Robotics 2

Hybrid Force/Motion Control

Prof. Alessandro De Luca

Dipartimento di Ingegneria Informatica
Automatica e Gestionale Antonio Ruberti

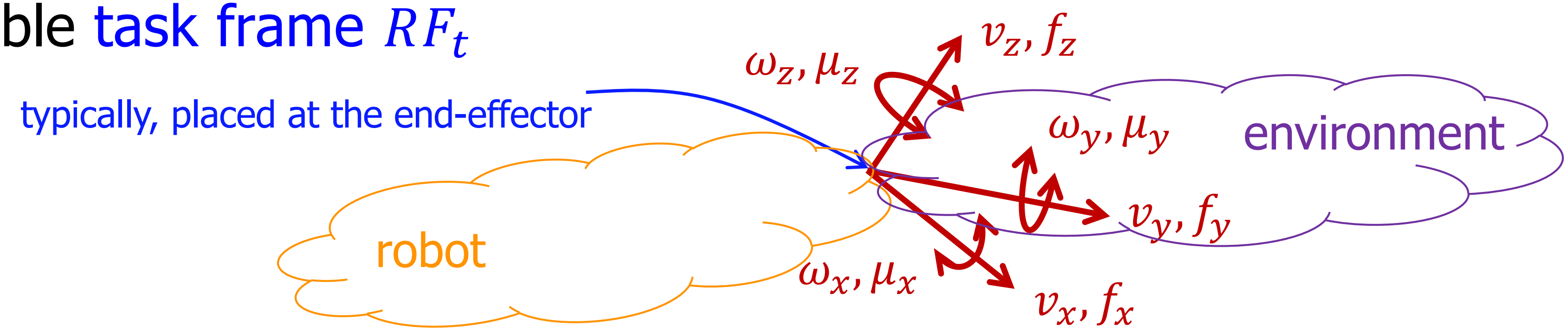
Sapienza
Università di Roma

Hybrid force/motion control

- we consider contacts/interactions between a robot and a stiff environment that naturally constrains the end-effector motion
- compared to an approach using the constrained/reduced robot dynamics with (bilateral) geometric constraints, the differences are
  - the hybrid control law is designed in ideal conditions, but now unconstrained directions of motion and constrained force directions are defined in a more direct way using a task frame formalism
  - all non-ideal conditions (compliant surfaces, friction at the contact, errors in contact surface orientation) are handled explicitly in the control scheme by a geometric filtering of the measured quantities
    - only signal components that should appear in certain directions based on the nominal task model are considered
    - those that should not be there are treated as disturbances to be rejected
- in this way, the hybrid control law avoids to introduce conflicting behaviors (force vs. motion control) in each task space direction!

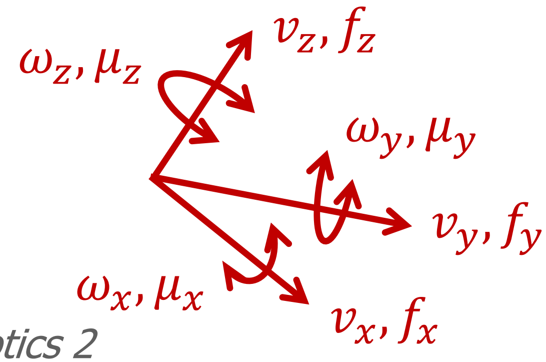
Natural constraints

- in ideal conditions (robot and environment are perfectly rigid, contact is frictionless), two sets of generalized directions can be defined in the task space
  - end-effector motion (v/ω) is prohibited along/around $6-k$ directions (since the environment reacts there with forces/torques)
  - reaction forces/torques (f/μ) are absent along/around k directions (where the environment does not prevent end-effector motions)
- these constraints have been called the natural constraints on motion and force associated to the task geometry
- the two sets of directions are characterized through the axes of a suitable task frame RF_t



