

Università di Pisa
Corso di Laurea Magistrale in Informatica



Corso di Robotica

**Fondamenti di controllo dei
robot**

Cecilia Laschi
cecilia.laschi@sssup.it



Modulo B. Fondamenti di controllo dei Robot

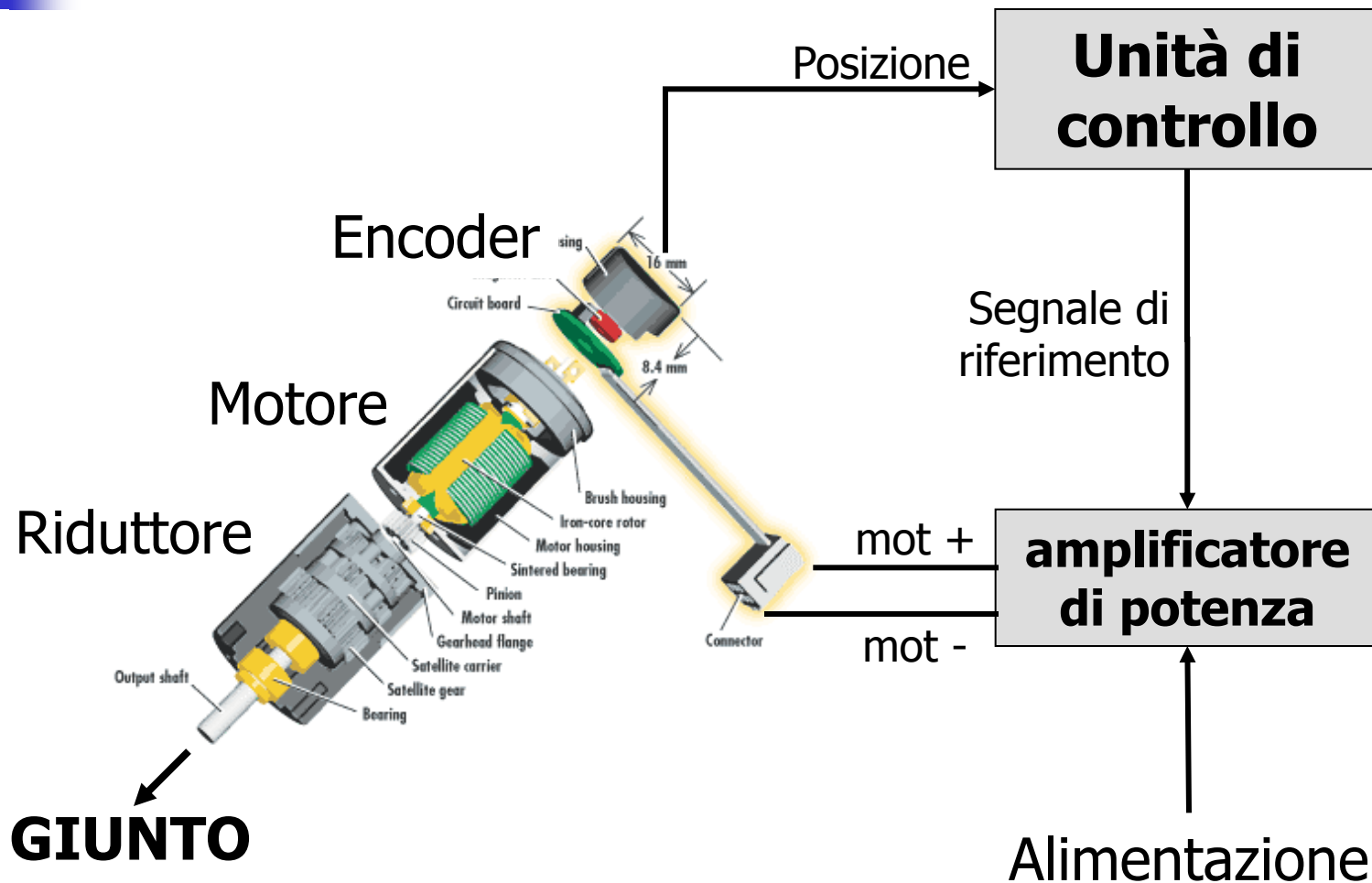
- Controllo del moto di un giunto:
 - Il controllo PID
- Controllo del moto di un manipolatore
 - Pianificazione delle traiettorie
 - Controllo del moto nello spazio dei giunti
 - Controllo del moto nello spazio operativo: cinematica differenziale e Jacobiano
- Esempio del Dexter Arm:
 - Meccanica, Cinematica, Controllo, Interfacce software



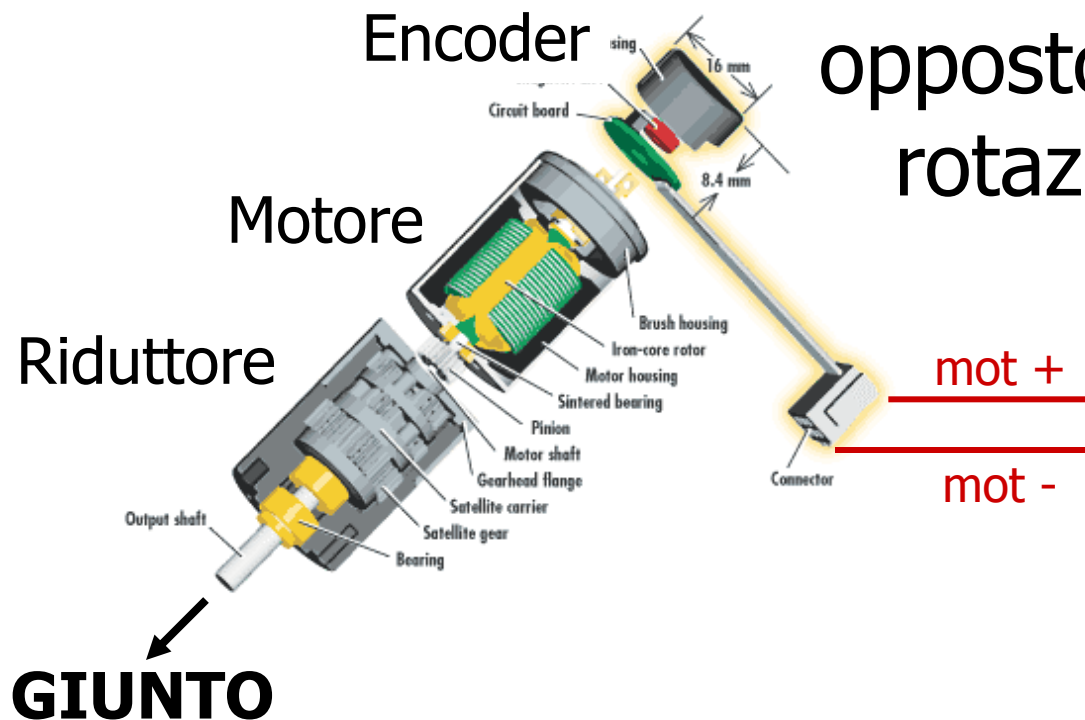
Sistema di controllo

- Un sistema di controllo fornisce un comando in tensione o in corrente agli attuatori (motori) in modo da far assumere ai giunti una configurazione desiderata

Schema di una unità di controllo



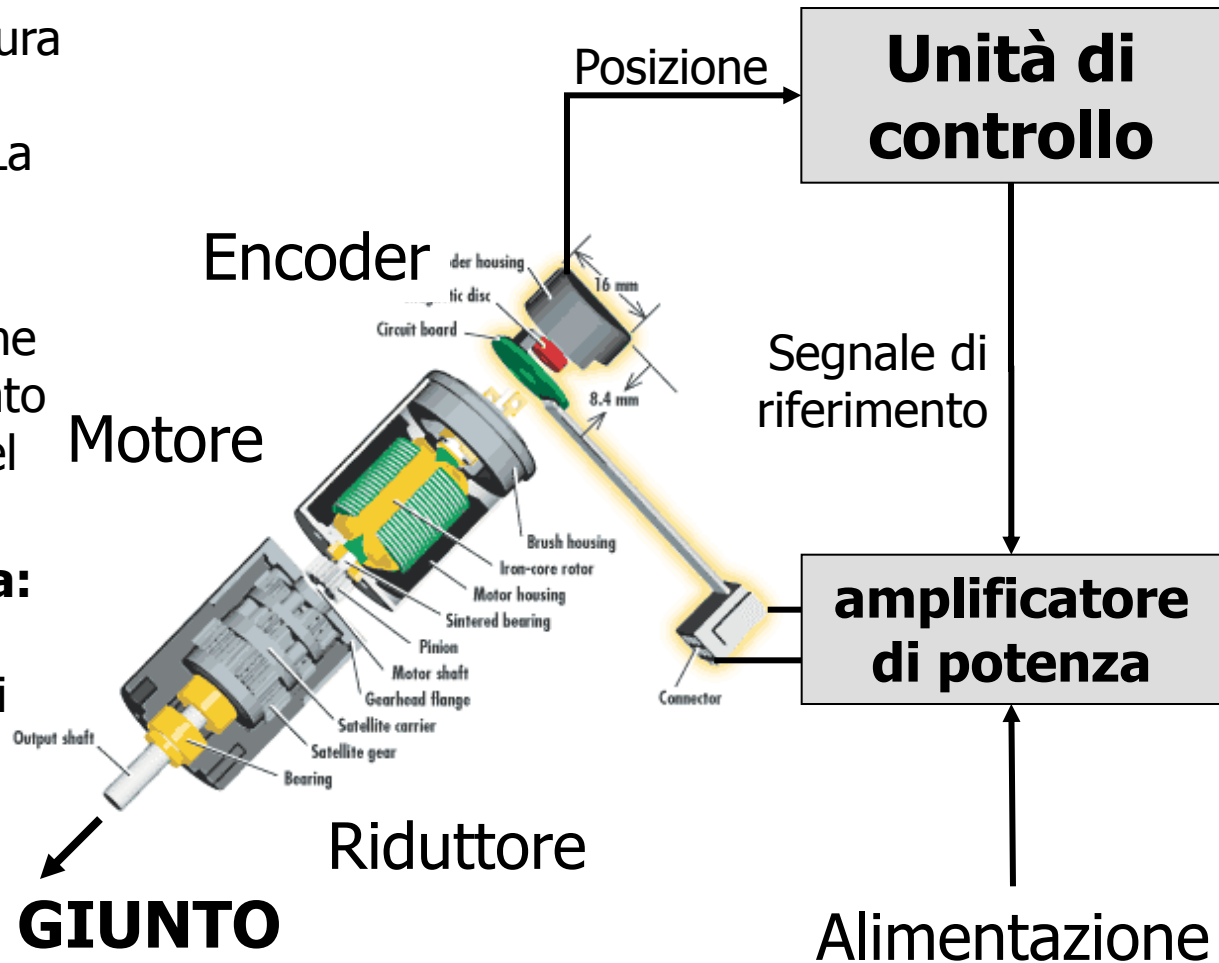
Schema di una unità di controllo



A tensioni in ingresso al motore di segno opposto corrispondono rotazioni opposte del motore

Schema di una unità di controllo

- **Encoder:** sensore che misura la rotazione dei giunti in valore relativo o assoluto. La misurazione avviene in "tacche di encoder"
- **Riduttore:** meccanismo che riduce i giri dell'asse montato sul giunto rispetto ai giri del motore (es. riduzione 1:k)
- **Amplificatore di potenza:** amplifica un segnale di riferimento in un segnale di potenza per muovere il motore
- **Unità di controllo:** unità che produce un segnale di riferimento per il motore





Relazione tra posizione del giunto e posizione encoder

- q : posizione angolare del giunto (in gradi)
- θ : posizione del giunto in tacche di encoder
- k : rapporto di riduzione del motore
- R : risoluzione dell'encoder (numero di tacche per giro)

$$q = \frac{\theta \times 360^\circ}{R \times k}$$



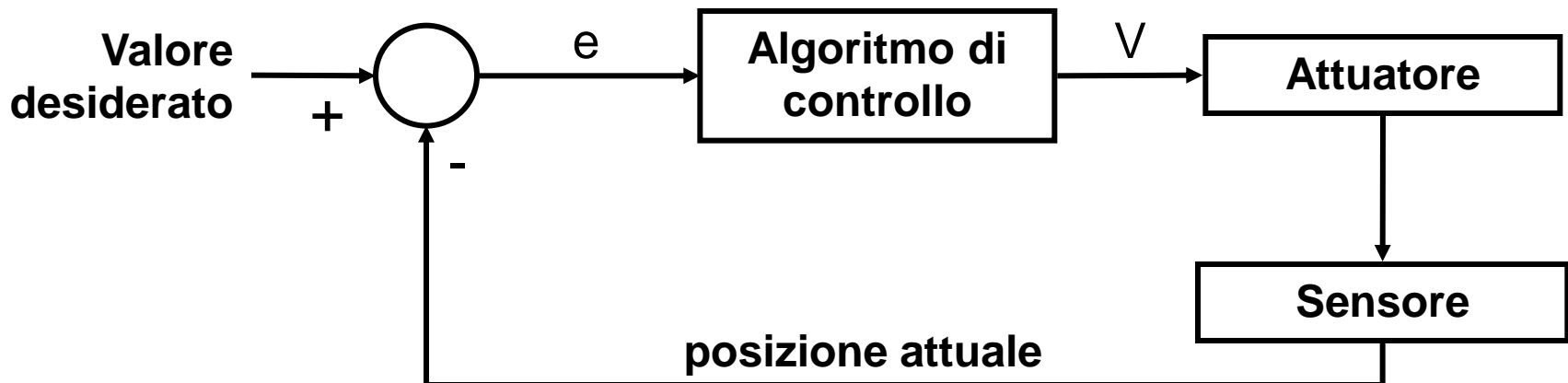
Controllo del moto di un giunto

- Obiettivo: muovere il giunto dalla posizione attuale q_i (espressa in gradi) alla posizione desiderata q_f in un intervallo di tempo t :

$$q_i \Rightarrow q_f$$

Controllo ad anello chiuso (feedback)

- La variabile da controllare è misurata e confrontata con il valore desiderato
- la differenza, o errore, è elaborata secondo un certo algoritmo
- il risultato di questa elaborazione costituisce il valore d'ingresso dell'attuatore





Controllo PID

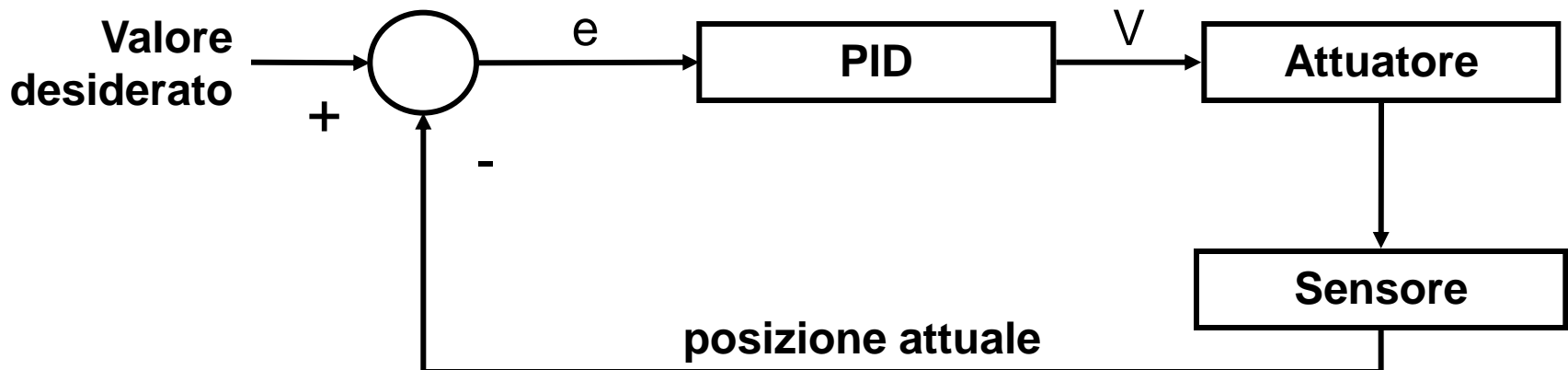
(Proporzionale, Integrativo e Derivativo):

- E' un sistema di controllo ad anello chiuso in cui l'errore è processato con un algoritmo di tipo **Proporzionale, Integrativo e derivativo**.
- L'algoritmo processa l'errore e fornisce un ingresso all'attuatore con 3 componenti:
 - **Proporzionale**, così detta perché proporzionale all'errore;
 - **Integrativa**, perché produce una correzione data dall'integrale dell'errore nel tempo;
 - **Derivativa**, perché genera una correzione che è funzione della derivata prima dell'errore.
- Non tutti i sistemi di controllo ad anello chiuso fanno uso di un algoritmo di tipo PID

Controllo PID

(Proporzionale, Integrativo e Derivativo):

- In un controllo PID l'errore è dato in ingresso al sistema di controllo il quale calcola le componenti derivativa e integrale e il segnale di uscita V





Controllo PID

(Proporzionale, Integrativo e derivativo):

$$V = K_p e_q + K_d \dot{e}_q + K_i \int e_q(t) dt$$

$$e_q = q_d - q_a$$

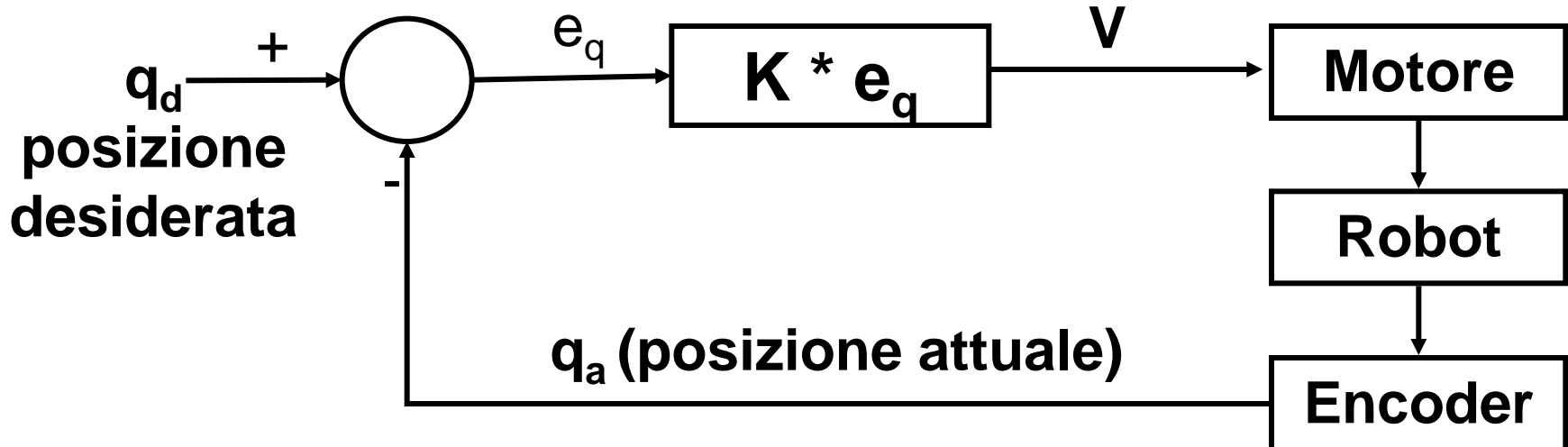
$$\dot{e}_q = \frac{de_q}{dt}$$

- K_p è il guadagno o costante *proporzionale*
- K_i è il guadagno o costante *integrale*
- K_d è il guadagno o costante *derivativa*
- e_q rappresenta l'errore, ovvero la differenza tra posizione desiderata e posizione attuale

