

---

## 0.1 Struttura del Sistema

Per gestire un robot sono necessari vari sottosistemi che, connessi tra loro, interagiscono avendo ognuno una funzione precisa.

Il sistema da noi utilizzato è principalmente suddiviso in tre parti:

- un calcolatore con DSP
- un pannello con convertitori e amplificatori
- il ROBOT vero e proprio.

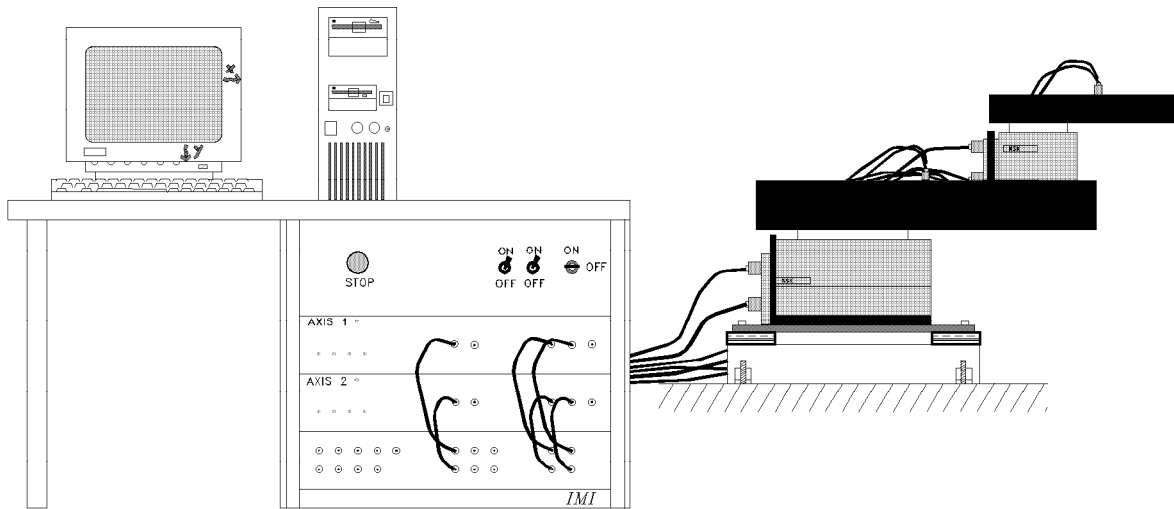


Figura 0.1. Stazione di controllo del manipolatore

Descriviamo ora i componenti dei vari sottosistemi.

### Il Calcolatore

Al calcolatore è affidata tutta la gestione del software e quindi l'interfaccia con l'utente.

Tramite il software si possono descrivere in vari modi i movimenti da far eseguire al robot, vengono generati i segnali di riferimento opportuni, si può simulare il sistema e, cosa più importante, si possono generare gli algoritmi di controllo. Questi ultimi sono scritti in linguaggio C e poi compilati con il compilatore *Texas Instruments TMS320C30*.

All'interno del calcolatore è inserita anche la scheda DSP (*Spectrum Signal Processing TMS320C30 Processor Board*) che è un po' il cuore di tutto il sistema, in quanto elaborando i segnali che riceve in ingresso e seguendo l'algoritmo di controllo genera i comandi da mandare ai motori che comandano il robot.

### Pannello Amplificatori

Questo pannello è suddiviso a sua volta in tre parti ben distinte:

- driver motore 1
- driver motore 2
- interfaccia DSP

I drivers per i motori 1 e 2 ricevono in ingresso il comando da mandare ai motori e, oltre che amplificarlo, lo modulano in modo opportuno per alimentare i motori (generano un segnale PWM trifase).

L'interfaccia per il DSP è costituita essenzialmente da convertitori A/D e D/A; essa serve per tradurre i segnali digitali provenienti dal DSP in analogici per i drivers e viceversa ritraduce i segnali analogici provenienti dai sensori per mandarli al DSP.

## II ROBOT

Il robot è un manipolatore a due giunti planare ed è quindi costituito da due motori (attuatori) e da due bracci (masse e inerzie).

Inoltre ogni giunto è dotato di un sensore (ENCODER) che permette in ogni istante di conoscere posizione e velocità del manipolatore e quindi di chiudere una retroazione per il controllo del robot.

Lo schema del sistema completo è riportato in figura 0.2.

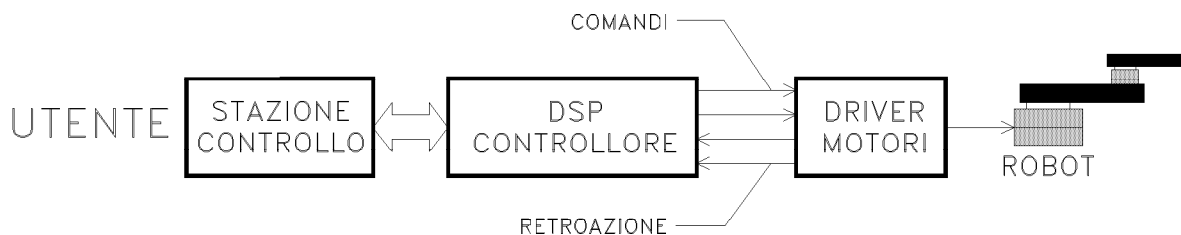


Figura 0.2. Schema funzionale del sistema

## 0.2 Struttura Hardware del sistema di controllo

Spieghiamo ora come funziona realmente tutto il sistema e in che modo interagiscono i sottosistemi che abbiamo visto.

