



**UNIVERSITÀ POLITECNICA DELLE MARCHE**

**FACOLTÀ DI INGEGNERIA**

---

**Studio ed integrazione di dispositivi indossabili per il controllo da remoto di veicoli autonomi**

---

**Study and integration of wearing devices for remote control of autonomous vehicles**

Relatore:

**Prof. Andrea Monteriù**

Tesi di Laurea di:

**Daniele Bruni**

Correlatore:

**Prof. Alessandro Freddi**

**Dedico questa tesi**  
ai miei genitori e fratelli,  
che per me vogliono sempre il meglio  
agli amici,  
che mi hanno sempre supportato  
e assecondato nei miei progetti e richieste;  
a tutti coloro che hanno il coraggio di mettersi in gioco per inseguire i propri sogni;  
a chi non si lascia abbattere dalle difficoltà;  
e a chi fa di un errore un insegnamento.

## Sommario

<b>Capitolo 1. Introduzione</b> .....	4
<b>Capitolo 2. Brain Computer Interface</b> .....	6
2.1 Cos'è la Brain-Computer Interface.....	6
2.2 Funzionamento delle BCI non invasive.....	8
2.3 Event Related Potential.....	8
2.4 Error-related Potential.....	9
2.5 Steady-State Visually Evoked Potential.....	11
<b>Capitolo 3. L'approccio proposto</b> .....	12
3.1 La BCI.....	12
3.2 Cos'è OpenViBE? .....	13
3.3 Cosa sono MATLAB & SIMULINK .....	13
3.4 Cos'è il protocollo TCP/IP.....	14
3.5 Modello di simulazione del drone.....	14
<b>3.5.1 Parametri</b> .....	15
<b>3.5.2 Variabili del drone</b> .....	15
<b>3.5.4 Vettori canonici e matrici trasposte</b> .....	16
<b>Capitolo 4. Controllore</b> .....	18
4.1 Inner loop .....	18
4.2 Outer loop.....	19
<b>Capitolo 5. Training della BCI &amp; testing</b> .....	21
5.1 Il training della BCI.....	21
5.2 Testing.....	22
<b>Capitolo 6. Conclusioni</b> .....	25
<b>Capitolo 7. Lavori futuri</b> .....	26
<b>Ringraziamenti</b> .....	27
<b>Bibliografia</b> .....	28

## Capitolo 1. Introduzione

Negli ultimi decenni, la ricerca nel campo delle interfacce cervello-computer (BCI) ha registrato progressi significativi, aprendo la strada a nuove applicazioni e opportunità in vari settori, tra cui quello della robotica assistiva.

È ben noto a tutti come il corpo umano possa essere soggetto di disabilità di vario tipo, a partire dalle disabilità intellettive a quelle motorie esse sono tutt'oggi oggetto di studio in tutto il mondo.

L'obiettivo di questa tesi è quello di applicare la BCI come dispositivo per controllare veicoli autonomi con l'intento di offrire alternative che risolvano i problemi di disabilità legati ai sintomi che affliggono la mobilità e cercare di migliorare la vita di chi ne è afflitto.

In dettaglio, le interfacce cervello-computer consentono la comunicazione e il controllo di dispositivi attraverso l'interpretazione dei segnali cerebrali. Un'applicazione promettente di questa tecnologia è il controllo dei droni. L'obiettivo è sviluppare un sistema che permetta agli operatori di pilotare un drone con la mente, consentendo movimenti fluidi e intuitivi basati sui pensieri e sulle intenzioni dell'operatore.

La metodologia di questo studio coinvolge la registrazione dei segnali EEG dall'operatore umano mediante una serie di elettrodi posizionati sul cuoio capelluto. I segnali EEG registrati vengono quindi analizzati e classificati per determinare le intenzioni del pilota. In particolare, vengono identificati i segnali EEG associati alle sei direzioni principali di volo: avanti, indietro, destra, sinistra, aumenta e scendi di quota. Obiettivo di questa tesi è dimostrare la fattibilità di tale approccio motivo per cui si limiterà a trattare tre movimenti: avanti, destra e sinistra.

La classificazione dei segnali EEG è una fase critica del sistema. Gli algoritmi di apprendimento automatico vengono addestrati utilizzando dati di addestramento acquisiti durante sessioni di calibrazione. Durante queste sessioni, l'operatore pensa intensamente a ciascuna direzione di volo mentre il sistema registra i segnali EEG corrispondenti. Questi dati vengono quindi utilizzati per addestrare il modello di classificazione.

Una volta che il modello di classificazione è stato addestrato, il drone può essere pilotato attraverso i segnali EEG rilevati. Quando l'operatore pensa a una direzione specifica, il sistema classifica il segnale e invia il comando corrispondente al drone. Questo processo avviene in tempo reale, consentendo un controllo diretto e responsivo del drone attraverso la mente.

I risultati di questo studio dimostrano certamente la fattibilità del pilotaggio di droni tramite una BCI basata su EEG, eppure ci sono sfide significative ancora da affrontare che potranno essere oggetto dei lavori futuri di questa tesi.

## Capitolo 2. Brain Computer Interface

### 2.1 Cos'è la Brain-Computer Interface

La Brain-Computer-Interface (BCI), tradotta in italiano con “Interfaccia cervello-computer”, è un sistema che misura l'attività del sistema nervoso centrale e, attraverso degli elettrodi, la converte in segnali elettrici che possono essere assimilati ad output artificiali utili al rimpiazzamento, il ripristino, il supporto e il miglioramento dei suoi normali output, alterando così le interazioni tra di esso e l'ambiente esterno [1].

Questa tecnologia permette la rilevazione di specifici schemi neurali grazie ai quali è stato possibile realizzare un canale di comunicazione tra il cervello umano e un dispositivo esterno quali, ad esempio, i computer dove è stato possibile realizzare algoritmi che impiegano tali schemi per pilotare il proprio mezzo con il pensiero. Grazie a questa caratteristica la BCI è stata utilizzata efficacemente in diversi ambiti dell'ingegneria biomedica e della neuroingegneria per lo sviluppo e la progettazione di sistemi di supporto funzionale e di ausilio per persone con disabilità.

I supporti automatizzati per le persone con disabilità richiedono l'impiego di sensori per la tutela e la sicurezza dell'utente i quali, però, non sono esenti da errori. Come soluzione a questa problematica numerose ricerche scientifiche propongono il controllo condiviso uomo-macchina grazie al quale l'utente diventa parte integrante del ciclo di controllo, lo studio avvenuto nelle ultime decenni ci consente oggi di impiegare la BCI come strumento di controllo utile alla realizzazione di tale soluzione. Tutto ciò reso possibile grazie a specifici segnali elettroencefalografici (EEG) emessi dall'utente che verranno adoperati per la correzione degli errori del supporto automatizzato.

Esistono due tipologie di BCI che si differenziano per metodo di applicazione, quelle invasive e quelle non invasive.