Controllo PID

Componente Proporzionale

- La tensione V imposta al motore è proporzionale alla differenza tra la posizione effettiva misurata dal sensore e la posizione desiderata





Controllo PID

Componente Proporzionale:

La tensione imposta al motore nell'unità di tempo è proporzionale alla differenza tra la posizione effettiva misurata dal sensore e la posizione desiderata

$$V = K_p e_q$$

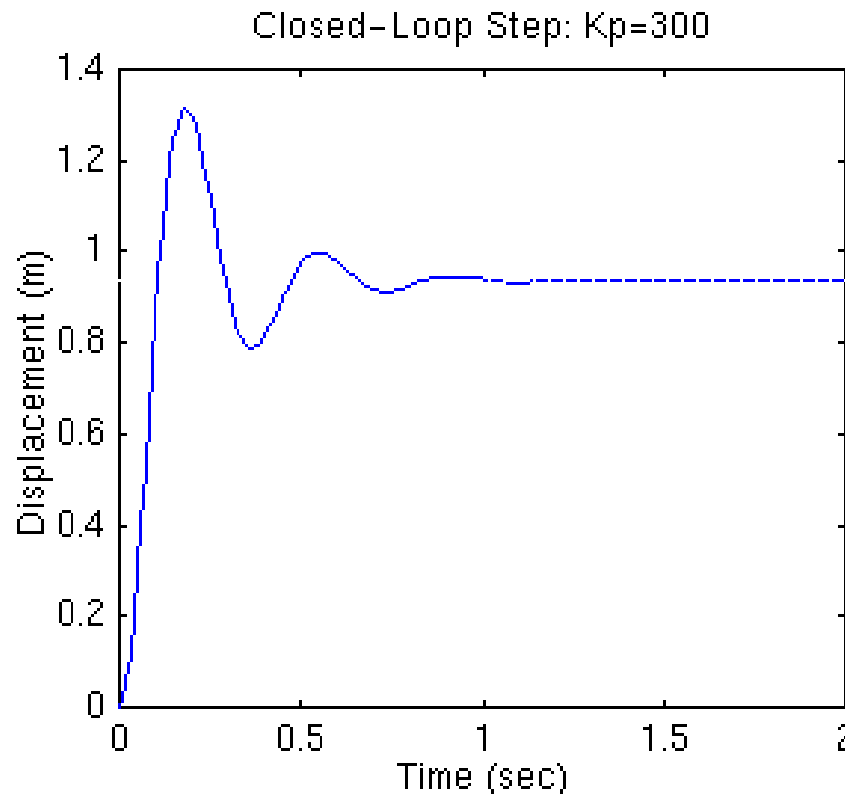
$$e_q = q_d - q_a$$

K_p : costante proporzionale

Controllo PID

Componente Proporzionale: comportamento del sistema

Posizione
desiderata: 1



- Il motore oscilla prima di convergere verso la posizione desiderata
- Il sistema si assesta senza annullare l'errore



Controllo PID

Componente derivativa:

$$\dot{e}_q = \frac{de_q}{dt} \quad \text{Derivata dell'errore nel tempo}$$

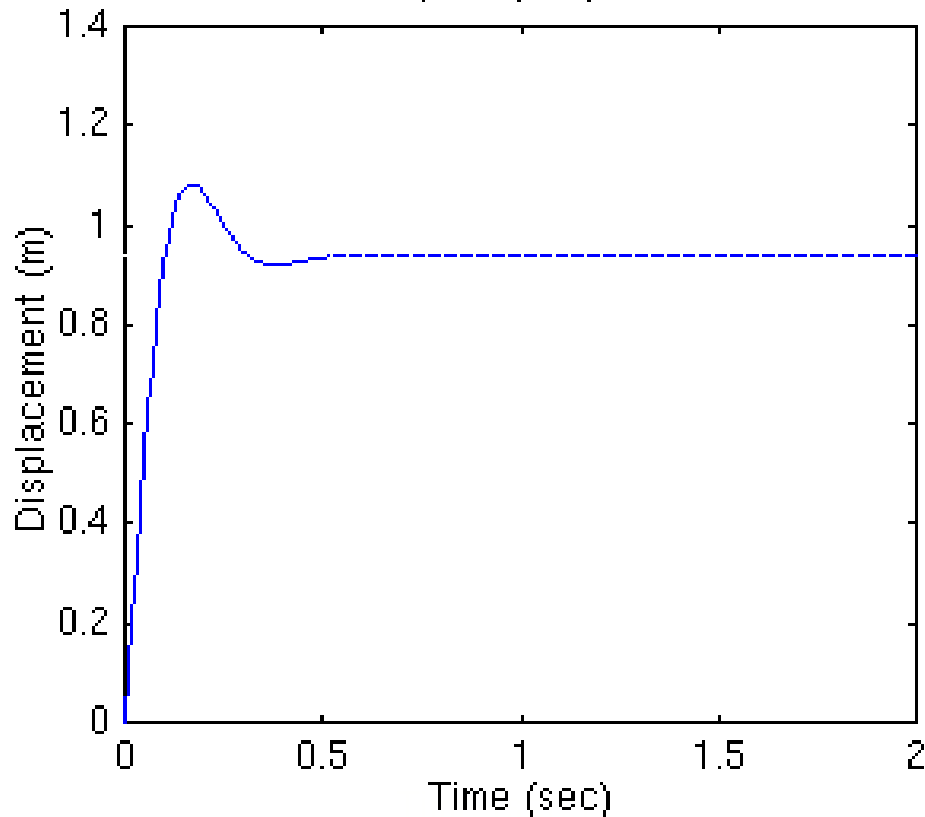
$$V = K_p e_q + K_d \dot{e}_q \quad K_d: \text{costante derivativa}$$

$$e_q = q_d - q_a$$

Controllo PID

Controllo Proporzionale e Derivativo:

Closed-Loop Step: $K_p=300$, $K_d=10$



- Riduzione delle oscillazioni
- Diminuzione del tempo di assestamento
- Il sistema si assesta senza annullare l'errore



Controllo PID

Componente integrativa:

$K_i \int e_q(t) dt$ Integrazione dell'errore nel tempo

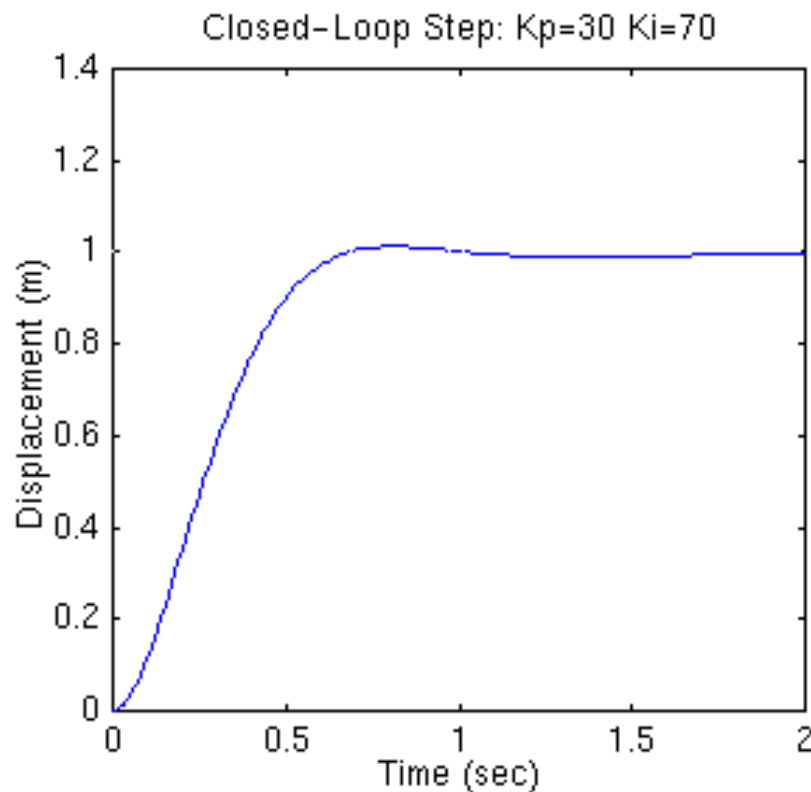
$$V = K_p e_q + K_i \int e_q(t) dt$$

$$e_q = q_d - q_a$$

K_i : costante integrativa

Controllo PID

Controllo Proporzionale e Integrativo:



- Il sistema si assesta annullando l'errore



Controllo PID

- **Controllo Proporzionale, Derivativo e Integrativo**

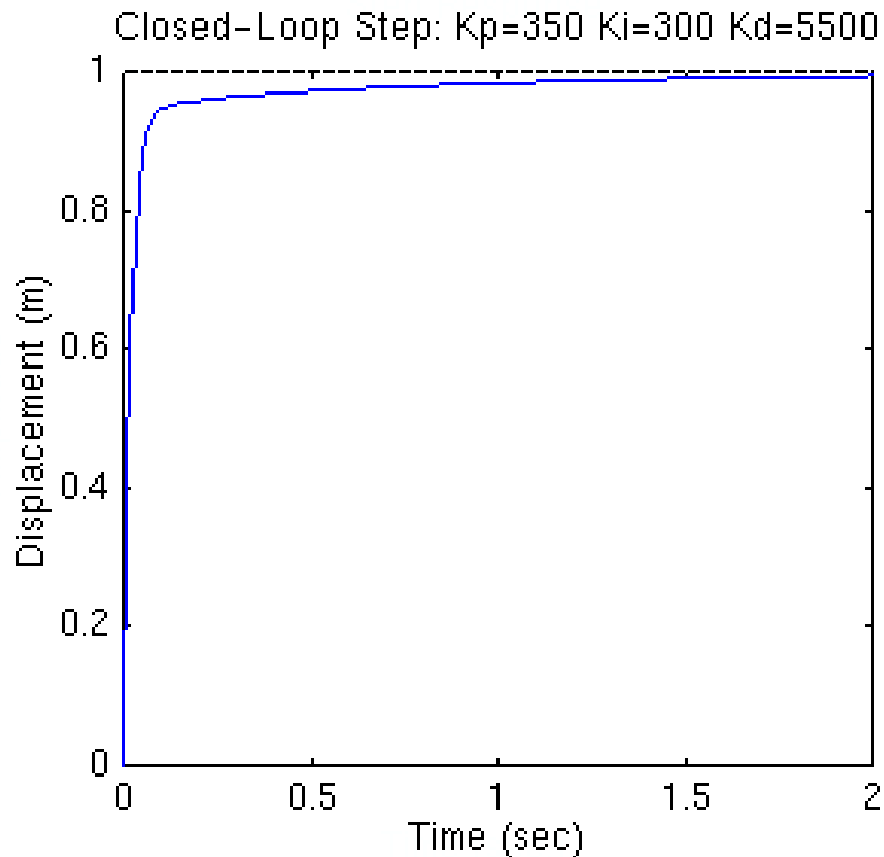
$$V = K_p e_q + K_d \dot{e}_q + K_i \int e_q(t) dt$$

$$e_q = q_d - q_a$$

$$\dot{e}_q = \frac{de_q}{dt}$$

Controllo PID

Controllo Proporzionale, Derivativo e Integrativo:



- Le costanti K_p , K_d , K_i vengono determinate in modo empirico o con metodi specifici



Controllo del moto di un manipolatore

- Obiettivo del controllo del moto di un manipolatore è muovere il braccio da una posizione iniziale ad una posizione finale espresse nelle coordinate dello spazio operativo
- In generale, il problema del controllo del moto di un manipolatore consiste nel determinare l'andamento delle coppie che gli attuatori devono applicare ai giunti in modo da garantire l'esecuzione di una traiettoria pianificata



Pianificazione di traiettorie

OBIETTIVO: generare gli ingressi di riferimento per il sistema di controllo del moto per muovere il braccio da x_{start} a x_f

PERCORSO: luogo dei punti dello spazio dei giunti o dello spazio operativo che il manipolatore deve descrivere nell'esecuzione del movimento assegnato

TRAIETTORIA: percorso su cui è specificata la legge oraria di moto (velocità ed accelerazione in ogni punto)



Pianificazione di traiettorie

OBIETTIVO: generare gli ingressi di riferimento per il sistema di controllo del moto:

$$X_{\text{start}} \dashrightarrow X_f$$

DATI IN INPUT:

definizione del percorso

vincoli del percorso

vincoli dovuti alla dinamica del manipolatore

DATI IN OUTPUT:

nello spazio dei giunti: traiettorie dei vari giunti

nello spazio operativo: traiettoria dell'organo terminale

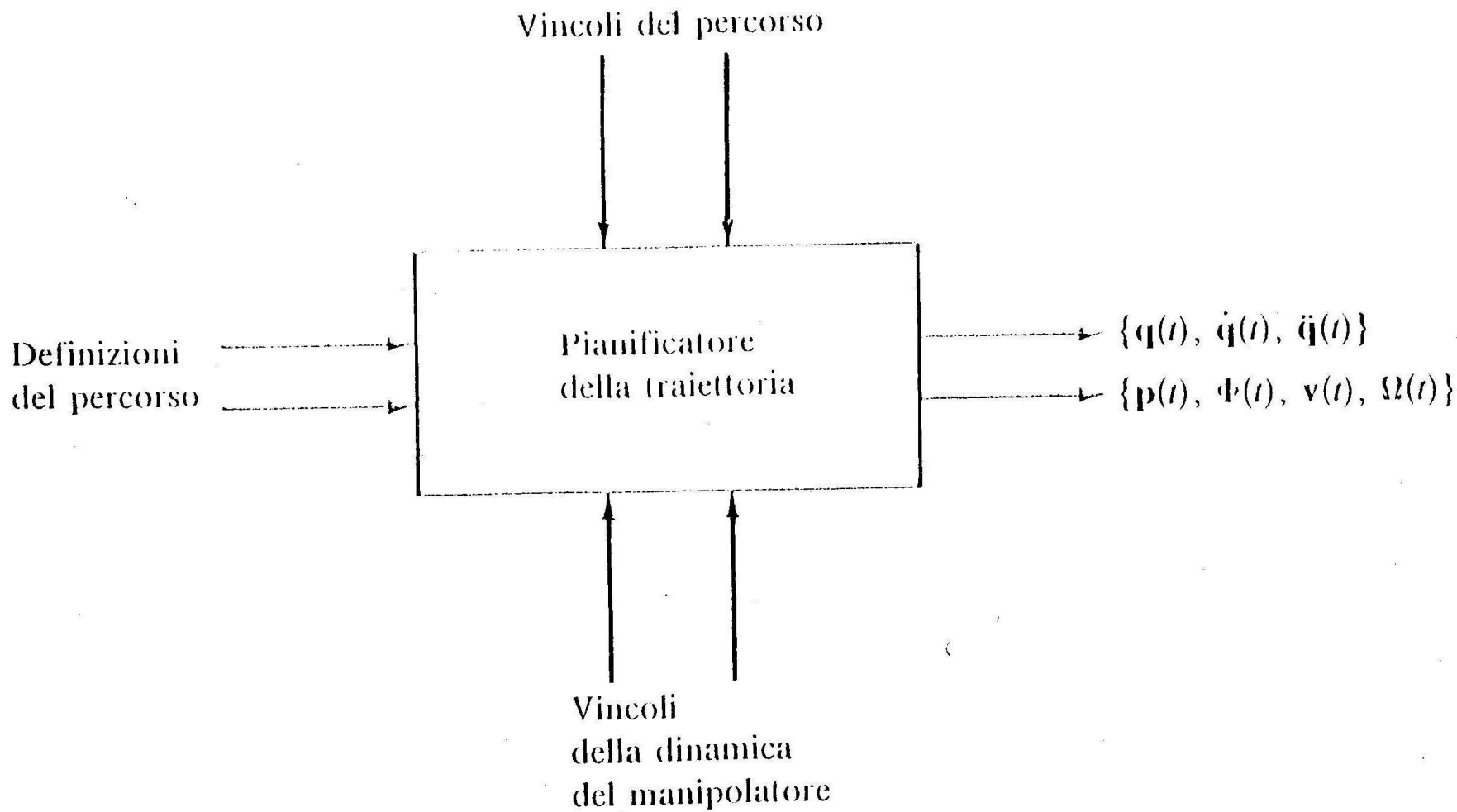


Figura 4.1 Diagramma a blocchi del pianificatore della traiettoria.



Traiettorie nello spazio dei giunti

- Moto punto-punto: il manipolatore deve muoversi da una configurazione iniziale delle variabili di giunto ad una finale, in un intervallo di tempo fissato t .
- Per imporre la legge di moto su un giunto si può scegliere un profilo di velocità trapezoidale, oppure si possono scegliere funzioni polinomiali:
 - Polinomio cubico: consente di imporre i valori iniziale q_i e finale q_d delle variabili di giunto e le velocità iniziale e finale (generalmente nulle).
 - Polinomio di quinto grado: consente di imporre i valori iniziale q_i e finale q_f delle variabili di giunto, le velocità iniziale e finale e il valore dell'accelerazione iniziale e finale.



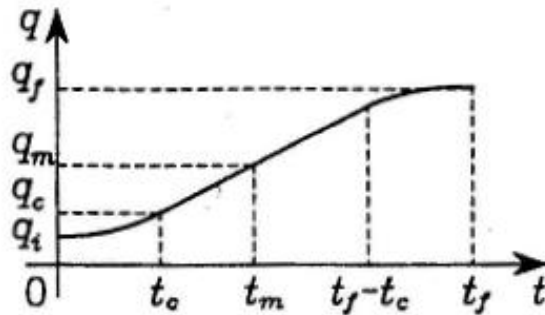
Traiettorie nello spazio dei giunti

Profilo di velocità trapezoidale

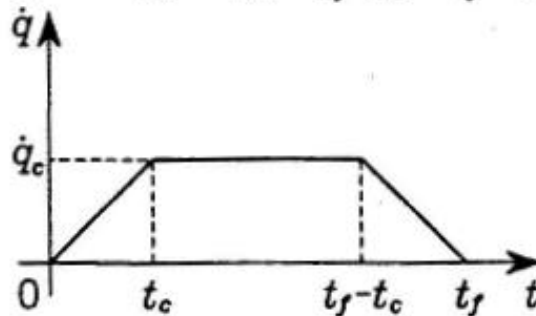
- Presenta una accelerazione costante nella fase di partenza, una velocità di crociera costante e una decelerazione costante nella fase di arrivo.
- La traiettoria corrispondente è di tipo polinomiale misto: un tratto lineare raccordato con due tratti parabolici nell'intorno delle posizioni iniziale e finale.

