Robotica ed Animazione Digitale
I comportamenti

Prof. Alberto Borghese
Dipartimento di Scienze dell'Informazione
borghese@dsi.unimi.it

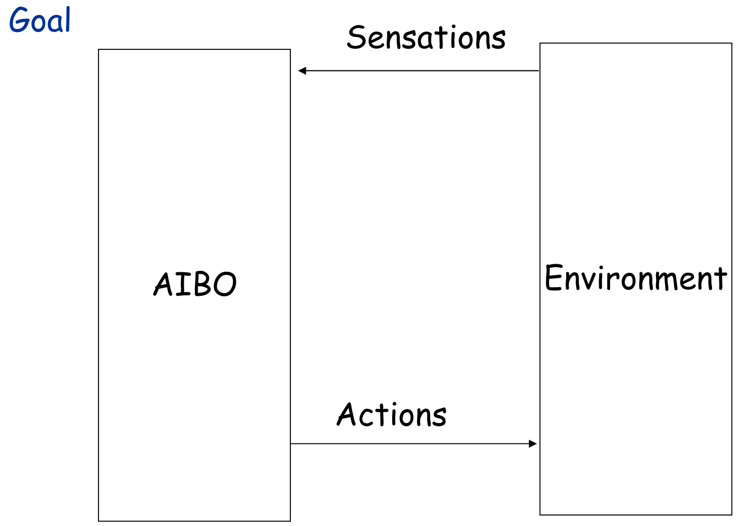
Università degli Studi di Milano

Slide in parte tratte da: http://www.andrew.cmu.edu/course/15-491

Sommario della lezione

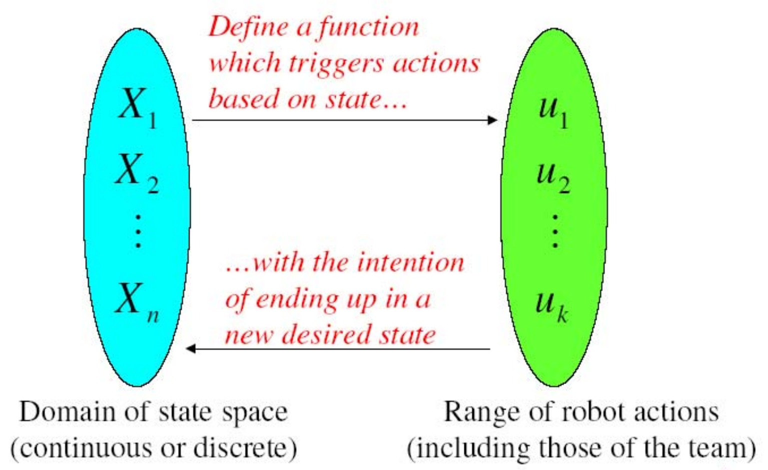
- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Gli scheletri
- Descrizione della posizioen di uno scheletro

AIBO as an agent



I comportamenti sono una mappatura dell'input in azioni, sono funzioni.

I comportamenti sono funzioni



Dalla misura alla percezione (feature)

- Perception takes raw sensor data and creates a set of useful features

Robot state — Environment state

$$X = \left[X_{R_1}^T \;\; \dots \;\; X_{R_m}^T \;\; X_{L_1}^T \;\; \dots \;\; X_{L_n}^T\right]^T$$

$\begin{bmatrix} x \\ y \\ \phi \end{bmatrix}$ Robot's position

$\begin{bmatrix} x \\ y \\ \phi \end{bmatrix}$ Teammate's position

$\begin{bmatrix} x \\ y \\ x' \\ y' \end{bmatrix}$ Tracked object

$\begin{bmatrix} x \\ y \\ \phi \end{bmatrix}$ Another robot

- Each component of the state vector has an associated **uncertainty model**

15-491 CMRoboBits

The different behaviors

There are two main ways to decide the AIBO's behavior:

Reactive

AIBO directly responds to the environment.

Deliberative

The reponse is mediated by the (internal) state of the AIBO (the state summarizes the past history) and it is based on the prediction of success of the selected choice.

Hybrid

Combination of Reactive and Deliberative behavior.

Sommario della lezione

- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Il planning
- Overview dei sensori ed attuatori di AIBO
- Gli scheletri
- Descrizione della posizione di uno scheletro

Reactive behavior

Action -> Reaction

There is no memory

Advantages:

Immediate response to environment variations.

Simple to program and analyze

The control can be smooth for smooth variations of the sensor input.

Disadvantages:

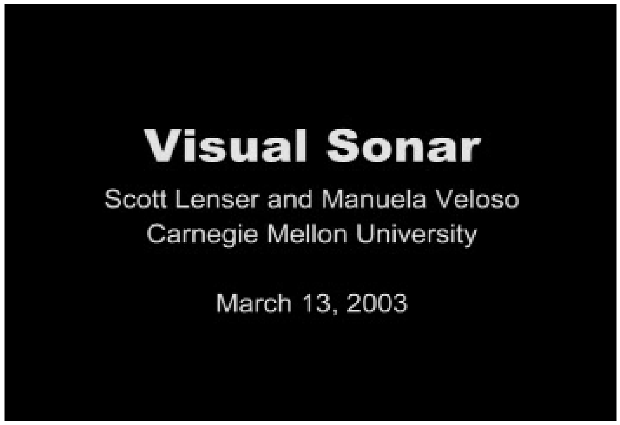
The response to a stimulus is stereotyped.

If an action does not work, there are not alternatives.

Complex tasks with many inputs, may create problems.

Rodney Brooks: http://people.csail.mit.edu/brooks/

Esempio di comportamento reattivo



Schemi motori

Mapping tra informazione proveniente dai sensori e vettore che definisce il movimento del robot.

Per movimento si intende un movimento vero e proprio (cf. Arbib) oppure un vettore spostamento (atto di moto).