Mentre quasi tutti i sistemi di controllo industriali hanno strutture chiuse basate su ROM ed è difficile o impossibile modificarle, questo sistema è basato su una struttura aperta e facilmente accessibile.

Durante le operazioni normali il programma di interfaccia con l'utente, le routine che gestiscono le equazioni della cinematica inversa e la pianificazione della traiettoria

(interpolazione) risiedono all'interno della memoria del computer; mentre l'algoritmo di controllo è caricato sul DSP.

Questo schema di funzionamento ottimizza il sistema, in quanto sfrutta le capacità di calcolo del microprocessore e la potenza ed efficienza nella manipolazione dei segnali I/O da parte del DSP.

La struttura della parte Hardware del sistema è rappresentata in figura 0.3 .

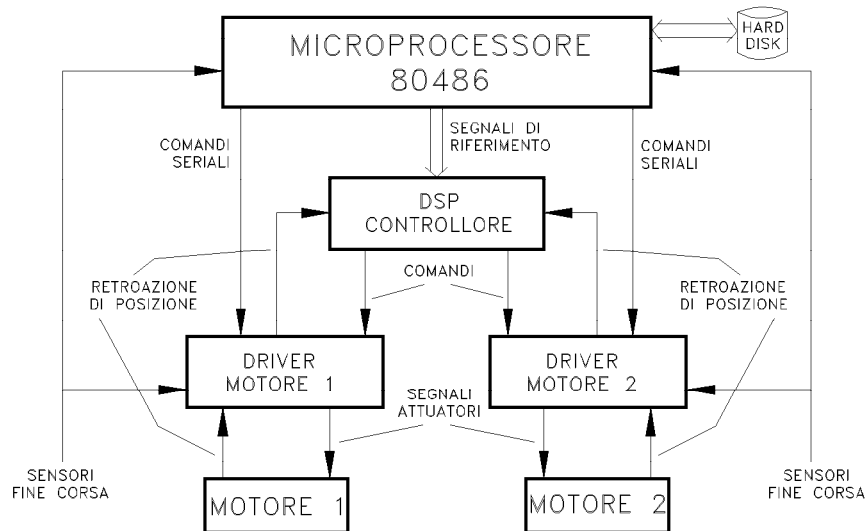


Figura 0.3. Struttura Hardware

## 0.2.1 Amplificatori dei motori

Gli amplificatori possono essere pensati come una "scatola nera" che trasforma il comportamento complesso dei motori brushless NSK in uno simile a quello di motori DC ad alta coppia e bassa velocità.

Inoltre funzionano da amplificatori di potenza in modo da portare il segnale di comando ad un livello in cui è in grado di comandare direttamente i motori.

Gli amplificatori possono anche provvedere a chiudere un anello di retroazione interno di velocità (vedi **Velocity mode**).

È possibile accedere direttamente all'ambiente di impostazione delle caratteristiche degli amplificatori attraverso la porta seriale del computer (RS232); si possono così inserire determinati filtri o imporre un certo modo di funzionamento.

### Torque Mode e Velocity Mode

Una differenza molto importante si ha tra i due modi di funzionare dei motori: Torque Mode e Velocity Mode.

Nel primo caso infatti i driver si comportano semplicemente come amplificatori di corrente e quindi i motori applicano una coppia che è proporzionale al segnale di ingresso.

Nel Velocity Mode invece gli amplificatori chiudono un anello di controllo di velocità che normalmente è un controllore di tipo proporzionale (P), ma che può essere anche fissato di tipo proporzionale-integrativo (PI) attraverso il terminale.

---

I due modi di funzionamento sono rappresentati dagli schemi a blocchi di figura 0.4

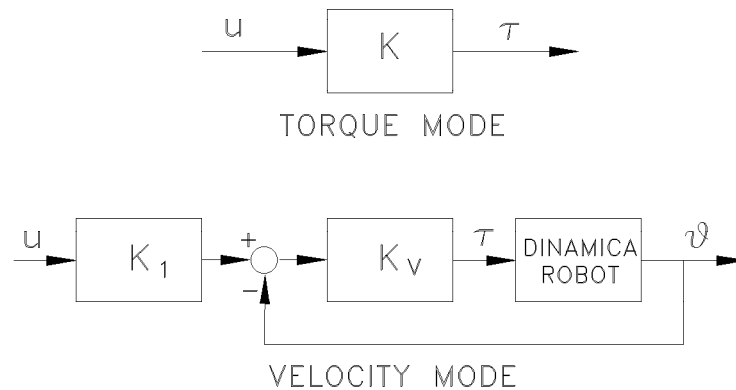


Figura 0.4. Torque Mode e Velocity Mode

L'anello di velocità è una retroazione digitale che opera con un periodo di campionamento di  $550\mu s$ .

Il guadagno dell'anello,  $K_v$ , si può variare via software per fissarlo al valore più opportuno.

Si parlerà ancora in seguito dei due modi di funzionamento degli amplificatori, soprattutto riguardo agli algoritmi di controllo che saranno necessariamente diversi per un modo e per l'altro.

### 0.2.2 DSP e interfaccia DSP

La scheda DSP è inserita in apposite sedi all'interno del corpo del calcolatore e normalmente in essa risiede l'algoritmo di controllo utilizzato che può essere quello standard (un controllore proporzionale che sfrutta l'anello di velocità degli amplificatori) oppure uno creato appositamente dall'utente.