Profilo di velocità trapezoidale

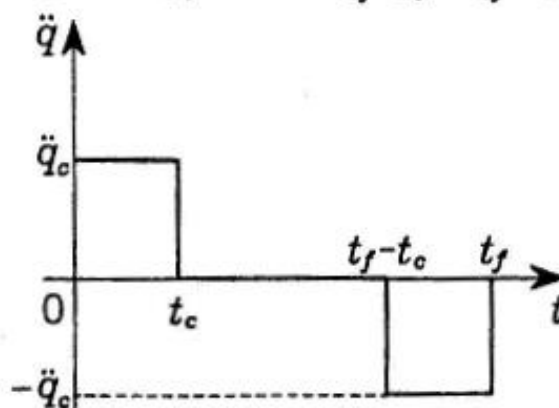
Posizione



Velocità

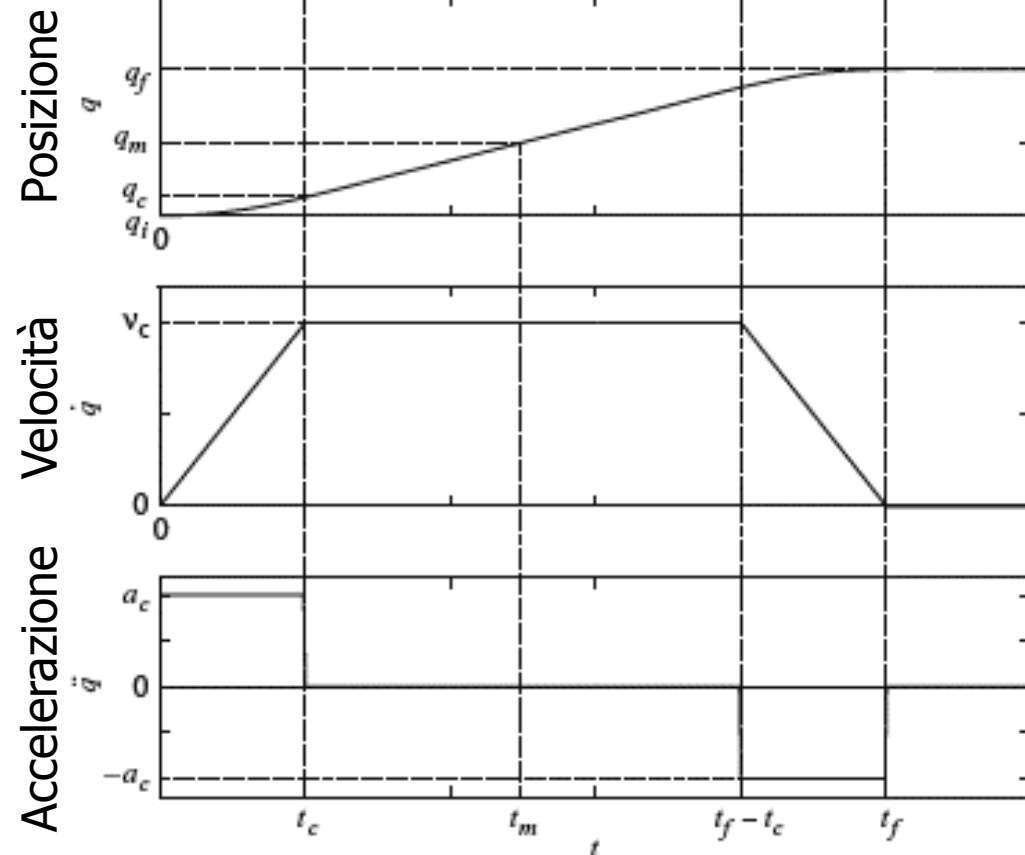


Accelerazione



NB: velocità ed accelerazioni all'istante finale ed iniziale possono essere diverse da zero

Profilo di velocità trapezoidale



Prima fase:

$$q_1(t) = q_i + \frac{1}{2}a_c t^2 \quad 0 \leq t \leq t_c$$

Seconda fase:

$$q_2(t) = q_i + a_c t_c \left(t - \frac{t_c}{2}\right) \quad t_c < t \leq (t_f - t_c)$$

Terza fase:

$$q_3 = q_f - \frac{1}{2}a_c (t - t_f)^2 \quad (t_f - t_c) < t \leq t_f$$

Interpolazione di traiettoria

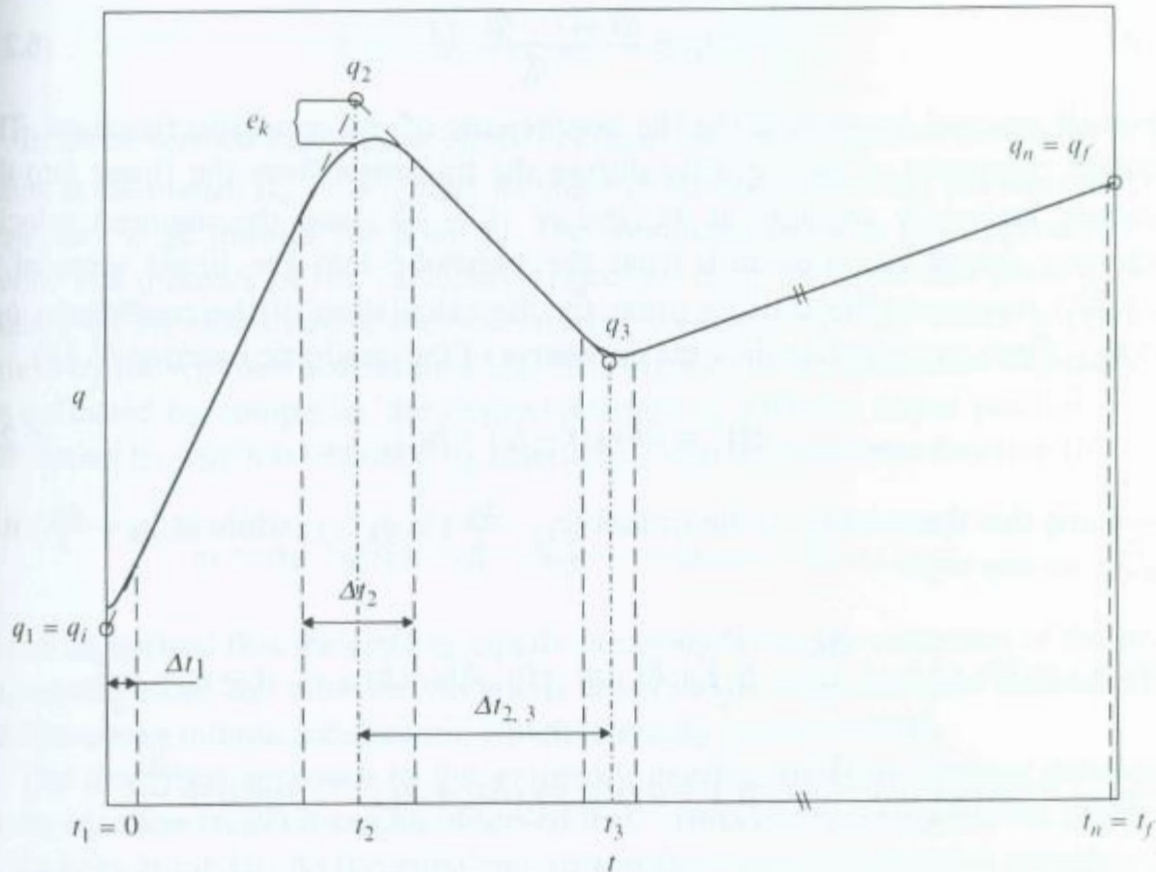


Fig. 6.2 Trajectory interpolation through n via points – linear segments with parabolic transitions are used



Traiettorie nello spazio operativo

- L'algoritmo di pianificazione di traiettoria nello spazio operativo genera la legge di moto dell'organo terminale, secondo un percorso di caratteristiche geometriche definite nello spazio operativo.
- Il risultato di una pianificazione è una sequenza di n-uple: $(p(t), \Phi(t), \dot{p}(t), \omega(t))$



Controllo del moto di un manipolatore

Il controllo del moto può essere realizzato nello

- **spazio dei giunti**
- **spazio operativo**



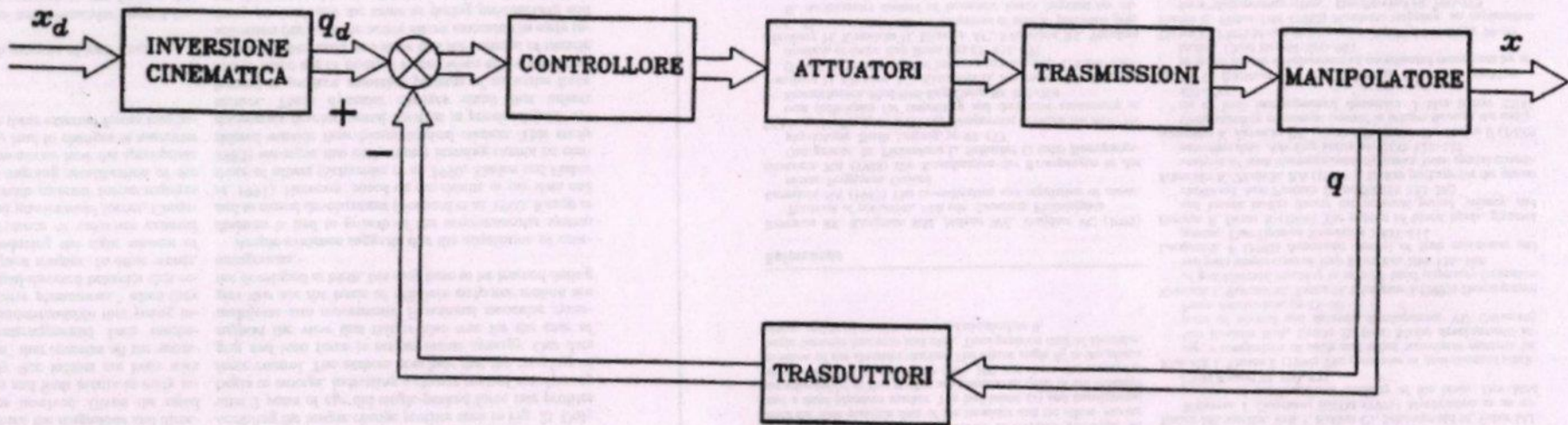
Controllo del moto nello spazio dei giunti

- Muovere il braccio da x_i a x_d espresse nello spazio operativo del robot senza interessarsi alla traiettoria percorsa dall'organo terminale del braccio
- Si determina la posizione finale del braccio nello spazio dei giunti q_d tramite la cinematica inversa

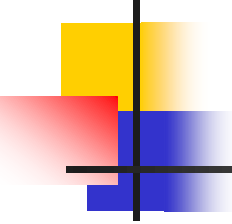
$$q_d = K^{-1}(x_d)$$

- Si muovono i giunti dalla posizione attuale q_i alla posizione desiderata q_d

Controllo del moto nello spazio dei giunti



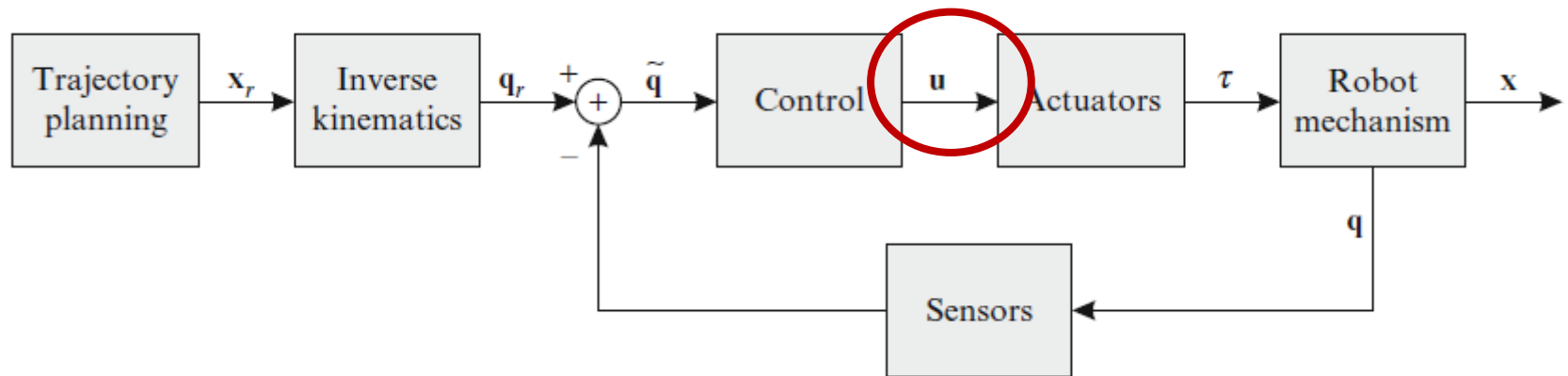
L'inversione cinematica viene effettuata al di fuori del ciclo di controllo



Controllo del moto nello spazio dei giunti

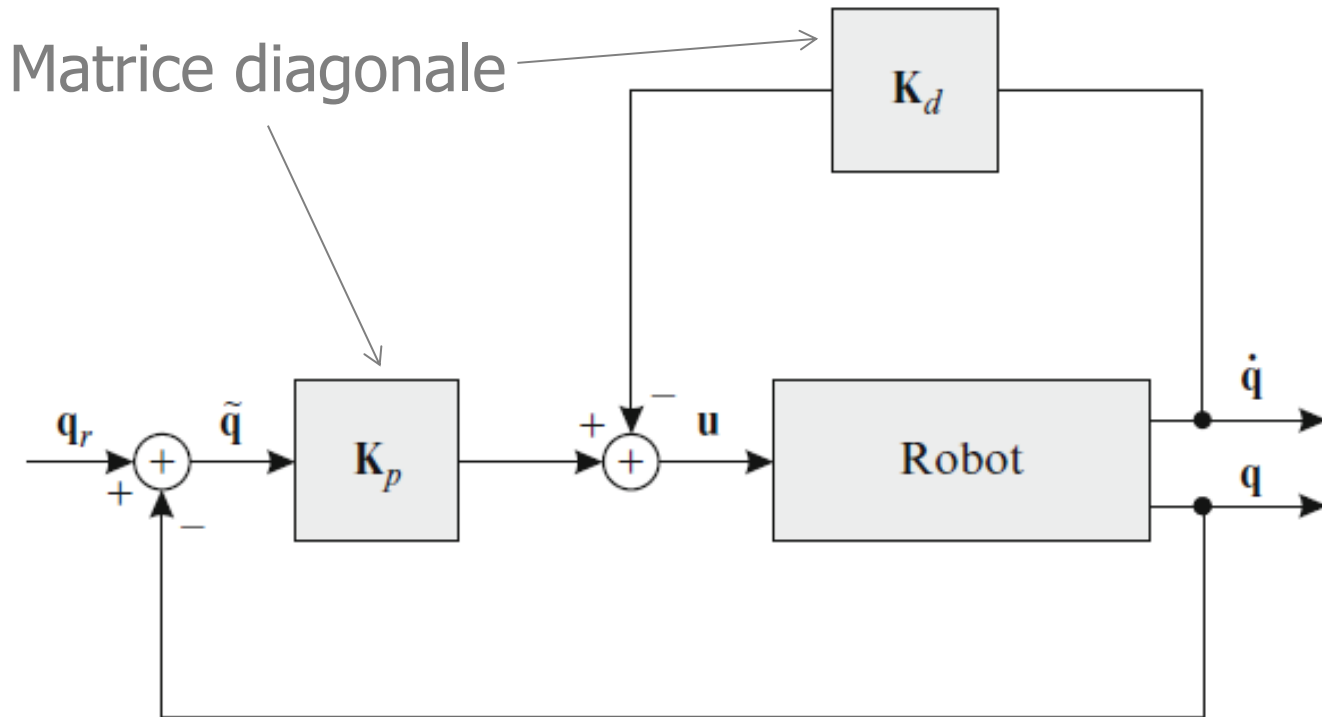
- Nell'effettuazione del movimento l'organo terminale del manipolatore esegue nello spazio operativo un percorso non prevedibile, a causa degli effetti non lineari introdotti dalla cinematica diretta

Schema generale di controllo di un robot nello spazio dei giunti



Schema di controllo nello spazio dei giunti

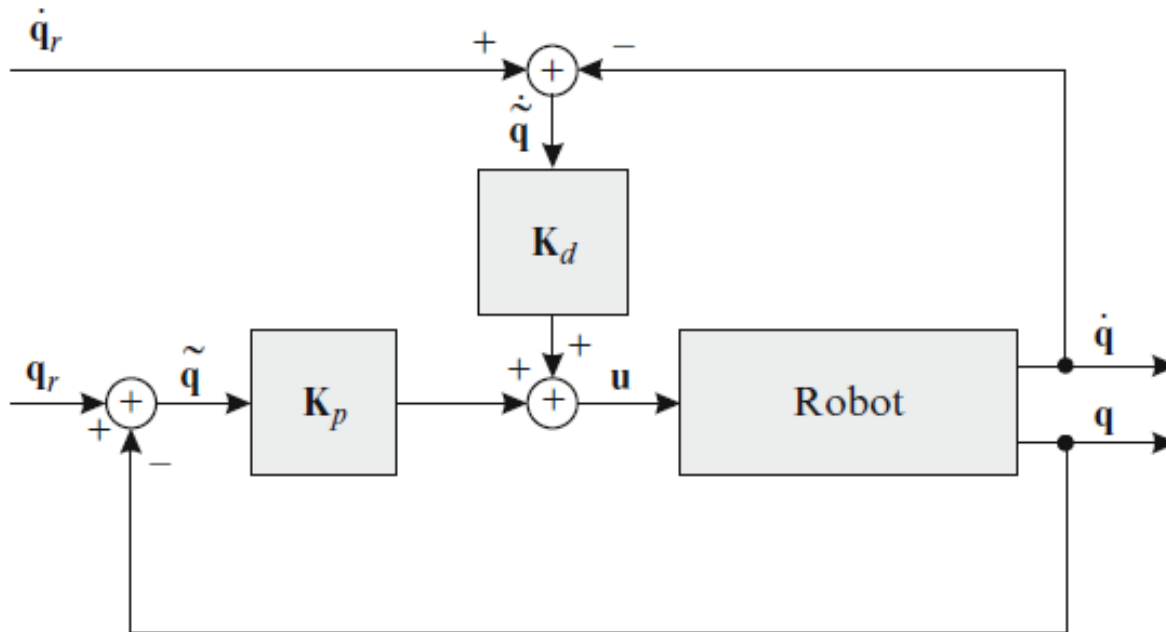
Controllo in posizione PD



$$\mathbf{u} = \mathbf{K}_p(\mathbf{q}_r - \mathbf{q}) - \mathbf{K}_d\dot{\mathbf{q}},$$

Schema di controllo nello spazio dei giunti

Controllo PD



$$\mathbf{u} = \mathbf{K}_p(\mathbf{q}_r - \mathbf{q}) + \mathbf{K}_d(\dot{\mathbf{q}}_r - \dot{\mathbf{q}})$$



