Robotics 1

Inverse differential kinematics
Statics and force transformations

Prof. Alessandro De Luca

Dipartimento di Ingegneria Informatica
Automatica e Gestionale Antonio Ruberti

Sapienza
Università di Roma

Inversion of differential kinematics

- find the joint velocity vector that realizes a desired task/ end-effector velocity ("generalized" = linear and/or angular)

generalized velocity

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

J square and non-singular at q →

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

- problems
  - near a singularity of the Jacobian matrix (too high $\dot{q}$)
  - for redundant robots (no standard "inverse" of a rectangular matrix)

in these cases, more robust inversion methods are needed

Incremental solution

to inverse kinematics problems

- joint velocity inversion can be used also to solve on-line and incrementally a "sequence" of inverse kinematics problems
- each problem differs by a small amount dr from previous one

$$r = f_r(q)$$

direct kinematics

$$dr = \frac{\partial f_r(q)}{\partial q} dq = J_r(q) dq$$

differential kinematics
(here with a square, analytic Jacobian)

current next

$$r \longrightarrow r + dr$$

$$r + dr = f_r(q) \quad \Rightarrow \quad q = f_r^{-1}(r + dr)$$

first, increment the desired task variables

then, solve the inverse kinematics problem

(possibly, with a numerical method from the current configuration)

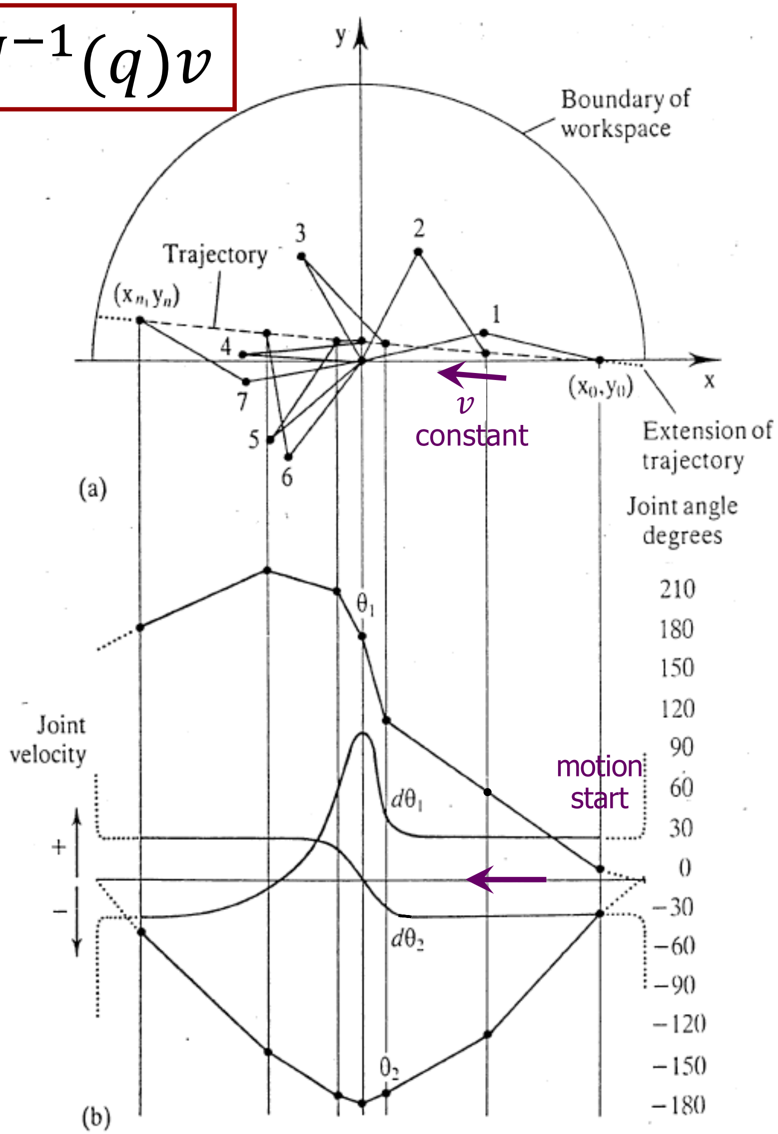
$$dq = J_r^{-1}(q) dr \quad \Rightarrow \quad q \longrightarrow q + dq$$