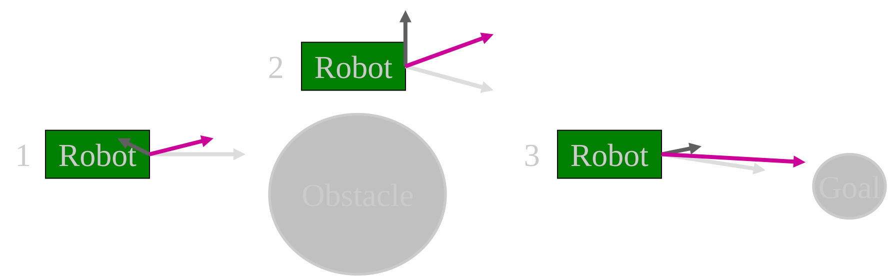
I movimenti prodotti da ciascun motor schema devono essere "*sommati*".

Il mapping deve tenere conto di eventuali ostacoli (o vincoli). Schemi motori che traducono vincoli.

Esempio: navigazione in presenza di ostacoli:

- Obstacle avoidance.
- Goal reaching.

Motor Schemas



Goal vector

Avoidance vector

Resulting vector

Come *sommare* i comportamenti?

Molti moduli possono essere attivi per un certo stimolo sensoriale. Come trattare ciò? Come arrivare ad un comportamento complesso.

- Operatori logici.
- Approccio fuzzy.

In modo più esplicito:

Blending (e.g. motor schema)

Competition - viene selezionato un comportamento mediante competizione.

Subsumption – viene selezionato un comportamento mediante criteri statici.

Sequencing – i comportamenti vengono eseguiti in sequenza.

Blending

E.g. atti motori.

Viene definito uno spazio continuo o campionato dei comportamenti.

Viene associato ad ogni comportamento un vettore (ampiezza e direzione).

I comportamenti vengono sommati vettorialmente ed eventualmente ricondotti al comportamento più vicino alla somma (approccio fuzzy-like).

Blending è facile da implementare quando l'informazione fornita dai sensori è vettoriale: (valori continui con modulo e verso, esempio: posizione, velocità).

Problema: forze con la stessa ampiezza e segno opposto si cancellano.

Similitudine: fuzzy system.

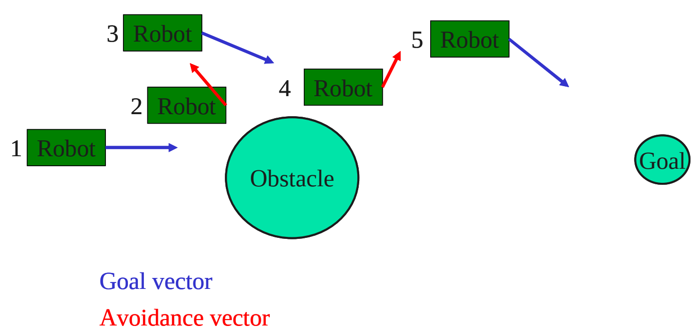
Competition

In questo caso si utilizza un meccanismo di "winner-take-all" per generare l'attivazione.

I comportamenti attivati in modo reattivo, competono; il vincente sarà il comportamento del robot.

E' facile adattarlo ad insiemi diversi di comportamento.

Problemi: oscillazioni tra due comportamenti con modulo molto vicino.



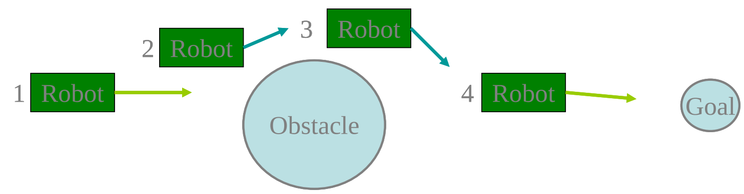
Subsumption

Simile alla Competition. Ma la scelta avviene mediante una priorità codificata a-priori, la scelta non avviene run-time a seconda delle condizioni che si verificano.

Tutti i moduli attivi output il comportamento reattivo previsto.

Il comportamento scelto è quello a priorità più elevata.

E' facile scalarlo, a patto di aggiungere i nuovi comportamento con le loro priorità nella tabella dei comportamenti.



Goal vector

Wall follow vector

Sommario della lezione

- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Gli scheletri
- Descrizione della posizione di uno scheletro

Agente

- Può scegliere un'azione sull'ambiente tra un insieme continuo o discreto.

- L'azione dipende dalla situazione. La situazione è riassunta nello **stato** del sistema.

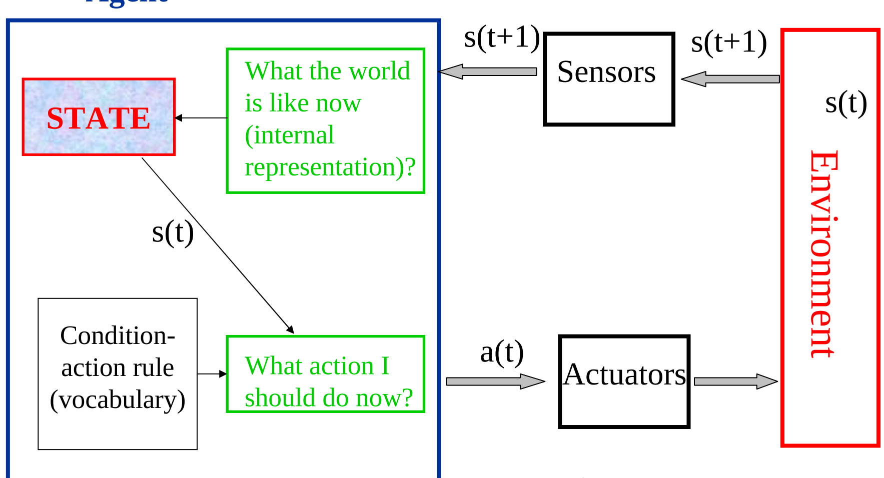
- L'agente monitora continuamente l'ambiente (input) e modifica continuamente lo stato.

- La scelta dell'azione è non banale e richiede un certo grado di "intelligenza".