Schema di controllo nello spazio dei giunti

Controllo PD

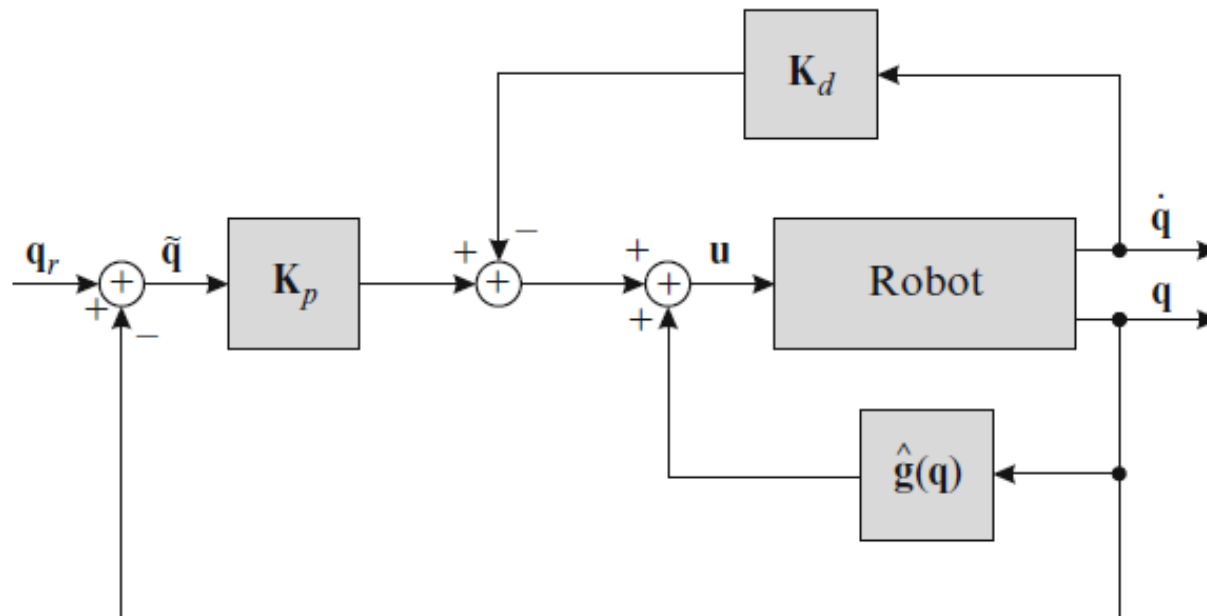
Determinazione delle matrici dei parametri K_p e K_d :

- Risposta veloce: K_p elevati
- K_d determina il damping migliore e garantisce risposta veloce senza oscillazioni
- I parametri K vanno determinati indipendentemente per ogni giunto

Schema di controllo nello spazio dei giunti

Controllo PD con compensazione di gravità

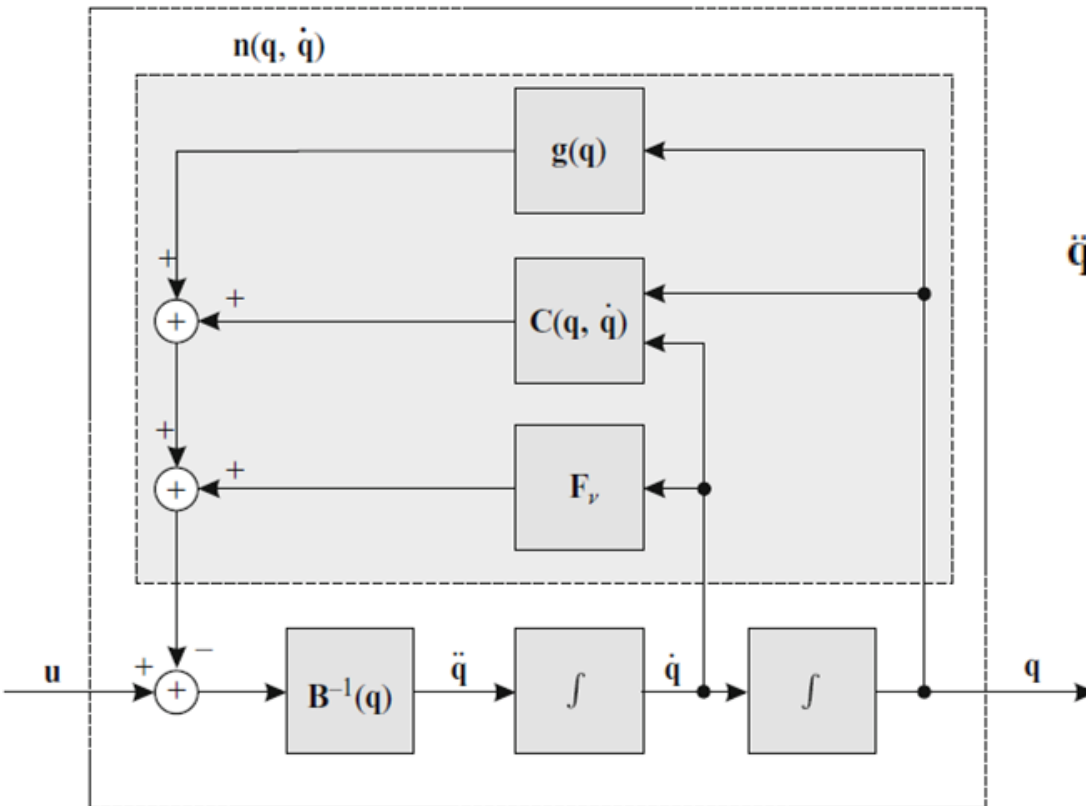
$$\mathbf{u} = \mathbf{K}_p(\mathbf{q}_r - \mathbf{q}) - \mathbf{K}_d\dot{\mathbf{q}} + \hat{\mathbf{g}}(\mathbf{q}).$$



Schema di controllo nello spazio dei giunti basato sulla dinamica inversa

Robot

$n(\mathbf{q}, \dot{\mathbf{q}})$



$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{u}.$$

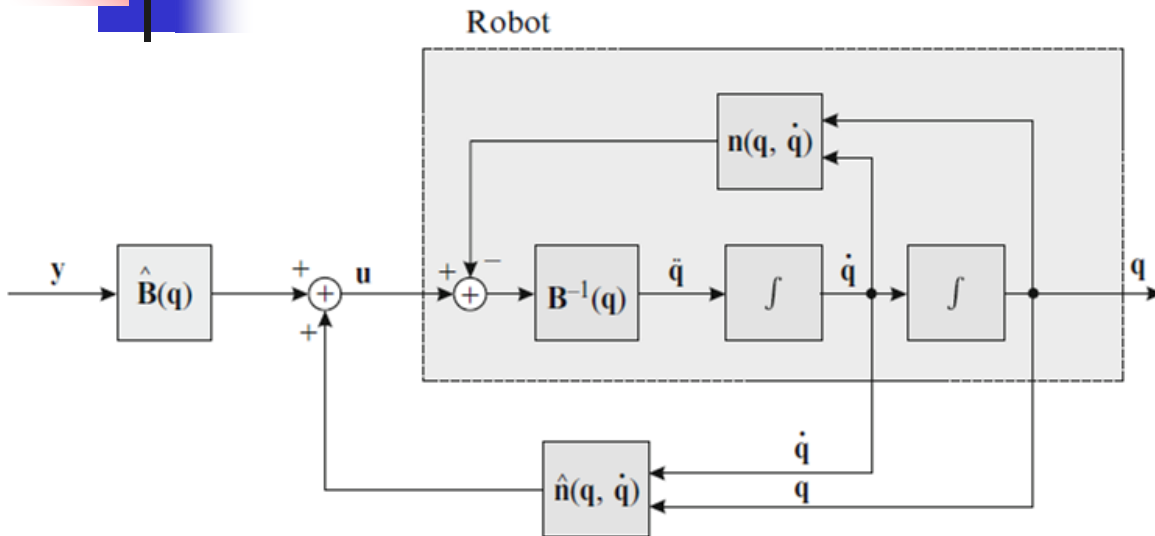
$$\ddot{\mathbf{q}} = \mathbf{B}^{-1}(\mathbf{q}) (\mathbf{u} - (\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}))).$$

$$\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}).$$

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}.$$

$$\ddot{\mathbf{q}} = \mathbf{B}^{-1}(\mathbf{q}) (\mathbf{u} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})).$$

Schema di controllo nello spazio dei giunti basato sulla dinamica inversa



$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{u}.$$

$$\ddot{\mathbf{q}} = \mathbf{B}^{-1}(\mathbf{q}) (\mathbf{u} - (\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}))).$$

Fig. 7.7 Linearization of the control system by implementing the inverse dynamic model

Let us assume that the robot dynamic model is known. The inertial matrix $\hat{\mathbf{B}}(\mathbf{q})$ is an approximation of the real values $\mathbf{B}(\mathbf{q})$, while $\hat{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})$ represents an approximation of $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$ as follows

$$\hat{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) = \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{F}}_v\dot{\mathbf{q}} + \hat{\mathbf{g}}(\mathbf{q}).$$

$$\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}).$$

The controller output \mathbf{u} is determined by the following equation

$$\mathbf{u} = \hat{\mathbf{B}}(\mathbf{q})\mathbf{y} + \hat{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}),$$

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}.$$

where the approximate inverse dynamic model of the robot was used.

$$\ddot{\mathbf{q}} = \mathbf{B}^{-1}(\mathbf{q}) (\mathbf{u} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})).$$

Schema di controllo nello spazio dei giunti basato sulla dinamica inversa

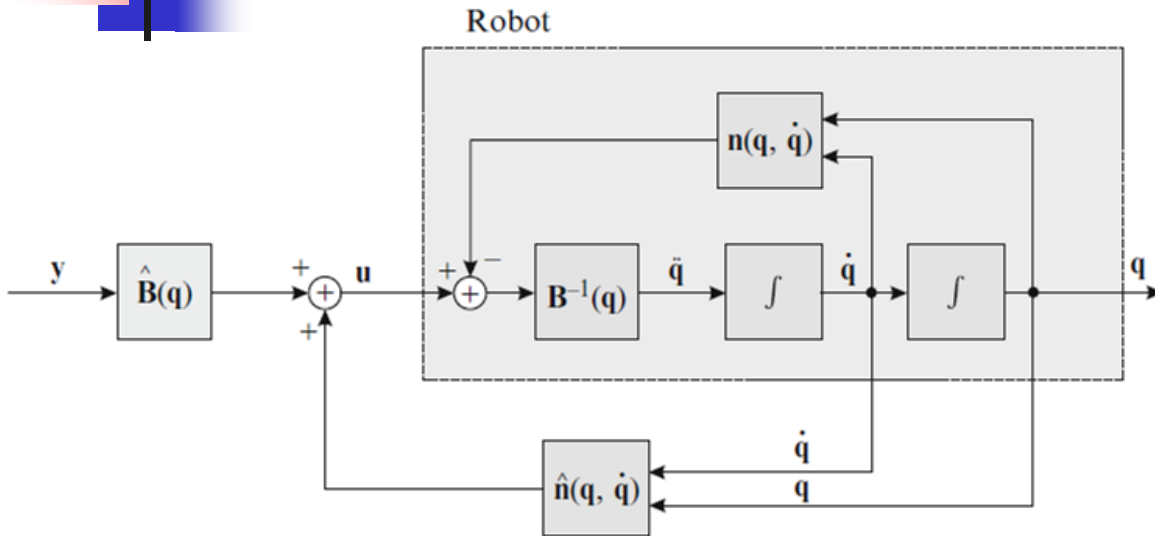
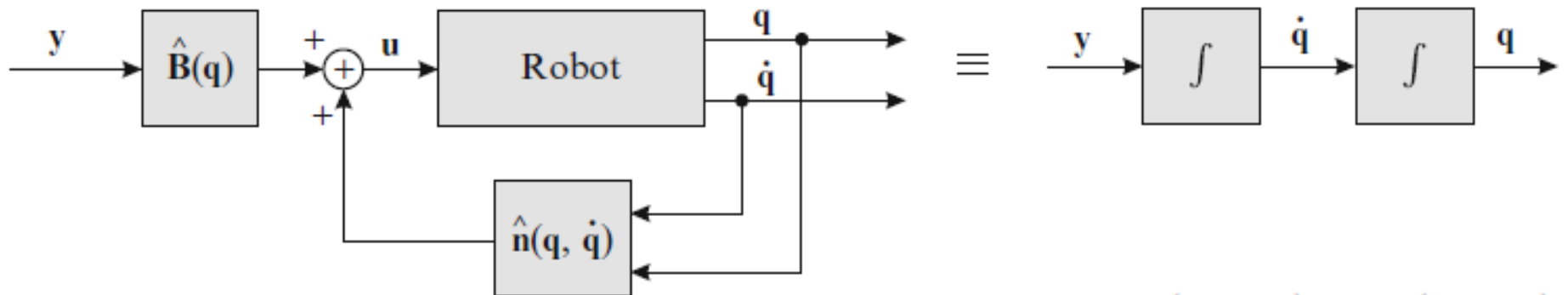


Fig. 7.7 Linearization of the control system by implementing the inverse dynamic model



$$y = \ddot{q}_r + \mathbf{K}_p(\mathbf{q}_r - \mathbf{q}) + \mathbf{K}_d(\dot{\mathbf{q}}_r - \dot{\mathbf{q}}).$$

Schema di controllo nello spazio dei giunti basato sulla dinamica inversa

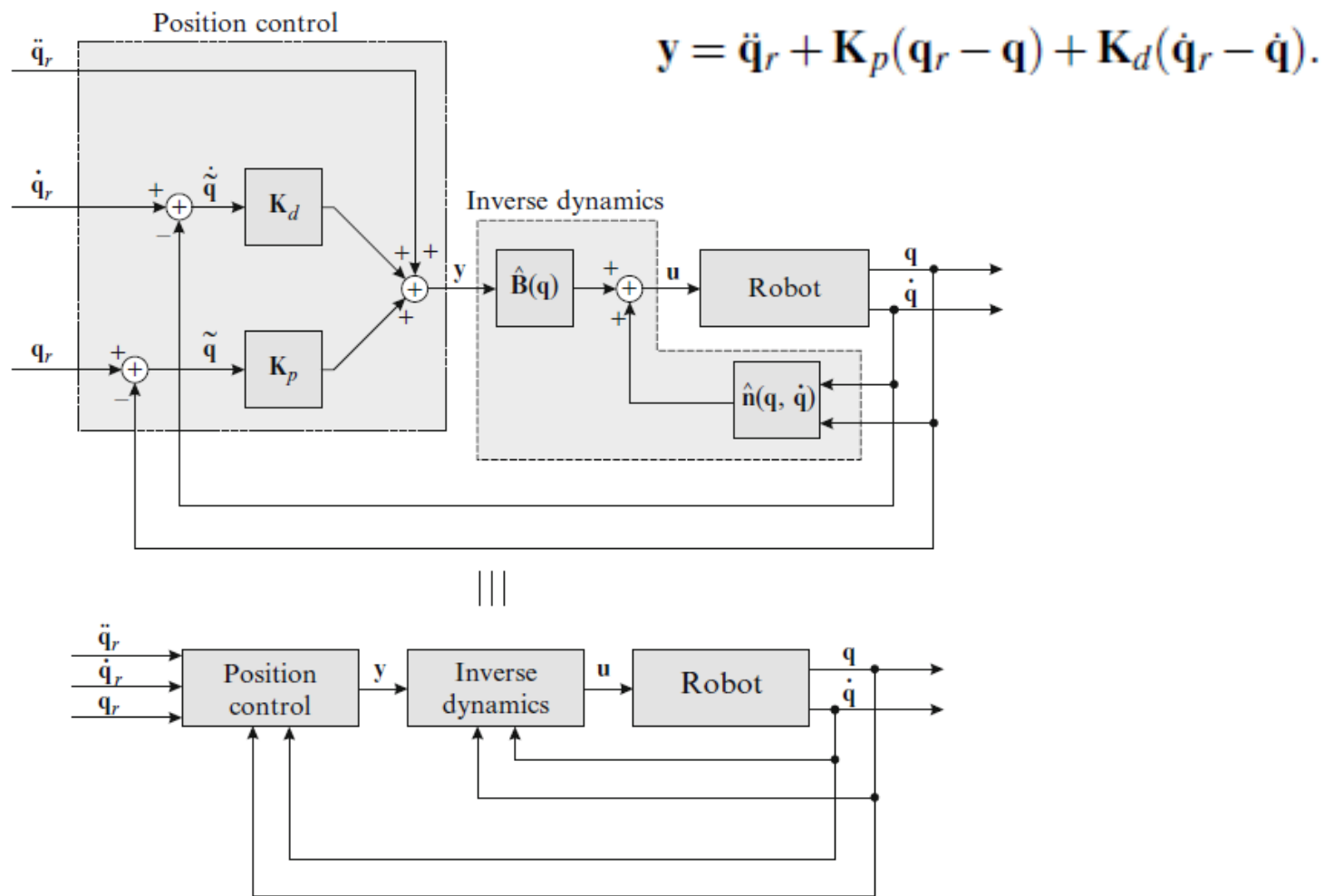


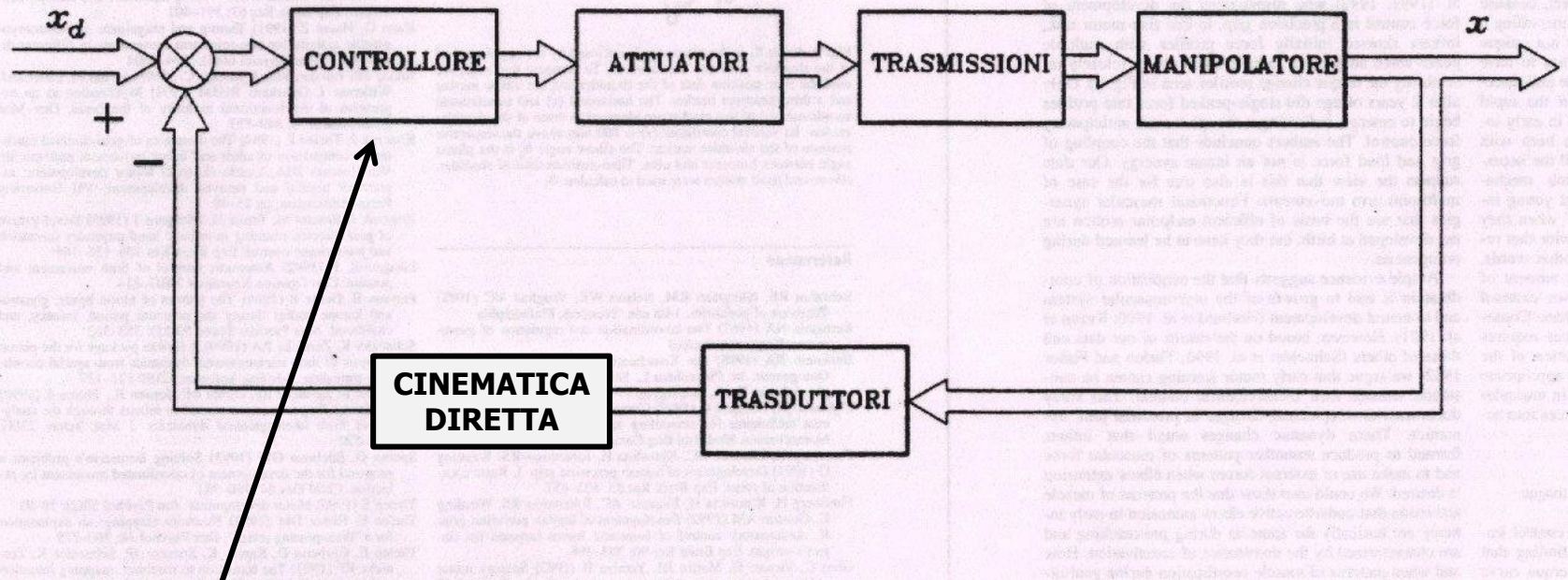
Fig. 7.9 Control of the robot based on inverse dynamics



Controllo del moto nello spazio operativo

- Nell'effettuazione del movimento da x_i a x_d l'organo terminale del manipolatore esegue nello spazio operativo una traiettoria in accordo ad una legge prestabilita
- Es. traiettoria lineare o curvilinea

Controllo del moto nello spazio operativo



L'inversione cinematica viene effettuata all'interno del ciclo di controllo



Controllo del moto nello spazio operativo

- far eseguire al manipolatore la traiettoria pianificata $(t, p(t), \Phi(t), \dot{p}(t), \omega(t))$
- determinare velocità e accelerazioni dei giunti nel tempo per raggiungere la posizione finale desiderata espressa nelle coordinate cartesiane (calcolo dello Jacobiano)
- determinare tensioni e correnti da applicare ai motori per far assumere ai giunti velocità e accelerazioni calcolate dallo Jacobiano



Cinematica differenziale

Determinazione della relazione tra le **velocità dei giunti** e le corrispondenti **velocità angolari e lineari** dell'organo terminale.