Artificial constraints

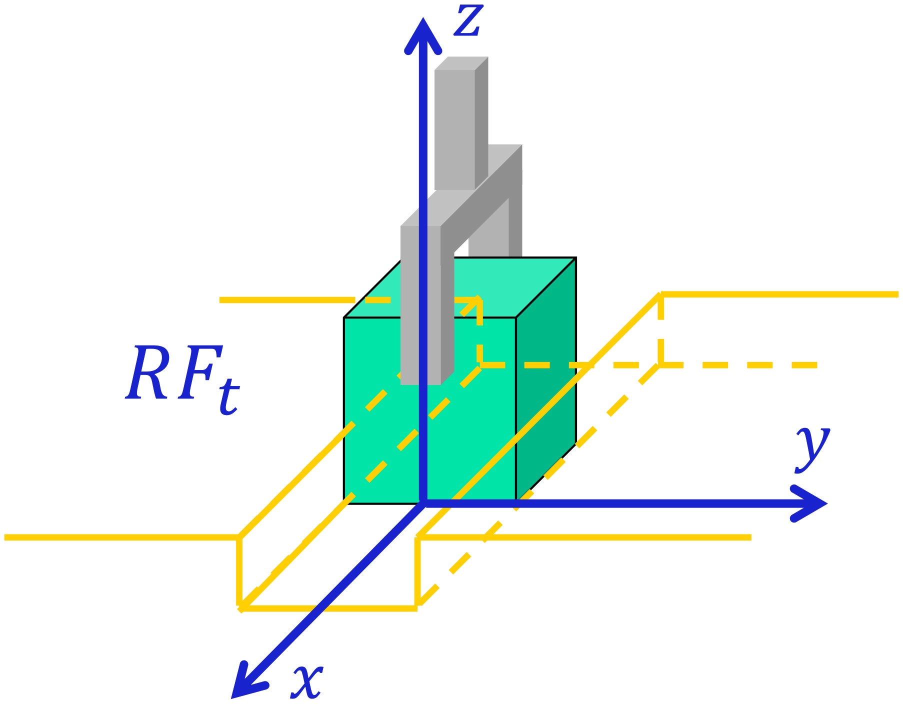
- the way task execution should be performed can be expressed in terms of so-called artificial constraints that specify the desired values (to be imposed by the control law)
  - for the end-effector velocities (v/ω) along/around k directions where feasible motions can occur
  - for the contact forces/torques (f/μ) along/around $6-k$ directions where admissible reactions of the environment can occur
- the two sets of directions are complementary (they cover the 6D generalized task space) and mutually orthogonal, while the task frame can be time-varying ("moves with task progress")
  - directions are intended as 6D screws: twists $V = (v^T \ \omega^T)^T$ and wrenches $F = (f^T \ \mu^T)^T$



$$F^T V = 0 \iff \text{orthogonality}$$

but ill-defined (don't use it!) for $V_1^T V_2$ or $F_1^T F_2$

Task frame and constraints - example 1



task: slide the cube along a guide

natural (geometric) constraints

$$\left.\begin{array}{l} v_y = v_z = 0 \\ \omega_x = \omega_z = 0 \end{array}\right\} 6 - k = 4$$

$$\left. f_x = \mu_y = 0 \right\} k = 2$$

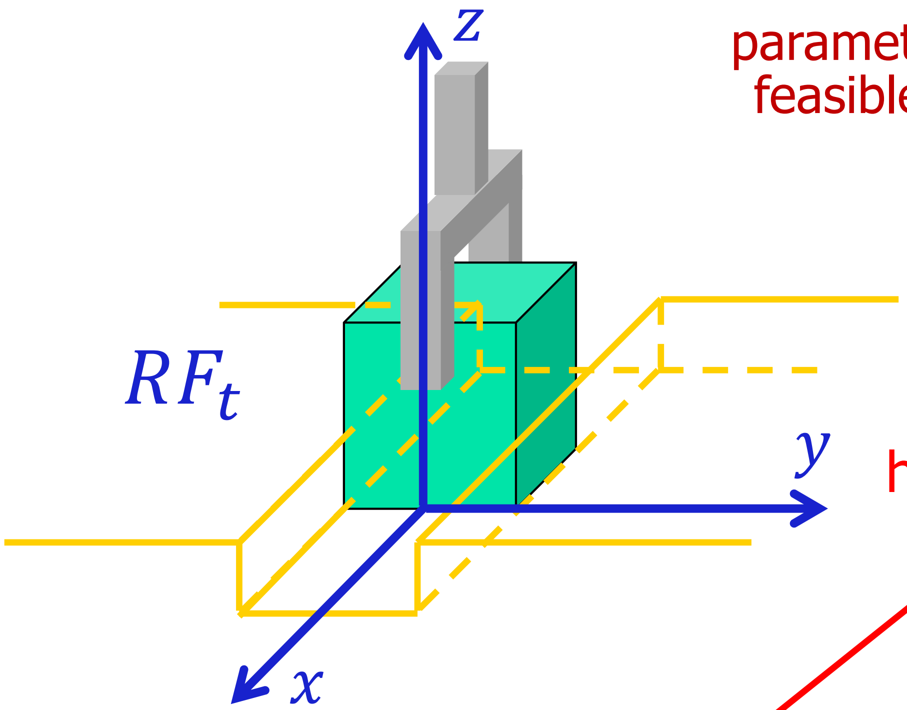
artificial constraints
(to be imposed by the control law)