Le BCI non invasive sono quelle più comuni soprattutto nell'ambito di ricerca di tecnologie di supporto, consistono nell'impiego di una cuffia di materiale isolante da posizionare sullo scalpo e di elettrodi posti su di essa che vengono messi a contatto con l'epidermide dell'utente (Figura 1).



Figura 1 - BCI non invasiva [2]

Le BCI invasive invece richiedono interventi chirurgici per l'installazione di una matrice di elettrodi a livello sottocutaneo (Figura 2). Questa versione di BCI è utilizzata principalmente per compensare la mancanza di alcune funzionalità cerebrali in caso di aree lesionate o parzialmente inattive a livello sinaptico.



Figura 2 - Installazione BCI invasiva [3]

Oggetto di questo studio saranno le BCI non invasive alle quali si farà sempre riferimento in seguito.

## 2.2 Funzionamento delle BCI non invasive

Le BCI non invasive si basano sul rilevamento del segnale bioelettrico di natura sinaptica per mezzo di elettrodi posizionate sullo scalpo, estraendo così il segnale elettroencefalografico (EEG) pronto per essere elaborato e analizzato da un calcolatore elettronico.

Un primo problema legato ad un corretto impiego delle BCI è legato al segnale stesso. Difatti non solo l'analisi grafica dell'EEG mostra una forte presenza di disturbi e rumori causati dai segnali bioelettrici emessi dagli altri neuroni presenti nel cervello, ma il segnale rilevato non porta con se tutto il contenuto informativo necessario alla sua trasduzione da parte del compilatore. Questa è la ragione per cui la corretta classificazione dei segnali EEG è la base per un controllo ottimale dei sistemi associati.

Di seguito sono stati trattati tre segnali importanti per l'analisi, tutti questi segnali sono generati da stimoli visivi.

## 2.3 Event Related Potential

Uno dei segnali che accomuna la maggior parte delle applicazioni della BCI è il "Event-Related Potential (ERP)", tale termine indica quei segnali prodotti in maniera sincrona dai neuroni corticali come risposte cerebrali derivanti da stimoli elettrofisiologici come possono essere degli stimoli visivi. Suscita particolare interesse di tali ERP lo studio di una particolare componente ondulatoria che prende il nome di "P300" poiché essa è responsabile della contrazione muscolare volontaria, tale segnale presenta il suo picco a 300ms dalla ricezione di uno stimolo e da tale cifra deriva il suo nome (Figura 3).



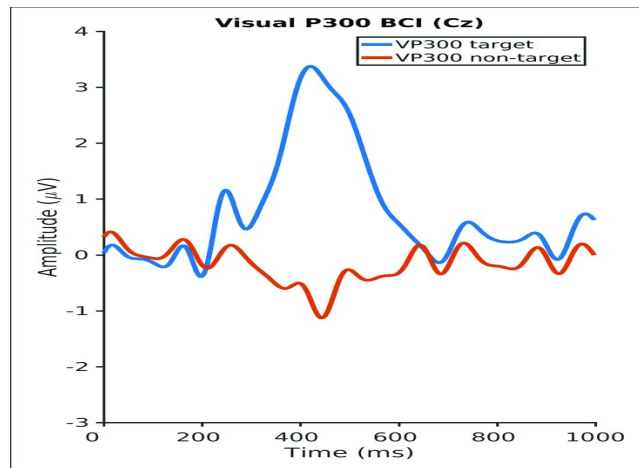


Figura 3 - Tipico segnale P300 estratto dall'elettrodo Cz [4]

Nella letteratura si trovano numerosi esempi di BCI basate sulla P300. B. Rebsamen et al. in [5], nel 2010, esplorano l'utilizzo di una BCI per navigare negli ambienti indoor: uno schermo diviso a griglia mostra all'utente vari ambienti, o oggetti preimpostati, come destinazioni predefinite. Questi quadrati della griglia vengono fatti illuminare in ordine casuale più volte per pochi millisecondi e il sistema BCI, a seconda di quale stimolo luminoso genera il P300, riconosce la destinazione. In questo modo e visto che l'ambiente è preimpostato, la Smart Wheelchair calcola una traiettoria e raggiunge il luogo desiderato.

C.Lopes et al. nel 2011 propongono in [6] un approccio differente, dove grazie al P300 è possibile muoversi anche in un ambiente non preimpostato. Lo schermo, in questo caso, ha comandi di tipo avanti, destra, sinistra, indietro e stop. Con lo stesso paradigma dove questi comandi vengono illuminati in modo casuale e intermittente, si riconosce quale impulso luminoso suscita il P300 e la carrozzina si muove automaticamente in quella direzione. Nello stesso studio viene evidenziato che, concentrarsi continuamente su comandi di tipo direzionali affatica molto l'utente, pertanto, aggiungere degli obiettivi predefiniti per ridurre gli sforzi al minimo, è preferibile.

## 2.4 Error-related Potential

Negli studi appena citati si parla di carrozzine a guida autonoma dove il percorso è pianificato sulla base di una posizione dello spazio da raggiungere oppure dall'indicazione di una direzione, il percorso che sarà fatto sarà dettato dalle conoscenze dello spazio congiuntamente alle continue rilevazioni effettuate dai sensori. Nell'ambiente reale in realtà non sempre si hanno informazioni

complete dell'ambiente, nonché tali sensori non sempre sono in grado di riconoscere tutti gli ostacoli presenti lungo il percorso pianificato (in particolare buche nella pavimentazione, piccoli oggetti e gradini possono sfuggire alle rilevazioni dei sensori).

Ferracuti et al. in [7] affrontano la problematica di errori percepiti dall'utente nella traiettoria degli "Assistive Robots employed to support the mobility of impaired users" pubblicato il 3 Maggio 2021, si fa presente che i sensori non sempre sono sufficienti per garantire la sicurezza dell'utente e l'inserimento di quest'ultimo nel sistema di controllo (tecnica anche detta "human-in-the-loop approach") può avvenire osservando dei particolari ERP generati quando l'utente percepisce un pericolo o un errore. Tali ERP prendono il nome di "Error-Related Potential (Errps)" e sono rilevati dalla EEG segnalando così l'avvenuta rilevazione di errori da parte dell'utente.

In Figura 4 è possibile osservare una forma d'onda di un Errp. A sinistra distinguiamo l'elettrodo che ha rilevato il segnale. Vari studi dimostrano come gli Errp si rilevano maggiormente nella zona parietale del capo.