Il DSP gestisce tutti i segnali tipici di un controllo applicato a due assi distinti, vale a dire le informazioni provenienti dal calcolatore e quelle provenienti dai sensori dei motori.

Tutti i segnali sopra menzionati passano attraverso convertitori A/D e D/A in quanto il DSP lavora nel digitale, mentre agli amplificatori deve arrivare un segnale analogico; inoltre è possibile accedere a questi segnali attraverso il pannello frontale dell'interfaccia del DSP. Si possono così misurare o visualizzare con un oscilloscopio i vari vari segnali di tensione secondo le necessità.

Per il funzionamento del robot devono essere collegate le uscite relative ai segnali di comando ed ai sensori con i corrispondenti morsetti sui pannelli degli amplificatori.

Per entrare un po' più nei dettagli tecnici della scheda DSP si può dire che lavora ad una frequenza di clock di  $33MHz$  ed esegue una operazione a virgola mobile per ciclo. Questo è in contrasto con quanto avviene normalmente con i microprocessori che

impiegano più di un ciclo per portare a termine l'operazione. L'efficienza di calcolo è ulteriormente migliorata con una struttura basata su 14 registri.

La grande potenza di calcolo del DSP permette di utilizzare un tempo di campionamento molto piccolo se necessario oppure applicazioni che girano in tempo reale e che richiedono una notevole mole di calcoli.

### 0.3 Struttura Software del sistema di controllo

Il software che provvede al controllo del robot può essere suddiviso in vari moduli, ognuno dei quali assolve una determinata funzione come l'interpolazione, la cinematica inversa, la simulazione o il controllo vero e proprio.

Tutti questi moduli sono programmi in linguaggio C separati ed eventualmente accessibili all'utente che li può modificare o sostituire. Tutti i vari moduli sono coordinati da un altro programma (che si può definire "Black Box" in quanto non è accessibile per modifiche e non si hanno a disposizione i files sorgente), che costituisce l'ambiente di lavoro quando si utilizza il robot.

La figura 0.5 rappresenta schematicamente i vari moduli e le connessioni tra di essi.

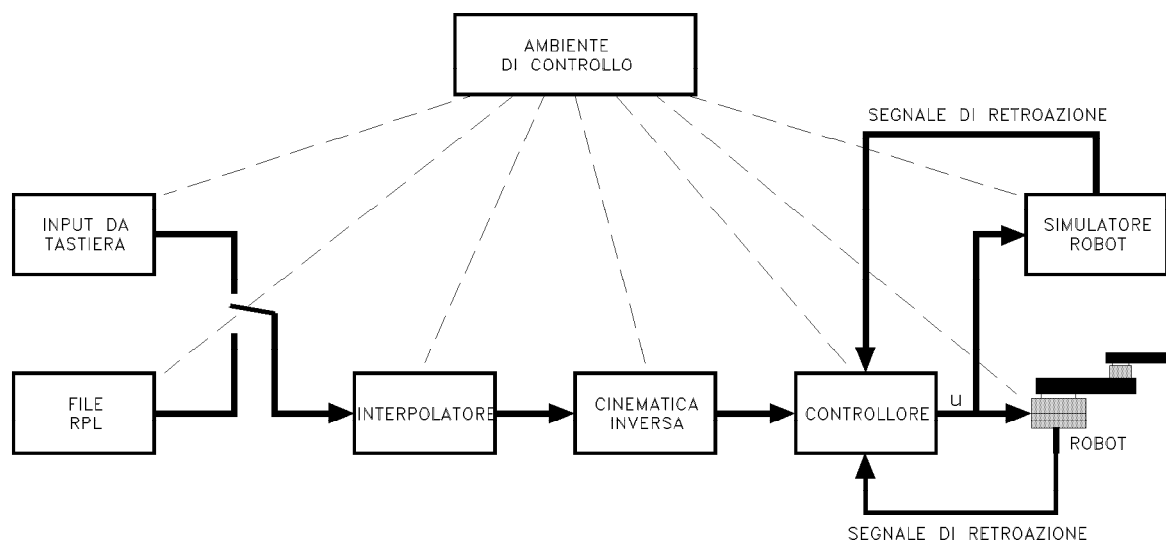


Figura 0.5. Struttura Software del sistema

Vediamo ora come interagiscono i vari moduli e come funziona nel suo insieme il software di controllo.

Innanzitutto si parte dal movimento che si vuole far compiere alla punta del manipolatore, si stabiliscono i punti che definiscono la traiettoria nello spazio giunti o nello spazio cartesiano attraverso comandi contenuti nei files **RPL** oppure tramite tastiera nel **Teach Mode**.

Una volta definita la sequenza dei movimenti, i punti estremi della traiettoria vengono raccordati con una linea retta (nello spazio giunti o in quello cartesiano, a seconda

---

di come si è definito il movimento) seguendo un profilo trapezoidale di velocità dall'interpolatore, che fornisce istante per istante come uscita i riferimenti per le posizioni.

Il modulo successivo è costituito dalla trasformazione della cinematica inversa che converte i riferimenti definiti nello spazio cartesiano  $(x,y)$  nei corrispondenti riferimenti nello spazio giunti  $(\theta_1,\theta_2)$ ; per i movimenti definiti nello spazio giunti il blocco della cinematica inversa non viene utilizzato in quanto i riferimenti corrispondenti sono già direttamente utilizzabili dal controllore.