$$6 - k = 4 \left\{ \begin{array}{l} f_y = f_{y,des}\ (= 0) \text{ (to avoid internal stress)} \\ \mu_x = \mu_{x,des}\ (= 0), \mu_z = \mu_{z,des} (= 0) \\ f_z = f_{z,des} \text{ (to keep contact)} \end{array}\right.$$

$$k = 2 \left\{ \begin{array}{l} \omega_y = \omega_{y,des} = 0 \text{ (to slide and not to roll !!)} \\ v_x = v_{x,des} \end{array}\right.$$

v = linear velocity
ω = angular velocity
f = force
μ = torque

Selection of directions - example 1



parametrization of feasible motions

$$\begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} v_x \\ \omega_y \end{pmatrix} = T \begin{pmatrix} v_x \\ \omega_y \end{pmatrix}$$

here, constant and unitary ("selection" of columns from the 6×6 identity matrix)

parametrization of feasible reactions

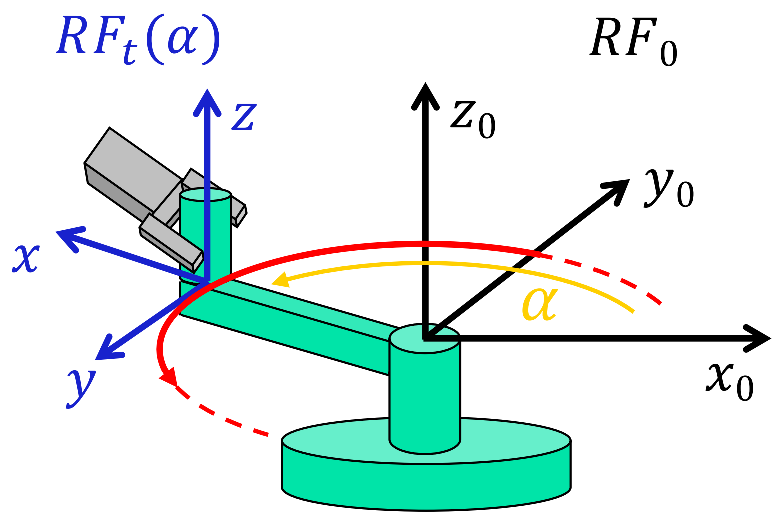
$$\begin{pmatrix} f \\ \mu \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_y \\ f_z \\ \mu_x \\ \mu_z \end{pmatrix} = Y \begin{pmatrix} f_y \\ f_z \\ \mu_x \\ \mu_z \end{pmatrix}$$

$$\boxed{T^T Y = 0}$$

reaction forces/torques do **not** perform work on feasible motions

$$\begin{pmatrix} f^T & \mu^T \end{pmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} = 0$$

Task frame and constraints - example 2



$$RF_0 \xrightarrow{R_z(\alpha)} RF_t(\alpha)$$

task: turning a crank (free handle)