first, solve the inverse differential kinematics problem

then, increment the original joint variables

Behavior near a singularity

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



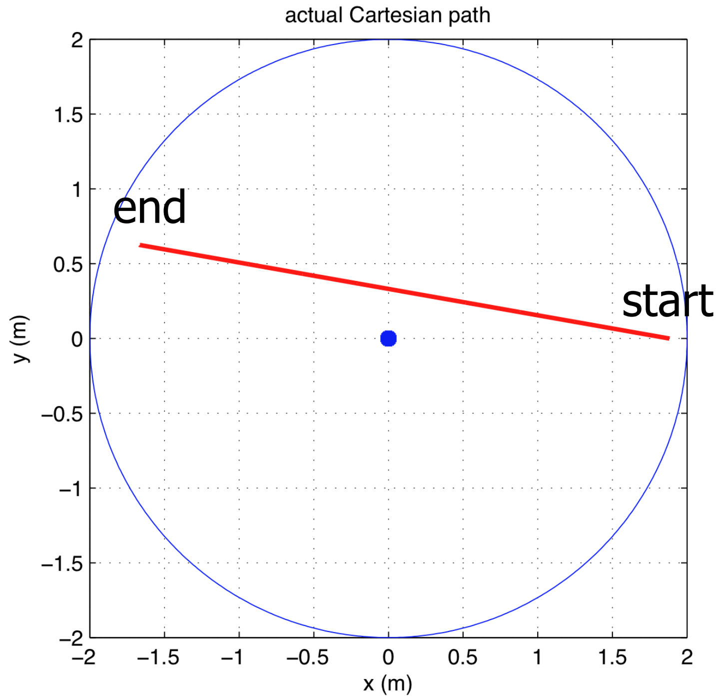
- problems arise only when commanding joint motion by inversion of a given Cartesian motion task
- here, a linear Cartesian trajectory for a planar 2R robot
- there is a sudden increase of the displacement/velocity of the first joint near $\theta_2 = -\pi$ (end-effector close to the origin), despite the required Cartesian displacement is small