Tali legami sono descritti da una matrice di trasformazione (Jacobiano) dipendente dalla configurazione del manipolatore.



Cinematica differenziale

Jacobiano geometrico = matrice di trasformazione dipendente dalla configurazione corrente del braccio

$$v = \begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} = J(q)\dot{q}$$

$J(q)$ = Jacobiano geometrico

\dot{p} = velocità dell'effettore finale

ω = velocità di rotazione dell'effettore finale

\dot{q} = velocità ai giunti



Cinematica differenziale

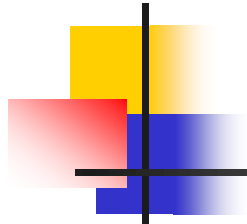
Determinare le velocità ai giunti data la velocità nello spazio operativo

$$v = \begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} = J(q)\dot{q}$$

$$\dot{q} = J^{-1}(q) \begin{bmatrix} \dot{p} \\ \omega \end{bmatrix} \quad J^{-1} \text{ è l'inversa dello Jacobiano}$$

Metodi di integrazione numerica consentono di ricavare il vettore q dal vettore delle velocità ai giunti

Cinematica e dinamica dei robot



Cinematica

$$x = k(q)$$
$$q = k^{-1}(x)$$

$$x = \begin{bmatrix} x \\ y \\ z \\ \varphi \\ \vartheta \\ \psi \end{bmatrix} \quad q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix}$$

$k(\cdot)$ = equazioni della cinematica diretta

Cinematica differenziale

$$\dot{x} = J(q)\dot{q}$$
$$\dot{q} = J^{-1}(q)\dot{x}$$

Spazio delle
velocità

$J(q)$ = matrice Jacobiana



Cinematica e dinamica dei robot

Dinamica

$$\tau = B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q)$$

τ = coppia

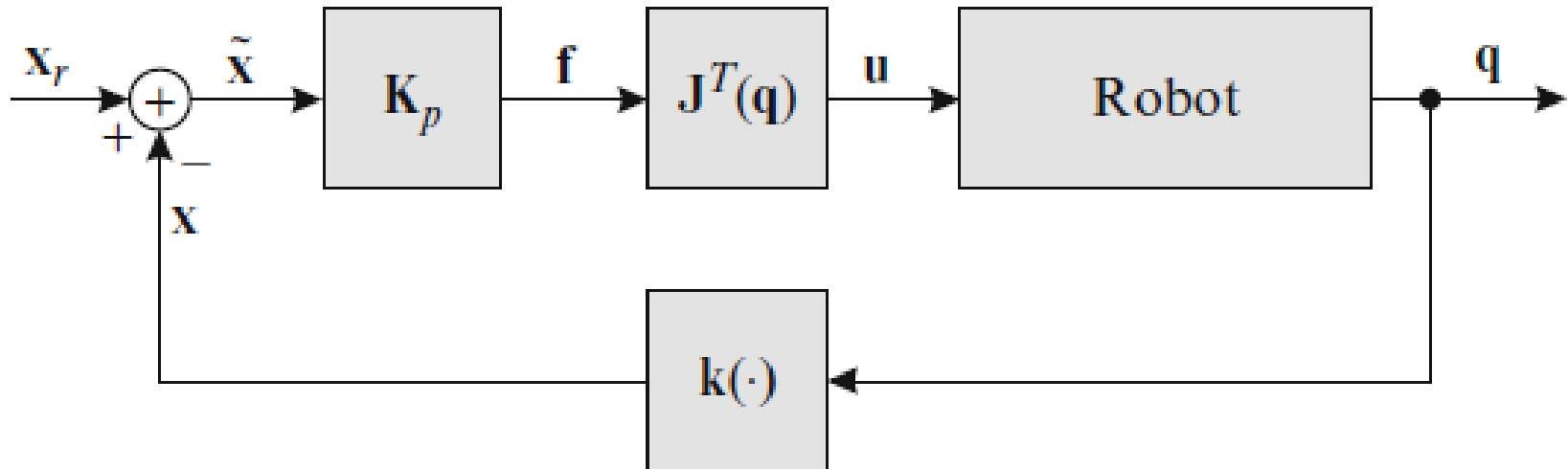
B = termine di inerzia

C = termine di Coriolis

g = termine di gravità

Schema di controllo nello spazio operativo

basato sulla matrice Jacobiana trasposta



Control based on the transposed Jacobian matrix

$$f = K_p \tilde{x}. \quad u = J^T(q) f.$$

f = forza all'end effector

Schema di controllo nello spazio operativo

basato sulla matrice Jacobiana inversa

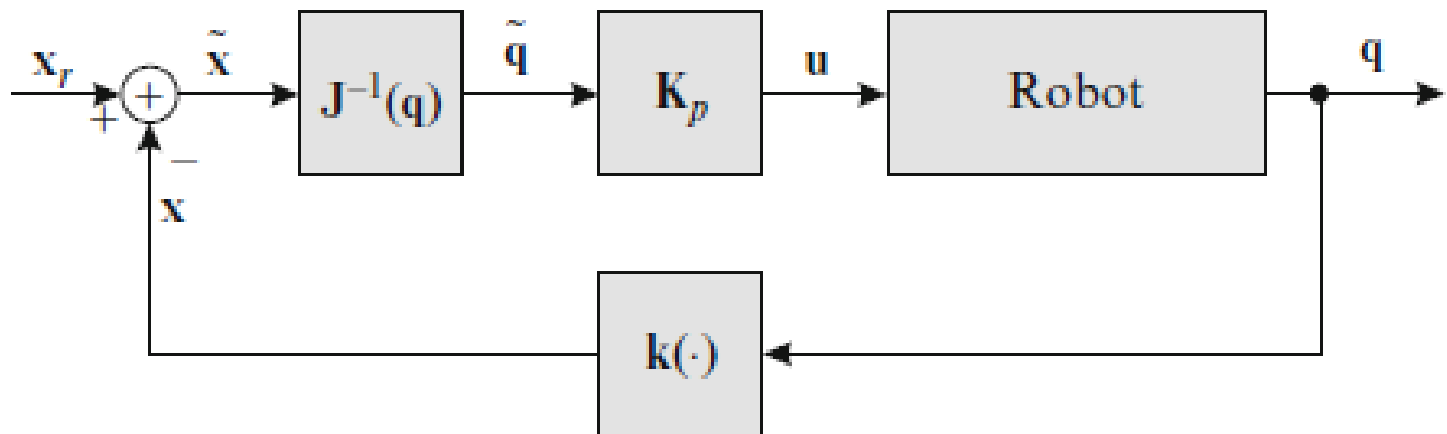


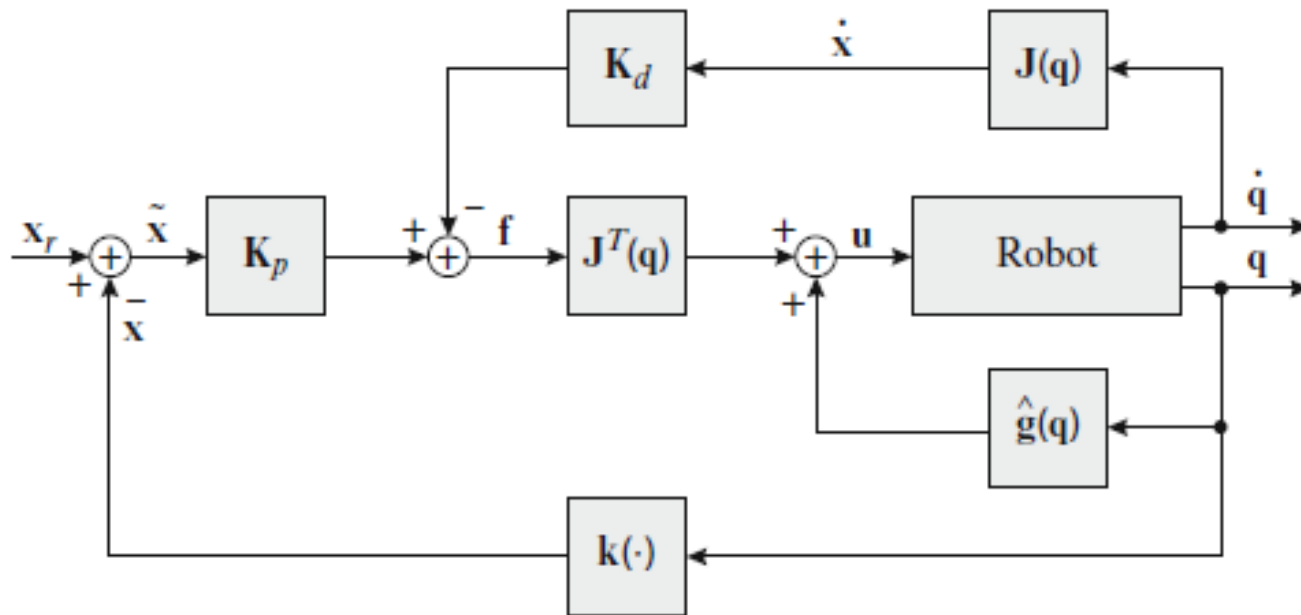
Fig. 7.11 Control based on the inverse Jacobian matrix

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad \Leftrightarrow \quad \frac{d\mathbf{x}}{dt} = \mathbf{J}(\mathbf{q})\frac{d\mathbf{q}}{dt} \quad \text{for small displacements} \quad \tilde{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{q})\tilde{\mathbf{x}}.$$

$$d\mathbf{x} = \mathbf{J}(\mathbf{q})d\mathbf{q} \quad \mathbf{u} = \mathbf{K}_p\tilde{\mathbf{q}}.$$

Schema di controllo nello spazio operativo

controllo PD con compensazione di gravità



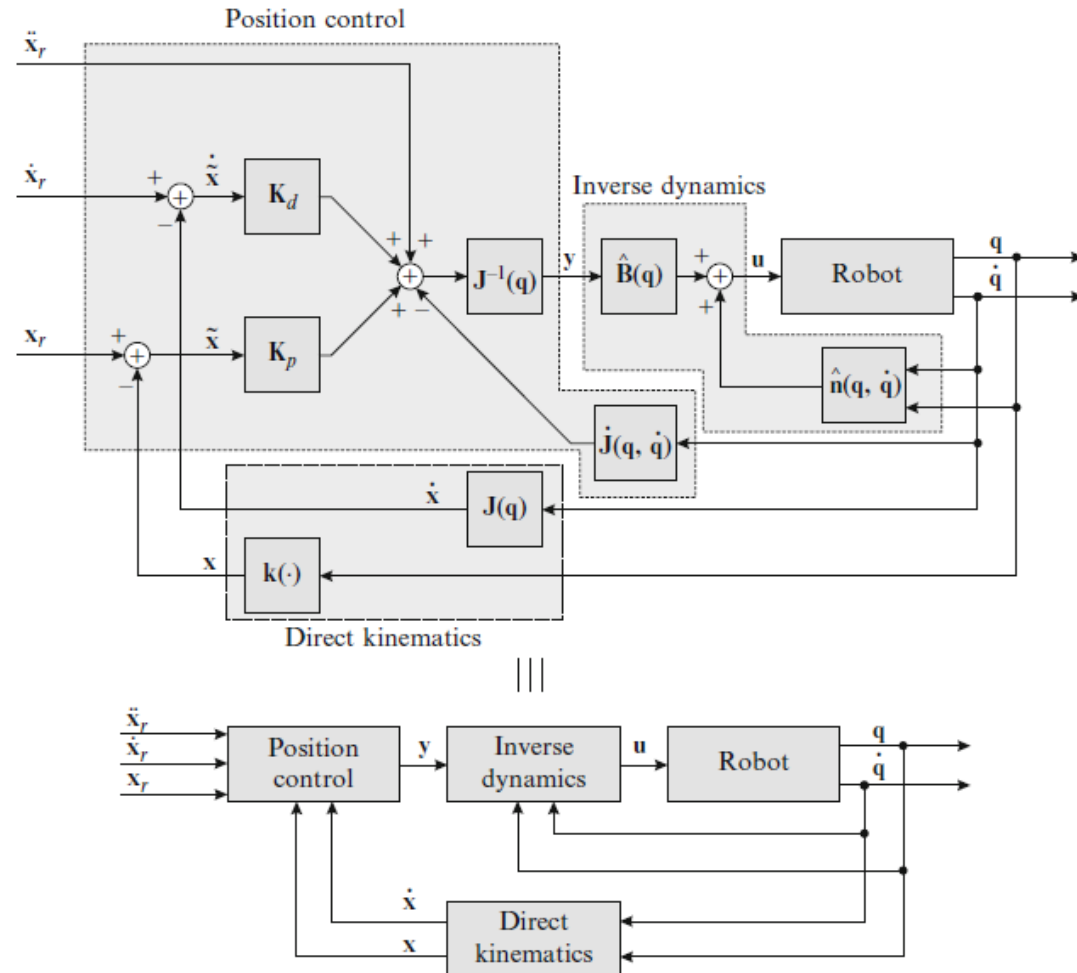
$$\dot{x} = J(q)\dot{q}.$$

$$f = K_p\tilde{x} - K_d\dot{x}.$$

$$u = J^T(q)f + \hat{g}(q).$$

PD control with gravity compensation in external coordinates

Schema di controllo nello spazio operativo basato sulla dinamica inversa



$$\dot{x} = J(q)\dot{q}. \quad (7.33)$$

By calculating the time derivative of equation (7.33), we obtain

$$\ddot{x} = \dot{J}(q)\dot{q} + J(q)\ddot{q}. \quad (7.34)$$

The error of the pose of the robot end-effector is determined as the difference between its desired and its actual pose

$$\tilde{x} = x_r - x = x_r - k(q). \quad (7.35)$$

In a similar way the velocity error of the robot end-effector is determined

$$\dot{\tilde{x}} = \dot{x}_r - \dot{x} = \dot{x}_r - J(q)\dot{q}. \quad (7.36)$$

The acceleration error is the difference between the desired and the actual acceleration

$$\ddot{\tilde{x}} = \ddot{x}_r - \ddot{x}. \quad (7.37)$$

When developing the inverse dynamics based controller in the internal coordinates, equation (7.19) was derived describing the dynamics of the control error in the form $\ddot{\tilde{q}} + K_d\dot{\tilde{q}} + K_p\tilde{q} = 0$. An analogous equation can be written for the error of the end-effector pose. From this equation the acceleration \ddot{x} of the robot end-effector can be expressed

$$\ddot{x} + K_d\dot{\tilde{x}} + K_p\tilde{x} = 0 \Rightarrow \ddot{x} = \ddot{x}_r + K_d\dot{\tilde{x}} + K_p\tilde{x}. \quad (7.38)$$

From equation (7.34) we express \ddot{q} taking into account the equality $y = \ddot{q}$

$$y = J^{-1}(q) (\ddot{x} - \dot{J}(q, \dot{q})\dot{q}). \quad (7.39)$$

By replacing \ddot{x} in equation (7.39) with expression (7.38), the control algorithm based on inverse dynamics in the external coordinates is obtained

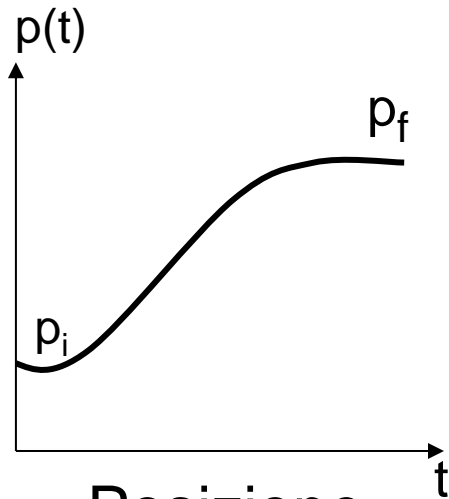
$$y = J^{-1}(q) (\ddot{x}_r + K_d\dot{\tilde{x}} + K_p\tilde{x} - \dot{J}(q, \dot{q})\dot{q}). \quad (7.40)$$

The control scheme encompassing the linearization of the system based on inverse dynamics (7.31) and the closed loop control (7.40) is shown in Figure 7.13.