natural constraints

$v_x = v_z = 0$

$\omega_x = \omega_y = 0$

$f_y = \mu_z = 0$

artificial constraints

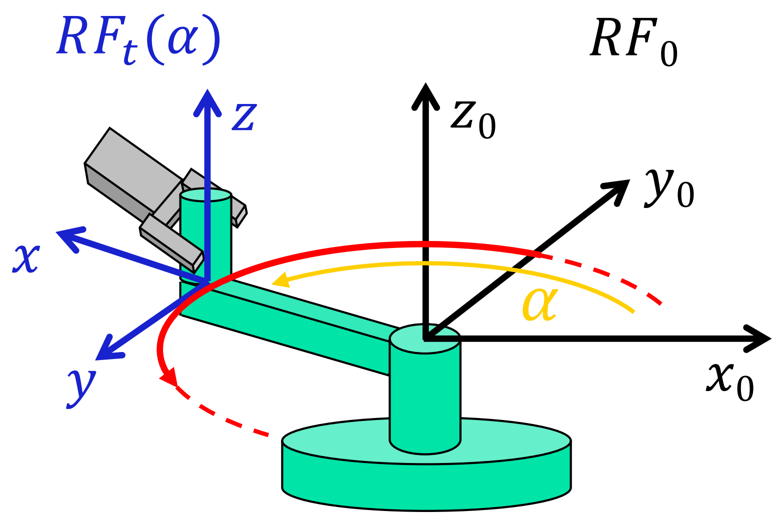
$f_x = f_{x,des}\ (= 0),\ f_z = f_{z,des}\ (= 0)$

$\mu_x = \mu_{x,des}\ (= 0),\ \mu_y = \mu_{y,des}\ (= 0)$

$v_y = v_{y,des}$ (the tangent speed of rotation)

$\omega_z = \omega_{z,des}$ ($=0$ if handle should not spin)

Selection of directions – example 2



parametrization of feasible motions

$$\begin{pmatrix} {}^0v \\ {}^0\omega \end{pmatrix} = \begin{pmatrix} R^T(\alpha) & 0 \\ 0 & R^T(\alpha) \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_y \\ \omega_z \end{pmatrix}$$

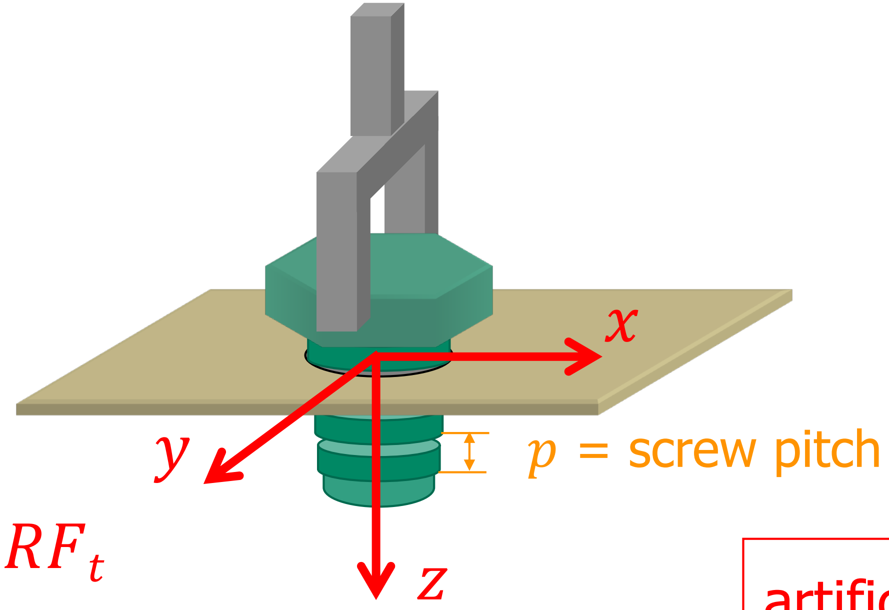
$$= T(\alpha) \begin{pmatrix} v_y \\ \omega_z \end{pmatrix}$$

$$\boxed{T^T(\alpha) Y(\alpha) = 0}$$

parametrization of feasible reactions

$$\begin{pmatrix} {}^0f \\ {}^0\mu \end{pmatrix} = \begin{pmatrix} R^T(\alpha) & 0 \\ 0 & R^T(\alpha) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} f_x \\ f_z \\ \mu_x \\ \mu_y \end{pmatrix} = Y(\alpha) \begin{pmatrix} f_x \\ f_z \\ \mu_x \\ \mu_y \end{pmatrix}$$

Task frame and constraints - example 3



task: insert a screw in a bolt

natural constraints (partial…)

$v_x = v_y = 0$

$\omega_x = \omega_y = 0$

artificial constraints (abundant…)

