Corso di Robotica 2

Robot con ridondanza cinematica

Prof. Alessandro De Luca

Dipartimento di Informatica e Sistemistica Antonio Ruberti

Sapienza Università di Roma

Robot ridondanti

- cinematica diretta del compito $r = f(q)$

$$f: Q \rightarrow R$$

spazio dei giunti (dim $Q = N$) — spazio del compito (dim $R = M$)

- robot ridondante se $N > M$ (più gradi di libertà di quelli necessari ad eseguire il compito)
- r può essere la posizione e/o orientamento dell'organo terminale o qualsiasi altra parametrizzazione del compito (anche non nello spazio di lavoro)
- si parla quindi propriamente di "ridondanza" solo rispetto a un compito (task)

Alcuni compiti e la loro dimensione

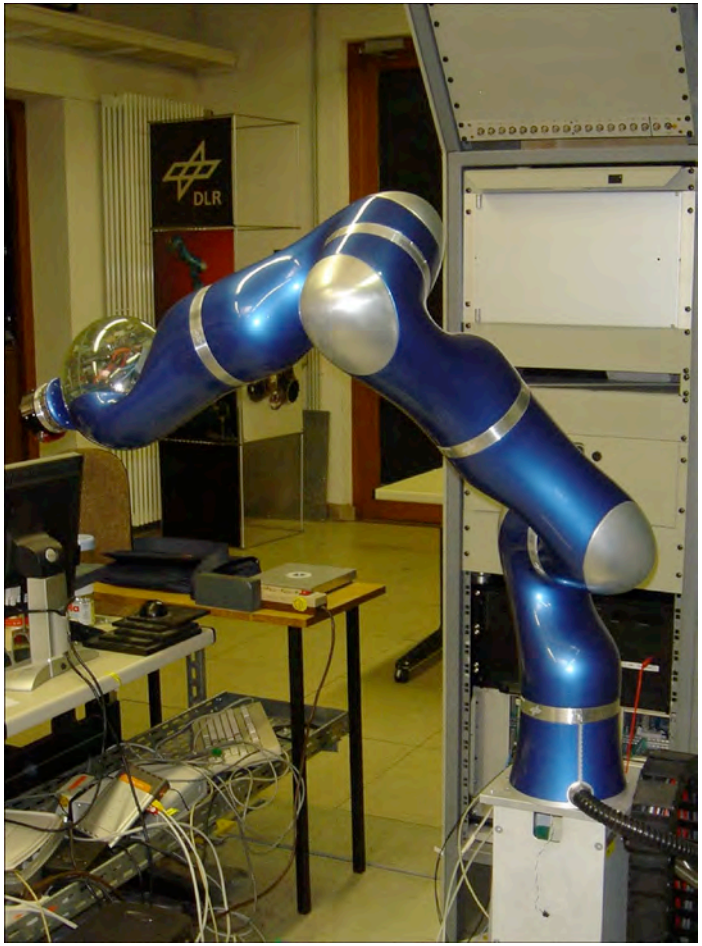
TASK (per l'organo terminale)	dimensione M
posizionamento nel piano →	2
posizionamento nello spazio →	3
orientamento nel piano →	1
puntamento nello spazio →	2
posizione e orientamento nello spazio →	6

un robot planare con N = 3 giunti è ridondante rispetto al posizionamento nel piano (M = 2), ma non rispetto al posizionamento + orientamento nel piano (M = 3)

Casi tipici di robot ridondanti

- robot 6R montato su binario
- robot a 6-dof usato per saldature ad arco
  - il compito non richiede di specificare l'angolo finale di roll
- manipolatore su base mobile
- mano robotica ad elevata destrezza
- squadre di robot mobili cooperanti
- …
- la ridondanza "cinematica" non è l'unico tipo…
  - ridondanza di componenti (attuatori, sensori)
  - ridondanza nell'architettura di controllo e supervisione