Schematic diagram of an agent

Agent

Fotografia della situazione ogni Δt



$s(t+1) = f[s(t),a(t)]$
$a(t) = g[s(t)]$

s, stato; a = uscita

s, stato;
a, ingresso

Comportamento reattivo basato su modello: sequencing

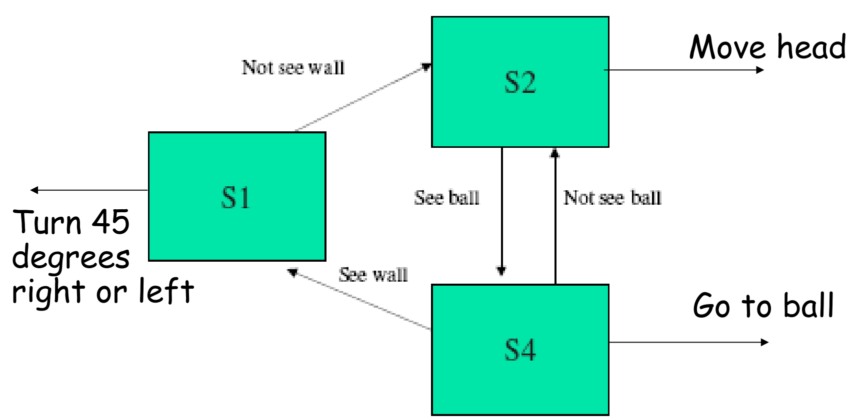
Viene eseguito un solo comportamento alla volta. Si passa da un comportamento ad un altro quando ci sono delle modifiche nell'input sensoriale.

Il modello migliore per catturare questa modalità di comportamento è una macchina a stati finiti (FSM).

Ciascuna macchina ha associato un repertorio di comportamenti.

Particolarmente utilizzato sia in robotica che in animazione digitale!

Rappresentazione dell'utilizzo dei sensori in una FSM



Macchina a Stati Finiti (di Moore)

La Macchina di Moore è definita, in teoria degli automi, dalla quintupla :
< X, I, Y, f(.), g(.) >

X: insieme degli stati (in numero finito).

I: alfabeto di ingresso: tutti i simboli che si possono presentare in ingresso.

Y: alfabeto di uscita: tutti i simboli che si possono generare in uscita.

f(.): funzione stato prossimo: X' = f(X,I). Definisce l'evoluzione della macchina nel tempo. L'evoluzione è deterministica.

g(.): funzione di uscita: Y= g(X) nelle macchine di Moore.

Per il buon funzionamento della macchina è previsto uno stato iniziale, al quale la macchina può essere portata mediante un comando di reset.

Vantaggi in questo approccio

Arricchimento del repertorio delle risposte comportamentali.

A stimoli identici possono corrispondere risposte diverse a seconda dello **stato**.

Esempio: palla vicina. Diverso se in difesa o in attacco (fallo laterale o tiro); se si vince o si perde (azione o melina).

Si possono creare facilmente sequenze di azioni.

Si possono creare dei cicli, ma esistono strumenti di analisi: le oscillazioni sono grandemente ridotte.

Altri vantaggi

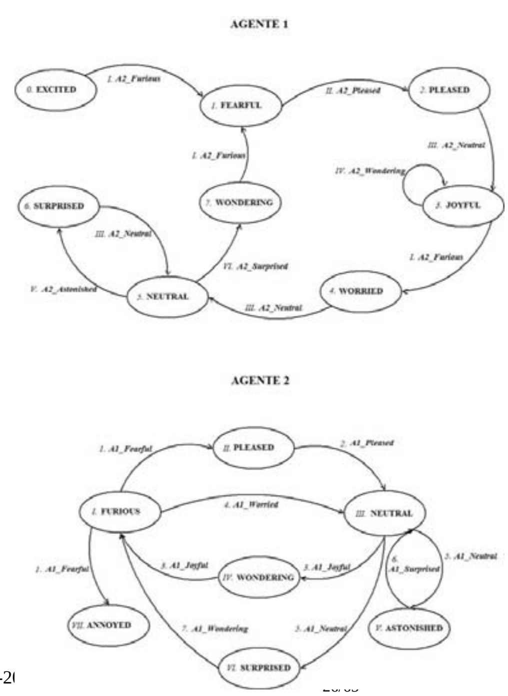
- Apprendimento
  - Per ogni stato, è possibile apprendere l'azione "ottima" per un certo goal.
  - Per ogni goal, le azioni associati ai diversi stati possono essere diverse.

Emotional interaction

Verso l'apprendimento

- Possibilità di arricchire il comportamento:
  - Stochastic automata (Transition function, probabilistica)
  - Attitudine del robot (ad esempio imitativa o compensativa)
  - Personalità (Transition function di base)

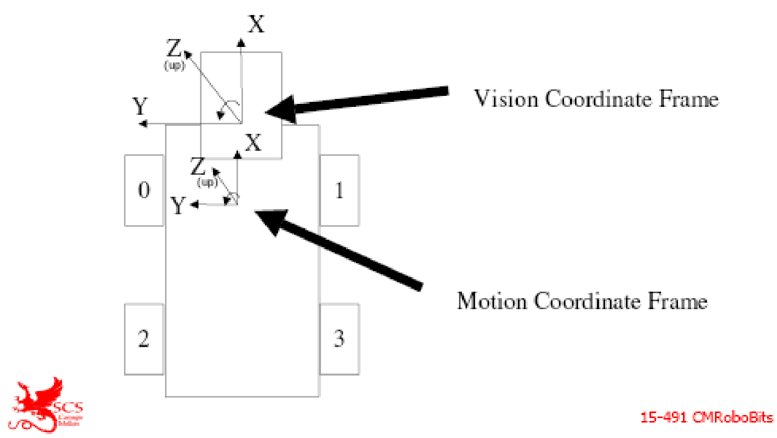
Esempio di FSM



Sommario della lezione

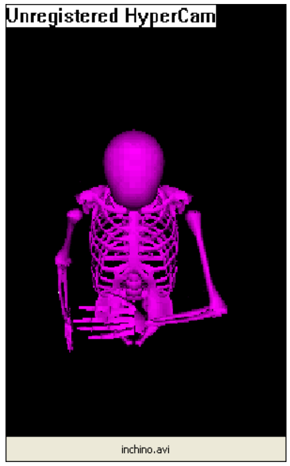
- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Gli scheletri
- Rappresentazione della posizione di uno scheletro

