.

### 4.3 Collegamento con filo

La trasmissione del moto, viene effettuata grazie all'utilizzo di tendini inestensibili (in questo caso del filo da pesca trecciato) collegando il motore con i polpastrelli. Per ogni dito, sono collegati due fili; uno dedicato alla contrazione, che accompagna il movimento di andata e un altro dedicato al rilassamento, che guida il movimento di ritorno. Entrambi i fili lavorano in sincronia; quando un filo viene teso e quindi accorciato, il dito si contrae e, come funziona per i tendini, il filo "antagonista" si deve allungare, mantenendo il sistema costantemente in tensione. Viceversa, per il movimento di ritorno, il primo filo andrà ad allungarsi, mentre il secondo si accorcerà, ritornando entrambi alla loro posizione originale.

La problematica maggiore di questa soluzione risiede nel fatto che, essendo i fili usati non elastici, per garantire le massime prestazioni nella fase di movimento è necessario che i fili siano costantemente in tensione, onde evitare che la struttura del dito diventi labile, influenzando in modo negativo la cinematica complessiva. Per risolvere il problema, è stato introdotto un elemento di forma circolare che faccia da guida al filo, e che possa mantenere distribuita la tensione in modo uniforme per tutta la durata del movimento.



**Figura 4.4:** Puleggia di accompagnamento - progettazione meccanica

Il filo scorre nella scanalatura visibile in figura, viene accompagnato durante la rotazione del servo motore, ed è collegato ad esso mediante un attacco in plastica bianca specifico, fornito insieme al servo motore (si veda figura 4.5). L'attacco permette di muovere solidalmente al motore anche la puleggia, mediante un incastro attraverso la scanalatura superiore del pezzo, visibile in figura 4.4. Per una visione di insieme si consulti l'appendice 7.3



**Figura 4.5:** Meccanismo accoppiato puleggia e servo motore - progettazione meccanica

## 4.4 Meccanica del dito

Come già descritto nelle sezioni precedenti, il movimento dal motore viene trasmesso tramite i tendini inestensibili fino al dito, che costituisce il meccanismo finale da mettere in movimento. La cinematica del dito è semplice ma efficace: esso è costituito da tre sezioni, oppure falangi, dette prossimale, media e distale ma conosciute anche come falange, falangina e falangetta; sono attaccate tra loro mediante delle coppie rotoidali alle estremità e sono collegate alla mano mediante una coppia rotoidale tra palmo e falange prossimale.

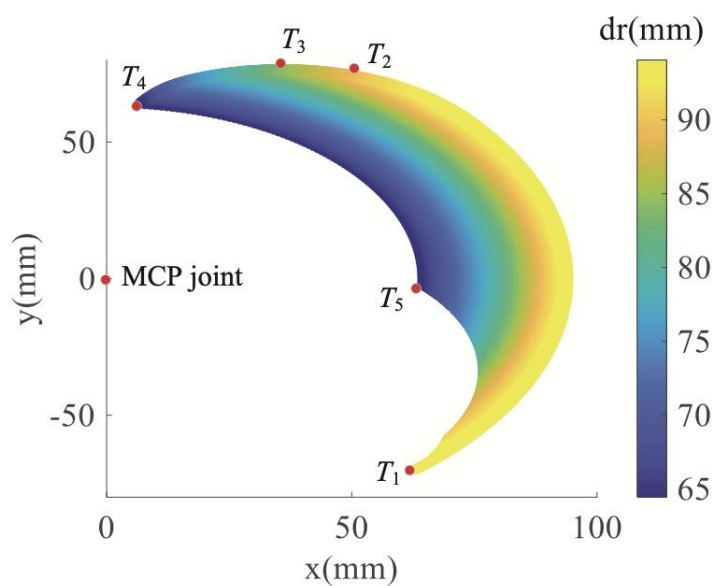


**Figura 4.6:** Meccanismo del dito, posizione contratta da visione laterale - progettazione meccanica

I due tendini raggiungono la punta del dito, dove vengono bloccati solidalmente al telaio in modo da poter contrarre e rilassare il meccanismo. Siccome viaggiano internamente sia al palmo che al dito, risultando nascosti dall'esterno e non visibili in figura.

### Meccanismo sottoazionato

La cinematica del dito è composta da 3 coppie rotoidali; tuttavia, a ogni dito è associato solamente un attuatore, quindi di 3 gradi di libertà, 2 rimangono liberi. Il meccanismo è quindi sottoazionato. Un'analisi completa di un dito robotico sottoazionato è stata svolta nella pubblicazione [20], dove in particolare viene analizzato lo spazio di lavoro del meccanismo (figura 4.7).



**Figura 4.7:** spazio di lavoro meccanismo sottoazionato - [20]

Il meccanismo analizzato nell'articolo è un dito guidato da componenti rigidi, e non tendini come nel progetto; tuttavia la cinematica rimane confrontabile essendo la catena cinematica estremamente simile. Si può osservare come, per un determinato angolo assunto dall'attuatore, è possibile che l'estremità finale del dito assuma varie configurazioni. Questa caratteristica dei meccanismi sottoattuati permette alla mano di afferrare una varietà di oggetti di diversa dimensione, adattando la presa in base al contesto.



Presa adattiva di diversi oggetti - nella prima immagine a sinistra, una pallina da Padel, il dito è parzialmente esteso.

Nell'immagine centrale, un bicchiere; il dito è parzialmente piegato.

Nell'immagine a destra, una confezione plastica, il dito risulta più allungato.

I punti di appoggio variano a seconda della forma dell'oggetto.



**Figura 4.8:** Presa adattiva coordinata delle dita su una pallina da Padel

Nonostante il numero ridotto di motori, è quindi possibile sfruttare questa proprietà per maneggiare oggetti di dimensione e forma differente in maniera semplice. Inoltre, un minor uso di attuatori si traduce in minor peso e soprattutto minor complessità, siccome il movimento è autoregolato e non necessita di essere controllato dall'esterno; costituisce una significativa semplificazione dell'utilizzo quotidiano da parte dell'operatore.

### Range di movimento

Il movimento è in parte limitato dalla geometria del dito: la massima contrazione è limitata dalle 3 componenti che, oltre un certo angolo limite, collidono tra loro impedendo l'ulteriore movimento. In fase di ritorno, invece, sono presenti dei fine-corsa in modo da impedire un eccessivo movimento inverso, che intralcia il successivo ciclo di contrazione, poiché potrebbe passare in una configurazione singolare a causa della forza peso. Inoltre, l'angolo massimo di estensione è determinato dall'Arduino che impone i valori da assumere (in dettaglio nella sezione (3.2.1)) e comunque non oltre i limiti di costruzione dell'attuatore, che possono essere usati come fine-corsa meccanici. L'angolo di massima contrazione non è limitato dal motore, poiché la geometria del dito è più restrittiva rispetto al limite massimo dell'attuatore; in linea teorica si potrebbe contrarre il dito molto di più di quanto lo conceda la struttura.



Figura 4.9: Dettaglio del meccanismo di fine corsa del dito medio - progettazione meccanica

## 4.5 Meccanica del polso

Il collegamento tra mano e avambraccio, dove alloggiavano i motori, è mediato dal polso, che ha il ruolo di tenere uniti i due componenti con un collegamento meccanico mobile. Infatti, seppur non motorizzato, il polso dispone di due gradi di libertà: può ruotare attorno all'asse dell'avambraccio e può basculare attorno all'asse che giace sul piano individuato dal palmo ed è ortogonale all'asse dell'avambraccio.

La scelta di non usare un polso rigido, ma dotarlo comunque di una certa libertà, permette di alleviare le tensioni che si genererebbero nella struttura, se paragonate all'uso di una geometria rigida e rende la mano più adattabile ai vari scenari. L'uso di una struttura labile rispetta maggiormente l'anatomia del corpo



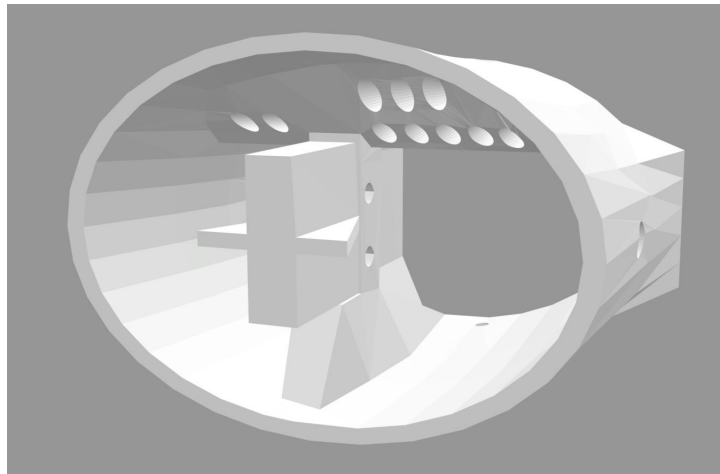
umano, se paragonare invece a una struttura fissa. Inoltre, è disponibile dello spazio all'interno del polso che lascia spazio all'implementazione di un ulteriore servo motore, che permetterebbe la rotazione controllata dello stesso in seguito ad una successiva sofisticazione.