Fig. 7.13 Robot control based on inverse dynamics in external coordinates

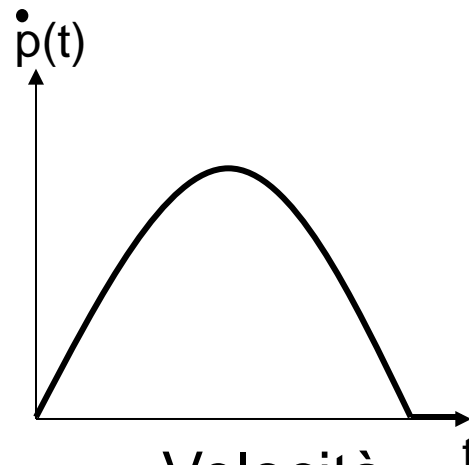
Controllo del moto nello spazio operativo

Pianificatore delle traiettorie



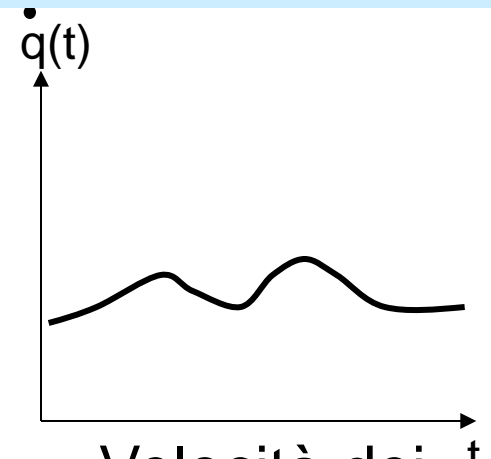
Posizione
nello spazio
operativo nel
tempo

$(t, p(t), \Phi(t), \dot{p}(t), \omega(t))$



Velocità
dell'effettore
finale nel
tempo

$J^{-1}(p(t))$



Velocità dei
giunti nel
tempo

$J^{-1}(p(t))$



$(t, \dot{q}(t))$

Controllo in forza

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} - \mathbf{J}^T(\mathbf{q})\mathbf{f} \quad \text{Effetto delle forze esterne}$$

$$\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}),$$

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} - \mathbf{J}^T(\mathbf{q})\mathbf{f}.$$

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{J}^T(\mathbf{q})\mathbf{f} = \mathbf{u}.$$

$$\ddot{\mathbf{q}} = \mathbf{B}^{-1}(\mathbf{q}) (\mathbf{u} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{J}^T(\mathbf{q})\mathbf{f}).$$

$$\mathbf{u} = \hat{\mathbf{B}}(\mathbf{q})\mathbf{y} + \hat{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{J}^T(\mathbf{q})\mathbf{f},$$

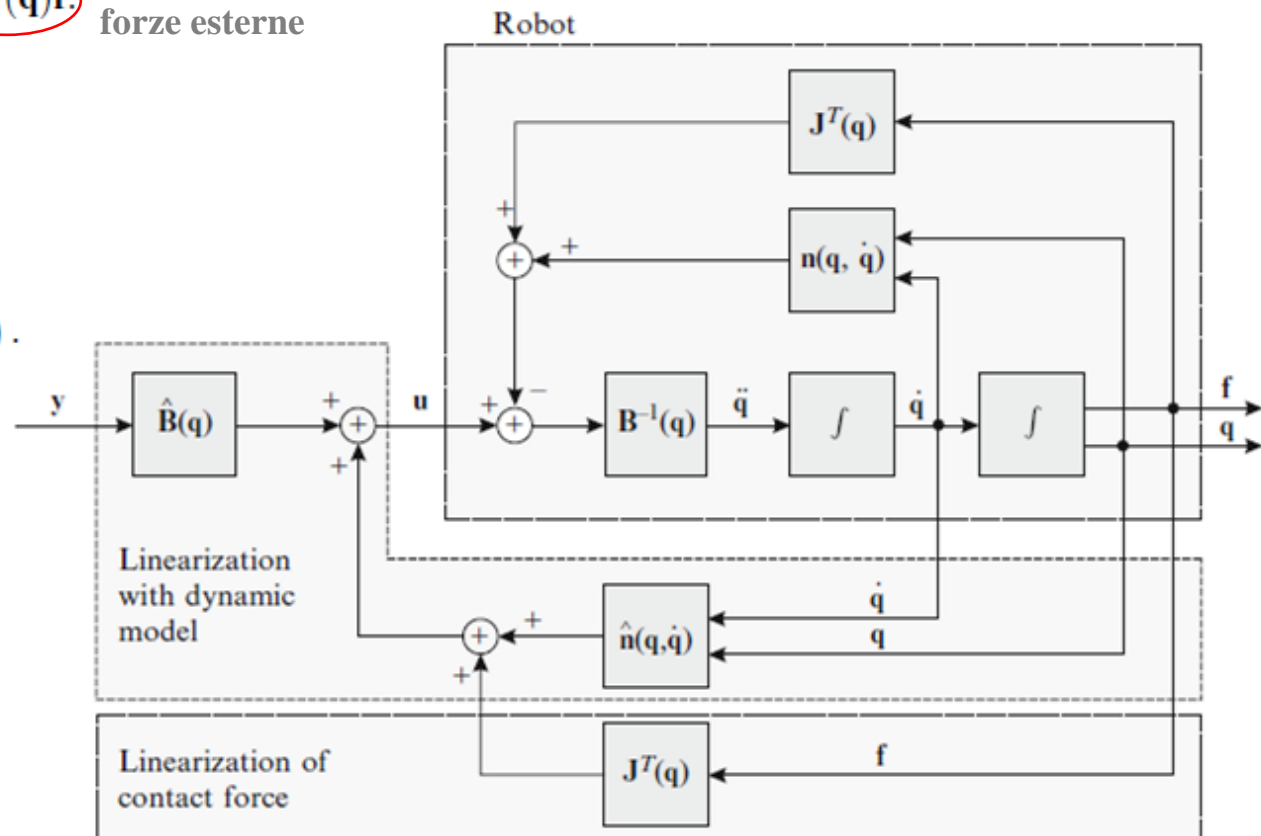


Fig. 7.14 Linearization of the control system by implementing the inverse dynamic model and the measured contact force

Controllo in forza

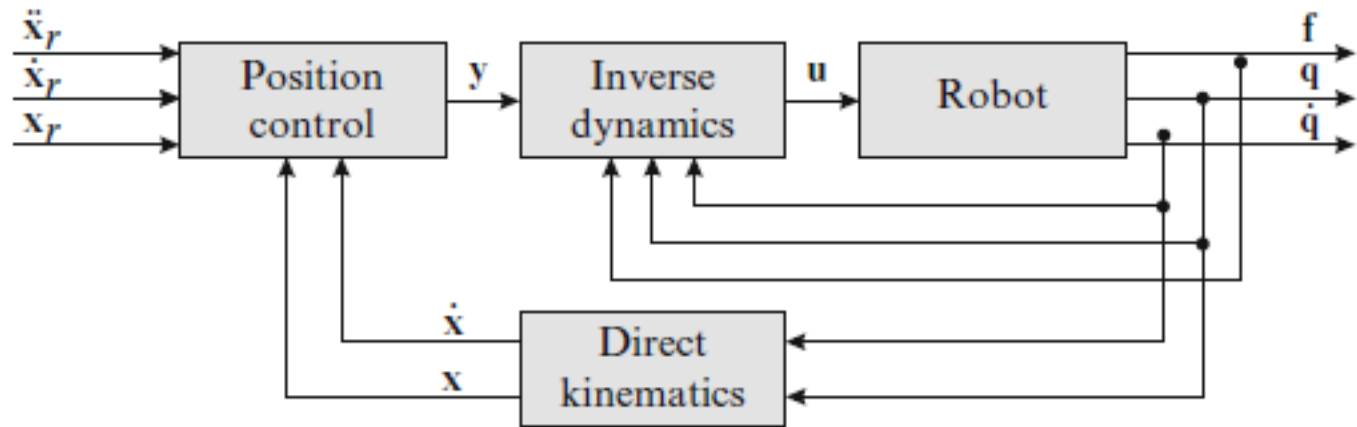
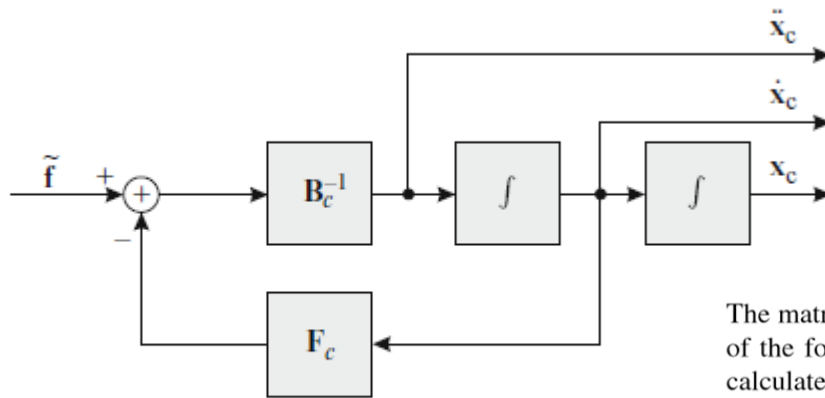


Fig. 7.15 Robot control based on inverse dynamics in external coordinates including the contact force

Controllo in forza



$$\tilde{\mathbf{f}} = \mathbf{B}_c \ddot{\mathbf{x}}_c + \mathbf{F}_c \dot{\mathbf{x}}_c. \quad (7.48)$$

The matrices \mathbf{B}_c and \mathbf{F}_c determine the movement of the object under the influence of the force $\tilde{\mathbf{f}}$. From equation (7.48) the acceleration of the virtual object can be calculated

$$\ddot{\mathbf{x}}_c = \mathbf{B}_c^{-1} (\tilde{\mathbf{f}} - \mathbf{F}_c \dot{\mathbf{x}}_c). \quad (7.49)$$

Fig. 7.16 Force control translated into control of the pose of robot end-effector

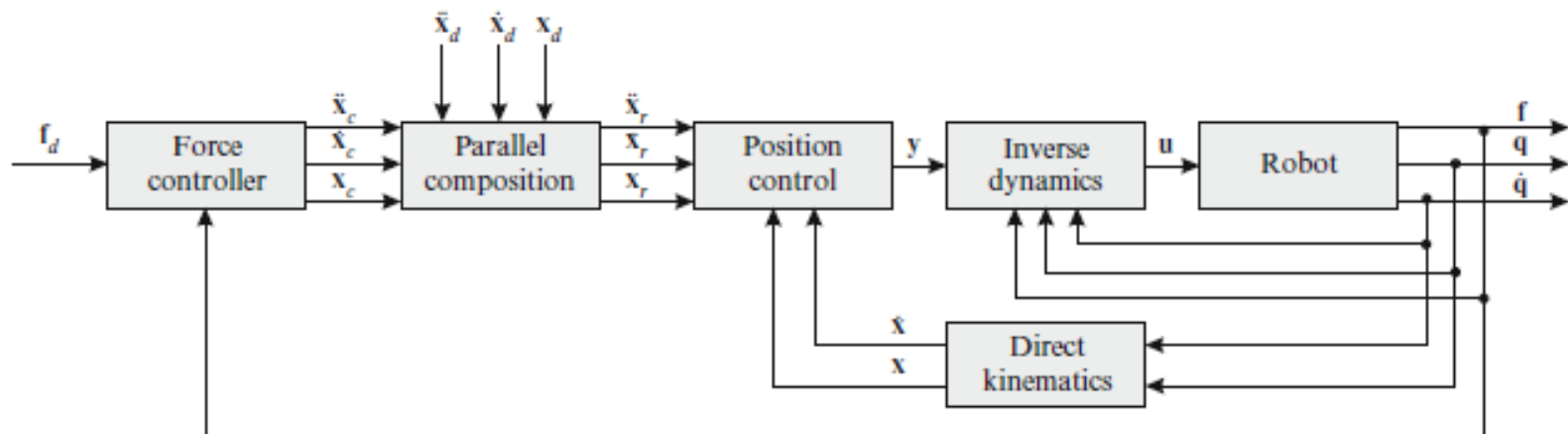


Fig. 7.17 Direct force control in the external coordinates



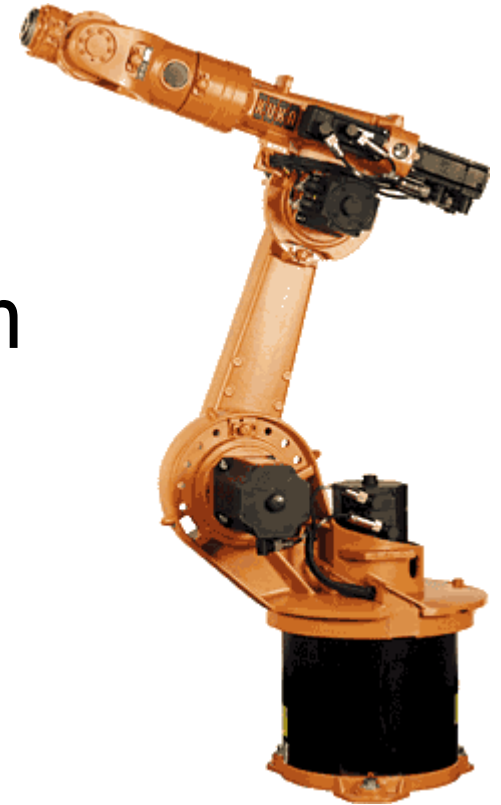
Performance di un manipolatore industriale

- **Payload:** massimo carico sollevabile
- **Velocità:** velocità massima di movimento nello spazio operativo
- **Accuratezza:** scostamento tra la posizione calcolata sulla base dei parametri di targa dal sistema di controllo e la posizione reale
- **Ripetibilità:** misura della capacità del manipolatore di tornare in una posizione precedentemente assunta (funzione del sistema e degli algoritmi di controllo, oltre che delle caratteristiche meccaniche del robot)



KUKA KR 15/2

- Dof: 6
- Payload: 15 kg
- Max. reach: 1570 mm
- Repeatability: $< \pm 0.1$ mm
- Weight: 222 kg



Il manipolatore PUMA 560

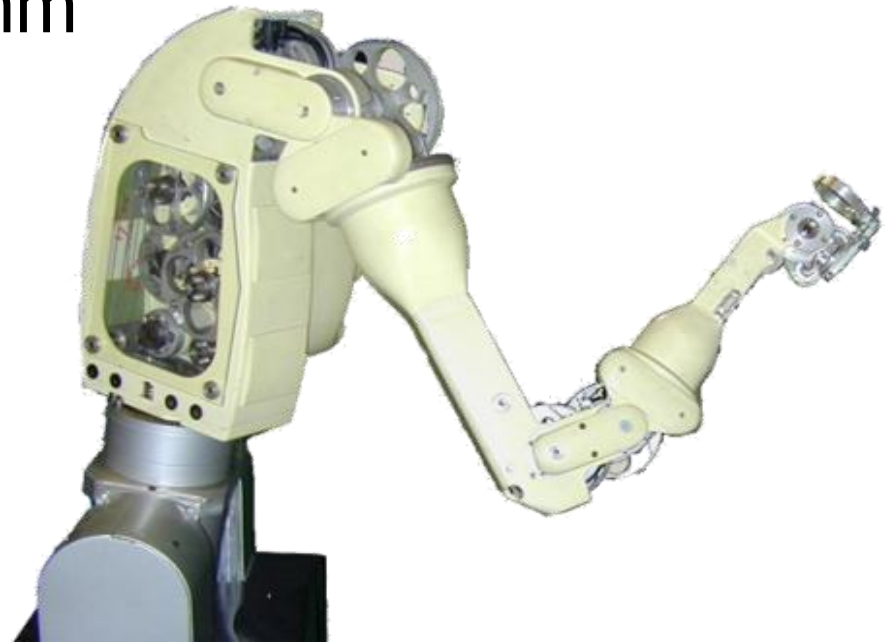
- Dof: 6
- Payload: 2 kg
- Velocità dell'effettore: 1.0 m/s
- Ripetibilità: $< \pm 0.1$ mm
- Peso: 120 lb = 55 Kg





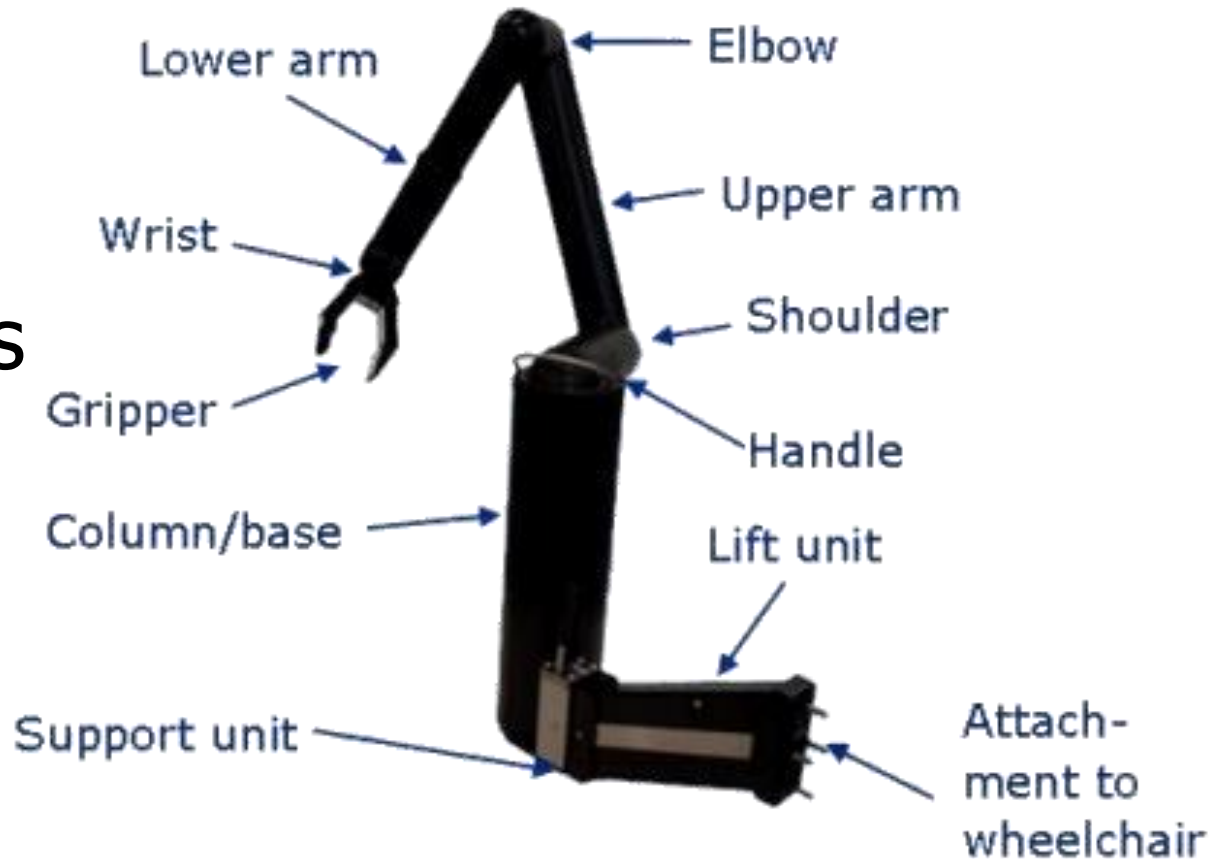
Dexter Arm

- Cable actuated
- d.o.f.: 8
- Workspace: 1200 mm x 350°
- Repeatability: ± 1 mm
- Velocity: 0.2 m/s
- Payload: 2 Kg
- Weight: 40 Kg



Manus

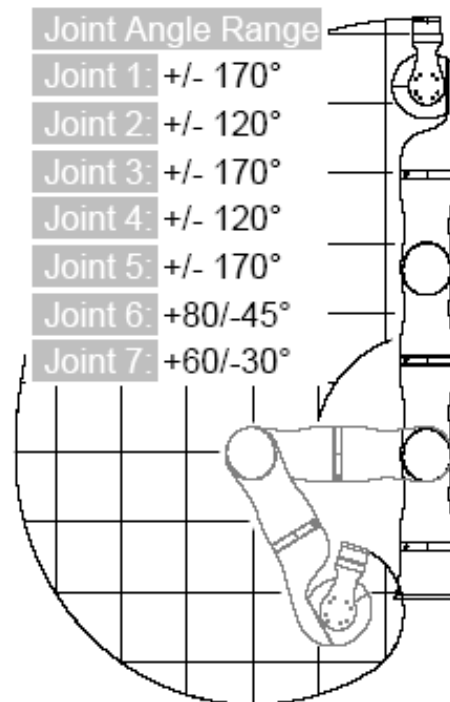
- Cable actuated
- d.o.f.: 6
- Velocity: 0.2 m/s
- Payload: 2 Kg
- Power: 24V DC
- Weight: 12 Kg