Robot del DLR: LWR-III e Justin

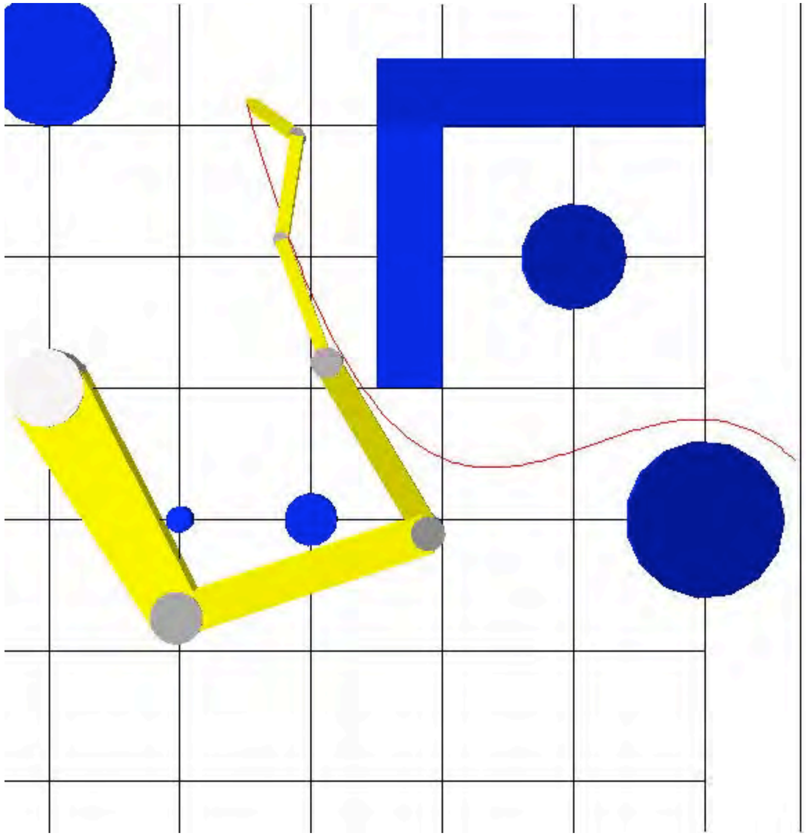


manipolatore 7R

2 bracci 7R + tronco 3R + 2 mani

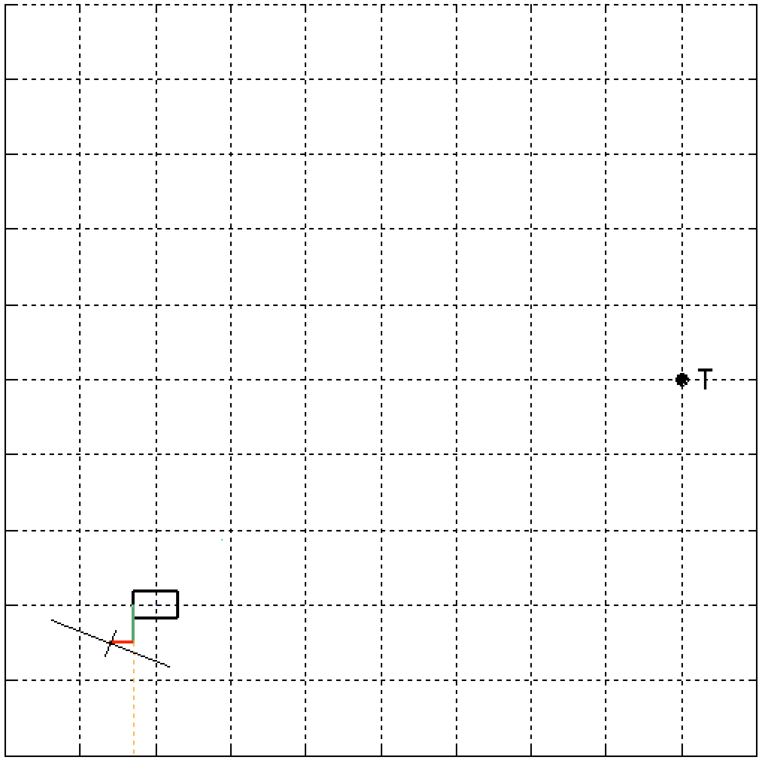
Esempi video - 1

Dexter 8R: automovimento

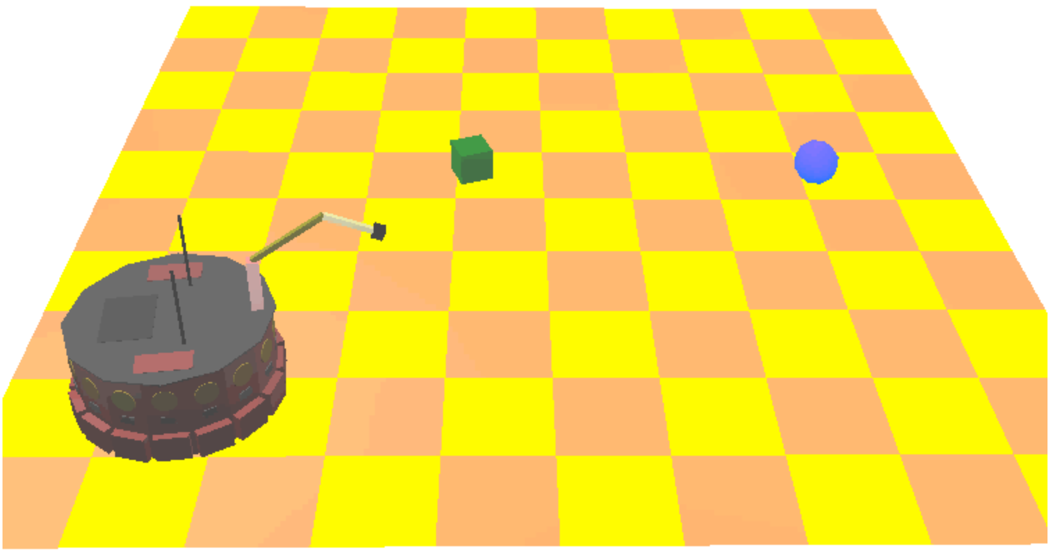


Planare 6R: obstacle avoidance su cammino assegnato dell'E-E

Esempi video - 2

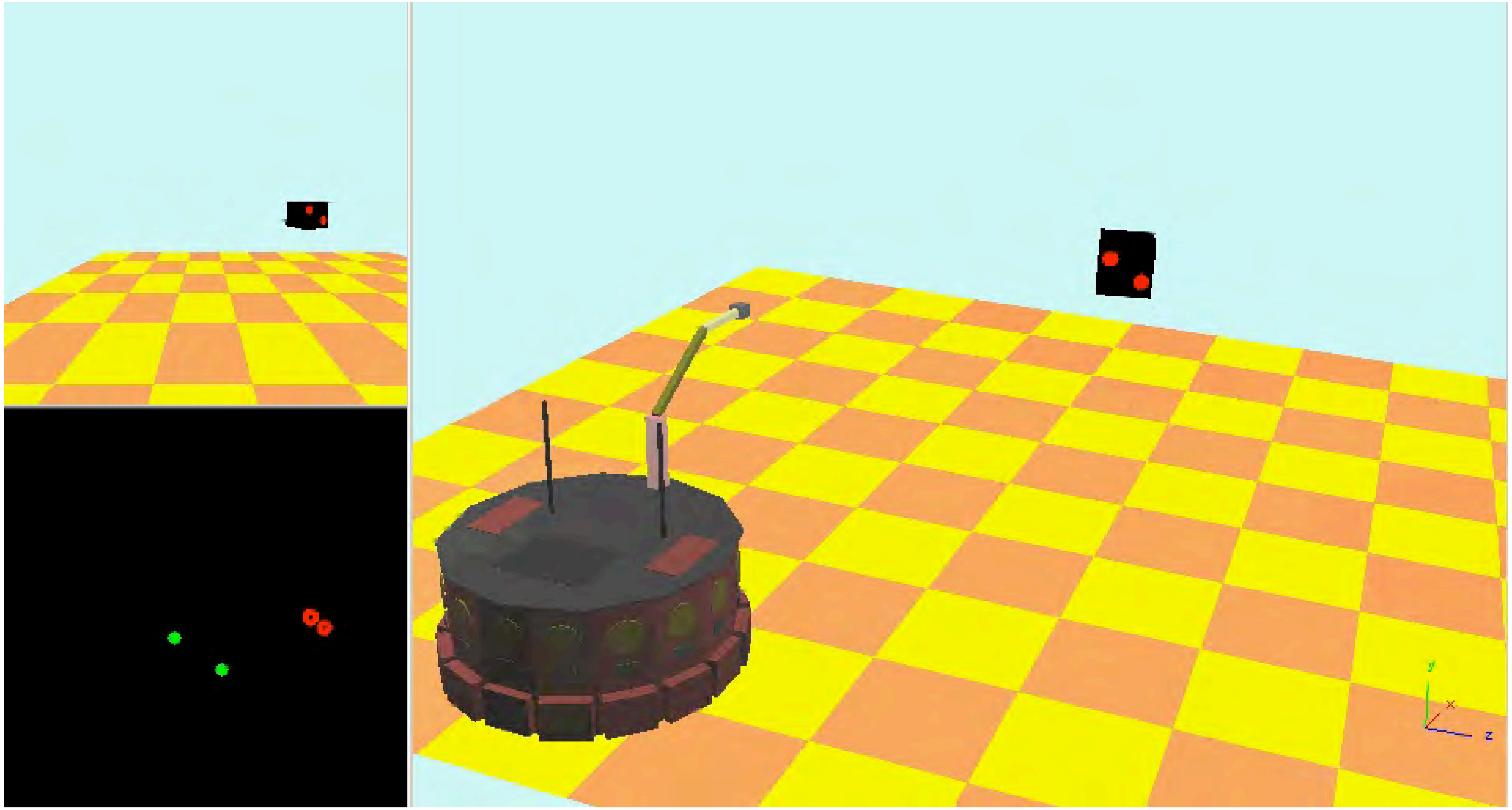


Uniciclo + planare 2R: controllo di traiettoria con compito su link 1



Magellan + 3R: controllo di traiettoria con puntamento dell'E-E

Esempi video - 3



Magellan + 3R: visual servoing nello spazio immagine di 2 features (M=4)

La ridondanza può essere usata per…

- evitare ostacoli (nello spazio cartesiano)…
- … o singolarità cinematiche (nello spazio dei giunti)
- aumentare la manipolabilità in determinate direzioni
- distribuire uniformemente/limitare le velocità di giunto
- rimanere entro i limiti di fondo corsa dei giunti
- minimizzare il consumo di energia
- ottimizzare il tempo di percorrenza, le coppie richieste…
- aumentare l'affidabilità rispetto a guasti

obiettivi "misurabili"

vantaggi ottenibili al costo di…

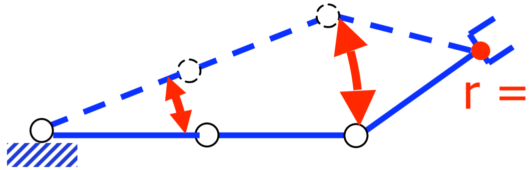
Svantaggi della ridondanza

- maggiore complessità strutturale
  - meccanica (bracci, trasmissioni)
  - di attuazione
- algoritmi di cinematica inversa e di controllo più complessi

Problema cinematico inverso

- trovare q(t) che realizzi f(q(t)) = r(t) (il compito nel tempo)
- esistono infinite soluzioni se il robot è ridondante
  (anche per r(t) = r = costante)

N = 3 > 2 = M



r = posizione E-E costante

- esistono "movimenti interni" che sono inosservabili dal punto di vista dell'esecuzione del compito (es., moto dell'E-E)
  - possono essere scelti in modo da ottimizzare in qualche modo il comportamento del sistema
- automovimento (self-motion): riconfigurazione del braccio che lascia inalterato il valore delle variabili r del compito

Risoluzione della ridondanza
(come ottimizzazione di una funzione obiettivo)

- **Locale**

note $\dot{r}(t)$ e $q(t)$, $t = kT_s$

(ottimizzazione di $H(q, \dot{q})$)

$\dot{q}(t)$ ← ON-LINE

$$q\big((k+1)T_s\big) = q(kT_s) + T_s \cdot \dot{q}(kT_s)$$

- **Globale**

note $r(t)$, $t \in [t_0, t_0 + T]$, $q(t_0)$

(ottimizzazione di $\int_{t_0}^{t_0+T} H(q,\dot{q})\,d\tau$)

$q(t)$, $t \in [t_0, t_0 + T]$

OFF-LINE

Metodi di risoluzione locale

Tre classi di metodi per risolvere $\dot{r} = J(q)\dot{q}$

1 metodi basati sullo Jacobiano (qui, analitico!)

tra le infinite soluzioni, si sceglie ad esempio quella che minimizza una certa norma (eventualmente) pesata

2 metodi con lo spazio nullo

si aggiunge alla soluzione a norma minima un contributo che non modifica la traiettoria del compito, ovvero che $\in \aleph(J(q))$

3 metodi basati sul compito aumentato

si riduce/elimina la ridondanza aggiungendo $S \leq N\text{-}M$ compiti ausiliari (se $S = N\text{-}M$, si "quadra" il problema)

1 Metodi Jacobian-based

si cerca una soluzione di $\dot{r} = J(q)\dot{q}$ nella forma

$J = \underbrace{\boxed{}}_{N}\}M$ $\qquad \dot{q} = K(q)\dot{r} \qquad$ $K = \underbrace{\boxed{}}_{M}\}N$

requisito minimo per K: J(q)K(q)J(q) = J(q)

(➡ K inversa generalizzata di J)

$$\forall \dot{r} \in \Re(J(q)) \;\Rightarrow\; J(q)[K(q)\dot{r}] = J(q)K(q)J(q)\,\dot{q} = J(q)\,\dot{q} = \dot{r}$$

ad es., se $J = [J_a \;\; J_b]$, $\det(J_a) \neq 0$, un'inversa (destra) di J è $K_r = \begin{pmatrix} J_a^{-1} \\ 0 \end{pmatrix}$