Gli angoli di riferimento vengono poi utilizzati dal controllore, assieme al segnale di retroazione, per generare il comando da fornire ai motori per controllare il robot.

Per i moduli visti fino ad ora il modo di operare è identico sia quando si lavora in simulazione che in implementazione e quindi con il robot in movimento, mentre i comandi che si ottengono in uscita dal controllore sono utilizzati in maniera diversa nei due casi.

Quando si lavora in implementazione i segnali di comando vengono inviati agli amplificatori e da questi ai motori; il robot di conseguenza si muoverà e i sensori posti sui due giunti rilevano la posizione reale corrente dei due motori. Il segnale dei sensori è il segnale di *feedback* e viene quindi utilizzato per chiudere la retroazione e permettere il controllo.

Nel caso invece si utilizzi il sistema in simulazione i comandi non vengono forniti agli amplificatori ma all'apposito programma software in cui è descritto il modello matematico del robot e degli amplificatori, il quale, tenendo conto del modello, fornisce le posizioni simulate dei motori.

Vediamo ora in modo più approfondito il funzionamento dei moduli relativi alla cinematica inversa, alla simulazione (quindi al modello matematico del robot contenuto in essa) e all'interpolazione.

Per quel che riguarda il modulo del controllore, invece, verrà approfondito in seguito in un capitolo apposito (Capitolo 4).

## 0.4 Cinematica del manipolatore

Ricaviamo le equazioni matematiche che sono utilizzate nel modulo della cinematica inversa per passare dai valori delle variabili espressi nello spazio cartesiano a quelli espressi nello spazio giunti. A tal scopo si terrà quindi conto della geometria del robot.

Approfondiamo inoltre un'altra funzione assolta da questo modulo che consiste nel verificare se un movimento impostato è realizzabile o meno tenendo conto dello spazio di lavoro del manipolatore.

### 0.4.1 Trasformazione dallo spazio cartesiano allo spazio giunti

Lo scopo di questo paragrafo è quello di trovare delle funzioni  $f(x,y)$  e  $g(x,y)$  tali che:

$$\theta_1 = f(x,y)$$

$$\theta_2 = g(x,y)$$

dove  $x$  e  $y$  sono le coordinate nello spazio di lavoro della punta operativa e  $\theta_1$  e  $\theta_2$  sono rispettivamente l'angolo del primo braccio (base) e del secondo braccio (elbow) con riferimento alla figura 0.6.

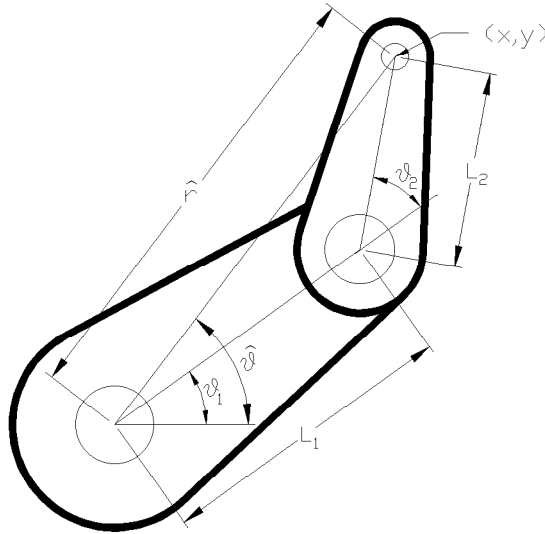


Figura 0.6. Geometria del manipolatore

Utilizziamo una trasformazione intermedia di coordinate

$$|\hat{r}| = (x^2 + y^2)^{\frac{1}{2}}$$

$$\hat{\theta} = \tan^{-1}\left(\frac{y}{x}\right)$$

Per la regola dei coseni si ha che:

$$|\hat{r}| = L_1^2 + L_2^2 + 2L_1L_2 \cos(\theta_2)$$

si ottiene così

$$\theta_2 = \cos^{-1}\left(\frac{|\hat{r}| - L_1^2 - L_2^2}{2L_1L_2}\right)$$

e

$$g(x,y) = \theta_2 = \cos^{-1}\left(\frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}\right) \quad (0.1)$$

Per determinare  $f(x,y)$  ricordiamo che, come si può verificare in figura 0.6 vale la relazione:

$$|\hat{r}| \sin(\hat{\theta} - \theta_1) = L_2 \sin(\theta_2)$$

quindi

$$\theta_1 = \hat{\theta} - \sin^{-1}\left(\frac{L_2 \sin(\theta_2)}{|\hat{r}|}\right) \quad (0.2)$$

e

$$f(x,y) = \theta_1 = \tan^{-1} \left( \frac{y}{x} \right) - \sin^{-1} \left\{ \frac{L_2 \sin \left[ \cos^{-1} \left( \frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1 L_2} \right) \right]}{(x^2 + y^2)^{\frac{1}{2}}} \right\} \quad (0.3)$$

Le espressioni contenute nelle equazioni (0.1) e (0.3) sono utilizzate nella cinematica inversa per convertire le coordinate nello spazio cartesiano nelle coordinate nello spazio giunti e rappresentano lo scopo finale di questo paragrafo.

Bisogna notare che l'equazione (0.1) ha due soluzioni distinte  $\theta_{2a}$  e  $\theta_{2b}$  dove  $\theta_{2a} = -\theta_{2b}$  ed un numero infinito di soluzioni  $2n\pi\theta_{2a,b}$ . La trasformazione quindi non è unica.