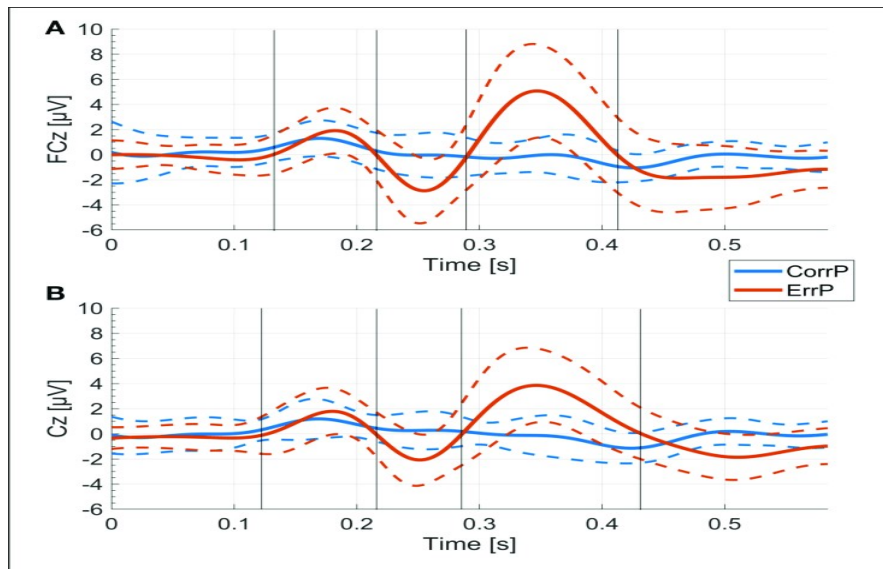


Figura 4 - Tipiche forme del segnale Errp rilevate dagli elettrodi Cz e FCz (in rosso) [8]

Come si può osservare dalla figura 4 gli Errp sono composti da un piccolo picco positivo che precede un picco negativo entro i 200ms dal riconoscimento dell'errore da parte dell'utente (il picco negativo prende il nome di "Error-related negativity, o ERN o Ne") e da un picco positivo con una latenza dal riconoscimento dell'errore da 200ms a 500ms che prenderà il nome di "Error-related positivity, o Pe".

## 2.5 Steady-State Visually Evoked Potential

Abbiamo precedentemente introdotto delle risposte cerebrali evocate, quali ERP e Errp, dagli stimoli visivi, tuttavia c'è un terzo tipo di segnale noto come "Steady-State Visually Evoked Potential" (SSVEP). Il SSVEP è prodotto come risposta alla frequenza di stimoli visivi periodici oscillanti che fungono da segnali di trigger per il loro rilevamento da parte dell'EEG. Questo segnale viene rilevato nella corteccia visiva e presenta armoniche che sono multipli della frequenza del segnale di trigger, confermando così la coerenza tra segnale generato e stimolazione visiva (Figura 5).

Duan in [9] applica una BCI multi-modale combinata con Motor-imagery-BCI e il SSVEP per utilizzare il battito delle ciglia come trigger per lo switch del controllo di un quadcopter a due gradi di libertà (Degree-of-freedom o "DOF").

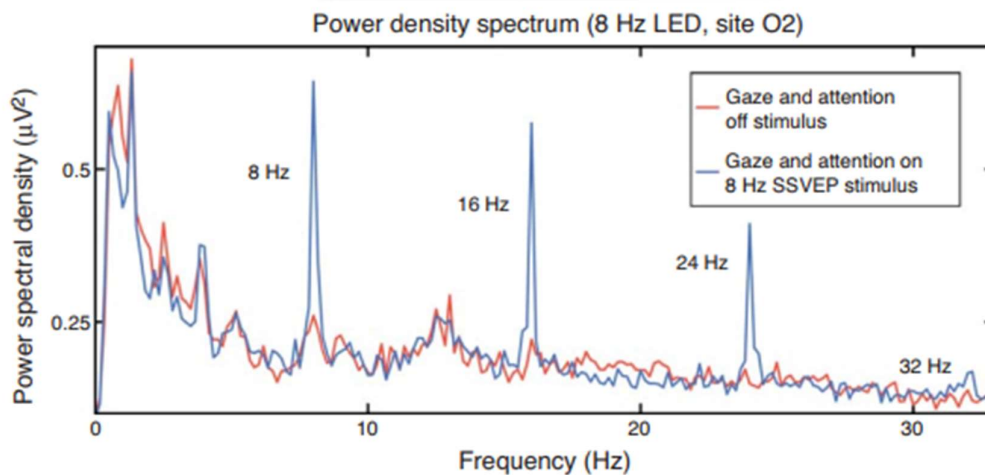


Figura 5 - Rappresentazione degli SSVEP stimolati da 3 "Red box" lampeggianti con frequenze diverse. Il grafo in blue rappresenta lo SSVEP stimolato dalle red box lampeggianti con una frequenza di 8Hz, 16Hz, 24Hz e 32Hz, il grafo in rosso invece rappresenta lo SSVEP stimolato quando l'utente non sta guardando nessuna di tali red box. [10]

## Capitolo 3. L'approccio proposto

L'attività di tesi proposta si pone come obiettivo il controllo di un drone impiegando la BCI come strumento di comando senza impiegare ulteriori controllers. Per realizzare ciò sono stati impiegati gli strumenti di OpenViBE e MATLAB & SIMULINK.

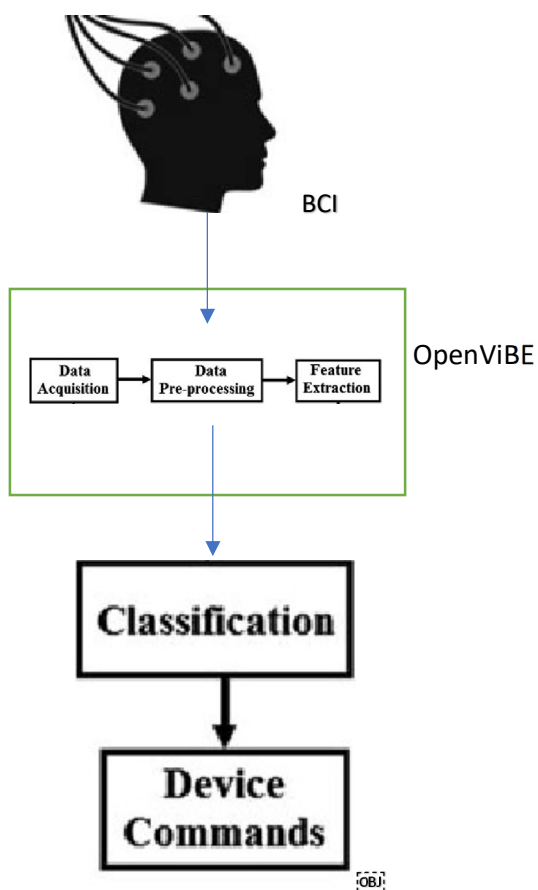


Figura 6 – Flowchart attività proposta

### 3.1 La BCI

Nell'attività proposta è stata impiegata una BCI non invasiva, insieme ad un amplificatore "g.Tec Mobilab Plus amplifiers" e di un "AntiCAP System" per l'acquisizione del segnale, inoltre viene rispettato il sistema internazionale 10/20 impiegando gli elettrodi Fz, Cz, CPz, Oz, PO7, PO8, POz, Iz. In Figura 7 vengono mostrate le mappature degli elettrodi di una classica BCI e quella di una che rispetta il sistema 10/20.



strumenti necessari per l'elaborazione di segnali ed immagini, sistemi di controllo, comunicazioni wireless, finanza computazionale, robotica, deep learning, AI e molto altro [16].