I diversi Reference Frame



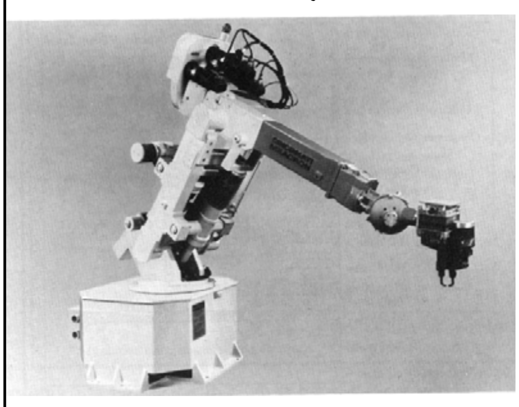
E anche: sensori di contatto, accelerometri, range sensor…
E' necessario ridursi ad un frame comune.

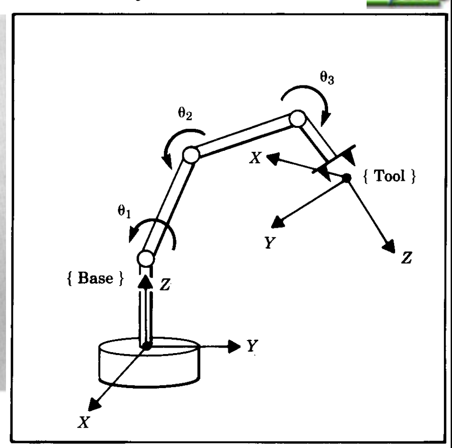
Animazione mediante rotazioni



Scheletro - segmenti incernierati

Descrizione della posizione (scheletro=robot)





Catena cinematica. Struttura gerarchica.

Posizione completamente definita dai gradi di libertà (movimenti concessi dai giunti articolari).

Frame. Sistema di riferimento connesso rigidamente con una parte del robot.

Joints and end-effector

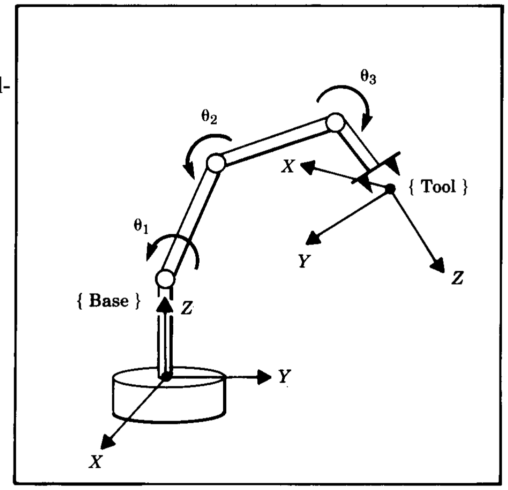
Il braccio è strumentale nel posizionamento ed orientamento dell'end-effector!

Tool frame viene associato all'*end-effector*.

Il base frame (o *root node*) è il sistema di riferimento della catena cinematica.

Joint prismatici o rotatori (*nodes*).

I segmenti sono chiamati anche *link*.



Rappresentazione grafica

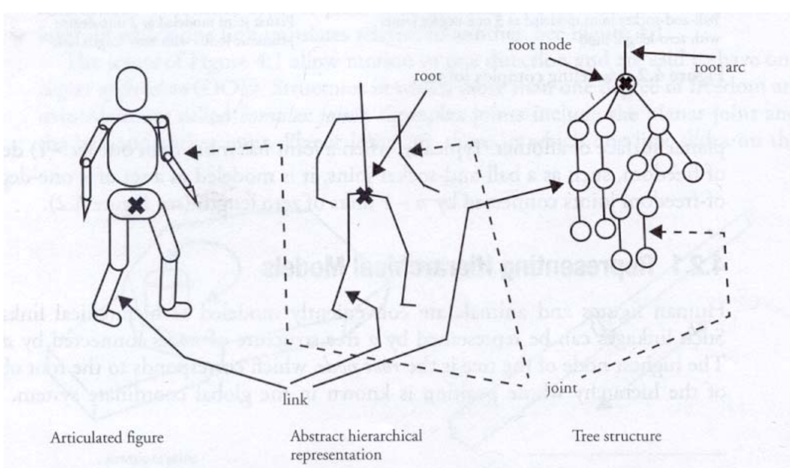
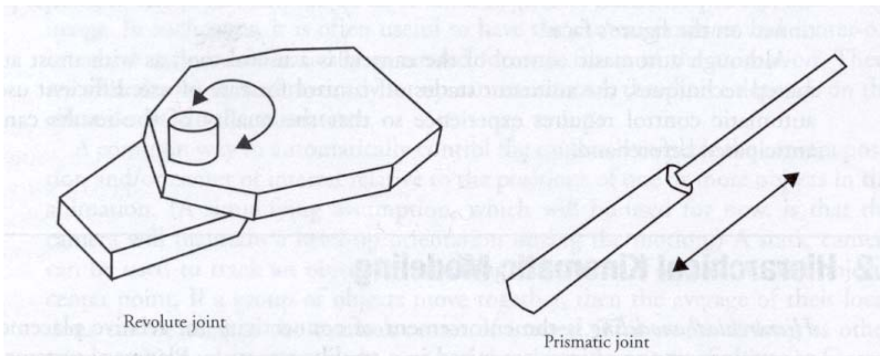
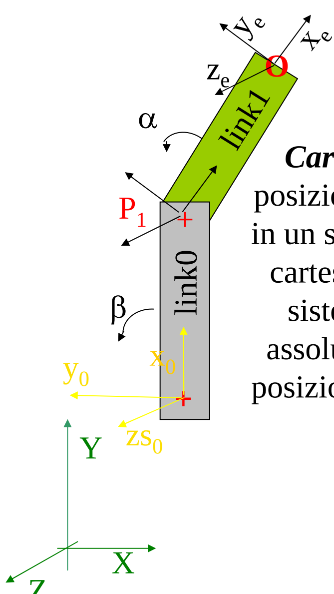


Figure 4.3 Example of a tree structure representing a hierarchical structure

Joint base

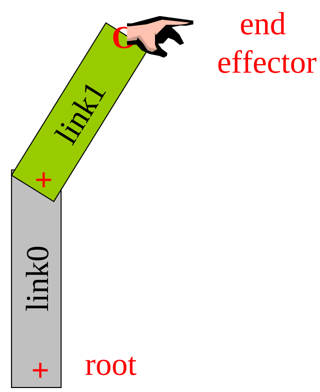


Spazi di movimento



Joint space. E’ lo spazio dei parametri liberi. In questo esempio: α e β.