**Figura 4.10:** Dettaglio del meccanismo del polso in posizione normale - progettazione meccanica

Tuttavia, è possibile rimuovere il movimento del polso attorno all'asse dell'avambraccio, mediante l'inserimento di una vite in una scanalatura attraverso il meccanismo di rotazione e il telaio, in modo da accoppiare le varie parti, impedendo qualsiasi tipo di movimento relativo. In figura (4.10) si può osservare che è presente un piccolo foro in prossimità del polso; la configurazione è libera ed è priva del blocco costituito dalla vite.

Attraverso il polso scorrono anche i tendini che permettono la contrazione della mano, dove sono presenti delle guide che impediscono ai vari fili di intrecciarsi su loro stessi durante la rotazione, oppure durante le varie fasi di movimento nelle quali i fili si muovono e potrebbero accavallarsi.

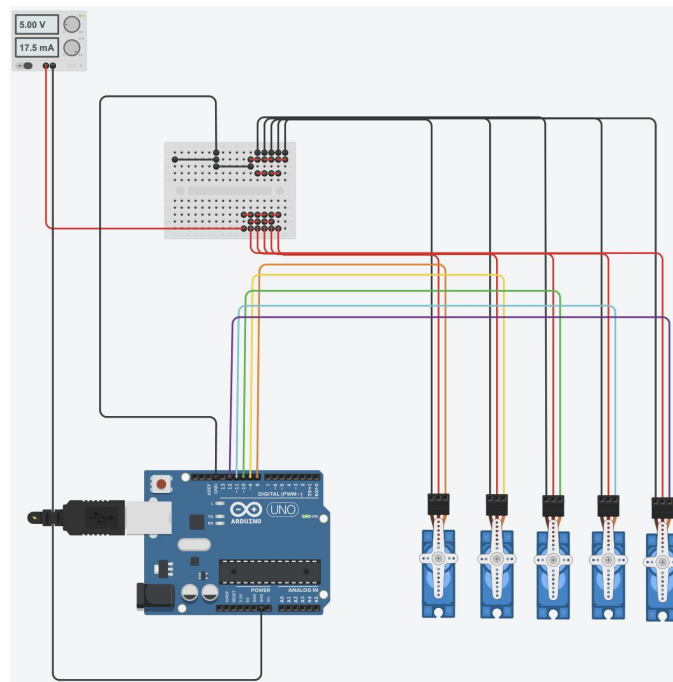


**Figura 4.11:** Dettaglio del polso, struttura esterna dove è visibile la guida interna per i fili, costituita dai buchi interni- progettazione meccanica

## 4.6 Collegamento ad Arduino

Per quanto riguarda la parte di circuiteria, a ogni uscita di segnale di ogni componente micro servo, è stata assegnata una posizione nel Pin digitale dell'Arduino uno, siccome ogni segnale deve essere comandato indipendentemente ed autonomamente rispetto agli altri; non possono e non devono sovrapporsi, in modo da non perdere la differenziazione e la gamma di movimenti. In questo modo, ogni motore può essere comandato autonomamente dall'Arduino facendo uscire un segnale specifico, corrispondente al relativo dito.

Per evitare di sovraccaricare la board con un'erogazione eccessiva di corrente ed evitando il rischio di danneggiarla, si è deciso di utilizzare un'alimentazione esterna per fornire energia ai micro servo; infatti, non è progettata per fornire grosse correnti, cosa che il collegamento di 5 motori potrebbe causare. L'alimentatore in questione utilizzato è in grado di fornire 5V di tensione in output, 450 mA di corrente, mentre necessita di una tensione in ingresso di 230V, adatto quindi al collegamento alla rete elettrica domestica.



**Figura 4.12:** Schema di collegamento elettrico intuitivo utilizzato nel progetto

Come illustrato nella figura (4.12), il collegamento dei 5 servo motori è stato effettuato tramite una connessione in parallelo, in modo da garantire un voltaggio uniforme per tutti i componenti, anche se si va ad aumentare la corrente totale fornita dall'alimentatore, che in questo caso non risulta un problema. Anche la messa a terra è stata effettuata mediante un parallelo, tuttavia, per questioni di trasmissione del segnale, è stato necessario far passare il collegamento anche per la board, mentre non è stato né per il morsetto positivo, come illustrato dal collegamento in figura. Per una descrizione elettrica più dettagliata si consulti l'appendice (7.1)



## Capitolo 5

# Programmazione dei movimenti

In questo capitolo vengono analizzati i metodi utilizzati per controllare il prototipo in maniera indiretta tramite i servo motori. Questi si possono muoversi all'interno di un range di  $180^\circ$ , potendo assumere in modo indipendente l'uno dall'altro una posizione intermedia arbitraria. Perciò, il movimento di ogni singolo dito è autonomo e indipendente dalle altre dita; la mano può assumere quindi una vasta gamma di configurazioni. Si è scelto di poter far assumere al prototipo delle posizioni preimpostate, mediante il codice che segue nel presente capitolo, tuttavia si vuole sottolineare come il prototipo non è limitato alle configurazioni qui descritte ed è in grado di eseguire movimenti delle dita diversificati.

### Scelta dell'ambiente di lavoro

Il dispositivo Arduino è predisposto per il suo utilizzo nel suo linguaggio di programmazione nativo, C. Tuttavia, mediante l'utilizzo di protocolli di comunicazione con la board, è possibile utilizzare ambienti di lavoro e linguaggi di programmazione differenti da quello di default.

MatLab permette di interfacciarsi al dispositivo mediante un'apposito Tool-Box, il pacchetto "MATLAB Support Package for Arduino Hardware", che permette di comunicare interattivamente con Arduino e allo stesso tempo utilizzare le librerie build-in.

### Fase preliminare di programmazione

L'obiettivo di assumere certe configurazioni alle dita della mano, in tempi ragionevoli e con una certa fluidità di movimento, è preceduto dalla fase di configurazione base dei motori. Infatti, per sfruttare tutto il loro range disponibile, nella fase di

collocazione è importante impostare la posizione di tutti e 5 in corrispondenza degli 0°. Di default, ogni volta che il servo motore viene resettato (quindi a ogni accensione), viene automaticamente spostato in corrispondenza dei 90°, ma siccome nel progetto non è fondamentale partire dalla configurazione preimpostata, si è scelto di usare tutto il range a disposizione.

Impostati i motori a 0 gradi, si può procedere al fissaggio in posizione completamente distesa delle dita, mediante il filo, ai motori, fissi all'interno degli alloggi.

## 5.1 Le funzioni di movimento

Una volta completata la progettazione fisica, è di fondamentale importanza muovere in tempo reale e in maniera soddisfacentemente fluida la mano artificiale. L'interfaccia di Arduino con i motori permette di modificare a piacimento l'andamento del movimento delle dita per ogni singolo grado di movimento del servo motore, programmando le tempistiche di movimento, la coordinazione tra i motori e la **velocità** di movimento dei singoli, eseguendo movimenti dolci e controllati. Questo permette alla mano di assumere le configurazioni prestabilite all'inizio del progetto.