$f_x = f_{x,des} = 0, f_y = f_{y,des} = 0$

$\mu_x = \mu_{x,des} = 0, \mu_y = \mu_{y,des} = 0$

$v_z = v_{z,des}, \omega_z = \omega_{z,des} = (2\pi/p)v_{z,des}$

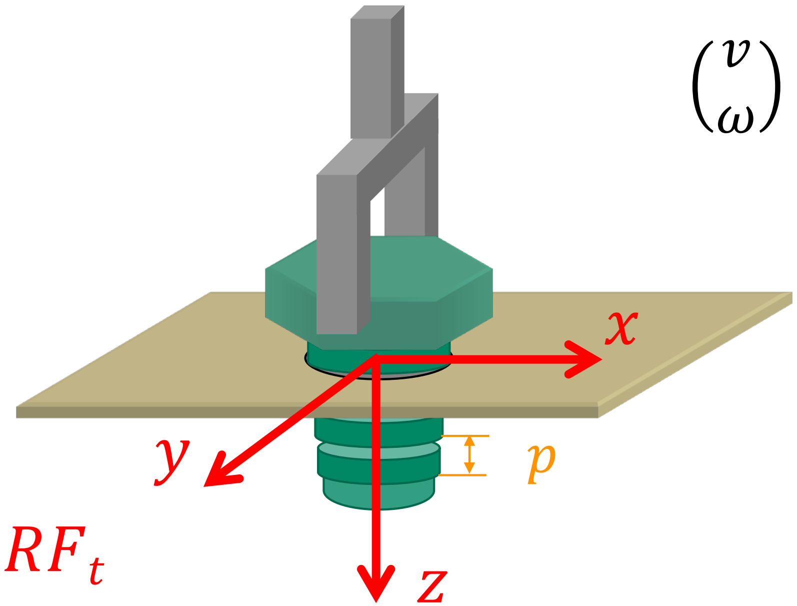
$f_z = f_{z,des}, \mu_z = \mu_{z,des}$ (one function of the other!)

the screw proceeds along and around the z-axis, but **not** in an independent way! (1 dof)

accordingly, f_z and μ_z **cannot** be independent

wrench (force/torque) direction should be orthogonal to motion twist!

Selection of directions – example 3



$$\begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & \dfrac{2\pi}{p} \end{pmatrix}^T v_z = T v_z \quad (k = 1)$$