Cartesian space. E’ la posizione di punti, cerniere in un sistema di riferimento cartesiano, ad esempio il sistema di riferimento assoluto. In particolare la posizione dell’end-effector.



Descrizione della posizione

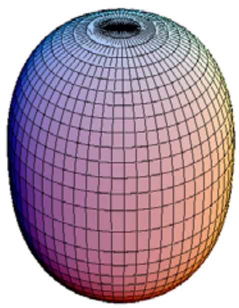
- Trasformazione da un frame all'altro.
- La trasformazione è funzione dei parametri liberi e dei parametri geometrici.
- Trasformazioni tra sistemi di riferimento: rototraslazione espressa mediante matrici affini (trasformazioni matriciali).

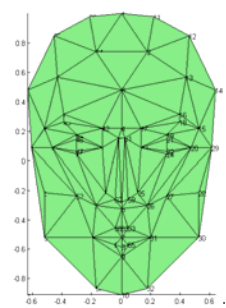
Sommario della lezione

- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Gli scheletri
- Rappresentazione della posizione di uno scheletro

Descrizione della posizione di un corpo rigido (non solo scheletri)

- Punto materiale: **P** = **P**(t) – 3 dof
- Corpo rigido: 6 dof [**R**(t), **T**(t)].
- Corpo deformabile: N dof **G**(t)





Coordinate omogenee

Spazio delle classi di equivalenza: ogni punto in coordinate carteziane 3D corrisponde a infiniti punti nello spazio omogeneo 4D che differiscono solo per un fattore moltiplicativo *w*:

$$V(x, y, z) \text{ corrisponde a :}$$
$$V(X, Y, Z, w)$$

Il passaggio tra lo spazio omogeneo e lo spazio 3D:

$$x = X / w$$
$$y = Y / w$$
$$z = Z / w$$

solitamente si sceglie *w=1*

w = 0 identifica il punto all'∞ sulla retta per l'origine, passante per V.

I coseni direttori saranno x/|V|, y/|V|, z/|V|.

Trasformazioni 3D

Traslazione – tutti i punti si spostano della stessa quantità (vettore spostamento). Di solito si considera la traslazione del baricentro.

Rotazione – tutti i punti lungo una retta chiamata asse non si spostano. Gli altri punti descrivono circonferenze perpendicolari all'asse.

Scala – variazione della dimensione lungo un asse.

Traslazione in coordinate omogenee

Vengono espresse come trasformazioni nello spazio di coordinate omogenee 4D come prodotto tra matrici.

Traslazione

$$T = \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad V' = TV = \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$x' = (x + 0 + 0 + T_x)$$
$$y' = (0 + y + 0 + T_y)$$
$$z' = (0 + 0 + z + T_z)$$
$$w' = (0 + 0 + 0 + 1)$$

coord. omogenee

$$x^t = x'/w' = (x + T_x)/1 = x + T_x$$
$$y^t = y'/w' = (y + T_y)/1 = y + T_y$$
$$z^t = z'/w' = (z + T_z)/1 = z + T_z$$

coord. cartesiane

Scala in coordinate omogenee

$$S = \begin{pmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad V' = SV = \begin{pmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$\begin{aligned} x' &= (x.S_x + 0 + 0 + 0) \\ y' &= (0 + y.S_y + 0 + 0) \\ z' &= (0 + 0 + z.S_z + 0) \\ w' &= (0 + 0 + 0 + 1) \end{aligned}$$

coord. omogenee

$$\begin{aligned} x^s &= x'/w' = (x.S_x)/1 \\ y^s &= y'/w' = (y.S_y)/1 \\ z^s &= z'/w' = (z.S_z)/1 \end{aligned}$$

coord. cartesiane

Traslazione + Scala

$$V' = TV = \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \qquad V'' = SV' = \begin{pmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix}$$

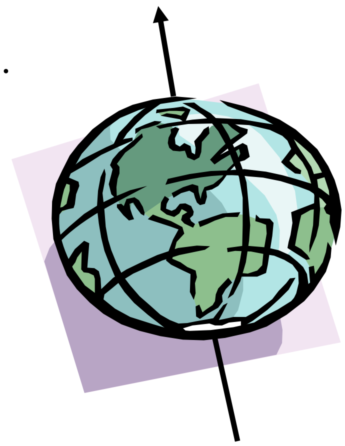
Traslazione — Scala

$$V'' = S(TV) = (ST)V = \begin{pmatrix} S_x & 0 & 0 & T_x \\ 0 & S_y & 0 & T_y \\ 0 & 0 & S_z & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Fattorizzazione delle trasformazioni: rappresentazione della trasformazione in un'unica matrice.

Traslazione + Scala

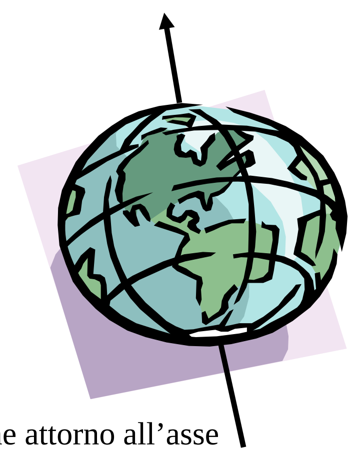
La rotazione



Ammette rappresentazioni diverse.