SIMULINK è un software di simulazione con diagrammi a blocchi che consente di studiare sistemi complessi di ampia gamma, a partire dai sistemi multi-frequenza, analogici, digitali e a segnale misto. I prodotti aggiuntivi di SIMULINK forniscono ulteriori funzionalità: dalla progettazione model-based, definizione dell'architettura del sistema, modellazione sistemi fisici tenendo conto di latenze e altri effetti di rete, valutazione efficienza del progetto attraverso strumenti specifici per i controlli, l'elaborazione dei segnali e applicazioni di comunicazione [17].

Nell'attività proposta è stato impiegato il SIMULINK come ambiente di simulazione per il drone da pilotare con la BCI, il che include una rappresentazione mediante schema a blocchi della fisica del drone, del suo sistema di controllo e la traduzione dei dati comunicati dall'OpenViBE (attraverso il TCP/IP che verrà descritto a breve) in comandi per il movimento del drone.

### 3.4 Cos'è il protocollo TCP/IP

Due persone possono comunicare tra loro se stabiliscono di utilizzare un linguaggio comune, un linguaggio altro non è che una serie di regole utili alla comunicazione. Il "Transmission Control Protocol/Internet Protocol (TCP/IP)" svolge la medesima funzione nella comunicazione tra due computer, nello specifico esso è un insieme di regole che definiscono come due computer si indirizzano reciprocamente e come si scambiano dati tra loro" [18].

### 3.5 Modello di simulazione del drone

L'attività proposta si basa sull'utilizzo di un drone nell'ambiente di simulazione di SIMULINK, l'intero sistema è approssimato come un unico corpo rigido. Si considerano quindi due terne di riferimento, denominate  $R_E$  e  $R_B$ . La terna  $R_E$  è considerata earth-fixed, e si assume sia inerziale, mentre la terna  $R_B$  è solidale al corpo (body fixed), centrata al centro di massa del frame (cioè, del corpo centrale).

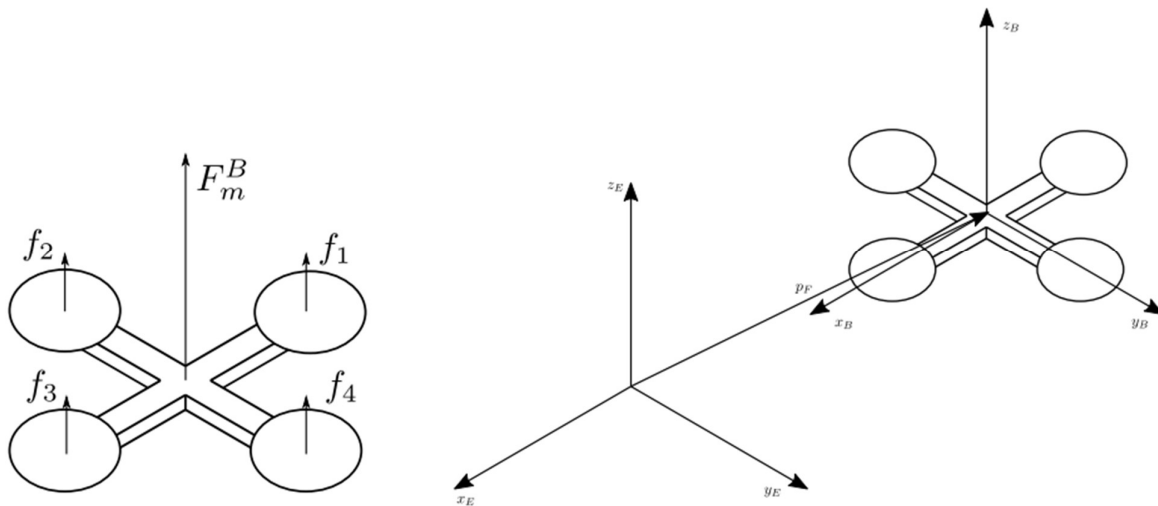


Figura 8 - Terne di riferimento del drone

### 3.5.1 Parametri

Si denota con  $m$  la massa totale del corpo, con  $k_t$  il coefficiente di attrito lineare, con  $k_r$  il coefficiente di attrito angolare, con  $g$  l'accelerazione gravitazionale, con  $J = \text{diag}(J_x, J_y, J_z)$  il tensore di inerzia. Le grandezze sono riportate nella tabella sottostante, dove tutte le unità di misura sono espresse nel sistema internazionale.

Parametro	Valore	Unità di misura
Massa totale ( $m$ )	3.95	[kg]
Coeff. di attrito lineare ( $k_t$ )	0	[N · s/m]
Coeff. di attrito angolare ( $k_r$ )	0	[N · s · m]
Inerzia lungo $x_B$ ( $J_x$ )	0.363	[kg · m <sup>2</sup> ]
Inerzia lungo $y_B$ ( $J_y$ )	0.363	[kg · m <sup>2</sup> ]
Inerzia lungo $z_B$ ( $J_z$ )	0.651	[kg · m <sup>2</sup> ]
Accelerazione gravitazionale ( $g$ )	9.81	[m/s <sup>2</sup> ]
Lunghezza braccio ( $l$ )	0.45	[m]
Coefficiente di lift ( $c_L$ )	$3.13 \cdot 10^{-5}$	[N · s <sup>2</sup> ]
Coefficiente di drag ( $c_D$ )	$7.5 \cdot 10^{-7}$	[N · m · s <sup>2</sup> ]

### 3.5.2 Variabili del drone

Si denomina con  $p_F = \text{col}(x_F, y_F, z_F)$  la posizione del frame rispetto alla terna  $R_E$ , con  $\eta = \text{col}(\phi, \theta, \psi)$  il vettore contenente gli angoli di roll, pitch e yaw che esprimono la rotazione che  $R_B$  ha rispetto a  $R_E$ , con  $\omega = \text{col}(p, q, r)$  la velocità angolare del corpo. Eventuali forze esterne (espresse in  $R_E$ ) sono denominate con  $F_e^E$ .

### 3.5.3 Ingressi di controllo

Si assume di poter controllare direttamente le forze sprigionate dai motori. Esse sono dunque ingressi di controllo, raggruppati nel vettore  $u = \text{col}(u_1, u_2, u_3, u_4)$ . Denominando con  $F_m^B$  e  $M_m^B$  la forza totale ed il momento totale agenti sul frame e dovuti ai motori (espressa in  $R_B$ ), si considerano le equazioni:

$$\begin{aligned} F_m^B &= F_1 u \\ M_m^B &= F_2 u \quad (1.1) \end{aligned}$$

dove:

$$F_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad F_2 = \begin{bmatrix} 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ -\frac{c_D}{c_L} & \frac{c_D}{c_L} & -\frac{c_D}{c_L} & \frac{c_D}{c_L} \end{bmatrix} \quad (1.2)$$

I parametri  $c_L$  e  $c_D$  sono detti coefficiente di lift (o thrust) e coefficiente di drag, mentre  $l$  è la distanza tra il centro di massa del sistema ed i motori, detto braccio.