Pseudoinversa

$$\dot{q} = J^{\#}(q)\dot{r}$$

scelta per $K = J^{\#}$

- $J^{\#}$ esiste sempre ed è l'unica matrice che soddisfa

$$J J^{\#} J = J \qquad J^{\#} J J^{\#} = J^{\#}$$

$$(J J^{\#})^T = J J^{\#} \qquad (J^{\#} J)^T = J^{\#} J$$

- se J è a rango pieno, $J^{\#} = J^T (J J^T)^{-1}$; altrimenti va calcolata numericamente mediante SVD (Singular Value Decomposition) della J
- la soluzione $\dot{q}$ minimizza la norma $\frac{1}{2}\|\dot{q}\|^2 = \frac{1}{2}\dot{q}^T\dot{q}$

Pseudoinversa pesata

$$\dot{q} = J_W^{\#}(q)\,\dot{p}$$

scelta per $K = J_W^{\#}$

- se J è a rango pieno, $\boxed{J_W^{\#} = W^{-1} J^T (J\, W^{-1} J^T)^{-1}}$
- la soluzione $\dot{q}$ minimizza la norma pesata

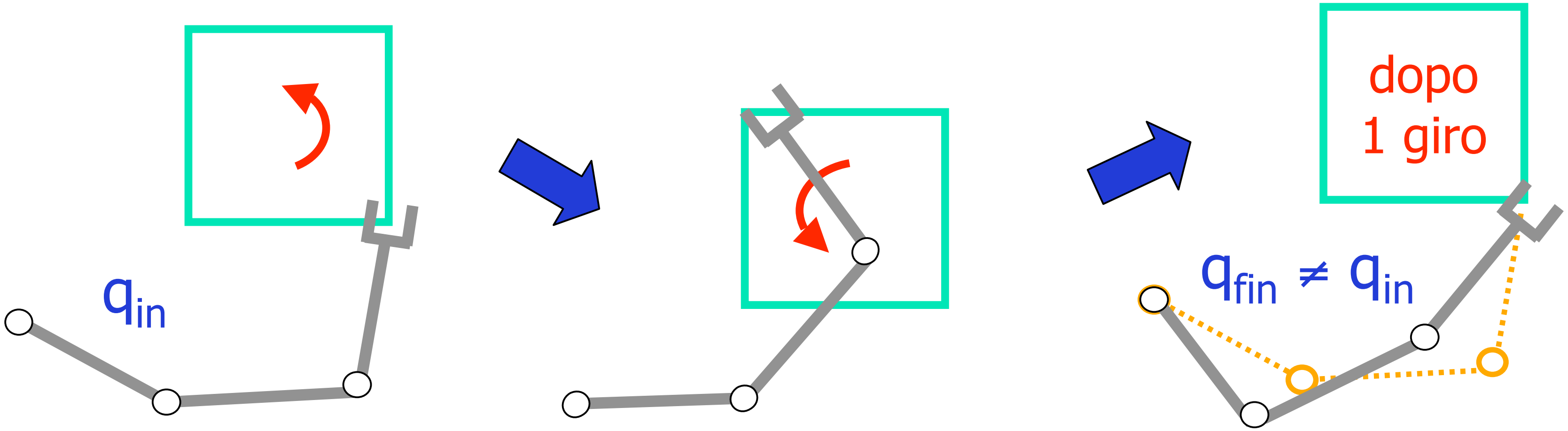
$$\frac{1}{2}\|\dot{q}\|_W^2 = \frac{1}{2}\dot{q}^T W \dot{q}$$

$W > 0$, simmetrica (spesso diagonale)

- peso W_i grande $\Rightarrow$ piccola $\dot{q}_i$ (ad es., pesi scelti inversamente proporzionali alle escursioni dei giunti)
- non è una "pseudoinversa" (non soddisfa la IV^a relazione)

Svantaggi dei metodi Jacobian-based

- non è detto che si evitino globalmente le singolarità durante l'esecuzione del compito
- potrebbero fornire soluzioni non ripetibili
  - cammini ciclici nello spazio del compito non corrispondono a cammini ciclici nello spazio dei giunti



$$q(t) = q_{in} + \int_0^t K(q)\dot{r}(\tau)d\tau$$

Robustezza a singolarità: Inversa Smorzata

ottimizzazione non vincolata

$$\min_{\dot{q}} \ \frac{\lambda}{2}\|\dot{q}\|^2 + \frac{1}{2}\|J\dot{q} - \dot{r}\|^2 = H(\dot{q})$$

minimi quadrati smorzati (DLS)

SOLUZIONE

$$\boxed{\dot{q} = J^T\left(JJ^T + \lambda I\right)^{-1}\dot{r}}$$

- è robusta rispetto alle singolarità (si può usare anche per N = M), ma ne risulta un errore sulla traiettoria del compito
- scelta del fattore di smorzamento $\lambda \geq 0$ di compromesso fra minima norma delle velocità di giunto $\|\dot{q}\|$ e minimo errore di traiettoria $\|\dot{r} - J\dot{q}\|$ (scelta ragionevole: $\lambda(q) \geq 0$ funzione del minimo valore singolare di J $\cong$ distanza da una singolarità)

2 Metodi con lo spazio nullo

soluzione generale di $J\,\dot{q} = \dot{r}$

$$\dot{q} = \underbrace{J^{\#}\dot{r}}_{\text{una soluzione particolare}} + \underbrace{\left(I - J^{\#}J\right)}_{\text{“proiettore” nel } \aleph(J)}\dot{q}_0$$

una soluzione particolare (qui la pseudoinversa)

“proiettore” nel $\aleph(J)$

$\longleftrightarrow$ tutte le soluzioni dell’equazione omogenea associata $J\dot{q} = 0$ (automovimenti)

proprietà di $[I - J^{\#}J]$

- simmetrica
- idempotente: $[I - J^{\#}J]^2 = [I - J^{\#}J]$
- $[I - J^{\#}J]^{\#} = [I - J^{\#}J]$

più in generale…

$$\dot{q} = K_1\,\dot{r} + \left(I - K_2 J\right)\dot{q}_0$$

K_1, K_2 inverse generalizzate di J ($J\,K_i\,J = J$)

come scegliere $\dot{q}_0$?

Ottimizzazione lineare-quadratica
Generalità

$$\min_x \frac{1}{2}(x-x_0)^T W(x-x_0) = H(x)$$

$$\text{s.t.} \quad \underset{\boxed{M \times N}}{J}\, x = y$$

$x \in \mathbb{R}^N$, $W > 0$ (simmetrica)

$y \in \mathbb{R}^M$, $\rho(J) = M$

$$L(x,\lambda) = H(x) + \lambda^T(Jx - y)$$ ← Lagrangiana

c. nec. minimo

$$\nabla_x L = \frac{\partial L}{\partial x}^T = W(x-x_0) + J^T\lambda = 0 \quad \Rightarrow \quad \boxed{x = x_0 - W^{-1}J^T\lambda}$$

$$\nabla_\lambda L = Jx - y = 0 \quad \Rightarrow \quad \boxed{Jx_0 - JW^{-1}J^T\lambda - y = 0}$$

\+ c. suff.

$$\nabla_x^2 L = W > 0$$

$$\boxed{\lambda = \big(\underbrace{JW^{-1}J^T}_{M \times M \text{ invertibile}}\big)^{-1}(Jx_0 - y)} \quad \Rightarrow \quad x = x_0 - W^{-1}J^T\left(JW^{-1}J^T\right)^{-1}(Jx_0 - y)$$

Ottimizzazione lineare-quadratica

Applicazione ai robot ridondanti

PROBLEMA

$$\min_{\dot q} \frac{1}{2}(\dot q - \dot q_0)^T W (\dot q - \dot q_0) = H(\dot q)$$

$$J(q)\dot q = \dot r$$

$\dot q_0$ è una velocità di giunto "privilegiata"

SOLUZIONE

$$\boxed{\dot q = \dot q_0 - \underbrace{W^{-1}J^T\left(JW^{-1}J^T\right)^{-1}}_{J_W^{\#}}\left(J\dot q_0 - \dot r\right)}$$

$$\dot q = \underbrace{J_W^{\#}\,\dot r}_{\text{soluzione a minima norma pesata (per } \dot q_0 = 0)} + \underbrace{\left(I - J_W^{\#}\,J\right)}_{\text{"proiettore" nel nucleo } \aleph(J)}\dot q_0$$

Gradiente proiettato (PG)