or $\boxed{\omega_z = 2\pi \dfrac{v_z}{p}}$

Y: such that $T^T Y = 0$

$$\boxed{f_z = -\frac{2\pi}{p}\mu_z}$$

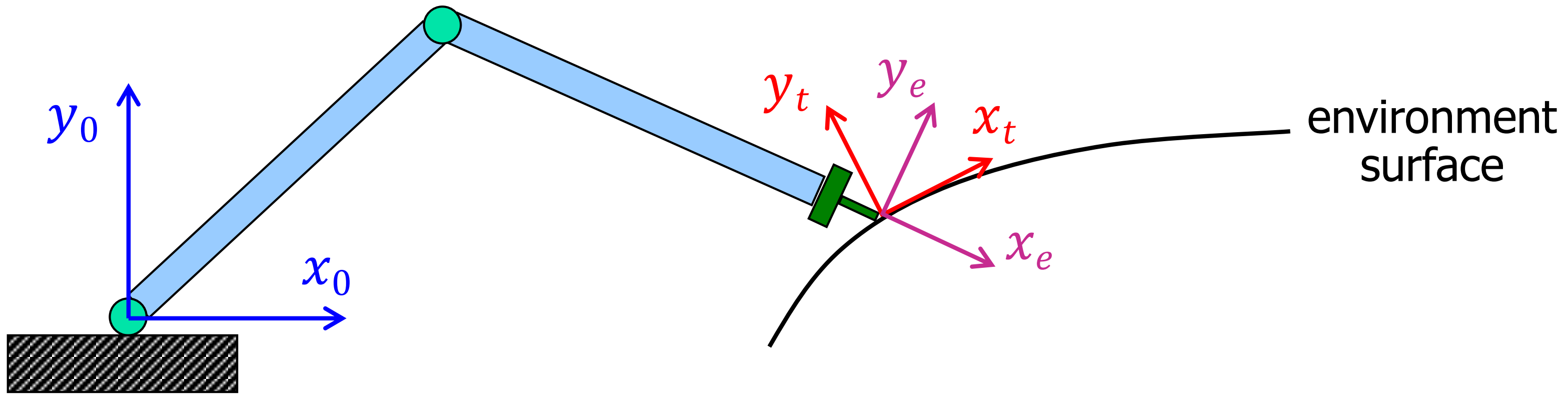
$(6 - k = 5)$

$$\begin{pmatrix} f \\ \mu \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2\pi/p \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_x \\ f_y \\ \mu_x \\ \mu_y \\ \mu_z \end{pmatrix} = Y \begin{pmatrix} f_x \\ f_y \\ \mu_x \\ \mu_y \\ \mu_z \end{pmatrix}$$

the columns of T and Y
do not necessarily coincide
with selected columns
of the 6×6 identity matrix
⇒ generalized (screw) directions

Frames of interest – example 4

planar motion of a 2R robot ($n = 2$) in pointwise contact with a surface (task dimension $m = 2$)



- task frame $\boldsymbol{RF_t}$ used for an independent definition of the hybrid reference values (here: ${}^t v_{x,des}$ [$k = 1$] and ${}^t f_{y,des}$ [$m - k = 1$]) and for computing the errors that drive the feedback control law
- sensor frame $\boldsymbol{RF_e}$ (here: RF_2) where the force ${}^e f = ({}^e f_x, {}^e f_y)$ is measured
- base frame $\boldsymbol{RF_0}$ in which the end-effector velocity is expressed (here: ${}^0 v = \left({}^0 v_x, {}^0 v_y\right)$ of O_2), computed using robot Jacobian and joint velocities

all quantities (and errors!) should be expressed ("rotated") in the same reference frame ⇒ the task frame!

General parametrization of hybrid tasks

a "description" of robot-environment contact type: it implicitly defines the task frame

$$\begin{cases} \begin{pmatrix} v \\ \omega \end{pmatrix} = T(s)\dot{s} & \boxed{s \in \mathbb{R}^k} \\ \begin{pmatrix} f \\ \mu \end{pmatrix} = Y(s)\lambda & \boxed{\lambda \in \mathbb{R}^{m-k}} \end{cases}$$

parametrizes robot E-E free motion

parametrizes reaction forces/torques

reaction forces/torques do not perform work on E-E displacements

$$\boxed{T^T(s)Y(s) = 0}$$

axes directions of task frame depend in general on s (i.e., on robot E-E pose in the environment)

in general, it is $m = 6$
(as in most of the previous examples)

$+$

robot dynamics

$$\boxed{M(q)\ddot{q} + S(q,\dot{q})\dot{q} + g(q) = u + J^T(q)\begin{pmatrix} f \\ \mu \end{pmatrix}}$$

robot kinematics

$$\boxed{\begin{pmatrix} v \\ \omega \end{pmatrix} = J(q)\dot{q}}$$

Hybrid force/velocity control

- control objective: to impose desired task evolutions to the parameters s of motion and to the parameters λ of force

$$s \longrightarrow s_d(t) \qquad \lambda \longrightarrow \lambda_d(t)$$

- the control law is designed in two steps
  1. exact linearization and decoupling in the task frame by feedback

     closed-loop model $\rightarrow$ $\begin{pmatrix} \ddot{s} \\ \lambda \end{pmatrix} = \begin{pmatrix} a_s \\ a_\lambda \end{pmatrix}$

  2. (linear) design of a_s and a_λ so as to impose the desired dynamic behavior to the errors $e_s = s_d - s$ and $e_\lambda = \lambda_d - \lambda$

- assumptions: $n = m$ ($= 6$ usually), $J(q)$ out of singularity

Note: in "simple" cases, $\dot{s}$ and λ drive single components of v or ω and of f or μ; accordingly, T and Y are just columns of 0/1 selection matrices