### 3.5.4 Vettori canonici e matrici trasposte

I vettori della base canonica di  $\mathbb{R}^3$  sono indicati con  $e_1, e_2, e_3$ , ovvero

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (1.3)$$

Se  $A$  è una matrice/vettore, allora  $A^T$  denota la sua trasposta.

### 3.5.5 Modello conclusivo

Sotto la nomenclatura descritta, il modello del quadrotor che si adotta è

$$\begin{aligned} m\ddot{p}_F &= -k_t \dot{p}_F - mg e_3 + R_B^E F_m^B + F_e^E \\ J\dot{\omega} &= -k_r \omega - \omega \times J\omega + M_m^B \\ \dot{\eta} &= T(\eta)\omega \quad (1.4) \end{aligned}$$

dove

$$R_B^E = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\psi s_\theta - c_\varphi s_\psi & s_\psi s_\varphi + c_\varphi c_\psi c_\theta \\ c_\theta s_\psi & c_\varphi c_\psi + s_\varphi s_\psi s_\theta & c_\varphi s_\psi s_\theta \\ -s_\theta & c_\theta s_\varphi & c_\varphi c_\theta \end{bmatrix}$$



$$T(\eta) = \begin{bmatrix} 1 & s_\varphi t_\theta & c_\varphi t_\theta \\ 0 & c_\varphi & -s_\varphi \\ 0 & \frac{s_\varphi}{c_\theta} & \frac{c_\varphi}{c_\theta} \end{bmatrix} \quad (1.5)$$

Le abbreviazioni  $\cos(\cdot) = c(\cdot)$ ,  $\sin(\cdot) = s(\cdot)$  e  $\tan(\cdot) = t(\cdot)$  sono considerate per compattezza.

## Capitolo 4. Controllore

### 4.1 Inner loop

L'inner loop prende in ingresso i riferimenti in altezza  $z_{F_r}(t)$  e di assetto  $\eta(t) = \text{col}(\varphi_r(t), \theta_r(t), \psi_r(t))$ , e genera di conseguenza l'ingresso di controllo  $u(t)$  affinché le variabili di stato  $z_F, \phi, \theta, \psi$  seguano i riferimenti (con ovvia nomenclatura).

Data la struttura di  $F_1$ , si ha

$$F_1 u = \begin{bmatrix} 0 \\ 0 \\ f_z \end{bmatrix} = f_z e_3 \quad (1.6)$$

dove  $f_z$  rappresenta l'espressione scalare della forza totale che i motori esercitano sul drone, ovvero

$$f_z = f_1 + f_2 + f_3 + f_4 = [1 \quad 1 \quad 1 \quad 1]u \quad (1.7)$$

Considerando che  $z_F = e_3^T p_F$  (ovvero, la sua terza componente), il modello del quadrotor ristretto all'inner loop è

$$\begin{aligned} m\ddot{z}_F &= -k_t \dot{z}_F - mg + r_{33} f_z + e_3^T e_e^E \\ J\dot{\omega} &= -k_r \omega - \omega \times J\omega + M_m^B \\ \dot{\eta} &= T(\eta)\omega \quad (1.8) \end{aligned}$$

dove  $r_{33} = e_3^T R_B^E e_3$  (ovvero, la componente (3, 3) di  $R_B^E$ ). La legge di controllo che si prende in considerazione è quindi

$$u = F^{-1}A^{-1}\left(\begin{bmatrix} -b_z \\ -b_\eta \end{bmatrix} + \begin{bmatrix} v_z \\ v_\eta \end{bmatrix}\right); \quad (1.9)$$

dove

$$b_z = -\frac{k_t}{m}\dot{z}_F - g + \frac{1}{m}e_3^T F_e^E \quad (1.10)$$

$$b_z = T(\eta)\omega - T(\eta)J^{-1}\omega - T(\eta)J^{-1}(\omega \times J\omega) \quad (1.11)$$

$$A^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & T(\eta)J^{-1} \end{bmatrix} \quad (1.12)$$

$$F^{-1} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ -\frac{c_D}{c_L} & \frac{c_D}{c_L} & -\frac{c_D}{c_L} & \frac{c_D}{c_L} \end{bmatrix} \quad (1.13)$$

Le quantità  $v_z(t) \in \mathbb{R}$  e  $v_\eta(t) \in \mathbb{R}^3$  sono ingressi ausiliari, da definire.

## 4.2 Outer loop

L'outer loop (anello esterno) viene progettato sotto l'approssimazione per piccoli angoli, ovvero si considerano

$\varphi(t) \approx 0,$	(1.16)
$\theta(t) \approx 0.$	(1.17)

Geometricamente parlando, queste due condizioni sono verificate quando il drone è inclinato di pochi gradi rispetto al suo piano  $(x_B, y_B)$  (appunto, piccoli angoli). Questa condizione è inoltre verificata in hovering (drone che è capace di restare fermo, orizzontalmente, in una posizione dello spazio), per cui è a volte detto che il drone è considerato vicino alla condizione di hovering (near hovering condition).

Si ricorda che per un piccolo  $x \approx 0$ , si hanno le approssimazioni (suggerite dallo sviluppo in serie di Taylor)  $\sin(x) \approx x$  e  $\cos(x) \approx 1$ . Segue quindi che

$\cos(\varphi(t)) \approx 1,$	$\sin(\varphi(t)) \approx \varphi(t),$	(1.18)
$\cos(\theta(t)) \approx 1,$	$\sin(\vartheta(t)) \approx \theta(t).$	(1.19)

Conseguenza diretta dell'approssimazione è

$$R_B^E F_m^B = \begin{bmatrix} \star & \star & s_\varphi s_\psi + c_\varphi c_\psi s_\theta \\ \star & \star & c_\varphi s_\psi s_\theta - c_\psi s_\varphi \theta \\ \star & \star & \star \end{bmatrix} \quad (1.20)$$

$$= f_z \begin{bmatrix} s_\varphi s_\psi + c_\varphi c_\psi s_\theta \\ c_\varphi s_\psi s_\theta - c_\psi s_\varphi \theta \\ \star \end{bmatrix} \quad (1.21)$$

$$\approx f_z \begin{bmatrix} \varphi s_\psi + c_\psi \theta \\ s_\psi \theta - c_\psi \varphi \\ \star \end{bmatrix} \quad (1.22)$$

$$\approx f_z \begin{bmatrix} s_\psi & c_\psi \\ -c_\psi & s_\psi \\ \star & \star \end{bmatrix} \begin{bmatrix} \varphi \\ \theta \end{bmatrix} \quad (1.23)$$