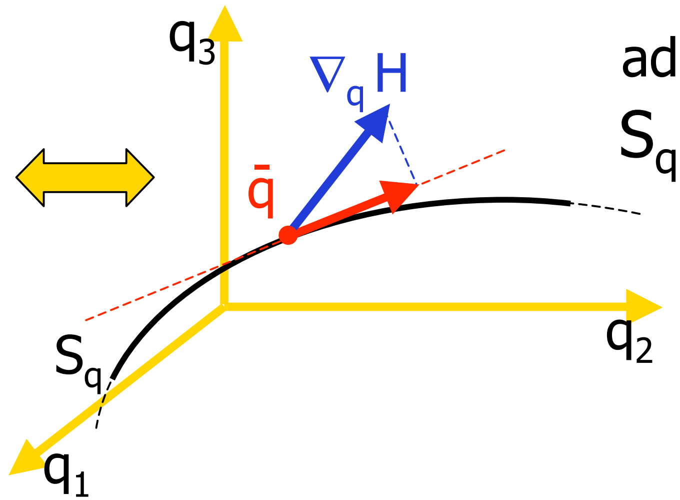
$$\dot{q} = J^{\#}\dot{r} + \left(I - J^{\#}J\right)\dot{q}_0$$

la scelta $\dot{q}_0 = \nabla_q H(q)$ → funzione obiettivo differenziabile

realizza un passo di un algoritmo di ottimizzazione vincolata

mentre si muove lungo il compito $r(t)$
il robot tende ad aumentare il valore di $H(q)$

gradiente
proiettato



ad es., per $\bar{r}$ fissato

$$S_q = \{q \in \mathbb{R}^N : f(q) = \bar{r}\}$$

$$\dot{q} = (I - J^{\#}J)\nabla_q H$$

Tipiche funzioni obiettivo H(q)

- manipolabilità (massimizza la "distanza" dalle singolarità)

$$H_{man}(q) = \sqrt{\det J(q)J^T(q)}$$

- joint range (minimizza la "distanza" dal centro delle escursioni di giunto)

$$H_{range}(q) = \frac{1}{2N}\sum_{i=1}^{N}\left(\frac{q_i - \bar{q}_i}{q_{M,i} - q_{m,i}}\right)^2$$

$$\dot{q}_0 = -\nabla_q H(q)$$

- obstacle avoidance (massimizza la "distanza" dagli ostacoli)

$$H_{obs}(q) = \min_{\substack{a \in robot \\ b \in ostacoli}} \|a(q) - b\|^2$$

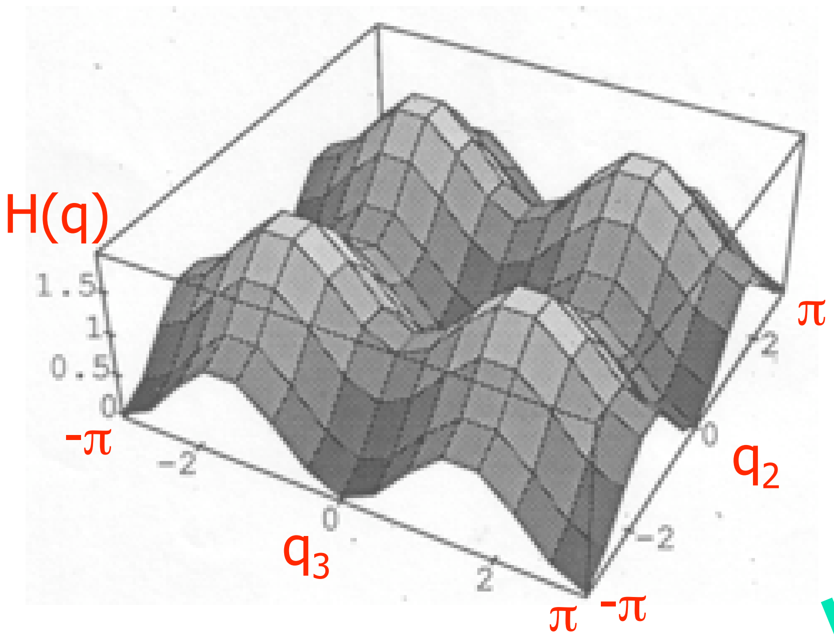
possibili problemi di differenziabilità in questo problema di max-min

Singolarità 3R planare

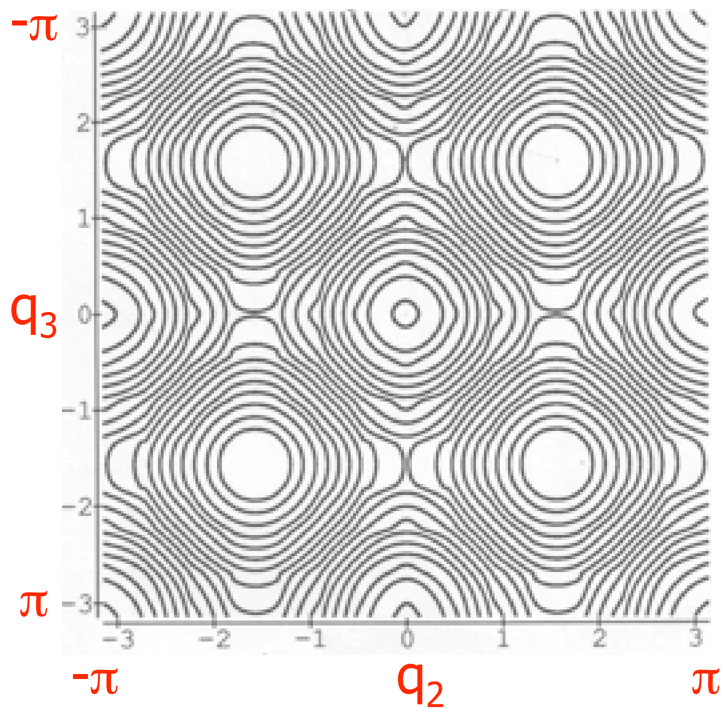
in un compito di posizionamento

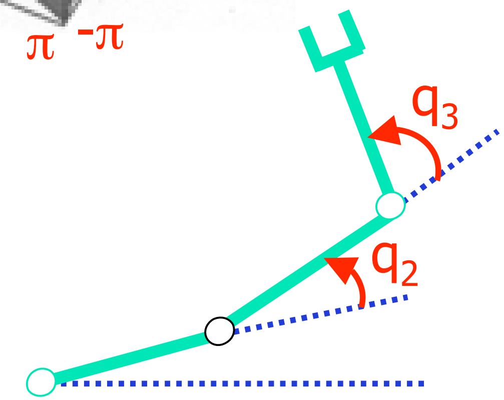
$$H(q) = \sin^2 q_2 + \sin^2 q_3$$

non è la H_{man}
ma ha gli stessi minimi!



curve di livello





indipendente da q_1!

Commenti sui metodi con lo spazio nullo

- il proiettore nel nullo $(I - J^{\#}J)$ ha dimensione N x N ma rango solo N – M (se J ha rango pieno), con spreco di informazione
- metodo effettivo di calcolo della soluzione

$$\boxed{\dot{q}} = J^{\#}\dot{r} + \left(I - J^{\#}J\right)\dot{q}_0 = \boxed{\dot{q}_0 + J^{\#}\left(\dot{r} - J\dot{q}_0\right)}$$

- viene comunque richiesto il calcolo della pseudoinversa, oneroso computazionalmente (sia se a rango pieno che non)
- la complessità di un metodo di risoluzione della ridondanza dovrebbe in effetti dipendere solo dal grado di ridondanza N – M
- esiste un metodo di ottimizzazione vincolata più efficiente del gradiente proiettato (PG) nel caso di Jacobiano a rango pieno …

Gradiente ridotto (RG)

- se $\rho(J(q)) = M$, allora esiste una partizione dei giunti (a valle di un eventuale riordinamento dei giunti)

$$q = \begin{pmatrix} q_a \\ q_b \end{pmatrix} \begin{matrix} \}\, M \\ \}\, N\text{-}M \end{matrix} \qquad \text{tale che } J_a(q) = \overbrace{\frac{\partial f}{\partial q_a}}^{M \times M} \text{ è nonsingolare}$$

- per il teorema delle funzioni implicite, $\exists$ una funzione g

$$f(q_a, q_b) = r \quad \Rightarrow \quad q_a = g(r, q_b)$$

$$\text{con } \frac{\partial g}{\partial q_b} = -\left(\frac{\partial f}{\partial q_a}\right)^{-1} \frac{\partial f}{\partial q_b} = -J_a^{-1}(q)\, J_b(q)$$

- si usano le N-M variabili indipendenti q_b per ottimizzare un criterio H(q) ("ridotto" tramite la g ad una funzione delle sole q_b) con il metodo del gradiente
- si scelgono poi le $q_a = g(r, q_b)$ per assicurare l'esecuzione del compito

Algoritmo del gradiente ridotto

- $H(q) = H(q_a, q_b) = H(g(r, q_b), q_b) = H'(q_b)$ (con r al valore corrente)
- gradiente ridotto (rispetto alle sole q_b)

$$\nabla_{q_b} H' = \begin{bmatrix} -\left(J_a^{-1} J_b\right)^T & I \end{bmatrix} \nabla_q H \qquad (\neq \nabla_{q_b} H\ !!)$$