Feedback linearization in task space

$$J(q)\dot{q} = \begin{pmatrix} v \\ \omega \end{pmatrix} = T(s)\dot{s} \;\Rightarrow\; J\ddot{q} + \dot{J}\dot{q} = T\ddot{s} + \dot{T}\dot{s} \;\Rightarrow\; \ddot{q} = J^{-1}(T\ddot{s} + \dot{T}\dot{s} - \dot{J}\dot{q})$$

robot side → ← task side

$$M(q)\ddot{q} + S(q,\dot{q})\dot{q} + g(q) = u + J^T(q)\begin{pmatrix} f \\ \mu \end{pmatrix} = u + J^T(q)Y(s)\lambda$$

nonsingular $n \times n$ matrix (under the assumption that $\boldsymbol{J^T Y}$ has full (column) rank)

$$\left(M(q)J^{-1}(q)T(s) \;\vdots\; -J^T(q)Y(s)\right)\begin{pmatrix} \ddot{s} \\ \lambda \end{pmatrix}$$

$$+ M(q)J^{-1}(q)\left(\dot{T}(s)\dot{s} - \dot{J}(q)\dot{q}\right) + S(q,\dot{q})\dot{q} + g(q) = u$$

$$u = \left(MJ^{-1}T \;\vdots\; -J^T Y\right)\begin{pmatrix} a_s \\ a_\lambda \end{pmatrix} + MJ^{-1}\left(\dot{T}\dot{s} - \dot{J}\dot{q}\right) + S\dot{q} + g$$

linearizing and decoupling control law

$$\Rightarrow \begin{pmatrix} \ddot{s} \\ \lambda \end{pmatrix} = \begin{pmatrix} a_s \\ a_\lambda \end{pmatrix} \begin{matrix} \}\, k \\ \}\, m-k \end{matrix}$$

s has "relative degree" = 2

λ has "relative degree" = 0

Stabilization with a_s and a_λ

as usual, it is sufficient to apply linear control techniques for the exponential stabilization of tracking errors (on each single, input-output decoupled channel)

$$\boxed{a_s = \ddot{s}_d + K_D(\dot{s}_d - \dot{s}) + K_P(s_d - s)}$$

$K_P, K_D > 0$ and diagonal

$$\Rightarrow \quad \ddot{e}_s + K_D \dot{e}_s + K_P e_s = 0 \quad \Rightarrow \quad e_s = s_d - s \longrightarrow 0$$

$K_I \geq 0$ diagonal

$$\boxed{a_\lambda = \lambda_d + K_I \int (\lambda_d - \lambda) dt}$$

$a_\lambda = \lambda_d$ would be enough, but adding an integral with the **force error** gives more robustness to (constant) disturbances

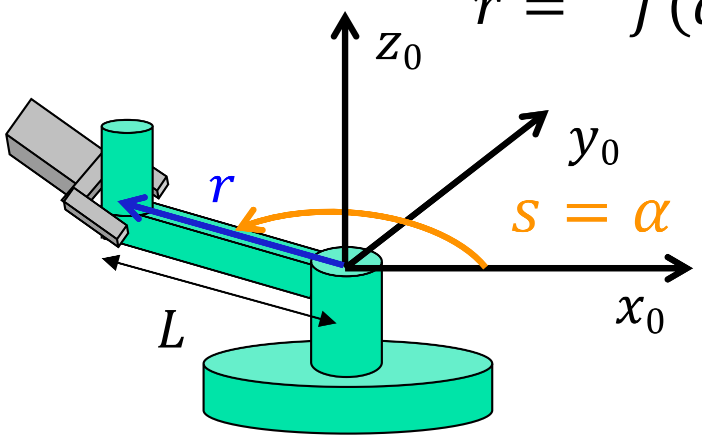
$$\Rightarrow \quad e_\lambda + K_I \int e_\lambda \, dt = 0 \quad \Rightarrow \quad e_\lambda = \lambda_d - \lambda \longrightarrow 0$$

we need "values" for s, $\dot{s}$ and λ to be extracted from actual measurements !