L'outer loop si progetta sulle equazioni dinamiche di  $x_F(t)$  e  $y_F(t)$ , che si ricordano essere le prime due componenti di  $p_F(t)$ . Tali equazioni sono quindi

$$m \begin{bmatrix} \dot{x}_F \\ \dot{y}_F \end{bmatrix} = -k_t \begin{bmatrix} \dot{x}_F \\ \dot{y}_F \end{bmatrix} + f_z \begin{bmatrix} s_\psi & c_\psi \\ -c_\psi & s_\psi \end{bmatrix} \begin{bmatrix} \varphi \\ \theta \end{bmatrix} + \begin{bmatrix} F_{e,x} \\ F_{e,y} \end{bmatrix} \quad (1.24)$$

dove si indica  $F_e^E = \text{col}(F_{e,x}, F_{e,y}, F_{e,z})$ . Supponendo di poter direttamente controllare  $\phi(t)$  e  $\theta(t)$  attraverso l'inner loop, si progetta una legge prendendo questi come ingressi. La legge presa in considerazione è quindi

$$\begin{bmatrix} \varphi \\ \theta \end{bmatrix} = \frac{m}{f_z} \begin{bmatrix} s_\psi & c_\psi \\ -c_\psi & s_\psi \end{bmatrix}^{-1} \left( + \frac{k_t}{m} \begin{bmatrix} \dot{x}_F \\ \dot{y}_F \end{bmatrix} - \frac{1}{m} \begin{bmatrix} F_{e,x} \\ F_{e,y} \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right) \quad (1.25)$$

dove  $v_x(t), v_y(t) \in \mathbb{R}$  sono ingressi ausiliari. Nella realtà non siamo in grado di comandare  $\phi(t), \theta(t)$  direttamente, quindi i valori ottenuti sono in realtà dei riferimenti per inner loop. Inoltre non conosciamo i valori delle componenti del disturbo esterno  $F_{e,x}(t), F_{e,y}(t)$ , quindi dovremo accontentarci di stime, ovvero  $\check{F}_{e,x}(t), \check{F}_{e,y}(t)$ . La legge implementabile è quindi

$$\begin{bmatrix} \varphi \\ \theta \end{bmatrix} = \frac{m}{f_z} \begin{bmatrix} s_\psi & c_\psi \\ -c_\psi & s_\psi \end{bmatrix}^{-1} \left( + \frac{k_t}{m} \begin{bmatrix} \dot{x}_F \\ \dot{y}_F \end{bmatrix} - \frac{1}{m} \begin{bmatrix} \check{F}_{e,x} \\ \check{F}_{e,y} \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right) \quad (1.26)$$

## Capitolo 5. Training della BCI & testing

### 5.1 Il training della BCI

Nella sezione “2”, in particolare dalla “2.3” alla “2.5”, sono stati descritti i segnali ERP menzionando gli SSVEP, il training della BCI significa addestrare la BCI a riconoscere determinati SSVEP, i quali saranno associati a istruzioni per il pilotaggio del drone. Tale attività è stata trattata attraverso la procedura “SSVEP: Steady-State Visual-Evoked Potentials” disponibile presso il sito di OpenViBE [19], consiste nella riproduzione video di 3 Red Boxes su sfondo nero e un indicatore giallo che indica su quale elemento del video l’utente deve concentrarsi per la durata di stimolazione (Figura 9).

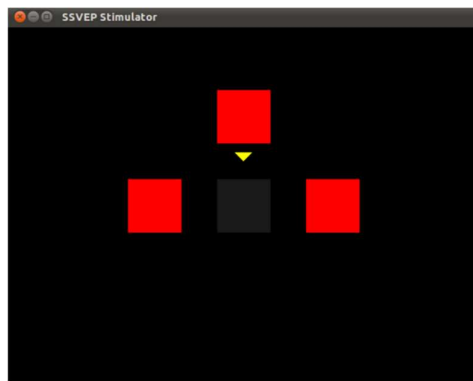


Figura 9 - Training BCI [19]

L’addestramento avviene nella seguente modo:

Gli elementi sono stati indicati con “0” per indicare il quadrato in nero attualmente evidenziato dall’indicatore in giallo in Figura 9 e ad esso è associata la volontà di non comunicare istruzioni, le red boxes invece sono indicate rispettivamente con “1,2,3” da sinistra a destra per un totale di 4 elementi

L’addestramento procede con le misure standard previste in [19], vale a dire una durata di stimolazione di 7 secondi in cui il target (0,1,2,3) lampeggia con la propria frequenza seguita da una durata di pausa di 4 secondi in cui i target smettono di lampeggiare, prima di ogni stimolazione il target verrà indicato con un cursore giallo e l’utente dovrà rivolgere su di esso la propria attenzione per tutta la durata di stimolazione. Ogni target verrà indicato diverse volte;

L’esperimento è pre-configurato per 3 frequenze di flickering e ognuna verrà stimolata per 8 volte con un ritardo di 1 secondo tra l’apparizione del cursore giallo e l’effettivo inizio di lampeggiamento del target. Alla fine della procedura verrà creato un file contenente le EEG rilevate (di default) nella cartella “OPENVIBE\_FOLDER/dist/share/opencv-scenarios/signals” folder.

## 5.2 Testing

Il testing dell'attività proposta è composta da due fasi, la prima coinvolge l'OpenViBE e la seconda il SIMULINK e si pone come obiettivo il movimento del drone. Il drone usato in simulazione è basato sul drone "Inserire\_modello\_drone\_e\_dettagli\_dello\_stesso".

### Fase 1:

Attraverso la procedura descritta in [19] viene utilizzato lo scenario "ssvep-bci-5-online-shooter" come interfaccia grafica per consentire all'utente di muoversi, in tale scenario viene rappresentato un velivolo composto da red boxes (Figura 10) dove le red boxes 1 e 3 sono le ali e vengono utilizzate per ruotare il velivolo stesso, mentre la red boxes 2 nello scenario serve per sparare. Attraverso l'aggiunta di un cursore circolare bianco che compare lungo i bordi l'obiettivo di questo scenario è quello di ruotare il velivolo e colpire tale cursore. Nell'attività proposta rivolgendo la propria attenzione ai 3 target viene generato un SSVEP specifico, ad esso viene associato un codice numerico (target 0->?, target 1->33025 , target 2->33024, target 3->33026) e, attraverso l'aggiunta di un blocco "TCP/IP Writer" questo codice viene inviato al SIMULINK dove fungerà da istruzione per il pilotaggio del drone.