- ALGORITMO

$$\dot{q}_b = \nabla_{q_b} H'$$

passo nella direzione del gradiente
nello spazio ridotto (N-M)-dim

$$J_a \dot{q}_a + J_b \dot{q}_b = \dot{r}$$

soddisfacimento degli M vincoli
del compito

$$\dot{q}_a = J_a^{-1}\left(\dot{r} - J_b \dot{q}_b\right)$$

Confronto tra i metodi PG e RG

- Projected Gradient (PG)

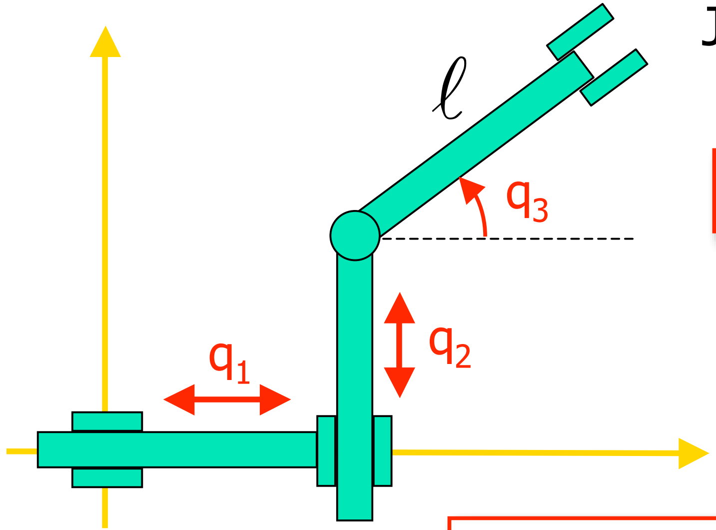
$$\dot{q} = J^{\#}\dot{r} + \left(I - J^{\#}J\right)\nabla_q H$$

- Reduced Gradient (RG)

$$\dot{q} = \begin{pmatrix} \dot{q}_a \\ \dot{q}_b \end{pmatrix} = \begin{pmatrix} J_a^{-1} \\ 0 \end{pmatrix} \dot{r} + \begin{pmatrix} -J_a^{-1}J_b \\ I \end{pmatrix} \left(-\left(J_a^{-1}J_b\right)^T \quad I \right) \nabla_q H$$

- RG più conveniente di PG sia dal punto di vista analitico che numerico, ma richiede l'individuazione di un minore non singolare (J_a) nello Jacobiano
- se r = cost & N-M=1 $\Rightarrow$ stessa direzione di $\dot{q}$, ma passo RG più lungo
- altrimenti, RG e PG forniscono evoluzioni diverse

Esempio analitico: robot PPR



$$J = \left(\begin{array}{cc|c} 1 & 0 & -\ell s_3 \\ 0 & 1 & \ell c_3 \end{array}\right) = \left(J_a \mid J_b\right) \qquad q_a = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} \qquad q_b = q_3$$

$$\dot{q} = \begin{pmatrix} J_a^{-1} \\ 0 \end{pmatrix}\dot{r} + \begin{pmatrix} -J_a^{-1}J_b \\ I \end{pmatrix}\left(-\left(J_a^{-1}J_b\right)^T \quad I\right)\nabla_q H$$

RG:

$$\dot{q} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}\dot{r} + \begin{pmatrix} \ell s_3 \\ -\ell c_3 \\ 1 \end{pmatrix}\left(\ell s_3 \quad -\ell c_3 \quad 1\right)\nabla_q H$$

PG: $\dot{q} = J^{\#}\dot{r} + \left(I - J^{\#}J\right)\nabla_q H$

$$J^{\#} = \frac{1}{1+\ell^2}\begin{pmatrix} 1+\ell^2 c_3^2 & \ell^2 s_3 c_3 \\ \ell^2 s_3 c_3 & 1+\ell^2 s_3^2 \\ -\ell s_3 & \ell c_3 \end{pmatrix} \qquad \left(I - J^{\#}J\right) = \frac{1}{1+\ell^2}\begin{pmatrix} \ell^2 s_3^2 & & \text{sym} \\ -\ell^2 s_3 c_3 & \ell^2 c_3^2 & \\ \ell s_3 & -\ell c_3 & 1 \end{pmatrix}$$

< 1!!

Limiti di giunto

$$q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \theta = T\theta$$

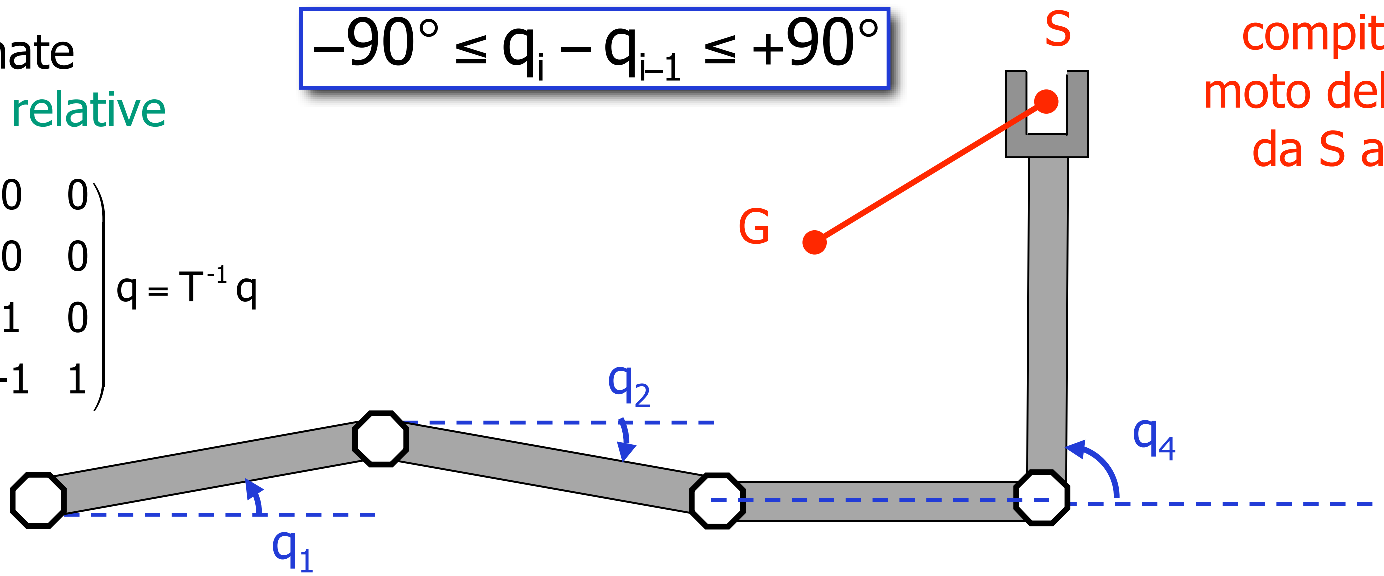
coordinate
assolute ⇔ relative

$$\theta = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix} q = T^{-1} q$$

$$\boxed{-90° \le \theta_i \le +90°}$$

$$\Updownarrow$$

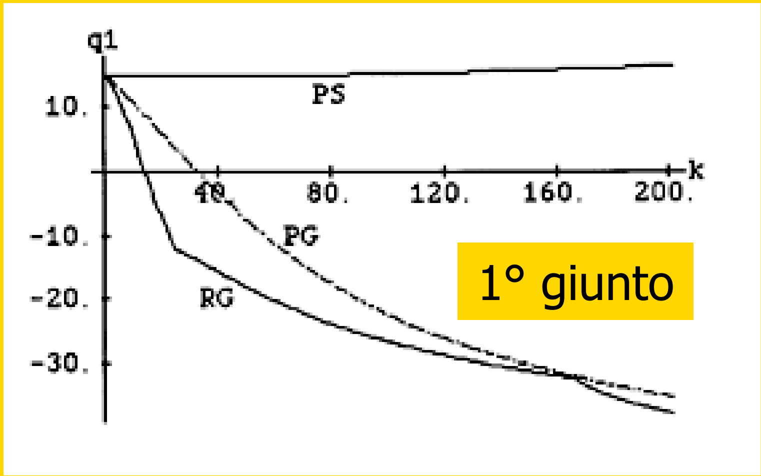
$$\boxed{-90° \le q_i - q_{i-1} \le +90°}$$

