



---

## *Robotics 2*

# Control in the Cartesian Space

Prof. Alessandro De Luca

DIPARTIMENTO DI INGEGNERIA INFORMATICA  
AUTOMATICA E GESTIONALE ANTONIO RUBERTI



SAPIENZA  
UNIVERSITÀ DI ROMA



# Regulation of robot Cartesian pose

- “PD +” type control for **regulation** problems
  - proportional to the **Cartesian pose error**, with a derivative term (on **velocity**) + cancellation/compensation of gravity **in joint space**
- robot
  - dynamics  $M(q)\ddot{q} + S(q, \dot{q})\dot{q} + g(q) = u$  dimension of spaces  
joint =  $n$
  - kinematics  $p = f(q) \rightarrow \dot{p} = J(q)\dot{q}$  Cartesian =  $m$
- **goal**: asymptotic stabilization of the end-effector pose

$$p = p_d, \dot{q} = 0 \rightarrow \dot{p}_d = 0$$

**Note**: if  $n = m$ , then  $\dot{q} = 0 \Leftrightarrow \dot{p} = 0$  up to **singularities**

if  $n > m$ , then the goal is **not** uniquely associated to a complete robot state:  $n - m$  joint coordinates are missing...



# A Cartesian regulation law

$$(*) \quad u = J^T(q)K_P(p_d - p) - K_D\dot{q} + g(q) \quad K_P, K_D > 0 \text{ (symmetric)}$$

## Theorem

under the control law (\*), the robot state will converge asymptotically

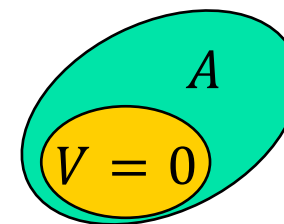
to the set  $A = \{\dot{q} = 0, q: K_P(p_d - f(q)) \in N(J^T(q))\}$   
 $\cong \{\dot{q} = 0, q: f(q) = p_d\}$

## Proof

define  $e_p = p_d - p$  (Cartesian error) and the associated Lyapunov-like candidate function

$$V = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}e_p^T K_P e_p$$

with  $V = 0 \Leftrightarrow (q, \dot{q}) \in \{\dot{q} = 0, q: f(q) = p_d\} \subseteq A$





## Proof (cont)

differentiating  $V = \frac{1}{2} \dot{q}^T M(q) \dot{q} + \frac{1}{2} e_p^T K_P e_p \geq 0$

$$\begin{aligned}\dot{V} &= \dot{q}^T \left( M\ddot{q} + \frac{1}{2} \dot{M}\dot{q} \right) - e_p^T K_P \dot{p} \\ &= \dot{q}^T \left( u - S\dot{q} - g + \frac{1}{2} \dot{M}\dot{q} \right) - e_p^T K_P \dot{p} \\ &= \dot{q}^T \left( J^T K_P e_p - K_D \dot{q} + g - g \right) - e_p^T K_P J \dot{q} \\ &= -\dot{q}^T K_D \dot{q} \leq 0\end{aligned}$$

with  $\dot{V} = 0 \Leftrightarrow \dot{q} = 0$

in this situation, the **closed-loop equations** become

$$M(q)\ddot{q} + g(q) = J^T(q)K_P e_p + g(q) \quad \Rightarrow \quad \ddot{q} = M^{-1}(q)J^T(q)K_P e_p$$

$$\Rightarrow \quad \ddot{q} = 0 \Leftrightarrow K_P e_p \in N(J^T(q))$$

by applying LaSalle theorem, the thesis follows





# Corollary

for a given initial state  $(q(0), \dot{q}(0))$ , if the robot **does not encounter any singularity** of  $J^T(q)$  (configurations where  $\rho(J^T) < m \leq n$ ) during its motion, then there is **asymptotic stabilization** to one single state ( $m = n$ ) or to a set of states ( $m < n$ ) such that