1) Quaternioni (asse + angolo)
2) Matrice di rotazione
3) Tre angoli di rotazione indipendenti

Quaternioni



Rappresentazione della rotazione mediante: 1 vettore + 1 scalare

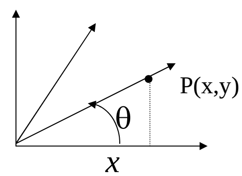
Asse di rotazione Angolo di rotazione

Si può dimostrare che data una rotazione attorno all'asse identificato dal versore **n**, di un angolo $-\pi <= \theta <= \pi$, questa può essere rappresentata dal quaternione: $q=(\cos \theta/2, \mathbf{n} \sin \theta/2)$

La rotazione attorno a z (forma matriciale)

$$P' = \begin{bmatrix} \cos\vartheta & \sin\vartheta & 0 \\ -\sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

P' = RP



Matrice di rotazione

$$\sum_{i=1}^{3} m^2_{ij} = 1$$

det(M) = 1

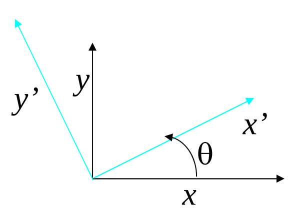
$$\sum_{i=1}^{3} m_{ij} m_{jk} = 0 \quad i \neq k$$

Matrice ortonormale

Significato geometrico della matrice di rotazione

Ruotiamo il sistema di riferimento xy in x'y' di un angolo -θ.

$$P' = \begin{bmatrix} \cos\vartheta & \sin\vartheta & 0 \\ -\sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



Matrice di rotazione

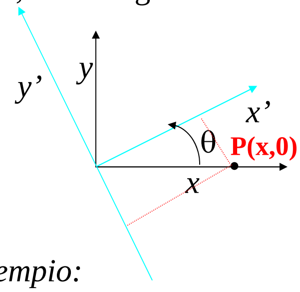
$$\mathbf{M} = \begin{bmatrix} x \bullet x' & x \bullet y' & 0 \\ y \bullet x' & y \bullet y' & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

M contiene la proiezione degli assi del sistema di riferimento xy sugli assi di x'y'.

Significato geometrico della matrice di rotazione

Consideriamo che il punto P -> P' sia un punto appartenente all'asse x, P(x,0) e che P' appartenga ad un asse x', ottenuto ruotando il sistema di riferimento xy in x'y', di un angolo -θ.

$$P' = \begin{bmatrix} \cos\vartheta & \sin\vartheta & 0 \\ -\sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



$$\mathbf{M} = \begin{bmatrix} x' \bullet x & x' \bullet y & 0 \\ y' \bullet x & y' \bullet y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

M contiene la proiezione degli assi (dei versori) del sistema di riferimento xy sugli assi di x'y' .

Esempio:

x' = | P | cos(θ) = xcos(θ)

y' = | P | cos[(90+θ)] = -xsin(θ)

Si può estendere a punti che non giacciano su uno dei due assi coordinati.

Rotazione attorno a z (coordinate omogenee)

$$V' = R_z V = \begin{pmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

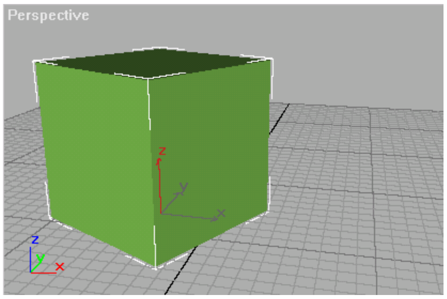
$$\begin{aligned} x' &= (x.\cos\theta + y.\sin\theta + 0 + 0) \\ y' &= (-x.\sin\theta + y.\cos\theta + 0 + 0) \\ z' &= (0 + 0 + z + 0) \\ w' &= (0 + 0 + 0 + 1) \end{aligned}$$

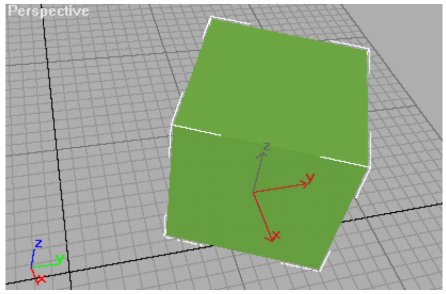
coord. omogenee

$$\begin{aligned} x^{R_z} &= x'/w' = (x.\cos\theta + y.\sin\theta)/1 \\ y^{R_z} &= y'/w' = (-x.\sin\theta + y.\cos\theta)/1 \\ z^{R_z} &= z'/w' = (z.1)/1 \end{aligned}$$

coord. cartesiane

Orientamento di un corpo rigido nello spazio



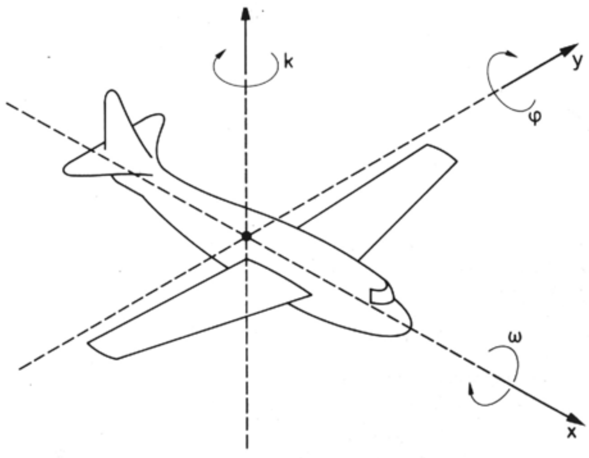


Tre parametri: tre rotazioni indipendenti.

Angoli di orientamento nello spazio 3D

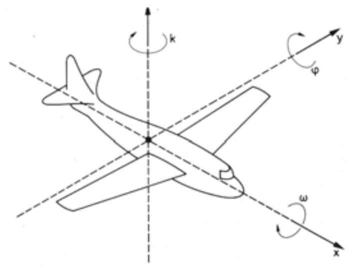
Modo generale: roll, pitch, e yaw.
(ω, ϕ, k): rollio, beccheggio e deriva.

Sono 3 rotazioni sequenziali, non commutative.