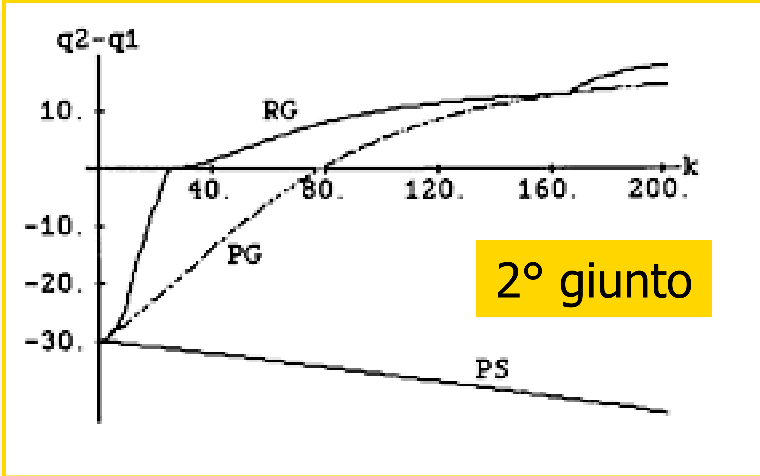


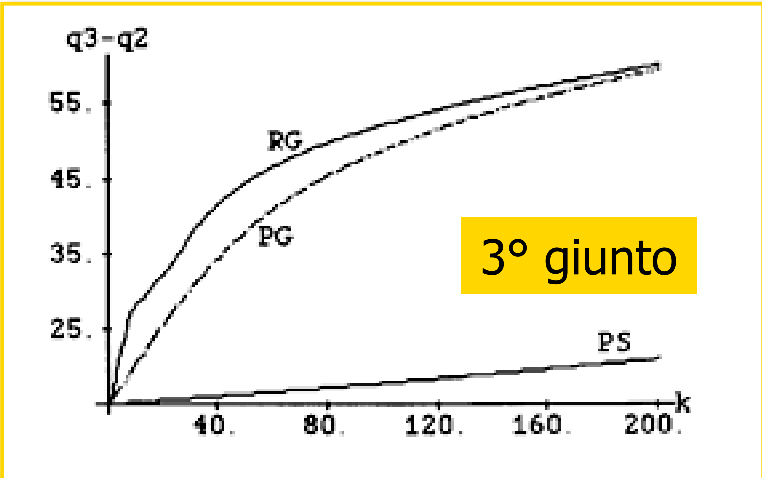
compito:
moto dell'E-E
da S a G

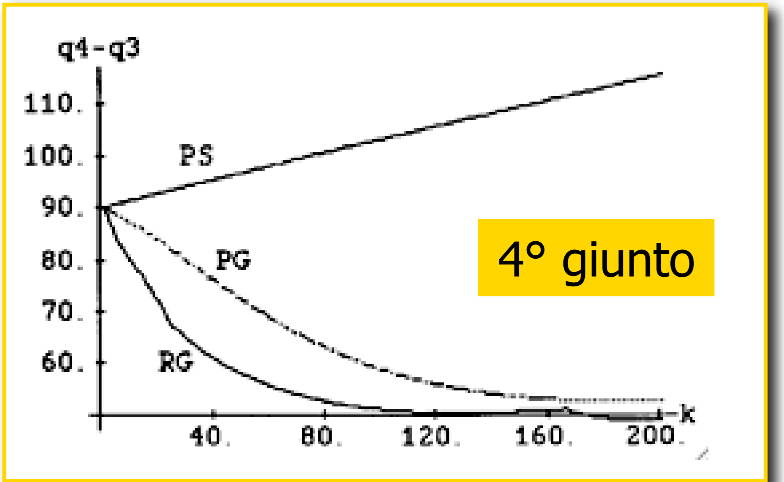
confronto fra metodo della pseudoinversa (PS), gradiente proiettato (PG) e gradiente ridotto (RG)

Simulazione numerica



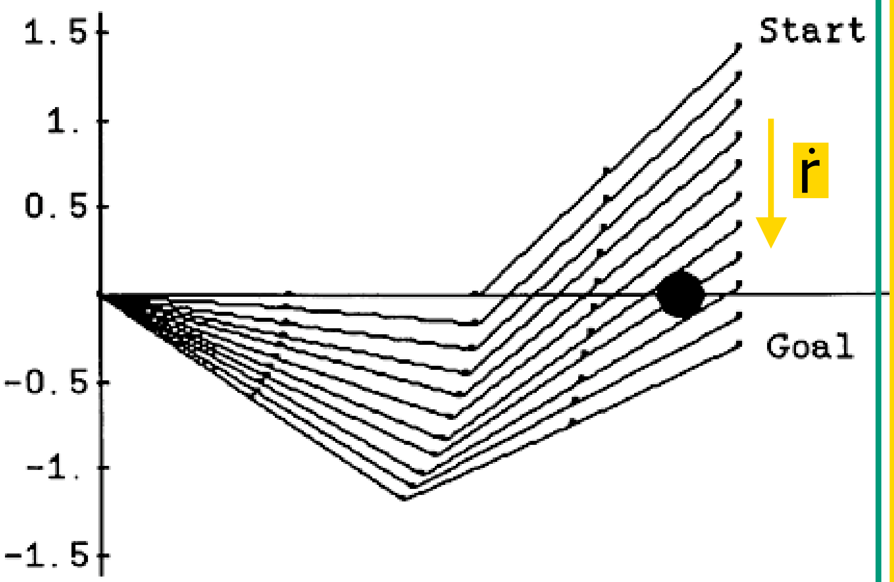




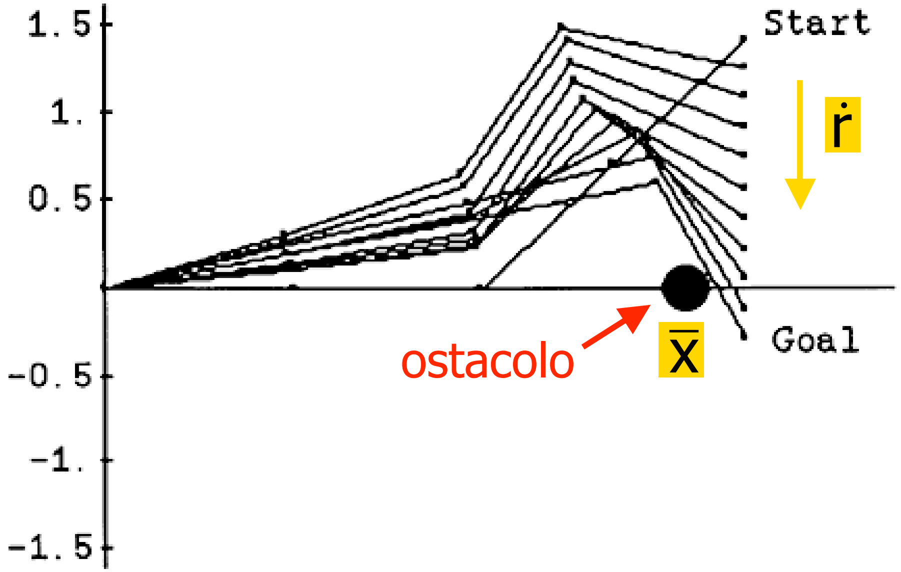


Obstacle avoidance

pseudoinversa $\dot{q} = J^{\#}\dot{r}$



gradiente ridotto



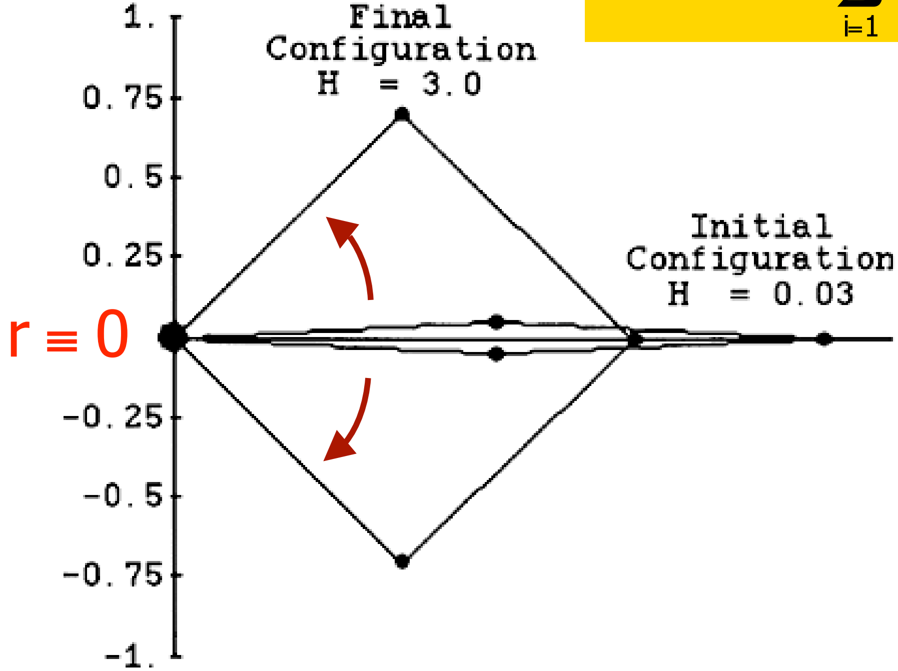
con massimizzazione di

$$H(q) = \left| \bar{x}\sin q_4 - \sum_{i=1}^{3} \ell_i \sin(q_4 - q_i) \right|$$