Usando l'equazione (0.2) per esprimere  $\theta_1$  in funzione di  $\theta_2$  e  $\hat{\theta}$  si trova che  $\theta_{2a}$  e  $\theta_{2b}$  forniscono due differenti valori di  $\theta_1$ . Ciò è rappresentato graficamente dalla figura 0.7

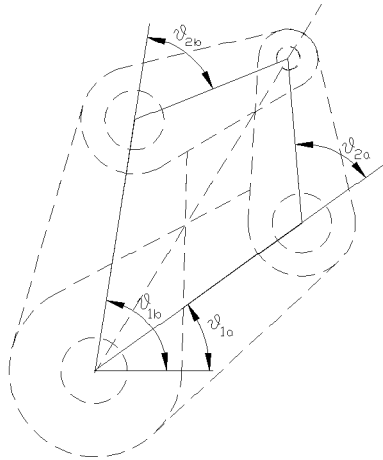


Figura 0.7. Descrizione grafica delle soluzioni delle equazioni 0.1 e 0.3

## 0.4.2 Singolarità della trasformazione dallo spazio cartesiano allo spazio giunti

La trasformazione dallo spazio giunti allo spazio di lavoro è più facile da ricavare. Dalla geometria del sistema si ha che:

$$\bar{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \bar{h}(\theta_1, \theta_2) = \begin{bmatrix} L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) \\ L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \quad (0.4)$$

Derivando l'equazione (0.4) si ottengono le velocità nello spazio di lavoro  $(\dot{x}, \dot{y})$  come funzioni delle velocità nello spazio giunti  $(\dot{\theta}_1, \dot{\theta}_2)$

$$\dot{\bar{x}} = J_0 \dot{\bar{\theta}}$$



dove

$$J_0 = \begin{bmatrix} \frac{\partial h_1}{\partial \theta_1} & \frac{\partial h_1}{\partial \theta_2} \\ \frac{\partial h_2}{\partial \theta_1} & \frac{\partial h_2}{\partial \theta_2} \end{bmatrix} = \begin{bmatrix} -L_1 \sin(\theta_1) - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (0.5)$$

è lo Jacobiano della trasformazione  $\bar{h}$ .

### Punti di singolarità

Il determinante di  $J_0$  è dato da

$$\det(J_0) = L_1 L_2 \sin(\theta_2) \quad (0.6)$$

è ovvio che la trasformazione  $\bar{h}$  ha una singolarità in ogni punto in cui  $\sin(\theta_2) = 0$  ossia  $\theta_2 = 0$  oppure  $\theta_2 = \pi$ . In questi punti le equazioni (0.1) e (0.3) non hanno soluzione.

Fisicamente se è richiesta una velocità nello spazio di lavoro costante, diversa da zero, quando il manipolatore si avvicina ad un punto di singolarità, la velocità nello spazio giunti necessaria tende a diventare infinita. Infatti si ha:

$$\dot{\theta} = J_0^{-1} \dot{x} = \frac{1}{\det(J_0)} \text{Adj}^T(J_0) \dot{x}$$

e  $\det(J_0)$  tende a zero quando ci si avvicina ad un punto di singolarità. In conclusione durante i movimenti nello spazio di lavoro la punta operativa deve essere mantenuta fuori dalle regioni di singolarità mostrate nella figura 0.8.

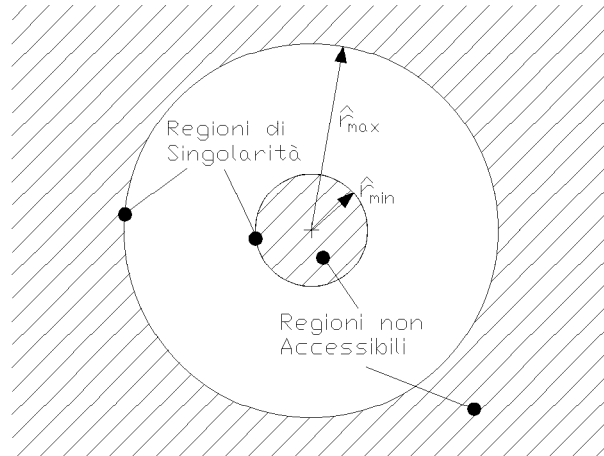


Figura 0.8. Regione di singolarità

Nella figura 0.8  $\hat{r}_{\min} = L_1 - L_2$  e  $\hat{r}_{\max} = L_1 + L_2$  delimitano le regioni di singolarità, in realtà è meglio tener conto di un fattore di sicurezza  $\delta$  facendo in modo che sia sempre verificato  $L_1 - L_2 + \delta \leq |\hat{r}| \leq L_1 + L_2 - \delta$ .

### Come evitare i punti di singolarità

Dopo che è stata specificato un movimento dall'utente o da un programma RPL, ma prima di eseguirlo, il percorso deve essere verificato affinché non includa alcun punto di

singularità. Se questo test rivela che il percorso non è valido, il movimento non viene effettuato.