### 5.1.1 Setup

Innanzitutto è necessario stabilire, come già spiegato al paragrafo 5, una connessione tra Arduino e Matlab:

```
1 port = "usbmodem11101";
2 a = arduino(port, "Uno");
```

La funzione `arduino()` connette una scheda Arduino Uno su una porta seriale e crea un oggetto di tipo 'arduino' (a) [2], che funge da tramite per il collegamento dei motori alla board:

```
1 function s = setup(a)
2 % riceve in ingresso un'oggetto di tipo arduino e
   % restituisce un'array di 5 oggetti di tipo servo.
3 % Collegamento dei motori alla board
4 s = [
5     servo(a, 'D10', 'MinPulseDuration', 544*10^-6, '
   MaxPulseDuration', 2400*10^-6);
```

```

6     servo(a, 'D11', 'MinPulseDuration', 544*10^-6, '
      MaxPulseDuration', 2400*10^-6);
7     servo(a, 'D9', 'MinPulseDuration', 544*10^-6, '
      MaxPulseDuration', 2400*10^-6);
8     servo(a, 'D8', 'MinPulseDuration', 544*10^-6, '
      MaxPulseDuration', 2400*10^-6);
9     servo(a, 'D12', 'MinPulseDuration', 544*10^-6, '
      MaxPulseDuration', 2400*10^-6)
10    ];
11    moveToOrigin(s);
12 end % function

```

Il primo parametro `a` specifica a quale oggetto di tipo arduino verranno connessi i motori.

Il Secondo parametro, `'D10'`, `'D11'`, `'D9'`, `'D8'`, `'D12'` specificano a quale PIN digitale dell'Arduino sono collegati i servo motori; in ordine, metteranno in moto pollice(1), indice(2), medio(3), anulare(4) e mignolo(5).

I parametri `'MaxPulseDuration'` e `'MinPulseDuration'` stabiliscono, rispettivamente la durata espressa in secondi dell'impulso a scalino che porta il motore ad assumere il valore di angolo massimo e minimo (parametri scelti da 3.2.1).

La funzione `moveToOrigin` verrà invece discussa nelle sezioni successive.

## 5.1.2 Movimento Base

La velocità di movimento del motore standard, come riportato nel paragrafo 3.2, è di 60 deg ogni 0.1 s, troppo elevata rispetto al movimento classico di una mano, portando a movimenti bruschi e impulsivi.

Siccome non è possibile modificare la velocità di rotazione del motore, è necessario compiere molti piccoli movimenti veloci seguiti da una breve pausa. Questa idea implementata nello script Matlab, sfocia nel seguente risultato:

```

1 function moveTo(s, maxAngle, verso)
2 %moveTo(s, angle) muove a velocita' ridotta il motore
  alla posizione
3 % <angle>
4     angle = readPosition(s);
5     speed = 0.01;

```

```
6   if verso == -1
7       while angle <= maxAngle
8           writePosition(s, angle);
9           %pause(0.1);
10          angle = angle + speed;
11      end
12  end
13  if verso == 1
14      while angle >= maxAngle
15          if angle < 0
16              angle = 0;
17          end
18          writePosition(s, angle);
19          angle = angle - speed*verso;
20      end
21  end
22 end %function
```

Dove è stato usato un ciclo for per iterare piccoli movimenti ad alta velocità, con risultato un movimento lento e fluido.

- La funzione `readPosition(s)` legge la posizione del servo motore collegato all'arduino e restituisce un valore percentuale, compreso tra 0 e 1, dove a 0 è associata la posizione 0 deg mentre a 1 è associata la posizione 180 deg.
- La funzione `writePosition(s)` muove il motore `s` all'angolo percentuale `angle`.
- La variabile `speed` indica di quale percentuale di angolo (rispetto al totale) viene ruotato il motore ad ogni iterazione.
- La variabile `verso` tiene conto del senso di rotazione del servo motore, se si trova in fase di andata oppure è in fase di ritorno ; è la responsabile del cambio del segno della velocità angolare una volta raggiunto l'angolo massimo.

### 5.1.3 Angolo di sfasamento - theta

Per tutta la programmazione, si è tenuto in considerazione un leggero sfasamento, che prenderà il nome di `theta`, al fine non avere mai completamente le dita distese,



che potrebbe dare problemi di chiusura ma anche di concedere al meccanismo maggiore spazio di manovra in caso di necessità.

Questo accorgimento fa assumere anche alla mano la tipica configurazione a riposo, leggermente contratta e non completamente distesa.

Il valore che assume `theta` è pari al 10% dell'angolo giro, ovvero 18°; di conseguenza, le dita a riposo in posizione standard saranno in corrispondenza dei motori impostati al 10% (dell'angolo piano).

#### 5.1.4 `moveToOrigin`

Lo scopo della funzione è quello di muovere, come specificato nella fase di costruzione di andare ad allineare le dita con la fase iniziale dei servo motori, in modo da riassumere, indipendentemente dalla posizione del motore, sempre la stessa configurazione iniziale, definita all'inizio del capitolo. Infatti, in seguito a ogni accensione, i motori si muovono automaticamente nella posizione di default in corrispondenza dei 90°, mentre si vuole portarli, mediante la funzione che segue, a una posizione di comodo, a  $0 + \text{theta}$  gradi.

```
1 function moveToOrigin(s)
2
3 % Sfasare i motori per usare tutto il range a
   disposizione
4
5     theta = 0.1; % Angolo percentuale di sfasamento
6     moveTo(s(1), theta, +1)
7     moveTo(s(2), theta, +1)
8     moveTo(s(3), theta, +1)
9     moveTo(s(4), theta, +1)
10    moveTo(s(5), theta, +1)
11
12 end %function
```

Per questioni di progettazione, si è introdotto l'angolo `theta` di sfasamento (5.1.3), in modo da impedire il movimento sotto una certa soglia (18 deg).

## 5.2 I movimenti predefiniti

Lo scopo ultimo della progettazione consiste nel muovere la mano in determinate posizioni, assumendo le configurazioni pre-stabilite oppure eseguire movimenti pre-impostati.

Le configurazioni da raggiungere sono state intitolate come segue, verranno spiegate nelle sezioni che seguono:

5.2.1 - Movimento standard

5.2.2 - Movimento sincrono

5.2.3 - Afferrare

### 5.2.1 moveStandard

La funzione `moveStandard` è la prima delle funzioni di base; permette il movimento di un dito alla volta, contraendolo fino al valore `maxAngle` del servo motore, avanzando di una frazione di angolo pari a `speed` alla volta (5.1.2), ripetendo il movimento completo, andata e ritorno, per `rep` volte. Per il codice completo, si veda appendice (7.5)

La funzione è suddivisa in tre sezioni principali:

1. `%forward` che si preoccupa della contrazione del dito.
2. `%backward` che si occupa del ritorno in posizione originale.
3. `%initialize` che si occupa del setup iniziale, per decidere se muovere i motori in senso orario oppure antiorario.

L'angolo `theta` tiene in conto le considerazioni fatte a (5.1.3).

La sezione `%initialize` si occupa del setup iniziale, verifica qual è l'angolo di partenza del servo motore `s`; se il suo valore è basso, circa 10% dell'angolo piano, allora si muoverà in senso antiorario, altrimenti se risulta alto, circa 80% dell'angolo giro, muoverà in senso orario. Serve per verificare se ci si trova in fase di andata o di ritorno.

La sezione `%forward` si occupa del movimento in senso antiorario. Tramite un espediente matematico, il ciclo viene eseguito fintanto che l'angolo del servo motore non ha raggiunto il valore desiderato di `'maxAngle'`, avanzando ad ogni ciclo di `speed` parti percentuali di angolo.

La sezione `%backward` è la parte complementare di quella precedente: si occupa del movimento orario del motore fintanto che non è stato raggiunto l'angolo `'minAngle'` di partenza, riportando il dito in configurazione iniziale.

All'inizio era stata implementata la funzione `pause()` al fine di rallentare l'esecuzione del ciclo e quindi rallentare il movimento delle dita. Tuttavia, è stato successivamente rimosso (nel codice in appendice 7.4 commentato) poiché dall'esperienza diretta è risultata un'accortezza eccessiva; inoltre complicava la modifica diretta della velocità di movimento al di fuori della funzione, una volta richiamata da altre funzioni.

Si osservi che la funzione muove solo un dito alla volta, singolarmente, non tutte le dita contemporaneamente come un blocco unico. Tuttavia, il movimento di tutte le dita può essere facilmente raggiunto richiamando più volte la funzione, indicando in successione gli oggetti servo su cui deve agire la funzione `moveStandard`.

### 5.2.2 Movimento sincrono

La funzione `moveHand()`, rispetto alla precedente, prevede il movimento simultaneo di tutte le dita fino a quando tutti i motori non sono arrivati alla posizione `maxAngle` gradi percentuali dell'angolo piano.

La fase di ritorno non avviene fintanto che tutte le dita non hanno raggiunto la posizione desiderata. Una volta compiuta la prima fase, si procede con il rilassamento delle dita, riportandole alla posizione di partenza.