Rotazione attorno ad un singolo asse

Modo generale: roll, pitch, e yaw.
(ω, ϕ, k): rollio, beccheggio e deriva.

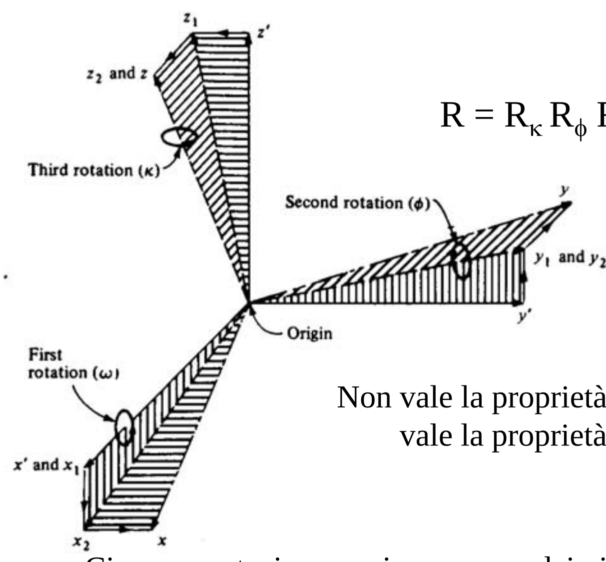


$$R_\omega = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix}$$

$$R_\phi = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}$$

$$R_\kappa = \begin{bmatrix} \cos k & \sin k & 0 \\ -\sin k & \cos k & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotazioni sequenziali



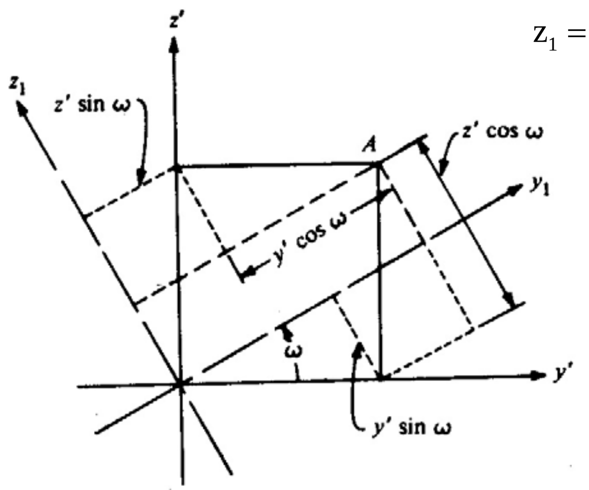
$$R = R_\kappa R_\phi R_\omega$$

Non vale la proprietà commutativa,
vale la proprietà associativa.

Ciascuna rotazione avviene su uno dei piani coordinati.

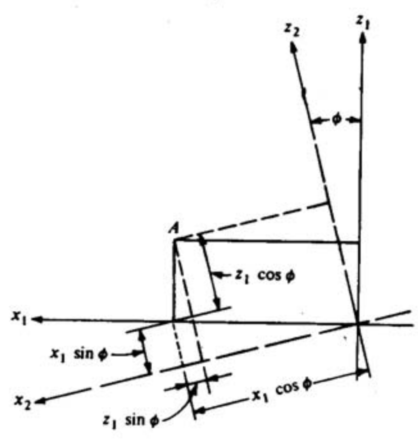
I) Rotazione attorno all'asse x (roll)

$$x_1 = x$$
$$y_1 = y'\cos\omega + z'\sin\omega$$
$$z_1 = -y'\sin\omega + z'\cos\omega$$



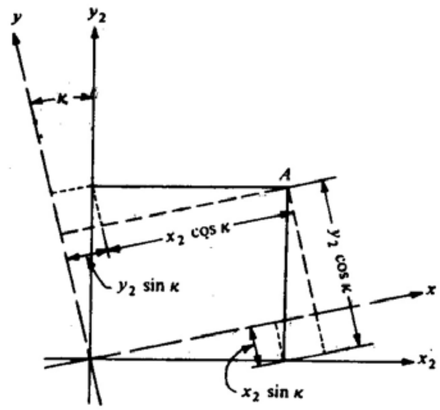
II) Rotazione attorno all'asse y (pitch)

$$x_2 = x_1\cos\phi - z_1\sin\phi$$
$$y_2 = y_1$$
$$z_2 = +x_1\sin\phi + z_1\cos\phi$$



$$x_2 = x'\cos\phi - (-y'\sin\omega + z'\cos\omega)\sin\phi$$
$$y_2 = y'\cos\omega + z'\sin\omega$$
$$z_2 = +x'\sin\phi + (-y'\sin\omega + z'\cos\omega)\cos\phi$$

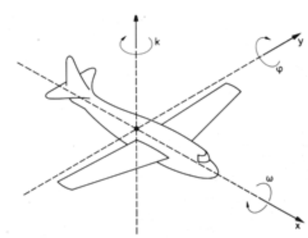
III) Rotazione attorno all'asse z (yaw)



$$x_3 = x_2 \cos k + y_2 \sin k$$
$$y_3 = -x_2 \sin k + y_2 \cos k$$
$$z_3 = z_2$$

$$x_3 = [x'\cos\phi - (-y'\sin\omega + z'\cos\omega)\sin\phi]\cos k + [y'\cos\omega + z'\sin\omega]\sin k$$
$$y_3 = -[x'\cos\phi + (-y'\sin\omega + z'\cos\omega)\sin\phi]\sin k + [y'\cos\omega + z'\sin\omega]\cos k$$
$$z_3 = +x'\sin\phi + (-y'\sin\omega + z'\cos\omega)\cos\phi$$

Dalle rotazioni alla matrice di rotazione



Come è legata R alle tre rotazioni indipendenti?

$$\mathbf{R} = \begin{bmatrix} \cos\phi\cos k & \sin\omega\sin\phi\cos k + \cos\omega\sin k & -\cos\omega\sin\phi\cos k + \sin\omega\sin k \\ -\cos\phi\sin k & -\sin\omega\sin\phi\sin k + \cos\omega\cos k & \cos\omega\sin\phi\sin k + \sin\omega\cos k \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{bmatrix}$$

Si ricava eseguendo le rotazioni sequenziali. Ogni rotazione tiene fermo un asse e agisce sul piano perpendicolare.