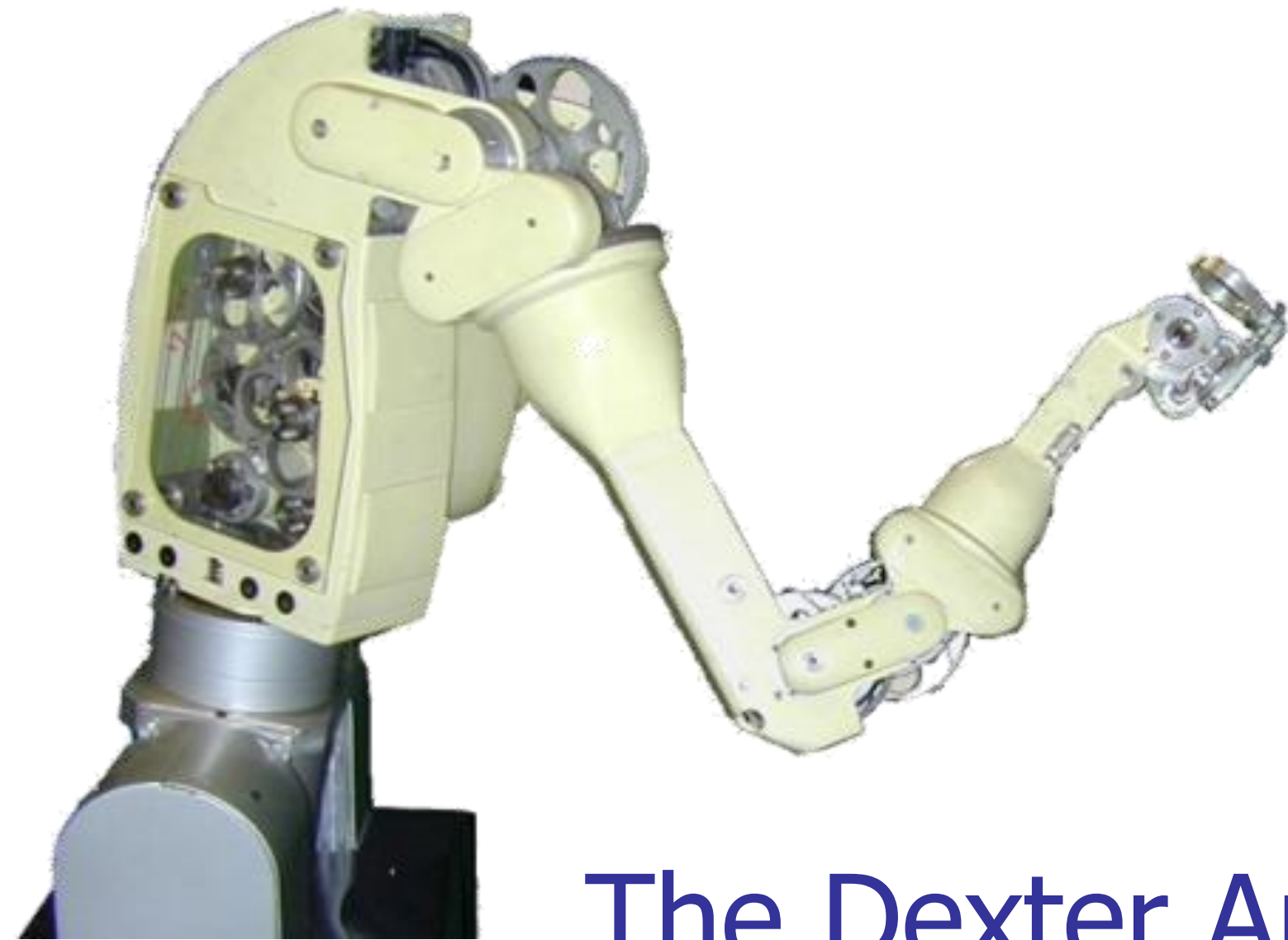


DLR Arm



Total Weight	14 kg
Max. Payload	14 kg
Max. Joint Speed	120°/s
Nr. of Axes	7 (R - P - R - P - R - P - P)
Maximum Reach	936 mm
Motors	DLR-Robodrive
Gears	Harmonic Drive
Sensors (each Joint)	2 Position, 1 Torque Sensor
Sensor (wrist)	6-DOF Force/Torque Sensor
Brakes	Electromagnetic Safety Brake
Power Supply	48 V DC
Control	Position-, Torque-, Impedance Control Control Cycles: Current 40 kHz; Joint 3 kHz; Cartesian 1 kHz
Electronics	Integrated Electronics, internal Cabling, Communications by optical SERCOS-Bus

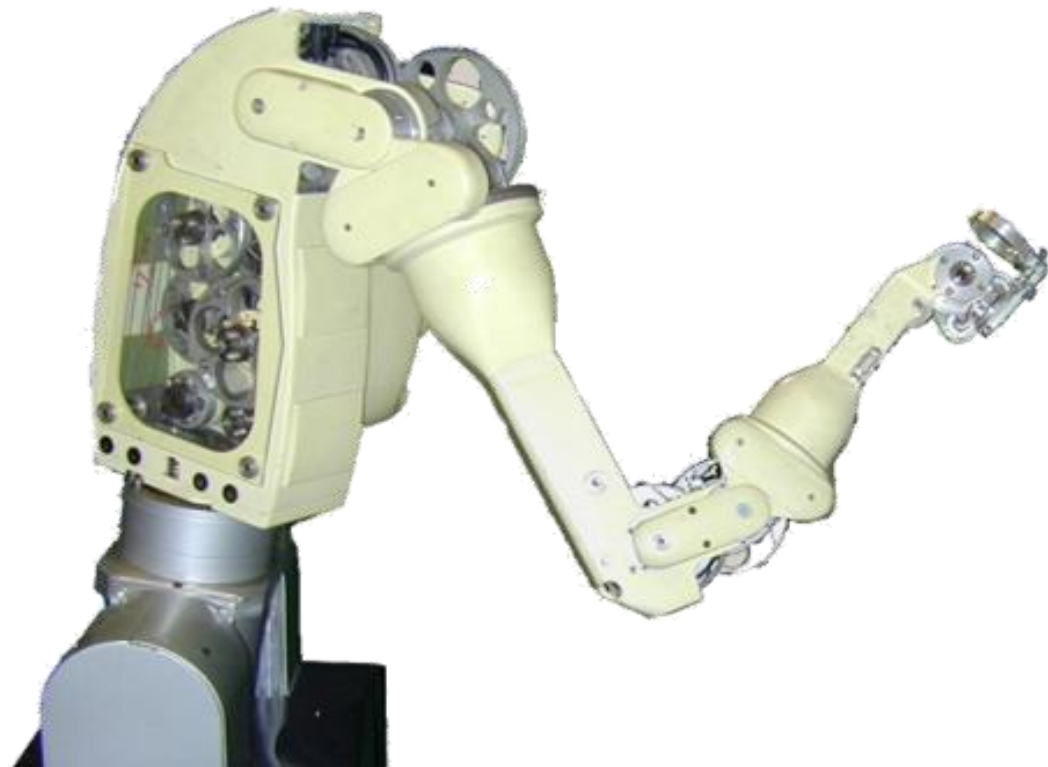




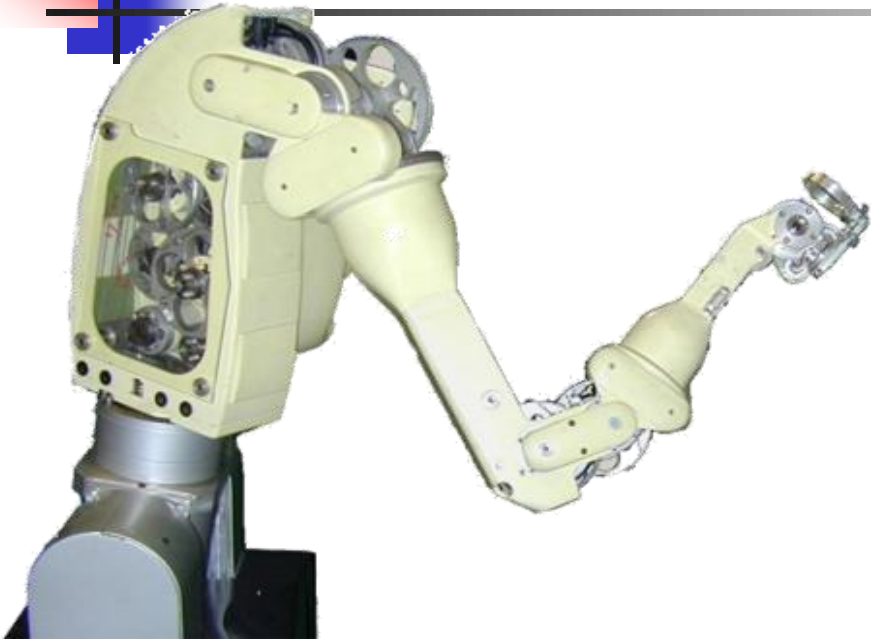
The Dexter Arm

The Dexter Arm

- Workspace: 1200 mm x 350°
- Repeatability: ± 1 mm
- Velocity: 0.2 m/s
- Payload: 2 Kg
- D.o.f.: 8
- Power: 24V DC
- Weight: 40 Kg



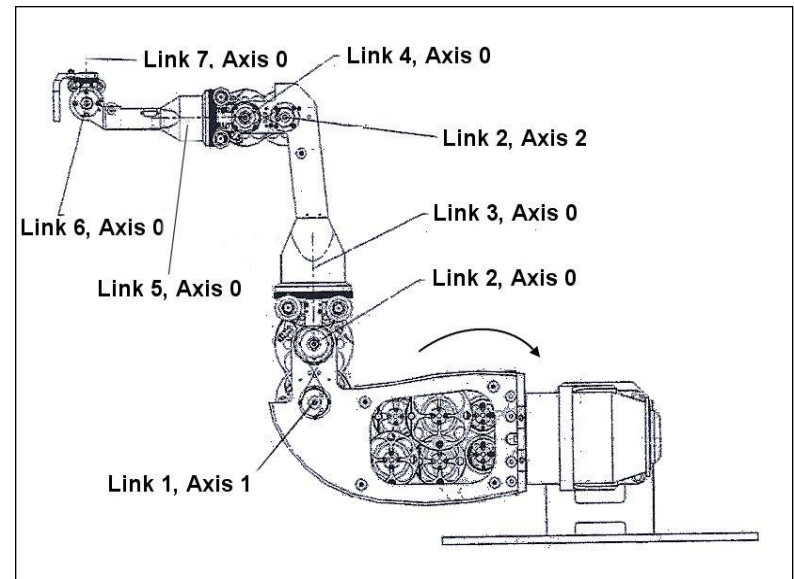
The Dexter Arm



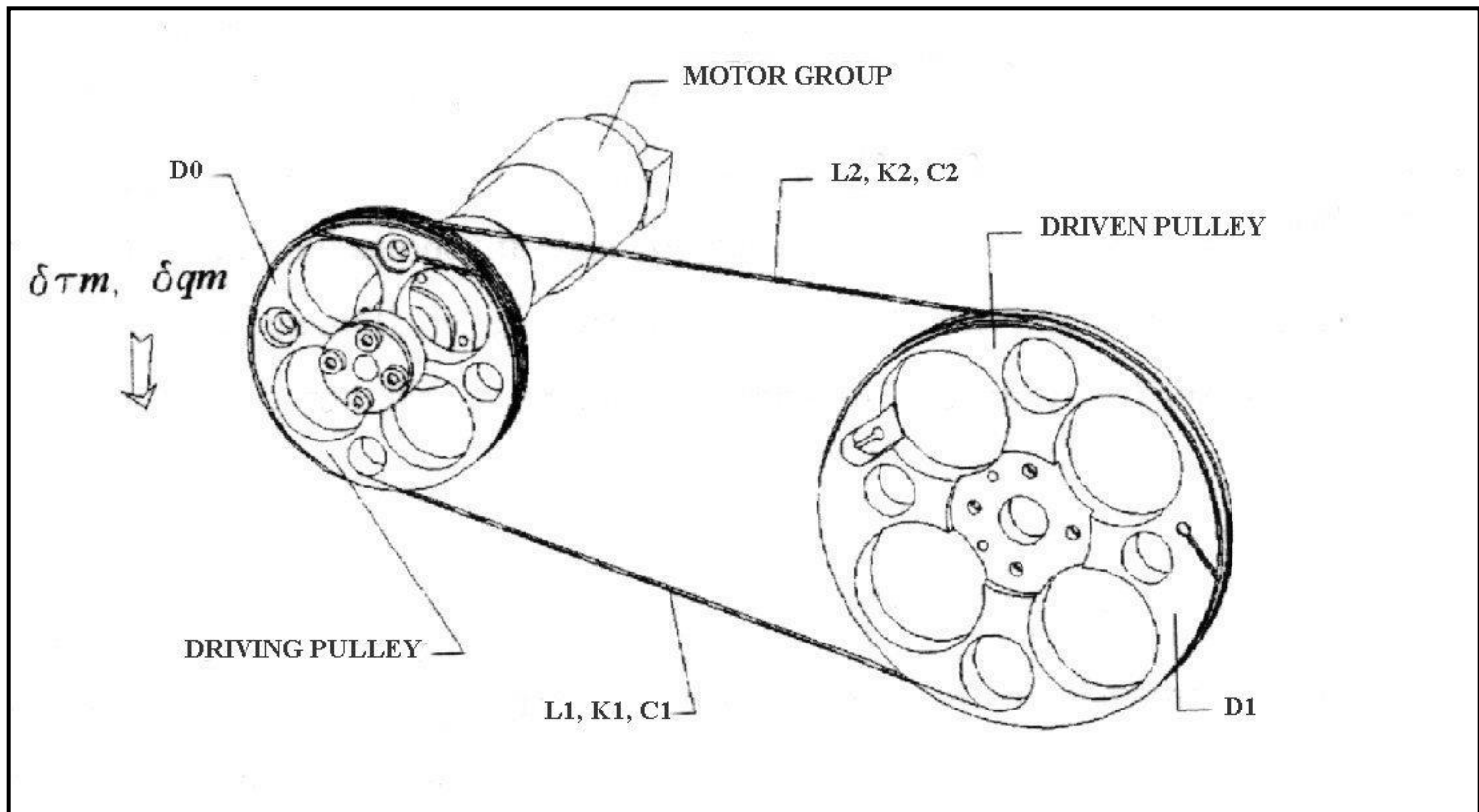
- 8-d.o.f. anthropomorphic redundant robot arm, composed of trunk, shoulder, elbow and wrist
 - designed for service applications and personal assistance in residential sites, such as houses or hospitals
-
- mechanically coupled structure: the mechanical transmission system is realized with pulleys and steel cables
 - main characteristics: reduced accuracy, lighter mechanical structure, safe and intrinsically compliant structure

The Dexter arm

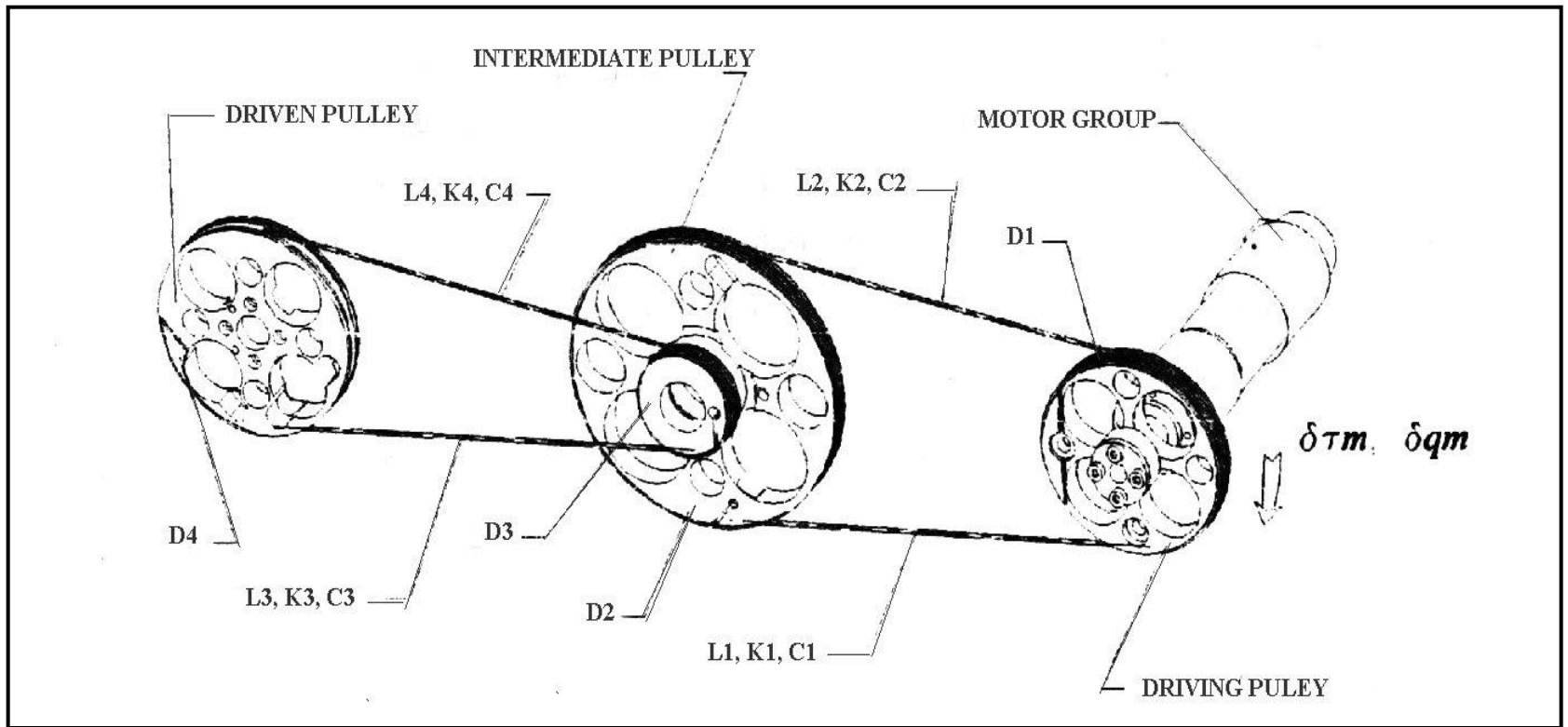
- Transmission system realized with pulleys and steel cables
- Joints J0 and J1 are actuated by motors and driving gear-boxes directly connected to the articulation axis
- Joints J2,...,J7 are actuated by DC-motors installed on link 1