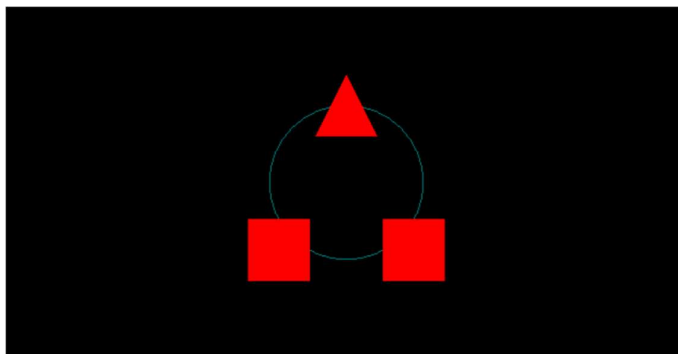


Figura 10 - Velivolo utilizzato nello scenario "ssvep-bci-5-online-shooter" [19]

### Fase 2:

Per mezzo del "TCP/IP Client Receive" il codice numerico che indica la SSVEP generata e quindi il target sul quale l'utente sta mantenendo la propria attenzione viene ricevuto dal software di SIMULINK dove è presente lo schema a blocchi per il controllo del drone (Figura 11).

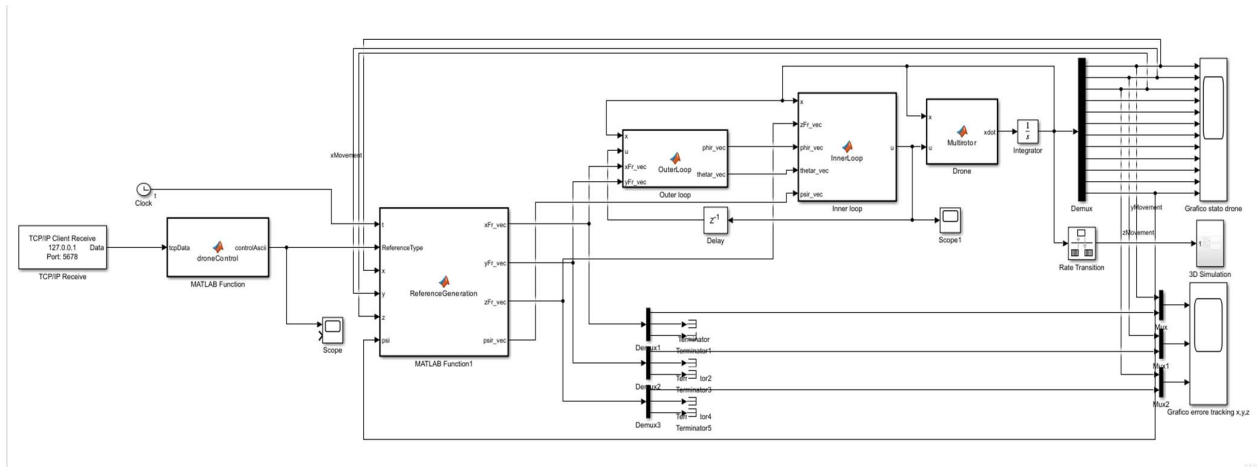


Figura 11 - Schema a blocchi simulink

Possiamo dividere tale schema a blocchi in due parti a cascata:

La prima parte si occupa della ricezione del codice numerico attraverso il blocco "TCP/IP Client Receive" setato sulla porta "5678" e indirizzo IP della "localhost" o "127.0.0.1" per stabilire che la comunicazione avviene tra applicazioni presenti sul medesimo PC (tale porta e indirizzo IP sono comuni al TCP/IP Writer con il quale sé in comunicazione, ossia quello presente in OpenViBE), tale codice viene poi consegnato al blocco "droneControl" che gli assegna uno specifico codice ASCII (?->49, 33024->119, 33025->97, 33026->100). Tale assegnazione è necessaria dal momento che nel successivo blocco "ReferenceGeneration" vengono presi come dati di input del "ReferenceType" valori ASCII;

La seconda parte consiste nella cascata di blocchi del ReferenceGeneration, OuterLoop, InnerLoop e multitrotor:

Per compiere il proprio movimento il drone utilizza un assegnamento di coordinate nello spazio x,y,z dove la sua posizione attuale coincide con l'origine, l'assegnamento di coordinate è implementato nel blocco del "ReferenceGeneration" attraverso un incremento costante nella direzione desiderata ripetuta ogni volta che viene focalizzata l'attenzione su di uno dei target della fig.6. Tale direzione è determinata dal codice ASCII ricevuto in input dal blocco "droneControl", in particolare al 119 corrispondente al target 2 viene associata l'istruzione "Vai avanti", al 97 corrispondente al target 1 viene associata l'istruzione "Muoviti a sinistra", al 100 corrispondente al target 3 viene associata l'istruzione "Muoviti a destra", all'ultima direzione prevista è prevista con il codice ASCII 49 corrispondente al target 0 e viene associata l'istruzione "Non sto comunicando istruzioni, continua il movimento fino al raggiungimento delle coordinate stabilite";

Dal punto di vista di controllo il volo del drone è un sistema instabile e per tanto necessita di un'appropriata azione di controllo. Tale azione di controllo viene realizzata attraverso un controllore previsto nello schema a blocchi attraverso la coppia a cascata di "OuterLoop e InnerLoop". Il controllore prende in input le coordinate calcolate dal blocco del ReferenceGeneration e l'attuale posizione del drone stesso e calcola la velocità di

rotazione delle eliche necessaria al compimento del movimento desiderato. Tale velocità prende il nome di “Vettore di spinta (Thrust Vector)”.

Le equazioni differenziali del moto del drone sono presenti nel blocco del Multirotor, esso riceve in input il vettore di stato del drone e il vettore di spinta e il suo compito è quello di calcolare le velocità e posizioni lineari e angolari che verranno poi esplicitate nel successivo blocco scope «Grafico stato drone». Di fatto l'insieme di queste variabili appena nominate formeranno il vettore di stato del drone.



## Capitolo 6. Conclusioni

In questa tesi è stato trattato un metodo per controllare un veicolo autonomo tramite BCI, si è proposto un approccio per il controllo in tempo reale del veicolo (nel caso proposto un drone). Dall'esperimento svolto e quelli usati come confronto emerge come ogni categoria di veicolo possa aver bisogno di un'azione ben specifica dettata dalla fisica dello stesso e dall'ambiente in cui viene impiegato (gli assistive robots vengono usati diversamente dai droni che a loro volta sono utilizzati diversamente dalle sedie a rotelle pilotate con la BCI, dette BCW).

Una delle problematiche rimaste da affrontare prima di poter installare tale software su di un dispositivo mobile è l'implementazione di un algoritmo per lo studio dei segnali di emergenza dell'utente. Il lavoro svolto voleva dimostrare la fattibilità dell'impiego della BCI come strumento di controllo per il movimento di un veicolo autonomo, per sua natura non tiene conto di vincoli da evitare e ostacoli da aggirare offerti da un ambiente reale.

I risultati di questo studio dimostrano la fattibilità del pilotaggio di droni tramite una BCI basata su EEG. Tuttavia, ci sono sfide significative da affrontare, come la riduzione del rumore nei segnali EEG e l'ottimizzazione degli algoritmi di classificazione per migliorare l'accuratezza e la velocità di risposta.