Supponiamo che venga specificato dall'utente un movimento nello spazio di lavoro tra i punti  $\hat{r}_0 = [x_0, y_0]^T$  e  $\hat{r}_1 = [x_1, y_1]^T$  (come in figura 0.9). Il percorso che unisce questi due punti non è valido se:

1. Il punto iniziale è fuori dallo spazio di lavoro :  $|\hat{r}_0| \leq \hat{r}_{\min}$  oppure  $|\hat{r}_0| \geq \hat{r}_{\max}$
2. Il punto finale è fuori dallo spazio di lavoro :  $|\hat{r}_1| \leq \hat{r}_{\min}$  oppure  $|\hat{r}_1| \geq \hat{r}_{\max}$
3. Qualsiasi altro punto sulla congiungente tra  $\hat{r}_0$  e  $\hat{r}_1$  è fuori dallo spazio di lavoro

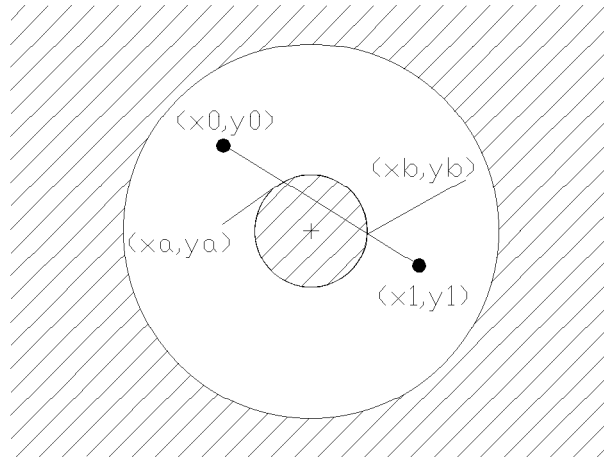


Figura 0.9. Percorso con valido con punti estremi validi

Le condizioni 1 e 2 sono facili da verificare, mentre la condizione 3, illustrata in figura 0.9 richiede ulteriori analisi.

Se le dimensioni dello spazio di lavoro sono misurate a partire dall'origine mostrata in figura 0.9, l'equazione per la regione di singularità interna è la seguente:

$$x^2 + y^2 = \hat{r}_{\min}^2 \quad (0.7)$$

e l'equazione per il percorso richiesto è

$$y = mx + b \quad (0.8)$$

dove

$$m = \frac{y_1 - y_0}{x_1 - x_0}$$

e

$$b = y_1 - mx_1$$

se non è verificata la condizione 3, ci devono essere due punti  $(x_a, y_a)$  e  $(x_b, y_b)$  dove il percorso interseca la circonferenza interna e si possono ricavare risolvendo simultaneamente le equazioni (0.7) e (0.8) .

$$x^2 + (mx + b)^2 = \hat{r}_{\min}^2 \quad (0.9)$$

$$(m^2 + 1)x^2 + 2mbx + (b^2 - \hat{r}_{\min}^2) = 0 \quad (0.10)$$

L'ascissa dei punti di intersezione si ricava dalla (0.10)

$$x_{a,b} = \frac{-mb \pm \sqrt{m^2b^2 - (m^2 + 1)(b^2 - \hat{r}_{\min}^2)}}{m^2 + 1} \quad (0.11)$$

La soluzione della (0.11) esiste solo se il radicando è maggiore o uguale a zero, così il percorso non è possibile solo se

$$m^2b^2 - (m^2 + 1)(b^2 - \hat{r}_{\min}^2) \geq 0 \quad (0.12)$$

$$\frac{b^2}{1 + m^2} \leq \hat{r}_{\min}^2 \quad (0.13)$$

Se la (0.13) non è verificata la retta e il cerchio non si intersecano e il movimento è valido. Se invece è verificata il movimento può essere valido o meno come mostrato nella figura 0.10

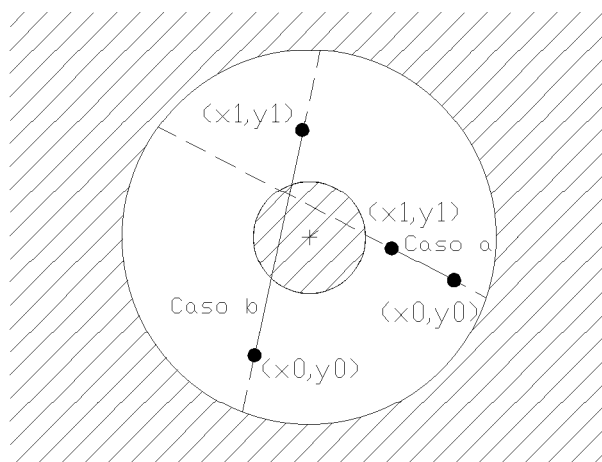


Figura 0.10. Percorsi che producono soluzioni reali dell'equazione 0.10

Una volta utilizzata l'equazione (0.13) per vedere se la retta che passa per  $(x_0, y_0)$  e  $(x_1, y_1)$  interseca la regione di singolarità è semplice verificare se il segmento compreso tra i due punti comprende punti di singolarità o meno.

Se si ha che vale la (0.13) e  $x_1 \geq x_a$  e  $x_0 \geq x_a$  oppure  $x_1 \leq x_b$  e  $x_0 \leq x_b$  allora il caso a è verificato e il movimento è ammesso, viceversa il caso b è vero ed il movimento non è ammesso.

---

## 0.5 Dinamica e modello matematico del manipolatore

Per modello matematico del manipolatore si intende una equazione differenziale che permetta di calcolare l'evoluzione temporale delle variabili che ci interessano (posizioni angolari di entrambi i bracci) rispecchiando il più possibile la realtà.