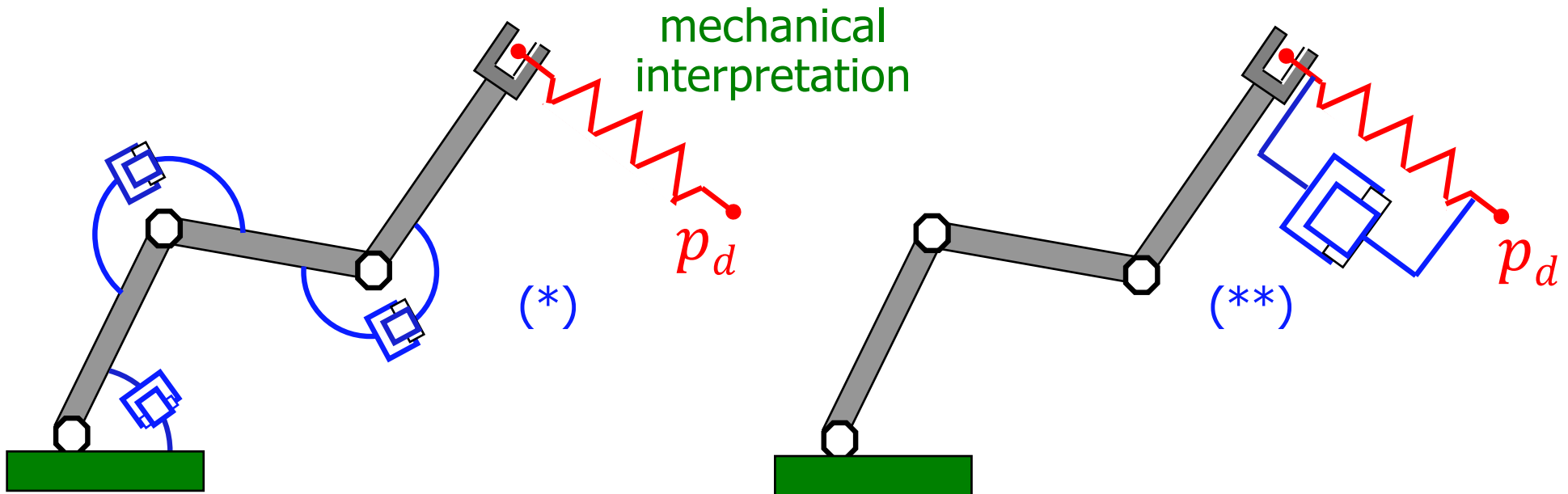
$$e_P = 0, \dot{q} = 0$$

**Note:** singular configurations  $q$  of  $J^T(q)$  coincide with those of  $J(q)$

# A possible variant for regulation

“all Cartesian” PD control + gravity cancellation in joint space

$$(**) \quad u = J^T(q)[K_P(p_d - p) - K_D\dot{p}] + g(q) \quad K_P, K_D > 0 \text{ (symmetric)}$$



$J^T$  transforms the “virtual” elastic, for (\*), or visco-elastic, for (\*\*), force/torque acting on the end-effector into control torques at the joints



# Feedback linearization in Cartesian space

robot  $M(q)\ddot{q} + c(q, \dot{q}) + g(q) = u$

output

$$y = p, \quad p = f(q)$$

Cartesian  
position/orientation

assume:  $M = N$

algorithm

differentiate the output(s) as many times as needed up to the appearance of (at least one of) the input torque(s), then verify if it is possible to solve for the input = "inversion"

uniform  
"relative degree"  
 $\rho = 2$   
for all outputs

$$y = f(q)$$

$$\dot{y} = J(q)\dot{q}$$

$$\ddot{y} = J(q)\ddot{q} + \dot{J}(q)\dot{q}$$

$$= J(q)M^{-1}(q)u - J(q)M^{-1}(q)[c(q, \dot{q}) + g(q)] + \dot{J}(q)\dot{q}$$

from the dynamic model

Theorem

for a non-redundant robot, it is possible to exactly linearize and decouple the dynamic behavior at the Cartesian level if and only if