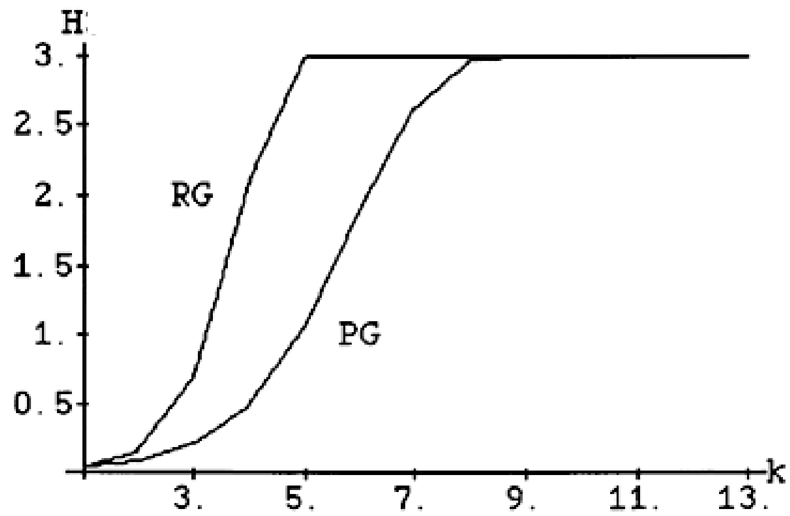
Automovimento con manipolabilità

$$\max H(q) = \sum_{i=1}^{3} \sin^2(q_{i+1} - q_i)$$

(ottima)



... in realtà, non è proprio la funzione di manipolabilità



passi di integrazione

RG più veloce di PG
(a parità di accuratezza su r)

3 Metodi con compito aumentato

- si aggiunge un compito ausiliario (task augmentation)

$$S \updownarrow \boxed{f_y(q) = y} \qquad S \leq N-M$$

corrispondente a qualche aspetto desiderabile della soluzione

$$r_A = \begin{pmatrix} r \\ y \end{pmatrix} = \begin{pmatrix} f(q) \\ f_y(q) \end{pmatrix} \Longrightarrow \dot{r}_A = \begin{pmatrix} J(q) \\ J_y(q) \end{pmatrix} \dot{q} = J_A(q)\dot{q} \qquad \underbrace{\boxed{J_A}}_{N} \left.\vphantom{\boxed{J_A}}\right\} M+S$$

- si cerca ancora una soluzione nella forma

$$\boxed{\dot{q} = K_A(q)\,\dot{r}_A}$$

o con l'aggiunta di un termine nel nullo di J_A

Con un compito aumentato…

- vantaggio: maggiore controllo sulla soluzione inversa

- svantaggio: si introducono singolarità algoritmiche quando

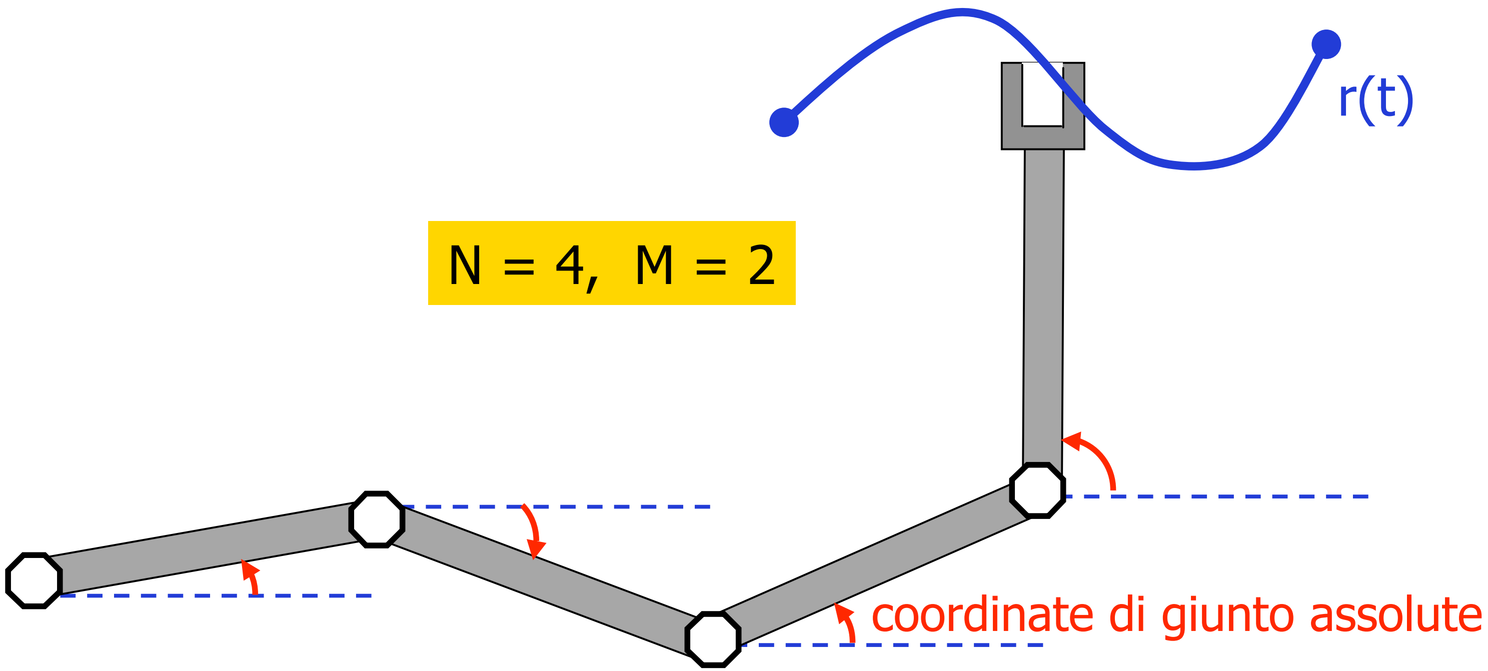
$$\rho(J) = M \quad \rho(J_y) = S \quad \text{ma} \quad \rho(J_A) < M+S$$

dovrebbe essere sempre $\boxed{\Re\left(J^T\right) \cap \Re\left(J_y^T\right) = \emptyset}$

difficile da ottenere globalmente!

righe di J e righe di J_y
linearmente indipendenti

Compito aumentato
esempio



$$f_y(q) = q_4 = \pi/2 \qquad (S = 1)$$

ultimo braccio mantenuto verticale

Jacobiano esteso (S = N– M)

- J_A quadrato: in assenza di singolarità algoritmiche, si può prendere

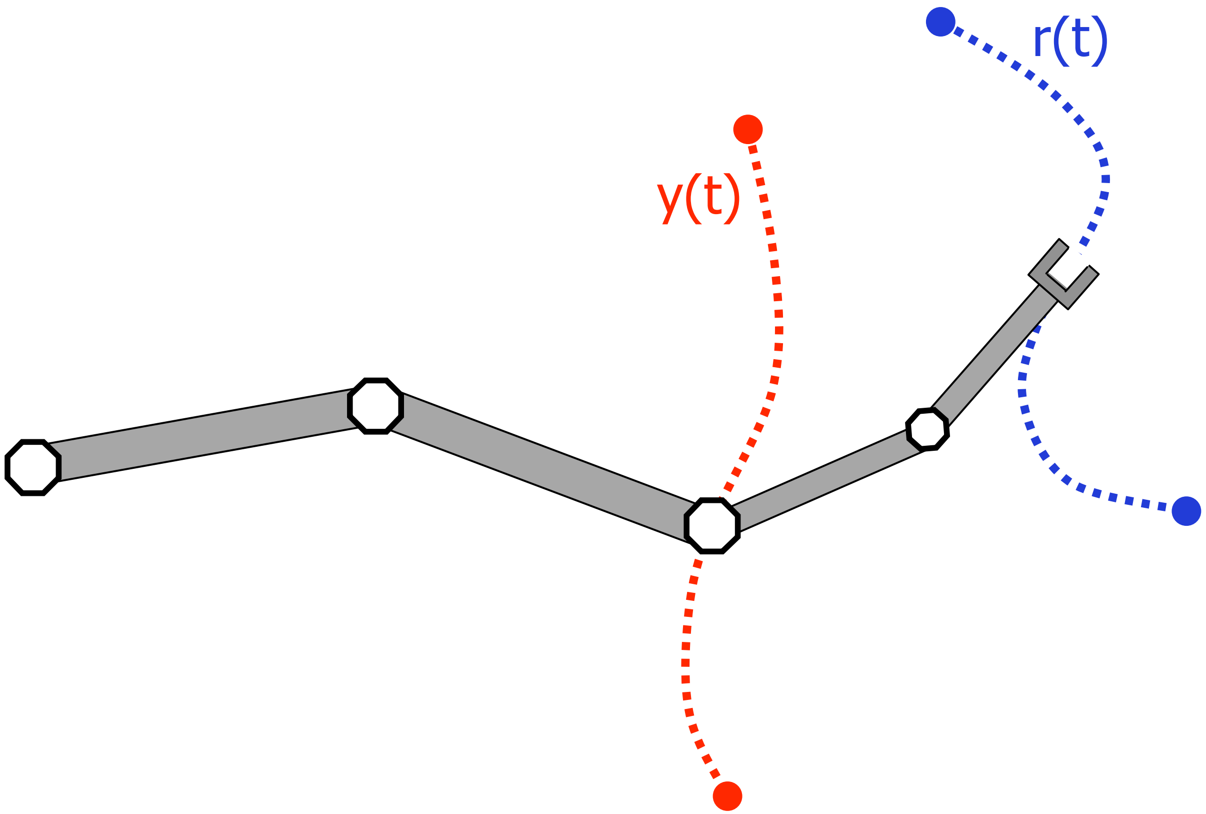
$$\dot{q} = J_A^{-1}(q)\dot{r}_A$$

- lo schema è evidentemente ripetibile
- se $f_y(q) = 0$ corrisponde a condizioni necessarie (& sufficienti) di ottimo vincolato per una certa funzione H(q), lo schema assicura permanenza dell'ottimalità durante l'esecuzione del compito
- in presenza di singolarità algoritmiche, l'esecuzione sia del compito originario che dei compiti aggiuntivi risulta affetta da errore