Il movimento simultaneo di 5 motori è complicato da effettuare da una singola board Arduino e un solo software Matlab, che di default lavora seguendo un'istruzione alla volta. Tuttavia, se piccoli movimenti vengono effettuati in successione, aspettando l'esecuzione dell'ultimo motore per ricominciare dalla prima, si può ottenere un effetto di movimento simultaneo. Una considerazione analoga era già stata fatta nelle sezioni precedenti, per operare un movimento più fluido delle singole dita. Per il codice completo, si consulti l'appendice (7.5).

La funzione è suddivisa in tre sezioni principali, strutturate come le precedenti della [5.2.1]. Tuttavia, è stato necessario introdurre una sezione di controllo :

1. `%forward` che si preoccupa della contrazione del dito.
2. `%backward` che si occupa del ritorno in posizione originale.
3. `%initialize` che si occupa del setup iniziale, per decidere se muovere i motori in senso orario oppure antiorario.

4. `%check` che si occupa di verificare se ogni dito ha raggiunto la posizione scelta in precedenza e, in seconda istanza, procedere con il movimento di ritorno.

Le sezioni ai punti (1), (2) e (3) dell'elenco precedente, sono analoghe a quelle trattate nella sezione (5.2.1). Tuttavia, il movimento delle dita in questa funzione avviene simultaneamente per tutte le dita, perciò la funzione avrà come ingresso non un oggetto `s` di tipo `servo` ma bensì un vettore di oggetti di tipo `servo`.

La sezione (3) dell'elenco sopra, comprende la preparazione delle variabili vettori `minAngle` e `maxAngle`, che avverrà quindi dentro un ciclo, dove verranno stabilite le posizioni, rispettivamente angoli di partenza e di arrivo, di ogni oggetto `servo` all'interno del vettore `s` che li contiene.

Una nuova variabile vettore `5x1`, chiamata `'sum'` è stata introdotta al fine di soddisfare molteplici richieste. La matrice è strutturata in modo che gli indici degli oggetti del vettore `s` coincidano con quelli del vettore `done`, che viene inizializzato come un vettore di zeri. Le funzioni che copre questa variabile sono le seguenti:

- Durante il ciclo di andata, una volta che un oggetto `servo` di indice `n` ha raggiunto la posizione `maxAngle`, il conteggio della prima riga e della colonna `n` di `done` incrementa di 1. Con una condizione `if` si verifica che l'ennesimo elemento di `done` sia diverso da 1, di conseguenza l'ennesimo oggetto `s` non abbia raggiunto la posizione desiderata.
- Durante il ciclo di ritorno, in modo analogo al punto precedente, viene verificato che il valore dell'angolo di partenza non sia stato raggiunto (o sorpassato) e, in caso contrario, incrementa di 1 il valore della ennesima colonna di `done`.
- Una volta che un oggetto `s` ha compiuto un ciclo di andata e un ciclo di ritorno, l'ennesimo elemento del vettore, `done(n)`, assume valore 2 e la sua posizione di arrivo è stata raggiunta. Mediante l'uso di un ciclo `while` (riga 35 del codice), una volta che tutti e 5 i valori di `done` hanno assunto valore 2, la funzione smette di agire e la mano resta immobile.

Riassumendo, il nuovo vettore `done` adempie ai seguenti scopi attraverso le varie sezioni del codice:

- Bloccare la contrazione disgiunta delle dita.

- Verificare che il ciclo di andata sia stato compiuto da tutte le dita.
- Iniziare il movimento di ritorno simultaneo.
- Fermare le dita una volta rilassate, arrivate alla posizione originale.

Il movimento di andata, della sezione al punto (1) , `%forward`, deve essere riadattato per lavorare con dei vettori, invece di singoli valori. A questo scopo, è ottimale l'uso di un ciclo `for` che itera la trattazione di andata, vista a (5.2.1), su tutti e 5 gli oggetti `servo`; è necessario e conveniente modificare alcune variabili, come `verso`, `angle` e convertirle in vettori contenenti, in ogni colonna, i singoli valori che avrebbero assunto quelle variabili se venisse fornito loro un singolo oggetto `servo`.

Il ciclo `while` è stato sostituito con una condizione `if` al fine di effettuare solo una variazione di angolo per motore alla volta, invece di iterare in continuo ('while') il movimento, lasciando indietro le altre dita.

Il movimento di ritorno, sezione (2) `%backward`, dell'elenco puntato, è simile al movimento di andata. Anche in questo caso sono stati adoperati degli accorgimenti per i valori assunti dalle variabili vettore introdotte, rispetto alla sezione (5.2.1)]a cui si fa riferimento.

La sezione `%check` è stata introdotta con lo scopo di verificare se l'ennesimo oggetto di tipo `servo` avesse raggiunto la posizione di arrivo desiderata. All'interno di una condizione `if`, si verifica se la posizione che ha assunto il motore sia almeno uguale a quella desiderata, se la rotazione avviene in senso antiorario. La moltiplicazione per `verso` inverte il segno della disuguaglianza in caso i motori muovano in senso opposto, orario. Se la condizione è soddisfatta, si procede "bloccando" l'ulteriore avanzamento del dito, tramite l'incremento di `done`.

### 5.2.3 Afferrare

La funzione `grab()` deriva dalla precedente, modificata in modo da poter afferrare oggetti di forma e dimensioni diverse. Per una visione del codice si consulti l'appendice (7.6).

In ingresso è necessario fornire un vettore `maxAngle` con i valori degli angoli percentuale massimi che possono assumere le dita, con la possibilità di attribuire valori diversi a ogni dito, in modo da potersi contrarre in maniera differente a seconda della situazione. Un'altra differenza rispetto alla funzione precedente consiste nel movimento di ritorno; non è presente una vera e propria fase di

ritorno ben definita come prima ma è gestita in base a piacimento dell'operatore. A questo scopo è stata introdotta la variabile `stop`, rappresentante la durata in secondi che devono intercorrere da quanto il movimento si è completato, e quindi l'oggetto è stato afferrato, a quando si procede con il ritorno in posizione predefinita della mano e il rilascio dell'oggetto. La struttura della funzione, è quindi suddivisa in:

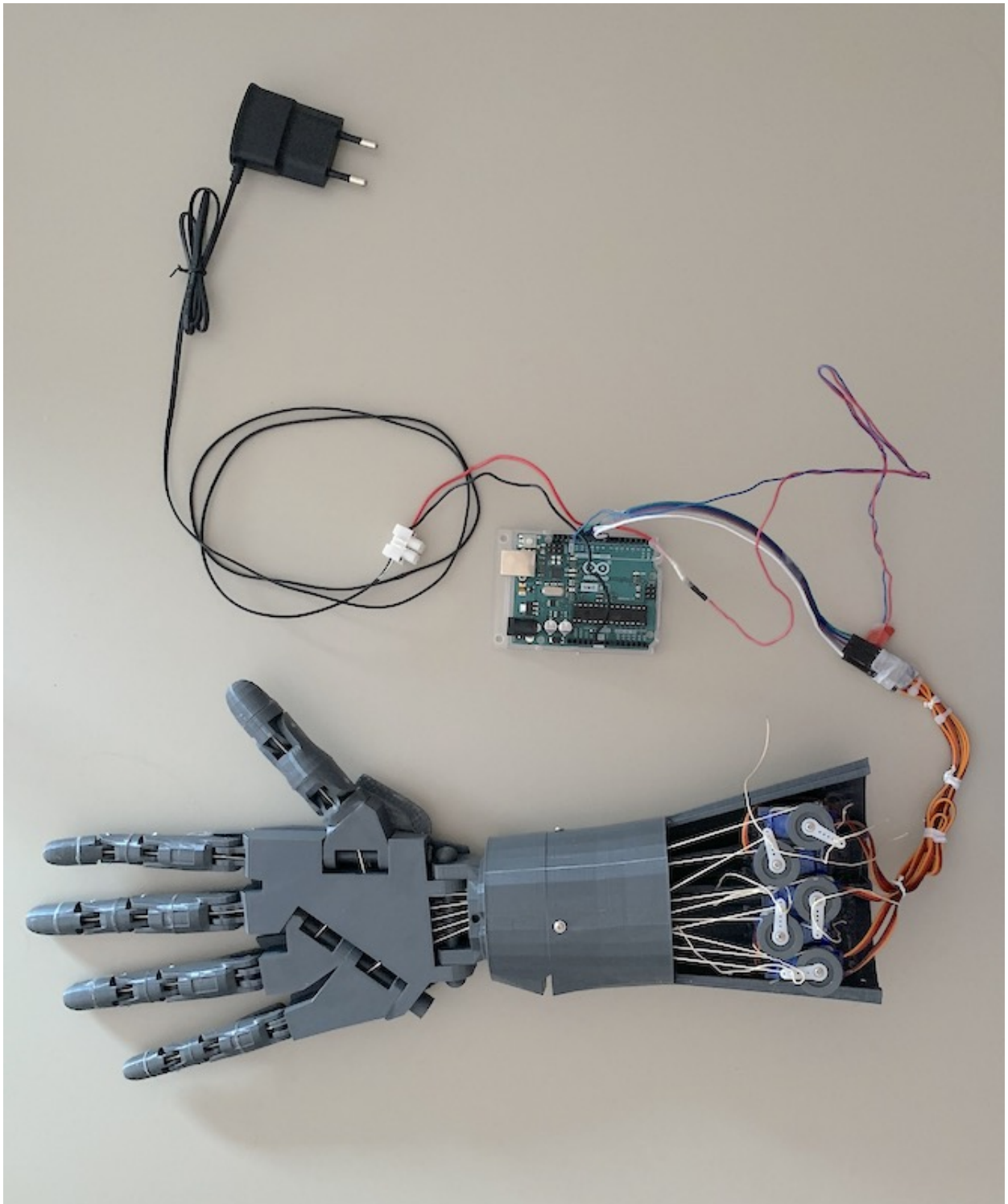
1. **Fase di inizializzazione**, dove si verifica che i motori ruotino in modo coerente con la posizione in cui si trovano.
2. **Movimento di chiusura**, dove ogni motore viene portato, in modo sincrono, alla posizione `maxAngle` preimpostata in ingresso.
3. **Plateau**, dove si attende il tempo impostato dall'utente, come parametro `stop`, attraverso l'uso della funzione `pause()`.
4. **Movimento di ritorno**, dove si è ricorso alla funzione `moveToOrigin()` per rilassare le dita, una alla volta con un movimento non impulsivo.