$$\det J(q) \neq 0$$

# Feedback linearization in Cartesian space (in the right coordinates!)



control law

$$u = M(q)J^{-1}(q)a + c(q, \dot{q}) + g(q) - M(q)J^{-1}(q)\dot{J}(q)\dot{q}$$

$$= \beta(q)a + \alpha(q, \dot{q})$$



$$\ddot{y} = \ddot{p} = J(q)M^{-1}(q)u - J(q)M^{-1}(q)[c(q, \dot{q}) + g(q)] + \dot{J}(q)\dot{q} = a$$

$p, \dot{p}$  are the so-called "**linearizing**" coordinates

closed-loop equations (in the **joint space**)

$$M^{-1} * M\ddot{q} + c + g = MJ^{-1}a + c + g - MJ^{-1}\dot{J}\dot{q}$$



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

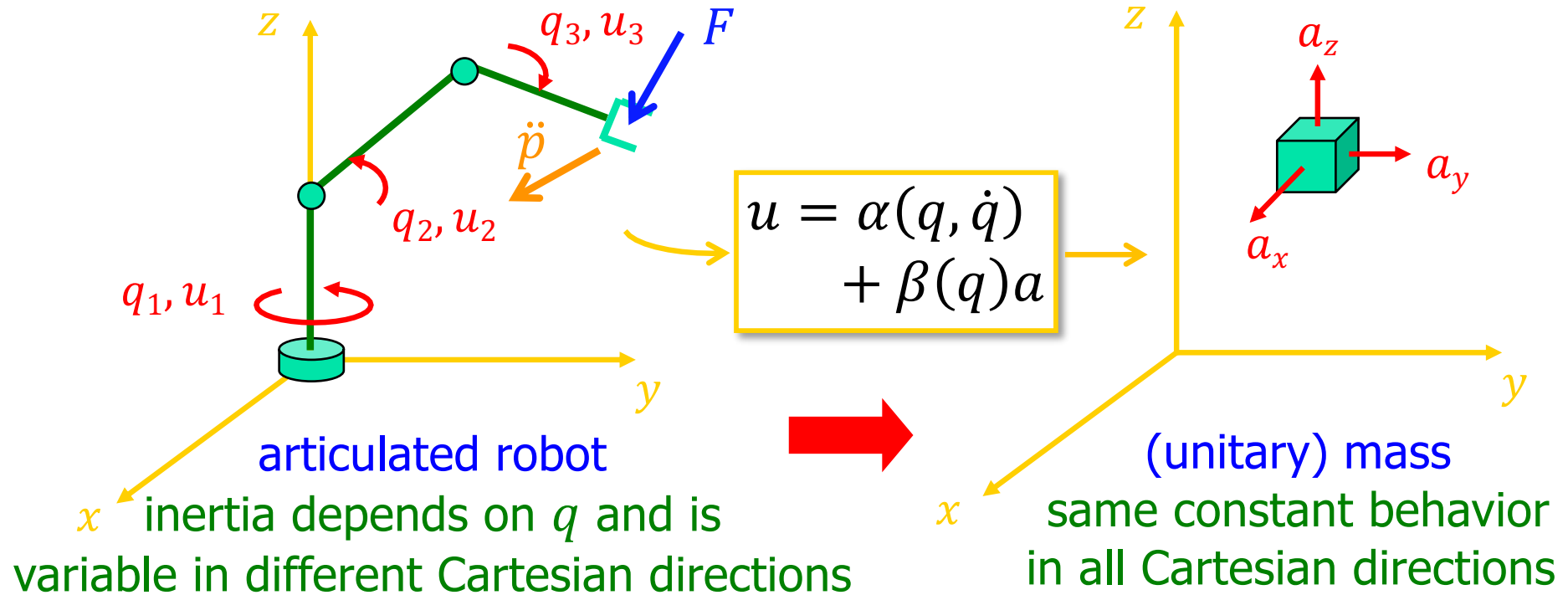
purely  
kinematic  
equations

(but still **nonlinear** and **coupled!!**)