Jacobiano esteso

esempio

manipolatore MACRO-MICRO



$N = 4, \quad M = 2$

$$\dot{r} = J(q_1, \ldots q_4)\dot{q}$$

$$\dot{y} = J_y(q_1, q_2)\dot{q}$$

$$\Downarrow$$

$$J_A = \left(\begin{array}{c|c} * & * \\ \hline * & 0 \end{array}\right) \quad 4 \times 4$$

Priorità di compito

se il compito originale $\dot{r} = J(q)\dot{q}$ ha una priorità maggiore del compito ausiliario $\dot{y} = J_y(q)\dot{q}$

- si fa **prima** in modo che venga eseguito il compito prioritario

$$\dot{q} = J^{\#}\dot{r} + (I - J^{\#}J)v$$

- e **poi** si sceglie v in modo da soddisfare (se possibile) anche il compito secondario

$$\dot{y} = J_y(q)\dot{q} = J_y J^{\#}\dot{r} + J_y(I - J^{\#}J)v = J_y J^{\#}\dot{r} + \tilde{J}_y v$$

la soluzione generale per v è nella forma usuale

$$v = \tilde{J}_y^{\#}(\dot{y} - J_y J^{\#}\dot{r}) + (I - \tilde{J}_y^{\#}\tilde{J}_y)w$$

a disposizione per eventuali ulteriori compiti a priorità più bassa

Priorità di compito (cont.)

- sostituendo l'espressione di v in $\dot{q}$

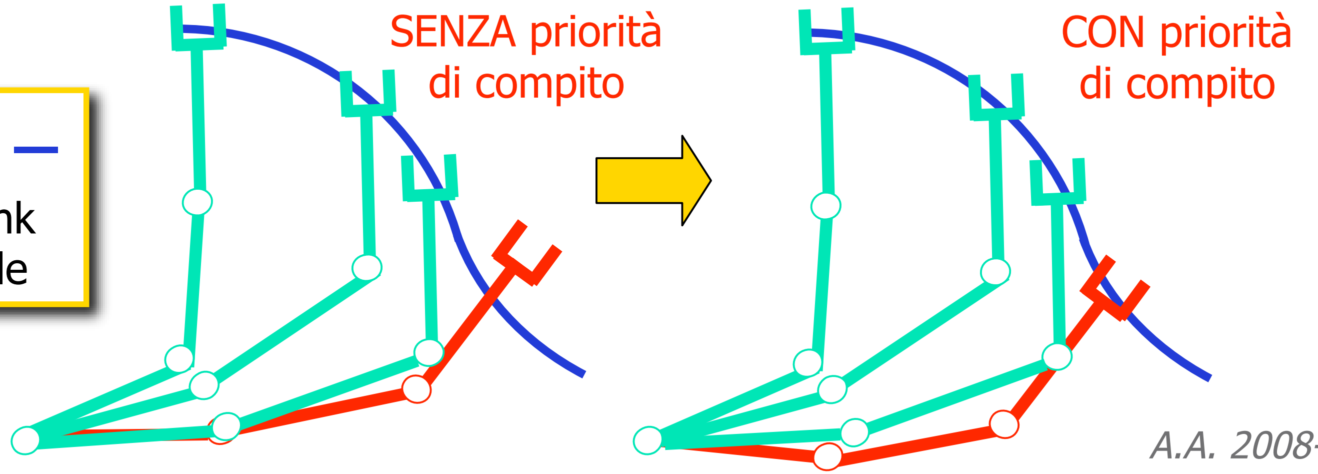
$$\dot{q} = J^{\#}\dot{r} + (I - J^{\#}J)\tilde{J}_y^{\#}(\dot{y} - J_y J^{\#}\dot{r}) + (I - J^{\#}J)(I - \tilde{J}_y^{\#}\tilde{J}_y)w$$

poiché $C\,[BC]^{\#} = [BC]^{\#}$
se C idempotente $\tilde{J}_y^{\#}$

eventualmente = 0

- vantaggio della strategia: compito principale non più affetto dalle singolarità algoritmiche

Task 1: seguire —
Task 2: terzo link verticale



Estensioni

- la trattazione dei metodi di risoluzione della ridondanza svolta finora:
  - è definita al primo ordine differenziale (velocità)
    - possibile lavorare in accelerazione
  - non tiene conto di errori nell'esecuzione del compito
    - possibile usare un controllo cinematico
  - si applica ai robot manipolatori a base fissa
    - possibile considerare manipolatori mobili (su ruote)

Risoluzione al secondo ordine

$$r = f(q) \;\Rightarrow\; \dot{r} = J(q)\dot{q} \;\Rightarrow\; \boxed{\ddot{r} = J(q)\ddot{q} + \dot{J}(q)\dot{q}}$$

- posto nella forma

$$\underbrace{J(q)\ddot{q}}_{\text{da scegliere}} = \underbrace{\ddot{r}}_{\substack{\text{assegnata}\\ \text{(al tempo t)}}} - \underbrace{\dot{J}(q)\dot{q}}_{\substack{\text{note q e } \dot{q}\\ \text{(al tempo t)}}} \triangleq \ddot{x}$$

il problema è formalmente equivalente a prima, con l'accelerazione al posto della velocità

- ad esempio, nel metodo con lo spazio nullo

$$\ddot{q} = \underbrace{J^{\#}(q)\ddot{x}}_{\substack{\text{soluzione a minima}\\ \text{norma di accelerazione } \frac{1}{2}\|\ddot{q}\|^2}} + \left(I - J^{\#}(q)J(q)\right)\underbrace{\ddot{q}_0}_{= \nabla_q H - K_D\dot{q}}$$

"stabilizzazione" automovimenti

Controllo cinematico

- dato un compito desiderato $r_d(t)$ (M-dimensionale), per recuperare un errore $e = r_d - r$, iniziale o successivo dovuto a
  - disturbi
  - linearizzazione intrinseca nello Jacobiano (moto al primo ordine)
  - implementazione a tempo discreto

  è necessario "chiudere" un anello di feedback sul compito, sostituendo (con guadagni K o K_P, K_V > 0 e diagonali)

$$\dot{r} \Rightarrow \dot{r}_d + K(r_d - r)$$ nei metodi in velocità

$$\ddot{r} \Rightarrow \ddot{r}_d + K_v(\dot{r}_d - \dot{r}) + K_P(r_d - r)$$ … in accelerazione

con $r = f(q)$, $\dot{r} = J(q)\dot{q}$

Manipolatori mobili

- si combinano coordinate della base q_b e del manipolatore q_m
- si utilizza il legame differenziale con i comandi effettivi u_b disponibili sulla base e u_m sul manipolatore

q_m

q_b

$r = f(q)$ (ad es., posa dell'E-E)

modello cinematico della base (con vincoli anolonomi)

$$q = \begin{pmatrix} q_b \\ q_m \end{pmatrix} \in \mathbb{R}^N \qquad \dot{q}_b = G(q_b)u_b \qquad \dot{q}_m = u_m$$

$$u = \begin{pmatrix} u_b \\ u_m \end{pmatrix} \in \mathbb{R}^{N_u} \qquad N_u \leq N$$

Jacobiano manipolatore mobile

$$r = f(q) = f(q_b, q_m)$$

$$\dot{r} = \frac{\partial f(q)}{\partial q_b}\dot{q}_b + \frac{\partial f(q)}{\partial q_m}\dot{q}_m = J_b(q)\dot{q}_b + J_m(q)\dot{q}_m$$

$$= J_b(q)G(q_b)u_b + J_m(q)u_m = \begin{pmatrix} J_b(q)G(q_b) & J_m(q) \end{pmatrix}\begin{pmatrix} u_b \\ u_m \end{pmatrix}$$

$$= \boxed{J_{NMM}(q)}u$$

Nonholonomic Mobile Manipulator
Jacobian ($M \times N_u$)

- … e tutto segue sostituendo

$J \Rightarrow J_{NMM}$ $\qquad \dot{q} \Rightarrow u$ $\qquad$ (ridondanza se $N_u - M > 0$)

ossia gli effettivi
comandi disponibili