Transmission #6



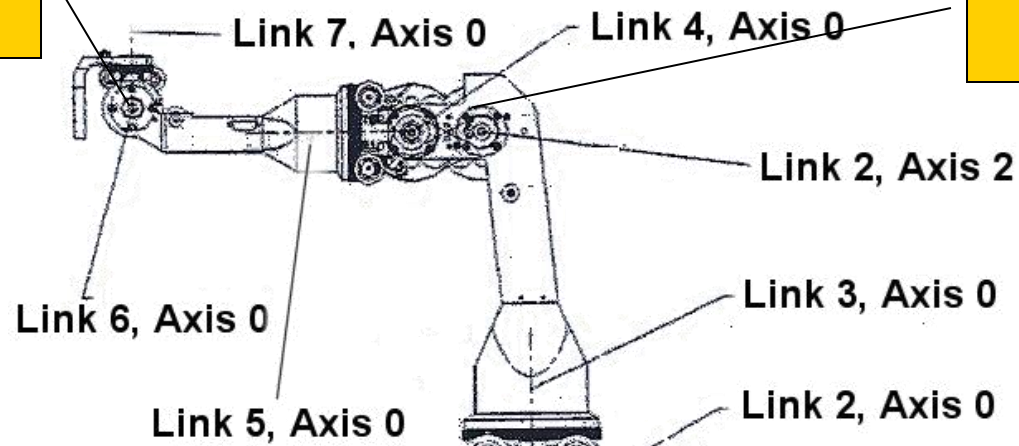
Transmissions #2-5 and 7



Anthropomorphic structure

Wrist

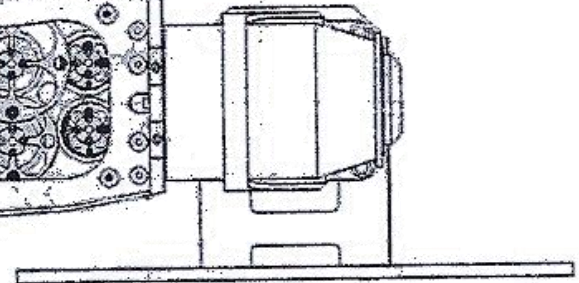
Elbow



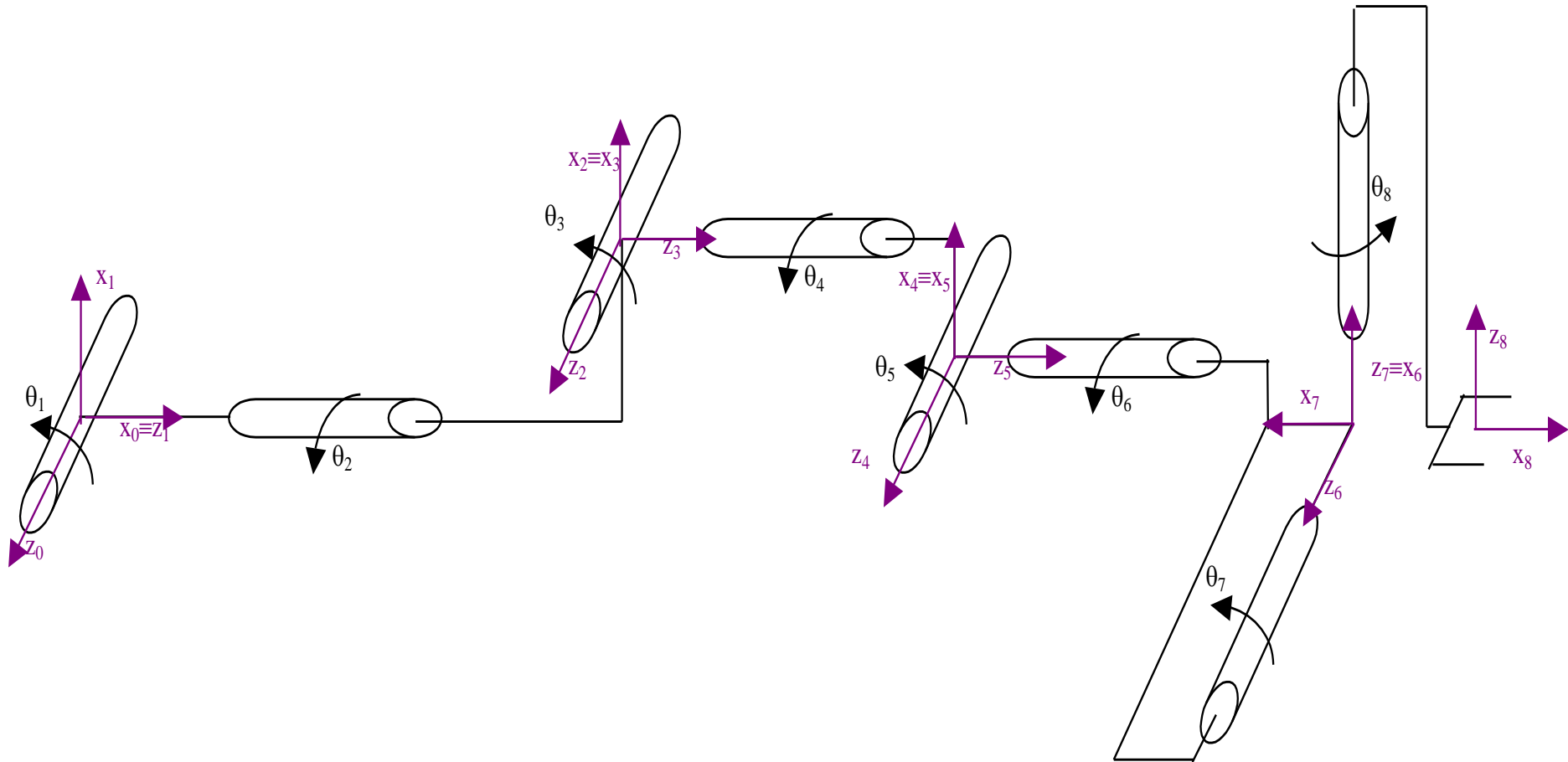
Shoulder

Link 1, Axis 1

Trunk



Kinematic Configuration

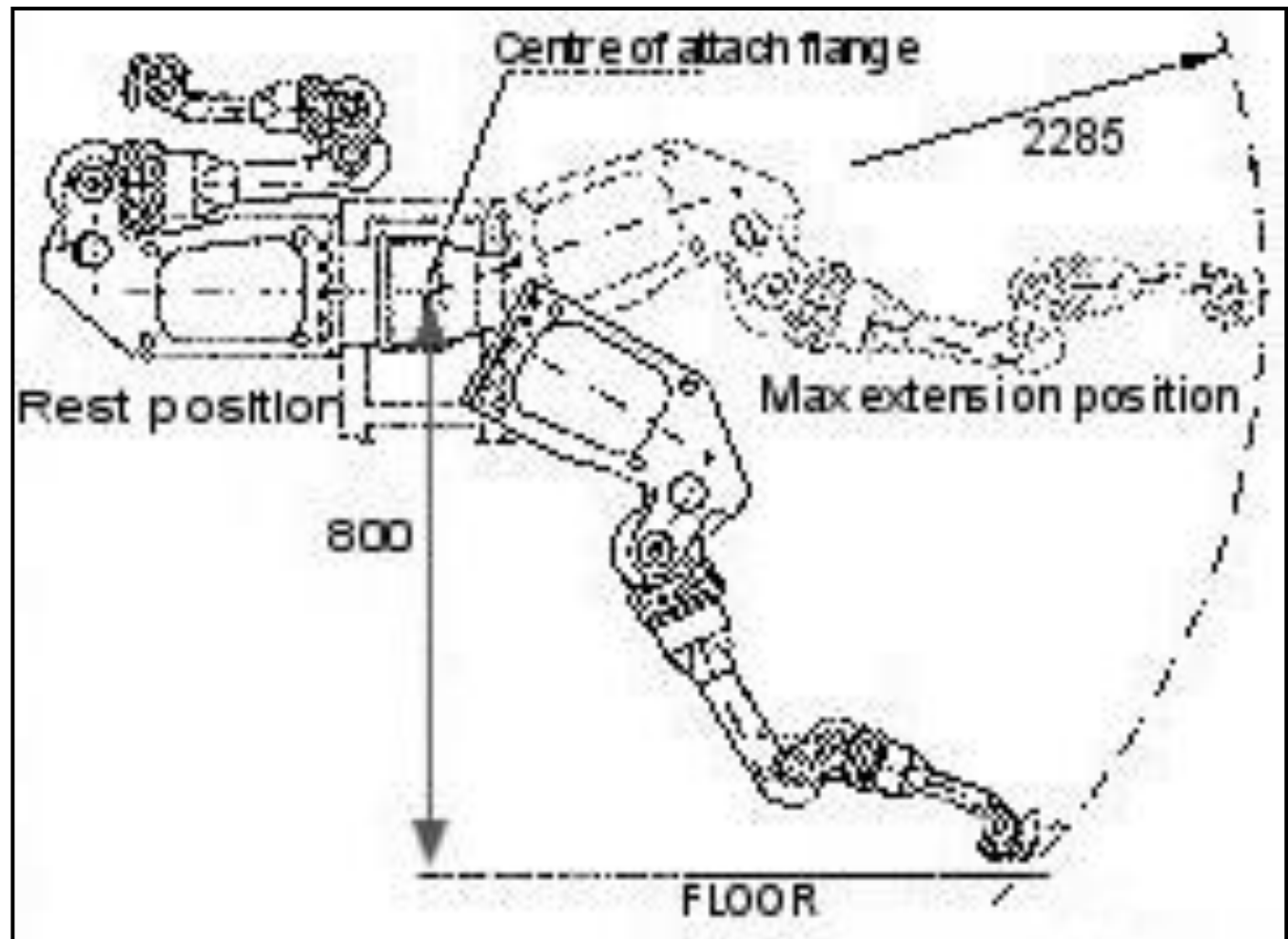




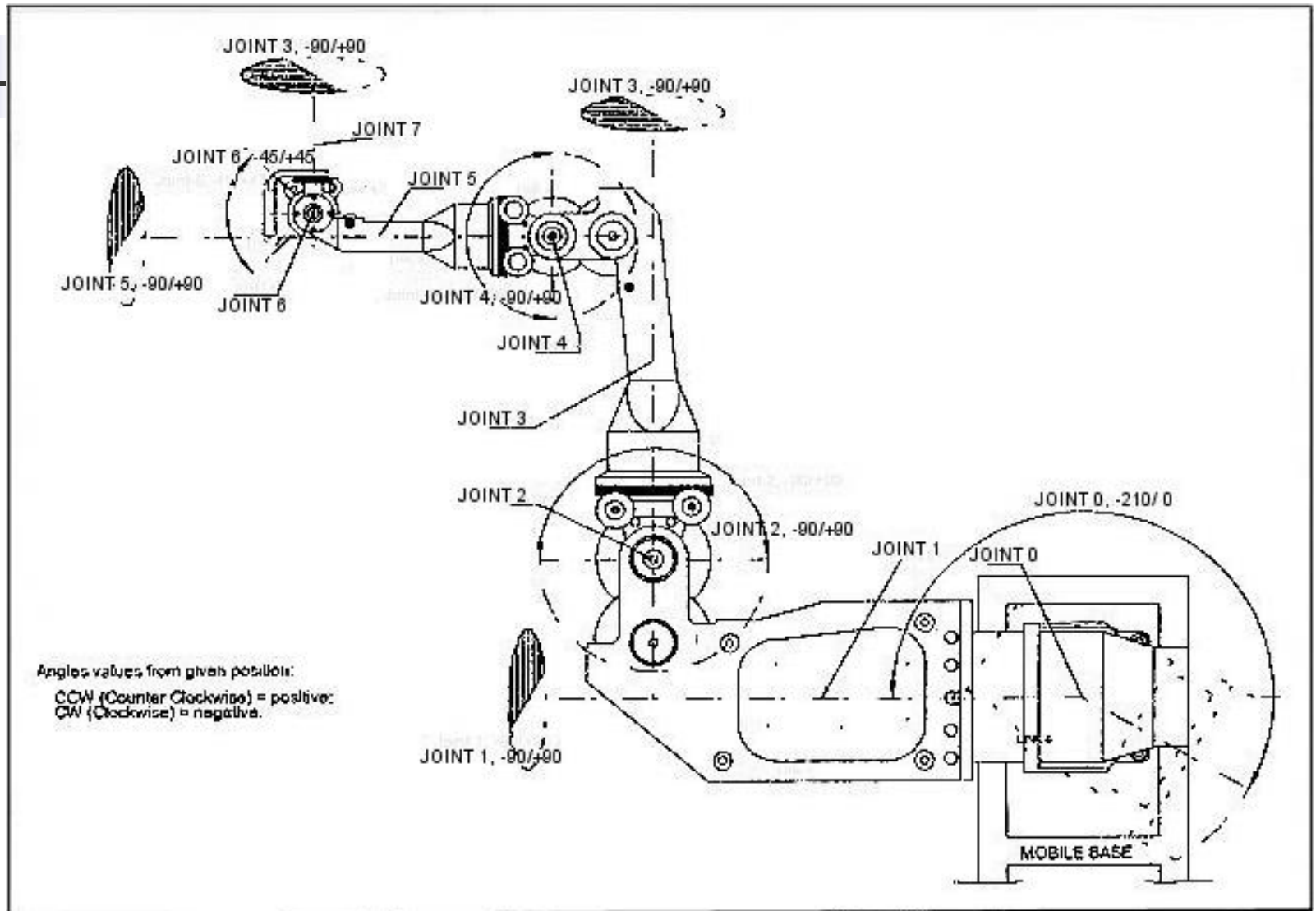
Denavit-Hartenberg Parameters

<i>Joint</i>	a_i [mm]	d_i [mm]	α_i [rad]	θ_i [rad]
1	0	0	$\pi/2$	θ_1
2	144	450	$-\pi/2$	θ_2
3	0	0	$\pi/2$	θ_3
4	-100	350	$-\pi/2$	θ_4
5	0	0	$\pi/2$	θ_5
6	-24	250	$-\pi/2$	θ_6
7	0	0	$\pi/2$	θ_7
8	100	0	0	θ_8

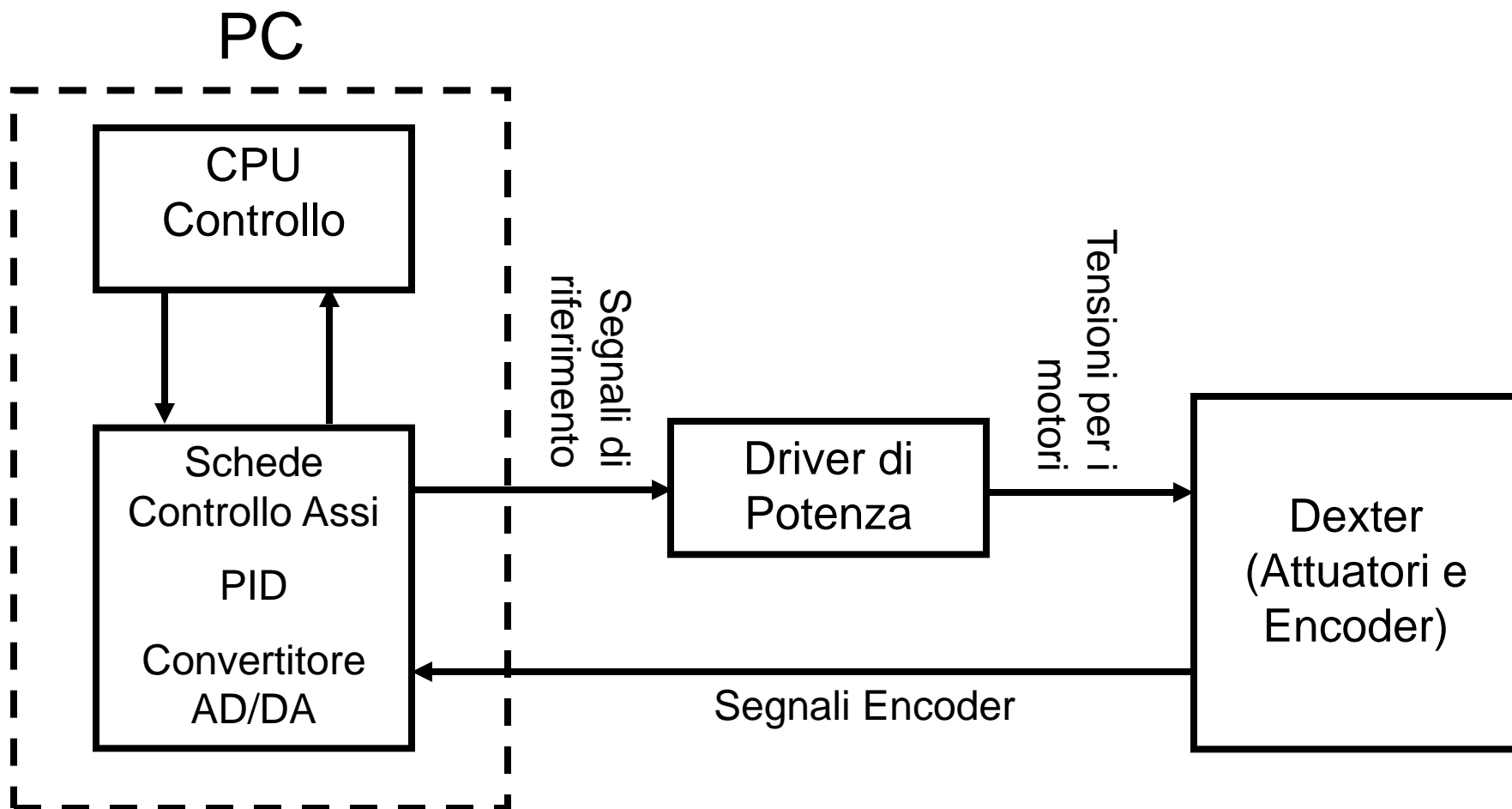
The Dexter Workspace



Joint Ranges

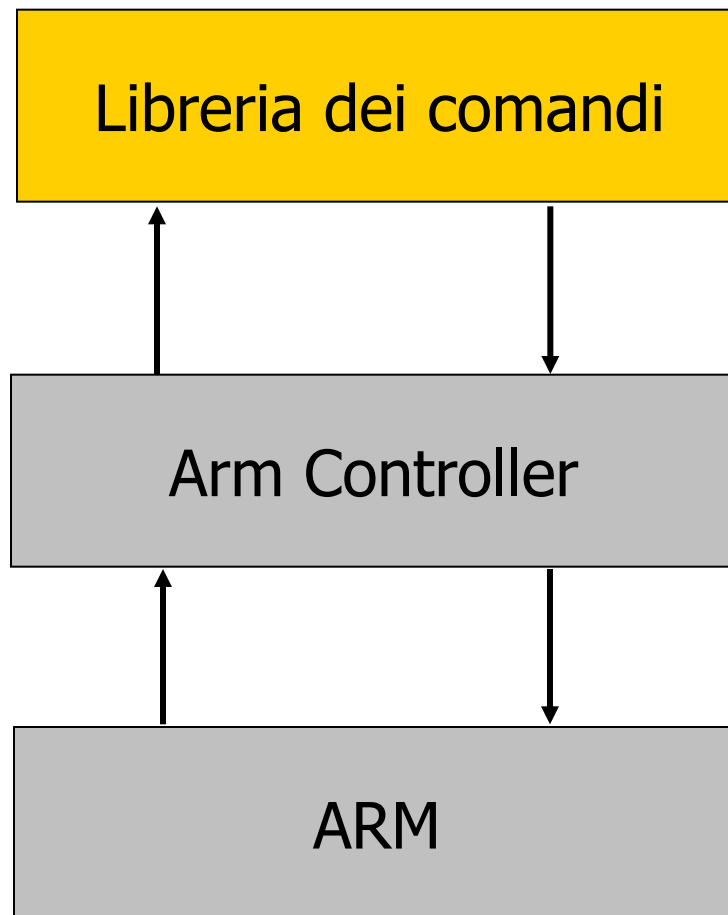


Sistema di controllo





Architettura software





Interfacce Software

Lettura della posizione corrente

- nello spazio dei giunti:

bool read_arm_q (double q)*

- *q*: puntatore ad un array di 8 double in cui viene restituita la posizione in gradi dei giunti del braccio

- nello spazio Cartesiano:

bool read_arm_c (double p)*

- *p*: puntatore ad un array di 6 double contenenti posizione in mm ed orientamento in gradi del braccio nello spazio cartesiano



Interfacce Software

Istruzioni di movimento
- nello spazio dei giunti:

bool move_arm_q(double q)*

- *q*: puntatore ad un array di 8 double contenente la posizione in gradi dei giunti del braccio



Interfacce Software

Istruzioni di movimento

- nello spazio Cartesiano:

bool move_arm_c7(double p, double elbow, double J0,
double velocity)*

- *p*: puntatore ad un array di 6 double contenenti posizione in mm ed orientamento in gradi nello spazio cartesiano
- *Elbow*: angolo del gomito espresso in gradi
- *J0*: posizione finale del giunto 0
- *Velocity*: frazione della velocità massima di movimento

Inversione cinematica su 7 dof



Interfacce Software

Istruzioni di movimento

- nello spazio Cartesiano:

bool move_arm_c(double p, double elbow, double velocity)*

- *p*: puntatore ad un array di 6 double contenenti posizione in mm ed orientamento in gradi nello spazio cartesiano
- *Elbow*: angolo del gomito espresso in gradi
- *Velocity*: frazione della velocità massima di movimento

Inversione cinematica su 8 dof