# Physical interpretation



when a control force  $F$  is applied at the end-effector

- the uncontrolled robot will accelerate with  $\ddot{p}$  in a different direction
- the (unitary) mass accelerates in the **same** direction of applied force  $F$



# Alternative derivation in purely Cartesian terms

the previous exact linearizing and decoupling law can be rewritten in **Cartesian terms** using a **control** force/torque  $F$

$$u = M(q)J^{-1}(q)a + c(q, \dot{q}) + g(q) - M(q)J^{-1}(q)\dot{J}(q)\dot{q}$$

joint torque  $u$  is moved to the **Cartesian space** as  $F = J^{-T}(q)u$  (for  $m = n$ )

$$\begin{aligned} F &= [J^{-T} M J^{-1}] a && \longrightarrow \text{Cartesian inertia [= } M_p(p)\text{]} \\ &+ [J^{-T} c - J^{-T} M J^{-1} \dot{J} \dot{q}] && \longrightarrow \text{Cartesian Coriolis/centrifugal terms} \\ &+ [J^{-T} g] && \longrightarrow \text{Cartesian gravity} \\ &= M_p a + c_p + g_p \end{aligned}$$

 this is the feedback linearization law applied to the **Cartesian dynamic model** of the robot

$$M_p(p)\ddot{p} + c_p(p, \dot{p}) + g_p(p) = F$$

  $\ddot{p} = a$



## Remarks - 1

- the design of a **Cartesian trajectory tracking control** is completed by **stabilizing** the tracking error in the  **$m$  independent** chains of double integrators, i.e., by setting

$$a_i = \ddot{p}_{di} + K_{Di}(\dot{p}_{di} - \dot{p}_i) + K_{Pi}(p_{di} - p_i)$$

$$\begin{array}{l} \text{scalars} \\ K_{Pi} > 0, K_{Di} > 0 \\ i = 1, \dots, m \end{array}$$

- in **redundant** ( $m < n$ ) robots: replacing  $J^{-1}$  with  $J^\#$  in the control law, we obtain **input-output** decoupling and linearization, but not exact linearization of the whole **state** dynamics
  - there is an remaining internal dynamics of dimension  $n - m$  left
- the Cartesian pose/velocity can either be directly **measured** by external sensors (cameras) or **computed** through the direct and differential kinematics of the robot arm



## Remarks - 2

---

- the transient behavior of the Cartesian error along a desired trajectory is **exponentially stable** (with arbitrary eigenvalues assigned by choosing the diagonal gains of  $K_P, K_D$ )
- when applied to the case  $p_d = \text{constant}$  (regulation task), the control law becomes

$$u = M(q)J^{-1}(q)[K_P e_P - K_D J(q)\dot{q}] + c(q, \dot{q}) + g(q) - M(q)J^{-1}(q)\dot{J}(q)\dot{q}$$

which is computationally more expensive than a control law designed directly for regulation, such as the previous laws (\*) or (\*\*), but keeps the additional property of obtaining an **exponentially stable** transient error



# Conclusion

---

- most of the control laws presented in the joint space (i.e., driven by a joint error) can be **translated** with relative ease to the Cartesian space, e.g.
  - regulation with constant gravity compensation
  - adaptive regulation
  - robust control for trajectory tracking
  - adaptive control for trajectory tracking
- the **main issues** are related to
  - presence of kinematic singularities, both for the Jacobian transpose or the Jacobian inverse control laws, needs some suitable **singularity robustness** modifications
  - presence of kinematic redundancy ( $m < n$ ) needs the use of an **additional stabilizing null-space torque** for the extra  $n - m$  generalized coordinates (locally,  $n - m$  joint variables)