Per ricavare questa equazione si deve partire dalla dinamica del sistema.

La figura 0.11 è un disegno schematico di un manipolatore a due bracci di tipo planare.

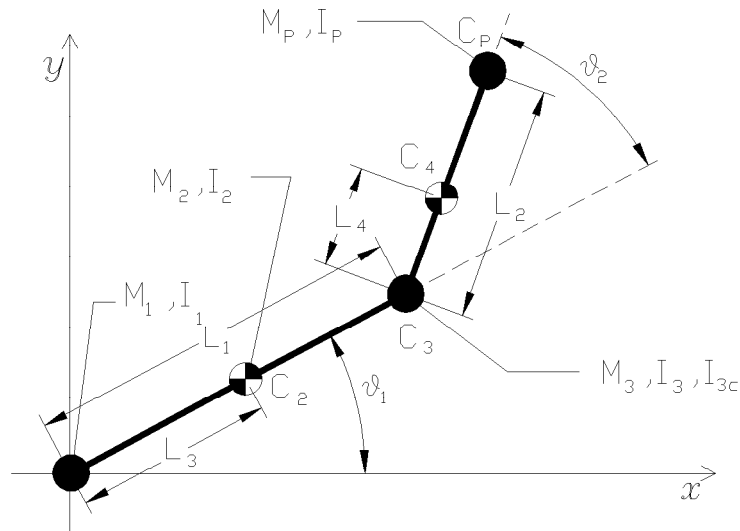


Figura 0.11. Manipolatore SCARA planare

I parametri definiti nella figura sono :

$I_1$  = Inerzia rotore motore 1

$M_1$  = Massa motore 1

$I_2$  = Momento centrale di inerzia braccio 1

$M_2$  = Massa braccio 1

$I_3$  = Inerzia rotore motore 2

$M_3$  = Massa motore 2

$I_{3c}$  = Inerzia statore motore 2

$M_4$  = Massa braccio 2

$I_4$  = Momento centrale di inerzia braccio 2

$M_p$  = Massa carico

$I_p$  = Momento di inerzia carico

---

$\theta_1$  = Angolo del braccio 1 rispetto all'orizzontale

$\theta_2$  = Angolo del braccio 2 rispetto al braccio 1

$L_1$  = Lunghezza braccio 1

$L_2$  = Lunghezza braccio 2

$L_3$  = Distanza tra il centro di massa (CG) del braccio 1 e l'asse di rotazione

$L_4$  = Distanza tra il centro di massa (CG) del braccio 2 e l'asse di rotazione

Altre quantità che verranno usate sono :

$x_{c2}$  = vettore posizione CG braccio 1

$x_{c3}$  = vettore posizione CG motore 2

$x_{c4}$  = vettore posizione CG braccio 2

$x_{cp}$  = vettore posizione CG carico

$v_{c1}$  = vettore velocità CG braccio 1

$v_{c2}$  = vettore velocità CG motore 2

$v_{c4}$  = vettore velocità CG braccio 2

$v_p$  = vettore velocità CG carico

$\tau_1$  = Coppia applicata dal motore 1

$\tau_2$  = Coppia applicata dal motore 2

$f_1$  = Coppia di attrito asse 1

$f_2$  = Coppia di attrito asse 2

L'equazione del moto del manipolatore sarà ricavata usando l'equazione di Lagrange:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}_i} \right) - \frac{dT}{\partial q_i} = Q_i$$

$$i = 1,2$$

dove  $T$  rappresenta la differenza tra l'energia cinetica e quella potenziale (in questo caso nulla per la geometria del manipolatore),  $q$  è un vettore di coordinate generalizzate che descrivono completamente la configurazione del manipolatore e  $Q$  è un vettore di forze generalizzate che sono applicate al manipolatore. In questo caso

$$q = \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

---


$$Q = \tau - f = \begin{bmatrix} \tau_1 - f_1 \\ \tau_2 - f_2 \end{bmatrix}$$

La posizione e la velocità del centro di massa del braccio 1 sono :

$$x_{c2} = \begin{bmatrix} L_3 \cos(\theta_1) \\ L_3 \sin(\theta_1) \end{bmatrix}$$

$$v_{c2} = \frac{dx_{c2}}{dt} = \begin{bmatrix} -L_3 \dot{\theta}_1 \sin(\theta_1) \\ L_3 \dot{\theta}_1 \cos(\theta_1) \end{bmatrix}$$

La posizione e la velocità del centro di massa del motore 2 sono :

$$x_{c3} = \begin{bmatrix} L_1 \cos(\theta_1) \\ L_1 \sin(\theta_1) \end{bmatrix}$$

$$v_{c2} = \frac{dx_{c3}}{dt} = \begin{bmatrix} -L_1 \dot{\theta}_1 \sin(\theta_1) \\ L_1 \dot{\theta}_1 \cos(\theta_1) \end{bmatrix}$$

La posizione e la velocità del centro di massa del braccio 2 sono :

$$x_{c4} = \begin{bmatrix} L_1 \cos(\theta_1) + L_4 \cos(\theta_1 + \theta_2) \\ L_1 \sin(\theta_1) + L_4 \sin(\theta_1 + \theta_2) \end{bmatrix}$$

$$v_{c4} = \frac{dx_{c4}}{dt} = \begin{bmatrix} -L_1 \dot{\theta}_1 \sin(\theta_1) - L_4 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_1 + \theta_2) \\ L_1 \dot{\theta}_1 \cos(\theta_1) + L_4 (\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_1 + \theta_2) \end{bmatrix}$$