## Capitolo 7. Lavori futuri

Questo lavoro di tesi rappresenta un passo avanti nella ricerca sull'interfacciamento uomo-macchina e nelle applicazioni pratiche nel campo dei droni. L'uso delle BCIs per il controllo dei droni apre la strada a numerose applicazioni, comprese operazioni di soccorso, sorveglianza e molto altro. Mentre sono necessari ulteriori sviluppi per migliorare l'efficienza e la precisione di tali sistemi, le prospettive future sono promettenti. In sintesi, questa tesi dimostra come la tecnologia BCI abbia il potenziale per rivoluzionare il modo in cui interagiamo con i droni e aprire nuove opportunità per l'automazione e il controllo intuitivo dei veicoli aerei senza equipaggio.

A tal riguardo, si desidera evidenziare che l'approccio proposto è stato impiegato in un ambiente di simulazione non tenendo conto dei vincoli e problematiche presenti nell'ambiente reale. Pertanto, prima di poter essere realmente installato su di un veicolo autonomo per l'assistenza dei disabili necessità dell'implementazione dei segnali di errore menzionati nel capitolo "2.3)" con l'Errp, il trattamento congiunto dei dati captati dai sensori menzionati e le pratiche del "Human-in-the-loop approach" in [7] così da poter comunicare istruzioni prioritarie, come un arresto improvviso, sia che si decida di mantenere l'attuale metodo di movimento per incremento sia che si impieghi in un metodo di tracciamento di un percorso utile al raggiungimento di una posizione specifica nello spazio.

## Ringraziamenti

Desidero ringraziare il mio relatore e correlatore, i professori Andrea Monteriù e Alessandro Freddi, i dottorandi Vella Francesco e Omer Karameldeen Ibrahim Mohamed per avermi dato l'opportunità di cimentarmi in quest'attività e per il sostegno datomi per la sua realizzazione. Ringrazio naturalmente anche tutti coloro che mi hanno supportato nel mio percorso, a partire dai miei genitori, i miei fratelli, parenti, amici e compagni di corso che più di chiunque altro, per natura dell'università, si sono dovuti confrontare con me nonostante il covid e le distanze reali che ci hanno separato.

## Bibliografia

- [1] Leuthardt EC, Moran DW, Mullen TR. "Defining Surgical Terminology and Risk for Brain Computer Interface Technologies". *Front Neurosci*. Ch. "Defining a Brain Computer Interface" 2021 Mar 26;15:599549. doi: 10.3389/fnins.2021.599549. PMID: 33867912; PMCID: PMC8044752.
- [2] <https://www.hsantalucia.it/laboratorio-immagini-neuroeletttriche-interfacce-cervello-computer>
- [3] <https://www.ingegneriabiomedica.org/news/biotech-support/brain-computer-interfaces-cervello-si-mette-moto/>
- [4] TY - JOUR. Eva M. Hammer, S. Halder et al. " Psychological Predictors of Visual and Auditory P300 Brain-Computer Interface Performance". DO - 10.3389/fnins.2018.00307 JO - *Frontiers in Neuroscience*.
- [5] B. Rebsamen, C. Guan, H. Zhang, C. Whang, C. Teo, M. H. Ang Jr. e E. Burdet, "A Brain Controlled Wheelchair to Navigate in Familiar Environments" , *IEEE Transaction on Neural Systems and Rehabilitation Engineering*, Vol. 18 no.6, Dicembre 2010. doi: 10.1109/TNSRE.2010.2049862.
- [6] A.C. Lopes, G. Pires, L. Vaz, U. Nunes, "Wheelchair Navigation Assisted by Human-Machine Shared-control and a P300-based Brain Computer Interface", 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 25-30 Settembre, 2011, San Francisco, CA, USA, doi: 10.1109/IROS.2011.6094748.
- [7] Ferracuti, F., Freddi, A., Iarlori, S. et al. "Augmenting robot intelligence via EEG signals to avoid trajectory planning mistakes of a smart wheelchair". *J Ambient Intell Human Comput* 14, 223–235 (2023). doi: 10.1007/s12652-021-03286-7.
- [8] P. Keyl , M. Schneiders et al. "Differences in Characteristics of Error-Related Potentials Between Individuals With Spinal Cord Injury and Age- and Sex-Matched Able-Bodied Controls". *Frontiers in Neurology* 9:1192, January 2019. DOI:10.3389/fneur.2018.01192.
- [9] X. Duan, S. Xie, X. Xie, Y. Meng, and Z. Xu, "Quadcopter Flight Control Using a Non-invasive Multi-Modal Brain Computer Interface," *Front Neurorobot*, vol. 13, p. 23, 2019
- [10] J. Wolpaw E. W. Wolpaw, "BCI principle and practice". Oxford University Press. 2012. Ch.-14, p.242.
- [11] Rene S. Skukies et al., "EEG-Data Lifecycle". Ch.-2. Aug. 21, 2020.
- [12] O. Attallah, "An Effective Mental Stress State Detection and Evaluation System Using Minimum Number of Frontal Brain Electrodes" by "Department of Electronics and Communications, College of Engineering and Technology, Arab Academy for Science,

- Technology and Maritime Transport, Alexandria 1029, Egypt Diagnostics 2020, 10(5), 292”, “  
<https://doi.org/10.3390/diagnostics10050292> “. Ch.-2. 9 May 2020.
- [13] “OpenViBE Software for Brain Computer Interfaces and Real Time Neurosciences “. Ch. -1 “What is OpenViBE?”. <http://openvibe.inria.fr/discover/> ”
- [14] Marco Congedo, Matthieu Goyat, Nicolas Tarrin, Gelu Ionescu, Léo Varnet, et al.. "Brain Invaders": un prototipo di un videogioco open source basato su P300 che funziona con la piattaforma OpenViBE. BCI 2011 - 5a conferenza internazionale sull'interfaccia cervello-computer, settembre 2011, Graz, Austria. pp.280-283. (hal-00641412).
- [15] Fabien Lotte, Josef Faller, Christoph Guger et al. “Combining BCI with Virtual Reality: Towards New Applications and Improved BCI”, Towards Practical Brain-Computer Interfaces, 2013. ISBN : 978-3-642-29745-8.
- [16] ”Mathworks”, video introduttivo al software di MATLAB.  
 “<https://it.mathworks.com/products/matlab.html>”.
- [17] ”Mathworks”, video introduttivo al software di SIMULINK.  
 “<https://it.mathworks.com/products/simulink.html>”.
- [18] Andew G. Blank “TCP/IP Foundation. Technology Fundamental for IT Success” Ch.-1, p. 4.  
<https://books.google.it/books?id=OB-k4OliTxIC&lpq=PR4&ots=uxPKfSXIN3&dq=what%20is%20tcp%2Fip&hl=it&pg=PA1#v=onepage&q=what%20is%20tcp/ip&f=false> .
- [19] “SSVEP: Steady-State Visual-Evoked Potentials“ <http://openvibe.inria.fr/steady-state-visual-evoked-potentials/#2nd+step:+Training>