Durante la fase di test (vedi sezione 6) si è ricorsi a questa funzione, con valori di angolo personalizzati per ogni oggetto in questione, al fine di avere una presa personalizzata per ogni singolo dito che rimanesse fissa nel tempo, e che fosse sufficientemente stretta per mantenere una presa salda sull'oggetto.

## Capitolo 6

### Test e risultati

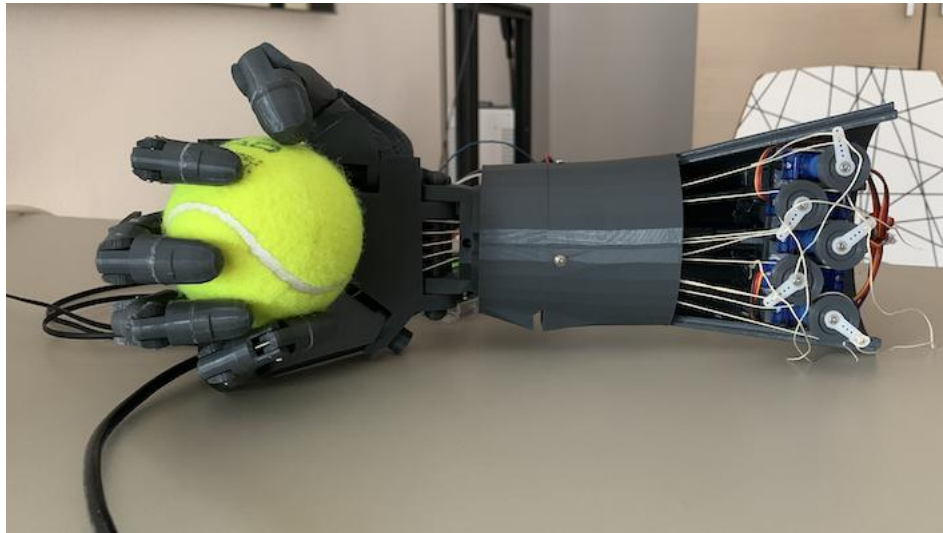
La mano robotica una volta ultimata è stata testata su oggetti di forma, materiale e peso diverso. Vengono ora riportate una serie di immagini che illustrano le configurazioni assunte dal prototipo, in seguito all'interazione con oggetti più o meno rugosi, rendendo più o meno complessa l'interazione con la mano. Per verificare la stabilità della presa, viene talvolta applicato un'inclinazione rispetto al piano orizzontale, per verificare al meglio le capacità e limiti della mano robotica.



**Figura 6.1:** Prototipo della mano robotica e strumentazione - appaiono visibili tutti i collegamenti elettrici dalla mano verso l'Arduino e verso l'alimentazione



## 6.1 Presa pallina da Padel



**Figura 6.2:** Presa di una pallina con angolo di inclinazione rispetto al piano - vista frontale



**Figura 6.3:** Presa di una pallina con la protesi in posizione eretta rispetto al terreno - vista laterale

In questo caso la rugosità e l'elasticità dell'oggetto facilità la presa con le dita rigide del prototipo.

## 6.2 Presa bicchiere in vetro

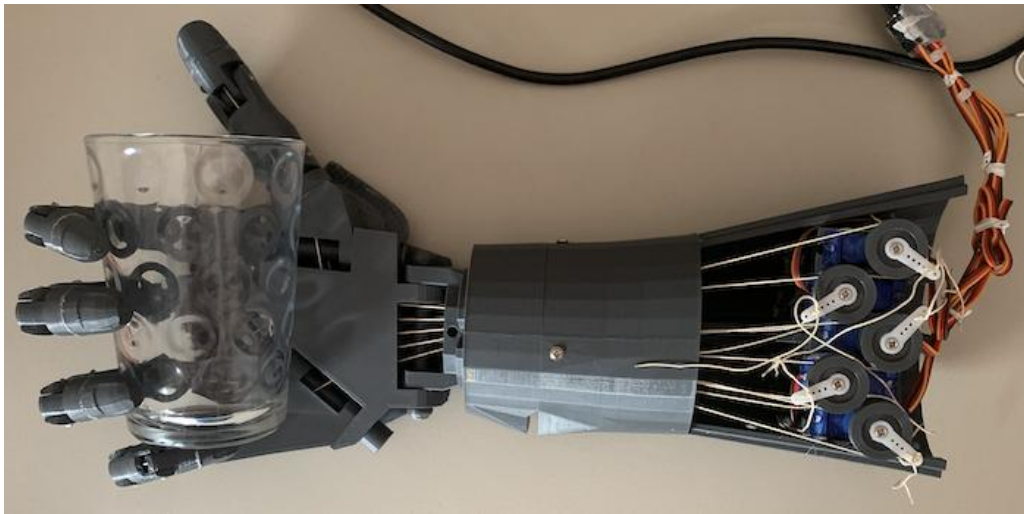


Figura 6.4: Presa di un bicchiere in vetro - vista dall'alto

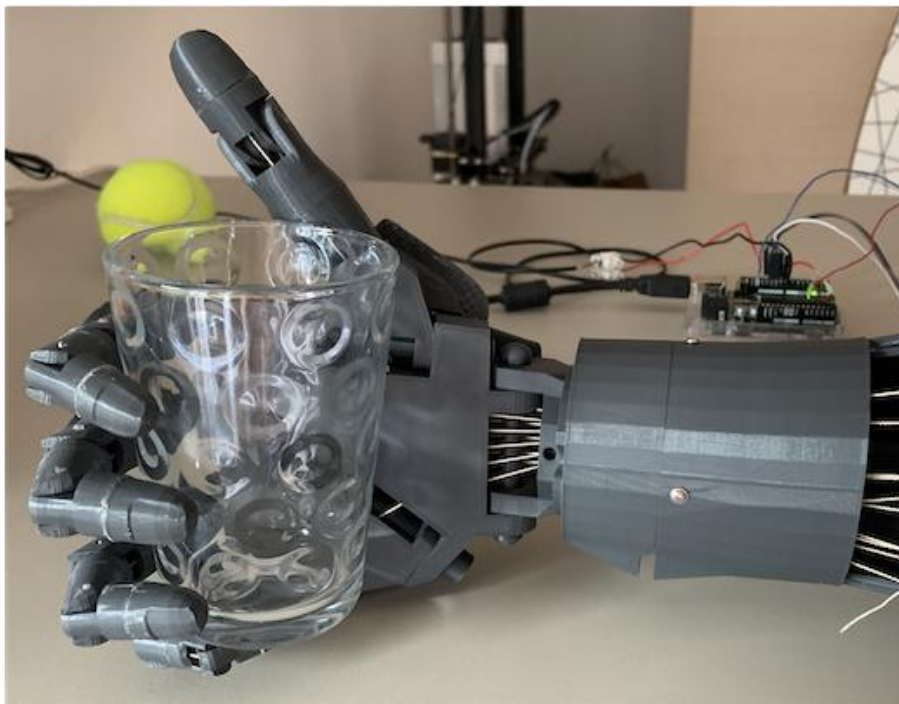


Figura 6.5: Presa di un bicchiere in vetro con angolo di inclinazione rispetto al piano - vista frontale

In questo caso l'oggetto non era elastico e la presa si è dovuta basare sulle reazioni vincolari piuttosto che sull'attrito, essendo l'oggetto duro e scivoloso. L'applicazione sulle punta delle dita di materiali che aumentino l'attrito, come la gomma avrebbe facilitato la presa. L'utilizzo di soft materials (1.4) avrebbe risolto il problema in modo più efficace.