Rotazioni "*semplici*" utilizzate dai programmi di animazione, gestione matriciale *efficiente* del calcolo.

Rotazione generica (coordinate omogenee)

$$V' = R_z V = \begin{pmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Trasformare gli oggetti

- i vertici dell'oggetto vengono trasformati (le loro coordinate modificate)
- denotiamo i vertici (punti) come vettore colonna ***V.***
- ***R, D*** e ***S*** sono matrici associate a rotazione, traslazione e scala
- Il punto trasformato si ottiene come:

V'=V+D traslazione, ***D*** è un vettore di traslazione
V'=SV scala, ***S*** è una matrice di scala
V'=RV rotazione, ***R*** è una matrice di rotazione

- Il punto trasformato si ottiene **in coordinate omogenee** come:

V'=V * D traslazione, ***D*** è una matrice 4x4 che contiene il vettore di traslazione
V'=S * V scala, ***S*** è una matrice di scala 4 x 4.
V'=R * V rotazione, ***R*** è una matrice 4x4 che contiene la matrice di rotazione

La rototraslazione in forma matriciale

P' = RP + T => P' = AP

$$\begin{bmatrix} X'_P \\ Y'_P \\ Z'_P \\ 1 \end{bmatrix} = \begin{bmatrix} \boxed{\begin{matrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{matrix}} & \boxed{\begin{matrix} T_x \\ T_y \\ T_z \end{matrix}} \\ \begin{matrix} 0 & 0 & 0 \end{matrix} & 1 \end{bmatrix} \begin{bmatrix} X_P \\ Y_P \\ Z_P \\ 1 \end{bmatrix}$$

Matrice di rotazione

Vettore di traslazione

Composizione di trasformazioni

- Si possono applicare trasformazioni in successione, moltiplicando in ordine opportuno le matrici.

  $$V'' = A_2 A_1 V = A_2(A_1 V) = (A_2 A_1)V$$

  - la trasf. A_1 viene applicata per prima!

- ricordiamo che il prodotto di matrici <u>non è *commutativo*</u>: $A_2A_1 \neq A_1A_2$, mentre vale la proprietà associativa: $A_2(A_1V) = (A_2A_1)V$.

- ***L'applicazione di trasformazioni dipende dall'ordine con cui sono applicate.***
- ***Tutte le traslazioni, rotazioni e variazioni di scala, possono essere rappresentata in un'unica matrice.***

Trasformazioni inverse

- La trasformazione inversa si ottiene invertendo l'ordine delle trasformazioni ed invertendo le singole matrici:

$$A = A_3 A_2 A_1 \Leftrightarrow A^{-1} = A_1^{-1} A_2^{-1} A_3^{-1}$$

- Denotiamo le inverse come le matrici di trasformazione: T^{-1}, S^{-1}, R^{-1}.
- La traslazione inversa si ottiene *negando* i coefficienti di traslazione.
- La scala inversa si ottiene prendendo il *reciproco* dei coefficienti.
- La rotazione inversa si ottiene *negando* l'angolo di rotazione. Matrice trasposta. Si può verificare invertendo il segno e l'ordine delle rotazioni:

$$R = R_\omega R_\phi R_\kappa \Rightarrow R^T = R_{-\kappa} R_{-\phi} R_{-\omega}$$

La rototraslazione inversa in forma matriciale

P' = RP + T => P' = AP

$$\begin{bmatrix} X'_P \\ Y'_P \\ Z'_P \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_P \\ Y_P \\ Z_P \\ 1 \end{bmatrix}$$

$R^TRP = +R^TP' - R^TT \Rightarrow P = A^{-1}P'$

Proiezione di **T** sugli assi di arrivo: $\mathbf{r_i} \bullet \mathbf{T}$

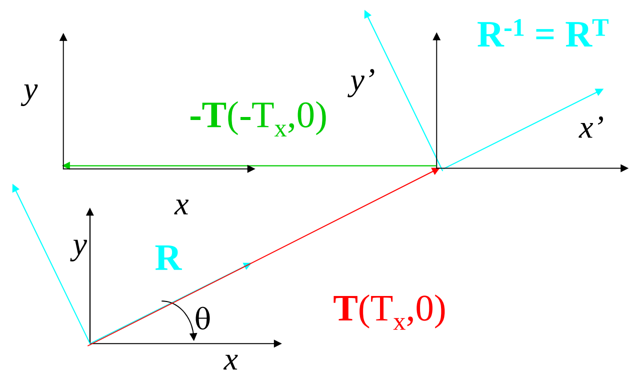
$$\begin{bmatrix} X_P \\ Y_P \\ Z_P \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & -(r_{11}T_x + r_{21}T_y + r_{31}T_z) \\ r_{12} & r_{22} & r_{32} & -(r_{12}T_x + r_{22}T_y + r_{32}T_z) \\ r_{13} & r_{23} & r_{33} & -(r_{13}T_x + r_{23}T_y + r_{33}T_z) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X'_P \\ Y'_P \\ Z'_P \\ 1 \end{bmatrix}$$

Matrice di rotazione (inversa)

Vettore di traslazione (inverso)

Perchè $-R^T T$?

Solo così applicando trasformata diretta ed inversa riportano un sistema di riferimento nella posizione iniziale.



$\mathbf{R^T T}$ è la proiezione del vettore traslazione sul sistema di riferimento ruotato.

Trasformazioni rigide

- rappresentate con matrici
- più trasformazioni possono essere combinate moltiplicando tra loro le matrici che rappresentano ciascuna trasformazione loro, creando una sola trasformazione matriciale.
- una trasformazione si ottiene in generale combinando trasformazioni di diverso tipo: rotazioni, scala, scala e traslazione.

Sommario della lezione

- I comportamenti
- Comportamento reattivo
- Comportamento deliberativo (FSM)
- Gli scheletri
- Rappresentazione della posizione di uno scheletro