"Filtering" position and force measures

- s and $\dot{s}$ are obtained from measures of q and $\dot{q}$, equating the descriptions of the end-effector pose and velocity "from the robot side" (direct and differential kinematics) and "from task/environment side" (function of $s, \dot{s}$)

example



$$ {}^0r = {}^0f(q) = \begin{pmatrix} L\cos s \\ L\sin s \\ 0 \end{pmatrix} \Rightarrow s = \operatorname{atan2}\{ {}^0f_y(q),\ {}^0f_x(q)\} $$

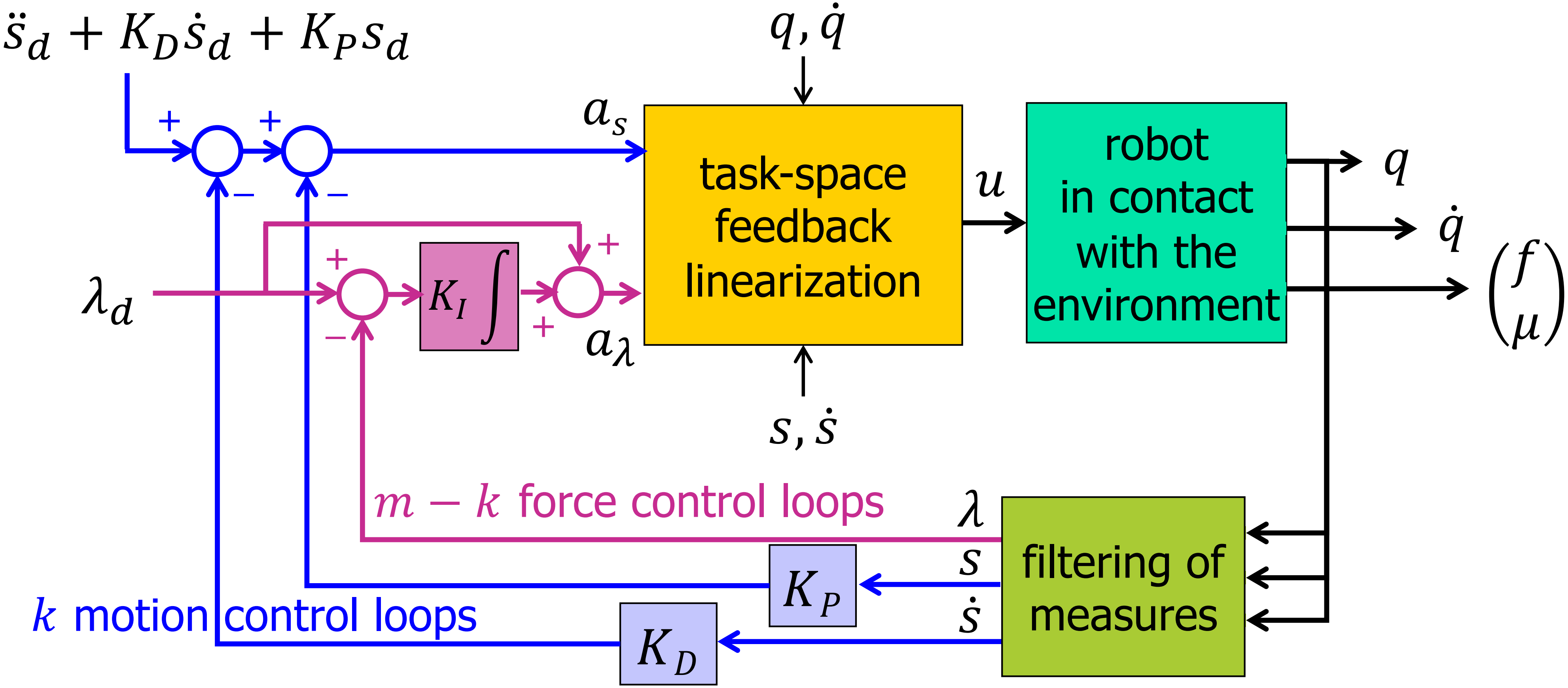
$$ J(q)\dot{q} = T(s)\dot{s} \Rightarrow \dot{s} = T^{\#}(s)J(q)\dot{q} $$

- λ is obtained from force/torque measures at end-effector

$$ \begin{pmatrix} f \\ m \end{pmatrix} = Y(s)\lambda \Rightarrow \lambda = Y^{\#}(s)\begin{pmatrix} f \\ m \end{pmatrix} $$

pseudoinverses of "tall" matrices having full column rank, e.g., $T^{\#} = (T^TT)^{-1}T^T$ (or weighted)

Block diagram of hybrid control



usually $m = 6$ (complete 3D space)

limit cases $k = m$: no force control loops, only motion (free motion)

$k = 0$: no motion control loops, only force (“frozen” robot end-effector)