## 6.3 Presa telecomando



Figura 6.6: Presa di un telecomando - vista dall'alto

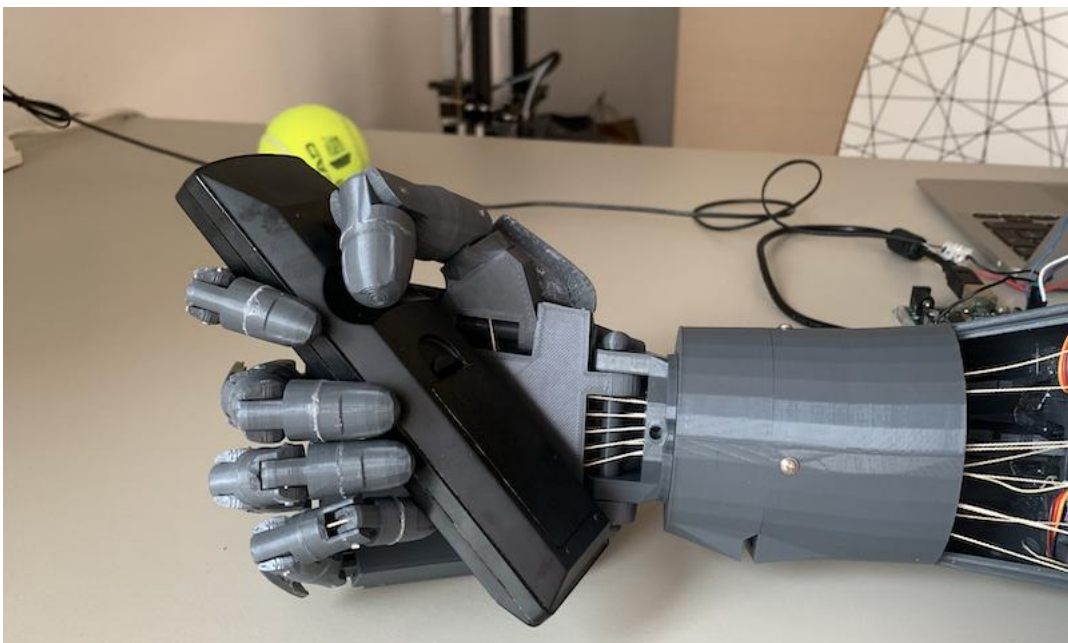


Figura 6.7: Presa di un telecomando con angolo di inclinazione rispetto al piano - vista frontale

Anche in questo caso, la geometria più particolare dell'oggetto, e il fatto che fosse fatto di plastica dura, hanno reso la sua manipolazione più difficoltosa. Ripiegare su una mano costruita con materiali morbidi, che siano in grado di adattarsi meglio alla forma dell'oggetto, avrebbe reso la presa più efficace. Afferrare l'oggetto sarebbe stato più semplice se sulle dita fosse applicato uno strato di materiale gommoso per aumentare l'attrito tra le plastiche.

## 6.4 Conclusioni

Il prototipo realizzato rispetta la geometria di una mano umana e imita la sua naturale meccanica e il suo naturale comportamento, mediante l'uso di articolazioni (coppie rotoidali) e di tendini (fili per trasmettere il movimento). Le priorità che sono state prese in considerazione all'inizio del progetto sono state rispettate: dai test effettuati la mano risulta prestante, sia per quanto riguarda l'interazione con oggetti di forme e materiali più o meno impegnativi, sia per quanto riguarda la velocità di movimento delle dita. Infatti, lungo la catena cinematica non sono presenti grosse forze di attrito che influenzano le prestazioni del sistema; le dita inseguono bene il movimento dei motori. Le prestazioni potrebbero essere incrementate ulteriormente, come accennato nella sezione (3.3) utilizzando un sistema che tenga ulteriormente in tensione i tendini, in ogni fase del movimento per avere un sistema più prestante. Per quanto riguarda l'interazione con gli oggetti, i risultati sono stati soddisfacenti, riuscendo ad afferrare vari oggetti di uso comune, come già visto a inizio capitolo (6). Il peso del prototipo finale è di 380 g.

Da subito si è cercato di sviluppare un prototipo a budget minimo, verificando la bontà delle prestazioni ottenibili facendo però fronte a dei costi moderati, con la possibilità di utilizzare dei componenti di seconda categoria. Il principale costo da affrontare è stato la board di controllo Arduino, poiché già a disposizione durante la fase di costruzione del prototipo. Il costo di un Arduino Uno attualmente è di 30.00 € ma, essendo il progetto OpenSource, è possibile acquistare delle schede di altre marche meno costose ma con stessa struttura e stesso software. Il costo dei servo motori ammonta a 3 € ciascuno, hanno fornito buone prestazioni per il ruolo che dovevano compiere. Il costo totale della plastica è di circa 6.50 € mentre i fili elettrici e il resto delle componenti meccaniche provengono da materiali di recupero e coprono un costo complessivo trascurabile. Il costo complessivo del prototipo, infine, è di poco più di 51.50 €, volendo sottolineare che più della metà è coperto dall'Arduino, che potrebbe essere sostituito senza alcuna sostanziale

modifica alle prestazioni da schede più economiche e che il costo derivato da tutti gli altri componenti del prototipo è di circa 21.50 €.

