

# Esercitazione con gli SCORBOT

Il laboratorio è equipaggiato di diversi manipolatori di tipo SCORBOT. Lo scopo dell'esercitazione è quello di verificare la correttezza delle formule della cinematica diretta e inversa studiate a lezione su un robot reale.

Al momento nel laboratorio sono disponibili quattro postazioni di due tipi diversi: tre di tipo più datato (in cui si fa uso di un'unità di potenza dedicata) e una più recente (in cui si utilizza una scheda Arduino Mega).

## 1 Istruzioni per l'utilizzo dei due tipi di postazione

### 1.1 Postazione con unità di potenza dedicata

I passi da fare in questo caso sono i seguenti:

1. Accendere il PC e l'unità di potenza (tasto sul retro)
2. Attendere la fine del boot del PC finché non appare il prompt di MS-DOS
3. Portarsi nella directory  $C : \backslash ROBOTICA \backslash ROBOT99 \backslash SCORB \backslash$  digitando:  

```
cd robot99  
cd scorb
```
4. Lanciare l'eseguibile `sbdemo.exe` digitando `sbdemo`
5. Una volta che il programma si è avviato ed è apparsa la schermata principale, in basso è riportato il menu dei comandi disponibili:
  - I tasti da 1 a 5 permettono di selezionare il giunto che si vuole muovere
  - I tasti  $+$  e  $-$  permettono di modificare lo step (in alcune postazioni conviene portarlo a 1, per default è 3)
  - I tasti shift (destra e sinistra) permettono di muovere il giunto selezionato (premerli per **breve tempo** per evitare escursioni eccessive del robot).
  - Sulla schermata principale sono riportati i passi encoder dei vari motori: il tasto F1 permette di mandare il motore del giunto selezionato al passo encoder desiderato. Con il tasto *return* si azzerano tutti i passi encoder.

### 1.2 Postazione con scheda Arduino Mega

Avviare Matlab ed eseguire l'app `hmi_scorbot` (disponibile andando sulla scheda APPS di Matlab). Selezionare, quando viene chiesto, la porta seriale a cui è stata collegata la scheda Arduino (la numero 2 in questo caso). Stabilito il collegamento (occorre qualche secondo), collegare l'alimentazione della scheda di potenza.

L'interfaccia del programma permette la movimentazione dei motori cliccando con il mouse sugli appositi campi presenti nella schermata di controllo e di leggere e azzerare i passi encoder. Si ha anche la possibilità di definire una home del robot e di tornarvi premendo con il mouse su un apposito pulsante. Il campo PWM, che può essere modificato tra 0 e 255, permette di variare l'entità dello spostamento ad ogni pressione del mouse sulle frecce.

## 2 Traccia dell'esercitazione

Qualunque sia la postazione, occorre venire equipaggiati di una calcolatrice (meglio se programmabile, meglio ancora con un PC), di un metro e della dispensa sullo SCORBOT. Qualche goniometro è invece disponibile in laboratorio. Si tratta di calcolare le formule dello SCORBOT (cinematica diretta e inversa) e si consiglia pertanto di venire all'esercitazione con le formule dello SCORBOT già implementate in un programma (per esempio in uno script Matlab) per velocizzare i calcoli.

Il primo problema da affrontare è la conversione dai passi encoder agli angoli. Infatti, i passi encoder dei vari motori non rappresentano direttamente gli angoli di giunto per due motivi. Innanzitutto occorre trovare la costante di proporzionalità tra passi encoder e angoli, diversa in generale per ogni giunto. Secondo, per via della struttura meccanica dello SCORBOT, la dipendenza di un angolo dai passi encoder è piuttosto complessa. Si riporta sotto la corrispondenza (in entrambi i versi) tra passi encoder  $p_i$ ,  $i = 0, 1, \dots, 4$  ed angoli di giunto  $\theta_i$ ,  $i = 1, 2, \dots, 5$ , in cui le costanti vanno determinate (vedasi il primo punto dell'esercitazione):

$$\begin{array}{ll}
\theta_1 = -\alpha \cdot p_0 & p_0 = -\theta_1/\alpha \\
\theta_2 = -\beta \cdot p_1 & p_1 = -\theta_2/\beta \\
\theta_3 = \beta(p_1 + p_2) & p_2 = (\theta_2 + \theta_3)/\beta \\
\theta_4 = -\beta \cdot p_2 + \gamma(p_3 - p_4) & p_3 = 0.5[(\theta_2 + \theta_3 + \theta_4)/\gamma + \theta_5/\delta] \\
\theta_5 = \delta(p_3 + p_4) & p_4 = 0.5[\theta_5/\delta - (\theta_2 + \theta_3 + \theta_4)/\gamma]
\end{array}
\Longleftrightarrow$$

Il valore di  $p_5$  è invece legato al movimento di apertura e chiusura della pinza (ottenibile pigiando rispettivamente i tasti O e C) e non rientra tra gli scopi della presente esercitazione.

## 2.1 Passi dell'esercitazione

I passi dell'esercitazione sono i seguenti:

1. Determinare innanzitutto le costanti  $\alpha$ ,  $\beta$ ,  $\gamma$  e  $\delta$  presenti nelle formule precedenti (si può far uso del goniometro ma si possono anche muovere i vari giunti di angoli noti facili da valutare, come per esempio  $\pi/2$ ). È probabile che si troverà  $\delta = \gamma$ .
2. Misurare la lunghezza dei vari link del robot (delle grandezze cioè che compaiono nella tabella di Denavit Hartenberg dello SCORBOT). I valori riportati nella dispensa potrebbero non essere precisi o variare da robot a robot.
3. Portare il robot nella configurazione in cui tutti gli angoli di giunto sono nulli (secondo la convenzione utilizzata a lezione), azzerare i passi encoder e fissare quindi anche il sistema di riferimento della base  $L_0$  (sempre in accordo con quello definito nella dispensa).
4. Muovere ora il robot a piacere e verificare se, in base alla lettura dei passi encoder a schermo, applicando le equazioni precedenti di conversione passi encoder  $\rightarrow$  angoli e quelle della cinematica diretta, le coordinate della pinza fornite dalle formule coincidono con quelle rilevabili a mano col metro. Eventualmente verificare anche l'orientamento della pinza (per esempio verificare se l'asse  $z_5$  è diretto così come ci si aspetta dalle formule).
5. Si scelga una posizione e un orientamento (in termini di angolo di beccheggio  $\beta_d$  ed eventualmente anche di angolo di rollio  $\omega_d$ ) desiderati per la pinza. Applicare le formule della cinematica inversa e quelle di conversione angoli  $\rightarrow$  passi encoder. Vedere se portando ciascun motore al passo encoder determinato, la pinza raggiunge la configurazione desiderata.