La posizione e la velocità della punta operativa sono :

$$x_{cp} = \begin{bmatrix} L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) \\ L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2) \end{bmatrix}$$

$$v_{cp} = \frac{dx_{cp}}{dt} = \begin{bmatrix} -L_1 \dot{\theta}_1 \sin(\theta_1) - L_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_1 + \theta_2) \\ L_1 \dot{\theta}_1 \cos(\theta_1) + L_2 (\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_1 + \theta_2) \end{bmatrix}$$

L'energia cinetica del manipolatore è espressa da :

$$T = \frac{1}{2} \left[ (I_1 + I_2 + I_{3c}) \dot{\theta}_1^2 + (I_3 + I_4 + I_p) (\dot{\theta}_1 + \dot{\theta}_2)^2 + M_2 v_{c2}^t v_{c2} + M_3 v_{c3}^t v_{c3} + M_4 v_{c4}^t v_{c4} + M_p v_{cp}^t v_{cp} \right]$$

Sostituendo le espressioni di  $v_{c2}, v_{c4}$  e  $v_p$  e con alcune regole trigonometriche si ottiene

$$T = \frac{1}{2} \left[ (I_1 + I_2 + I_{3c} + M_3 L_1^2 + M_4 L_1^2 + M_p L_1^2 + M_2 L_3^2) \dot{\theta}_1^2 \right. \\ \left. + (I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2) (\dot{\theta}_1^2 + \dot{\theta}_2^2) \right. \\ \left. + 2(M_4 L_1 L_4 + M_p L_1 L_2) \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_2) \right]$$

Derivando

$$\frac{\partial T}{\partial \dot{\theta}_1} = (I_1 + I_2 + I_{3c} + M_3 L_1^2 + M_4 L_1^2 + M_p L_1^2 + M_2 L_3^2) \dot{\theta}_1 \\ + (I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2) (\dot{\theta}_1 + \dot{\theta}_2) \\ + (M_4 L_1 L_4 + M_p L_1 L_2) (2\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_2) \\ = p_1 \dot{\theta}_1 + p_2 \dot{\theta}_2 + p_3 (2\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_2)$$

dove le costanti valgono

$$p_1 = I_1 + I_2 + I_{3c} + I_3 + I_4 + I_p + (M_3 + M_4 + M_p) L_1^2 + M_2 L_3^2 + M_4 L_4^2 + M_p L_2^2$$

$$p_2 = I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2$$

$$p_3 = M_4 L_1 L_4 + M_p L_1 L_2$$

analogamente si ottiene

$$\frac{\partial T}{\partial \dot{\theta}_2} = p_2 \dot{\theta}_1 + p_2 \dot{\theta}_2 + p_3 \dot{\theta}_1 \cos(\theta_2)$$

e

$$\frac{\partial}{\partial t} \left( \frac{\partial T}{\partial \dot{\theta}_1} \right) = p_1 \ddot{\theta}_1 + p_2 \ddot{\theta}_2 + p_3 (2\ddot{\theta}_1 + \ddot{\theta}_2) \cos(\theta_2) - p_3 (2\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_2)$$

$$\frac{\partial}{\partial t} \left( \frac{\partial T}{\partial \dot{\theta}_2} \right) = p_2 \ddot{\theta}_1 + p_2 \ddot{\theta}_2 + p_3 \ddot{\theta}_1 \cos(\theta_2) - p_3 \dot{\theta}_1 \dot{\theta}_2 \sin(\theta_2)$$

$$\frac{\partial T}{\partial \theta_1} = 0$$

$$\frac{\partial T}{\partial \theta_2} = -p_3 \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_2)$$

Sostituendo queste espressioni nell'equazione di Lagrange, l'equazione del moto risulta

$$M(\theta) \ddot{\theta} + V(\theta, \dot{\theta}) = \tau(t) - f \quad (0.14)$$

dove

$$V(\theta, \dot{\theta}) = \begin{bmatrix} -\dot{\theta}_2 (2\dot{\theta}_1 + \dot{\theta}_2) p_3 \sin(\theta_2) \\ \dot{\theta}_1^2 p_3 \sin(\theta_2) \end{bmatrix}$$

---

è il vettore che contiene i termini di Coriolis e

$$M(\theta) = \begin{bmatrix} p_1 + 2p_3 \cos(\theta_2) & p_2 + p_3 \cos(\theta_2) \\ p_2 + p_3 \cos(\theta_2) & p_2 \end{bmatrix}$$

è la matrice di inerzia (variabile con l'angolo  $\theta_2$ ).

Partendo dall'equazione (0.14), riordinando ed invertendo la matrice  $M$  si ottiene

$$\ddot{\theta} = M^{-1}(\theta) [-V(\theta, \dot{\theta}) + \tau(t) - f] \quad (0.15)$$

che è un'equazione differenziale vettoriale del secondo ordine che ha come incognita il vettore delle posizioni angolari dei due bracci.

L'equazione (0.15) rappresenta proprio il modello matematico del robot.

All'interno della routine che effettua la simulazione si utilizza questa equazione che esprime  $\ddot{\theta}$  in funzione della posizione e della velocità attuali per calcolarsi, integrando due volte con l'algoritmo di Runge-Kutta, la posizione al passo successivo tenendo conto della dinamica del robot.