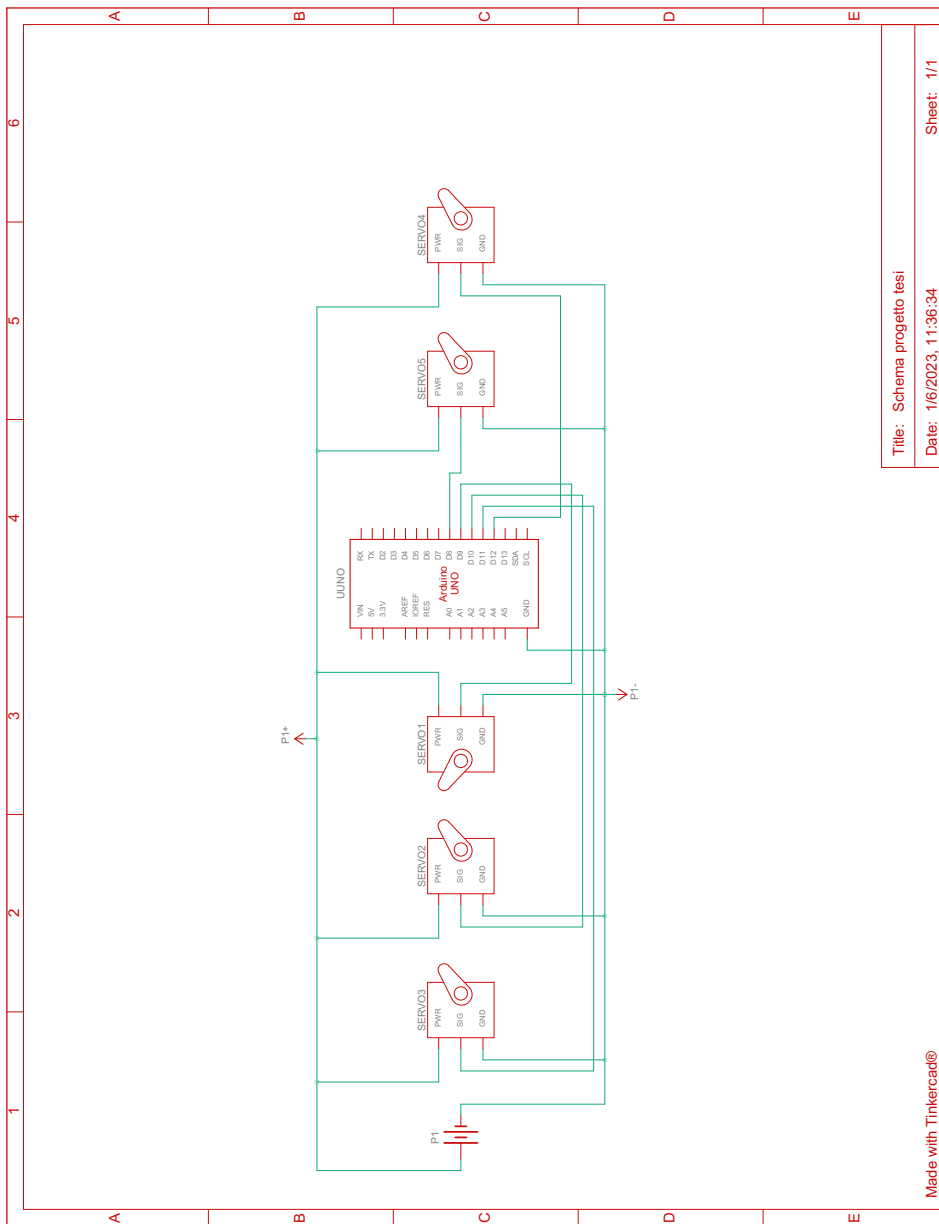
In futuro, sarebbe possibile migliorare il controllo della mano con l'implementazione di sensori di prossimità e di forza(3.3) che possano controllare direttamente lo spostamento dei motori in funzione della pressione applicata sull'oggetto. Se si dovesse rivedere la meccanica della mano, è probabile che si cercherebbe l'utilizzo di soft materials (1.4) o di articolazioni "morbide" (2.1.3) e comunque si preferirebbe l'utilizzo di materiali elastici piuttosto che pezzi completamente rigidi.



# **Capitolo 7**

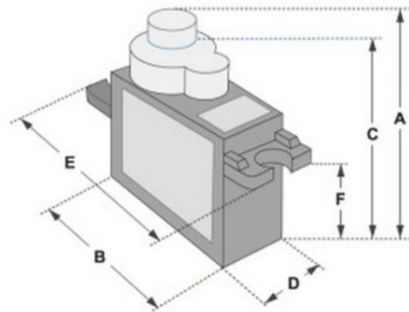
## **Appendice**

# 7.1 Schema elettrico - Arduino



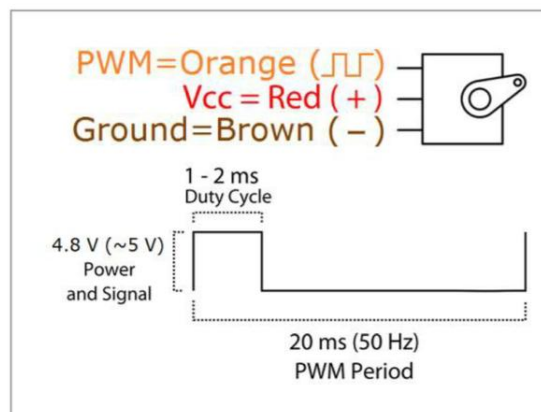


## 7.2 Scheda tecnica - Servo Motore



Dimensions & Specifications	
A (mm) :	32
B (mm) :	23
C (mm) :	28.5
D (mm) :	12
E (mm) :	32
F (mm) :	19.5
Speed (sec) :	0.1
Torque (kg-cm) :	2.5
Weight (g) :	14.7
Voltage :	4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.



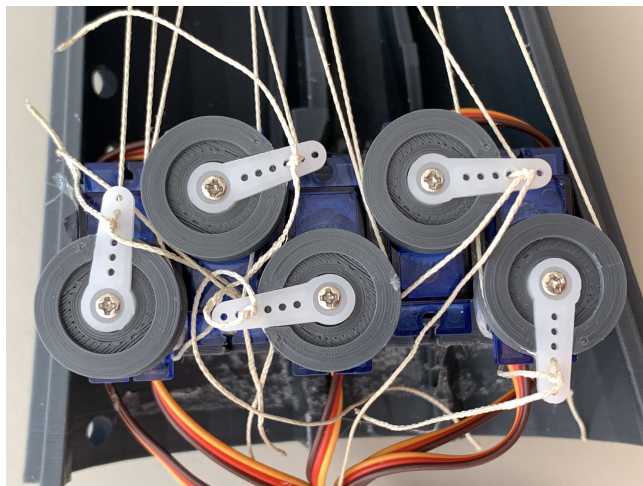
- Modulation: Analog
- Torque: 4.8V : 2.5 kg – cm
- Speed: 4.8V : 0.10sec/60°
- Weight: 0.9 g
- Length: 0.91 in (23.1 mm)
- Width: 0.48 in (12.2 mm)
- Height: 1.14 in (29.0 mm)
- Motor Type: 3-pole

- Gear Type: Plastic Rotation/Support: Bushing
- Rotational Range: 180°
- Pulse Cycle: ca. 20 ms
- Pulse Width: 500-2400  $\mu s$

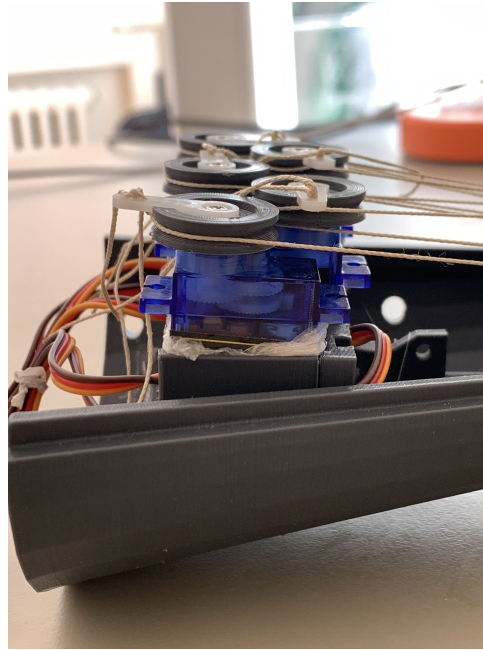
## 7.3 Meccanismo motori

Immagini scattate del prototipo concluso.

### 7.3.1 Visione d'insieme laterale - blocco motore



### 7.3.2 Visione d'insieme superiore - blocco motore



## 7.4 Codice MatLab - moveStandard

Vengono ora inclusi per completezza i codici utilizzati per comandare i motori tramite Arduino.

```
1
2 function moveStandard(s, maxAngle, speed, rep)
3 % moveStandard Muove tutti gli oggetti di s da 0 fino
4   a maxAngle, andando avanti speed parti di angolo
5   alla volta, in senso orario o antiorario in base a
6   verso (+-1)
7 %
8 % INPUT -----
9 % s = S [servo] Oggetto contenente i motori
10 % maxAngle [double] - percentuale max di movimento (da
11   0 a 1)
12 % speed [double] - percentuale di angolo alla quale si
13   procede (da 0 a 1)
14 % rep [integer] - ripetizioni del movimento
15
16     theta = 0.1; % Angolo percentuale di sfasamento
```

```
12     minAngle = readPosition(s);
13     angle = minAngle;
14     cicloCompleto = 0;
15
16     %initialize
17     if angle < 0.1 + theta
18         verso = 1;
19         angle = minAngle;
20     elseif angle > 0.9 - theta
21         verso = -1;
22         maxAngle = angle-maxAngle;
23     end
24
25
26     for k = 1:rep
27
28         %forward
29         while verso*angle <= verso*maxAngle
30             if angle < 0
31                 angle = 0;
32             end
33             writePosition(s, angle);
34             %pause(0.1);
35             angle = angle + verso*speed;
36         end
37
38         %backward
39         while verso*angle >= verso*minAngle
40             if angle < 0
41                 angle = 0;
42             end
43             writePosition(s, angle);
44             %pause(0.1);
45             angle = angle - speed*verso;
46         end
47     end
```

```
48
49 end %function
```

## 7.5 Codice MatLab - moveHand

```
1
2 function moveHand(s, maxAngle, speed)
3 % moveHand Muove tutti gli oggetti di s
4   CONTEMPORANEAMENTE da 0 fino a
5 % maxAngle, andando avanti speed parti di angolo alla
6   volta.
7 %
8 % INPUT -----
9 % s = S [servo] Oggetto contenente i motori
10 % maxAngle [double] - percentuale max di movimento (da
11   0 a 1)
12 % speed [double] - percentuale di angolo alla quale si
13   procede (da 0 a 1)
14
15   maxAngle = ones(1, 5)*maxAngle;
16   done = zeros(1, 5);
17   theta = 0.1; % Angolo percentuale di sfasamento
18   finish = false;
19
20   %initialize
21   for g = 1:5
22       minAngle(g) = readPosition(s(g));
23       angle(g) = minAngle(g);
24
25       if angle(g) < 0.1 + theta
26           verso(g) = 1;
27           angle(g) = minAngle(g);
28           maxAngle(g) = maxAngle(g) + 0.15
29       elseif angle(g) > 0.9 - theta
```

```
28         verso(g) = -1;
29         maxAngle(g) = angle(g) - maxAngle(g);
30     end
31
32 end
33
34 k = 0;
35 while sum(done) < 2*5
36     for g = 1:5
37
38         %forward
39         if verso(g)*angle(g) < verso(g)*maxAngle(g)
&& done(g) == 0
40             if angle(g) < 0
41                 angle(g) = 0;
42             end
43             angle(g) = angle(g) + verso(g)*speed;
44             writePosition(s(g), angle(g));
45
46             %check
47             elseif verso(g)*angle(g) >= verso(g)*
maxAngle(g)
48                 done(g) = 1;
49
50         end
51
52         %backward
53         if verso(g)*angle(g) >= verso(g)*minAngle(g
) && done(g) == 1
54
55             if angle(g) < 0
56                 angle(g) = 0;
57             end
58             angle(g) = angle(g) - speed*verso(g);
59             writePosition(s(g), angle(g));
60             elseif verso(g)*angle(g) <= verso(g)*
```

```
        minAngle(g) && done(g) == 1
61         done(g) = 2;
62         end
63
64         k = k+1;
65     end
66
67 end
68 end
```

## 7.6 Codice MatLab - grab

```
1
2 function grab(s, maxAngle, speed, stop)
3 % moveHand Muove tutti gli oggetti di s
4 % CONTEMPORANEAMENTE da 0 fino a
5 % maxAngle(i), andando avanti speed parti di angolo
6 % alla volta.
7 %
8 % INPUT -----
9 % s = S [servo] Oggetto contenente i motori
10 % maxAngle [double] - percentuale max di movimento (da
11 % 0 a 1) per ogni dito
12 % speed [double] - percentuale di angolo alla quale si
13 % procede (da 0 a 1)
14 % stop [int] - numero di secondi prima di tornare in
15 % posizione predefinita
16
17 done = zeros(1, 5);
18 theta = 0.1; % Angolo percentuale di sfasamento
19 finish = false;
20
21 for g = 1:5
22     minAngle(g) = readPosition(s(g));
23     angle(g) = minAngle(g);
24 end
```

```
20     if angle(g) < 0.1 + theta
21         verso(g) = 1;
22         angle(g) = minAngle(g);
23         % maxAngle(g) = maxAngle(g) + 0.15;
24     elseif angle(g) > 0.9 - theta
25         verso(g) = -1;
26         maxAngle(g) = angle(g) - maxAngle(g);
27     end
28
29 end
30
31
32 k = 0;
33 while sum(done) < 5
34     for g = 1:5
35
36         %forward
37         if verso(g)*angle(g) < verso(g)*maxAngle(g)
38 && done(g) == 0
39             if angle(g) < 0
40                 angle(g) = 0;
41             end
42             angle(g) = angle(g) + verso(g)*speed;
43             writePosition(s(g), angle(g));
44
45         %check
46         elseif verso(g)*angle(g) >= verso(g)*
maxAngle(g)
47             done(g) = 1;
48
49         end
50     end
51
52     pause(stop)
53     moveToOrigin(s)
```



54 `end`



# Bibliografia

- [1] Liang, Xu and Gao, Yifan and Gao, Zhengrun and Li, Guotao and Su, Tingting and He, Guangping, *Analysis of a Novel Underactuated Robotic Finger*, 2022.
- [2] The MathWorks Inc. , *Connect to Arduino Hardware*, <https://it.mathworks.com/help/supportpkg/arduinoio/ug/connect-to-arduino-hardware.html>
- [3] Montagnani, Federico & Controzzi, Marco & Cipriani, Christian. , *Non-back-drivable rotary mechanism with intrinsic compliance for robotic thumb abduction/adduction. Advanced Robotics.* , 2015.
- [4] *SG90 Datasheet, Equivalent, Micro Servo.*, <https://datasheetspdf.com/pdf/791970/TowerPro/SG90/1>
- [5] Negrello Francesca, Stuart Hannah S., Catalano Manuel G., *Hands in the Real World* , *Frontiers in Robotics and AI*. 2020. <https://www.frontiersin.org/articles/10.3389/frobt.2019.00147>
- [6] Controzzi, Marco & Cipriani, Christian & Carrozza, Maria Chiara. *Miniaturized non-back-drivable mechanism for robotic applications.*, 2010
- [7] Butterfass, J. & Grebenstein, Markus & Liu, Hangzi & Hirzinger, G.. *DLR-Hand II: next generation of a dextrous robot hand. Proceedings 2001 ICRA, IEEE International Conference on Robotics and Automation.* 2001.
- [8] Escribà Montagut G. *Inmoov robot: building of the first open source 3D printed life-size robot*, 2016.
- [9] Gael Langevin, *InMoov, open source 3D painted life-size robot*, <https://inmoov.fr>

- [10] Bulgarelli A, Toscana G, Russo LO, Farulla GA, Indaco M, Bona B. *A Low-Cost Open Source 3D-Printable Dexterous Anthropomorphic Robotic Hand with a Parallel Spherical Joint Wrist for Sign Languages Reproduction. International Journal of Advanced Robotic Systems.*, 2016.
- [11] Shahid, Zaryab & Glatman, Abigail & Ryu, Seok. *Design of a Soft Composite Finger with Adjustable Joint Stiffness.* Soft Robotics. 2019
- [12] Schmitt François, Piccin Olivier, Barbé Laurent, Bayle Bernard *Soft Robots Manufacturing: A Review*, Frontiers in Robotics and AI volume 5, 2018 <https://www.frontiersin.org/articles/10.3389/frobt.2018.00084>
- [13] Laschi, Cecilia & Cianchetti, Matteo & Mazzolai, Barbara & Margheri, Laura & Follador, Maurizio & Dario, Paolo. *Soft Robot Arm Inspired by the Octopus.* Advanced Robotics. 2012
- [14] Piazza, C. & Grioli, G. & Catalano, M.G. and Bicchi, A., *A Century of Robotic Hands*, Annual Review of Control, Robotics, and Autonomous Systems, 2019. <https://doi.org/10.1146/annurev-control-060117-105003>
- [15] Controzzi, M., Cipriani, C. and Carrozza, M. C. “*Miniaturized non-back-drivable mechanism for robotic applications*”, Mechanism and Machine Theory. 2010.
- [16] Zhe Xu and Todorov, Emanuel, *Design of a highly biomimetic anthropomorphic robotic hand towards artificial limb regeneration*, 2016 IEEE International Conference on Robotics and Automation (ICRA). 2016.
- [17] Piazza, C. and Catalano, M.G. and Godfrey, S.B. and Rossi, M. and Grioli, G. and Bianchi, M. and Zhao, K. and Bicchi, A., *The SoftHand Pro-H: A Hybrid Body-Controlled, Electrically Powered Hand Prosthesis for Daily Living and Working*, IEEE Robotics & Automation Magazine. 2017.
- [18] Cindy Chestek and Paul Cederna, Università del Michigan, <https://spotlight.engin.umich.edu/mind-control-prosthesis/>
- [19] Biddiss, Elaine PhD & Chau, Tom PhD. *Upper-Limb Prosthetics: Critical Factors in Device Abandonment. American Journal of Physical Medicine & Rehabilitation.* December 2007.

- 
- [20] Liang, Xu and Gao, Yifan and Gao, Zhengrun and Li, Guotao and Su, Tingting and He, Guangping, *Analysis of a Novel Underactuated Robotic Finger*, 2022 China Automation Congress (CAC). 2022.