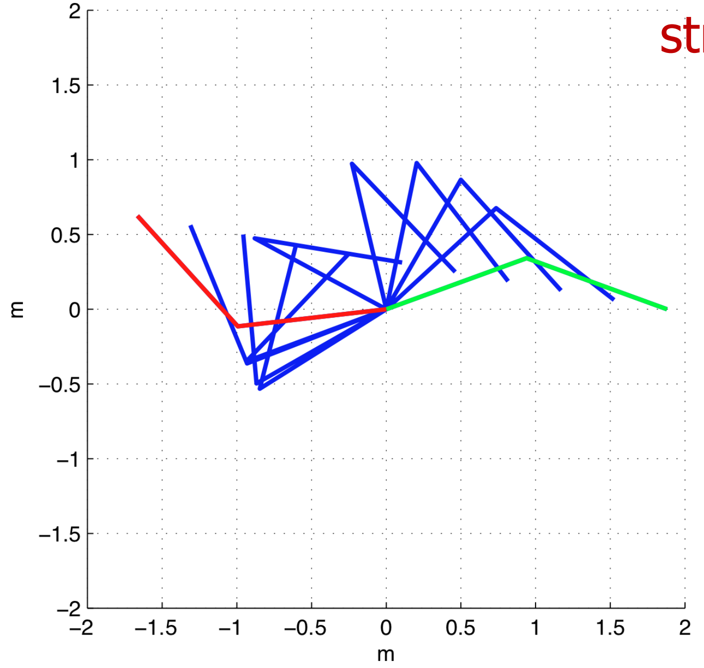
Simulation results

planar 2R robot in straight line Cartesian motion

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

regular case





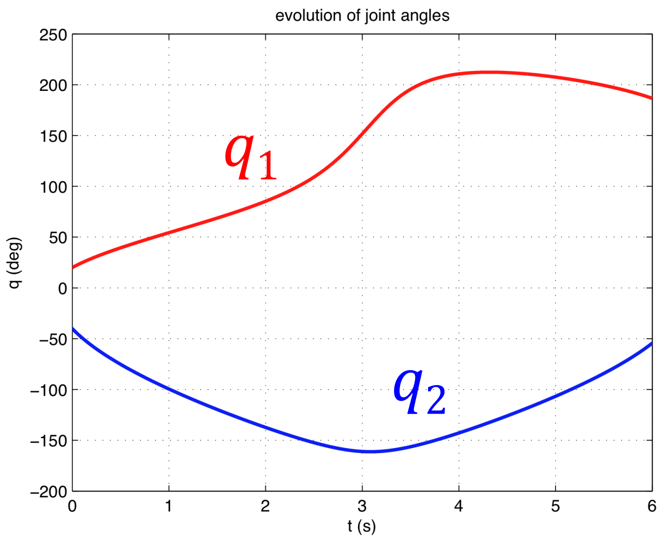
stroboscopic view

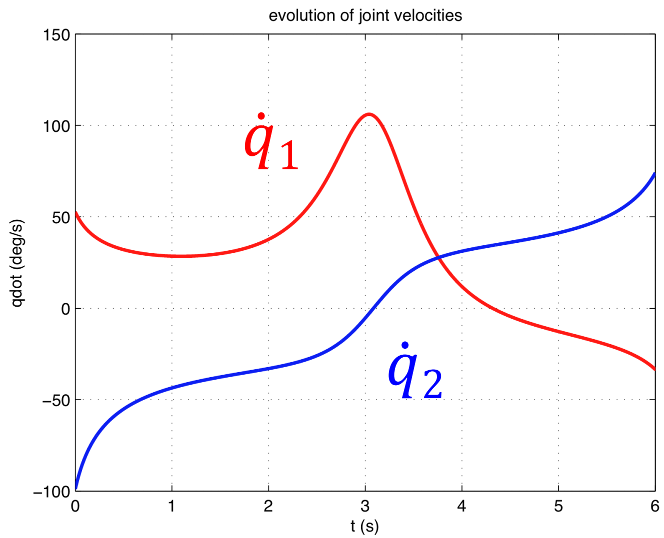
a line from right to left, at $\alpha = 170°$ angle with x-axis, executed at constant speed $v = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

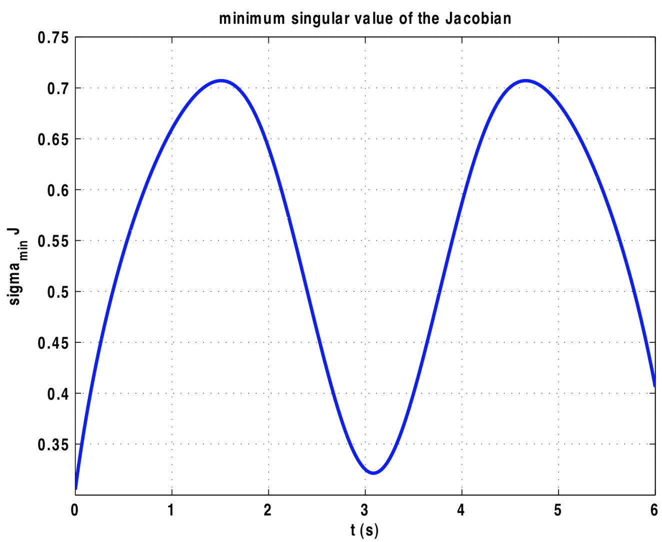
path at
$\alpha = 170°$

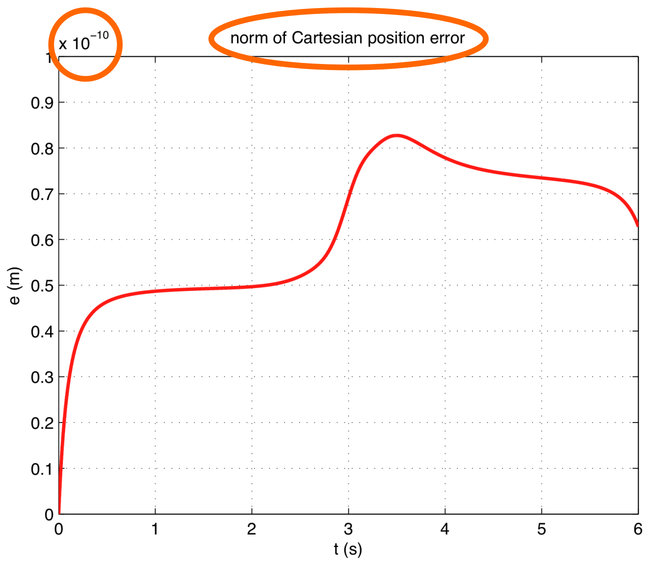




regular
case

distance to singularity by the minimum singular value $\sigma_{min}\ (= \sigma_2) > 0$ of Jacobian J





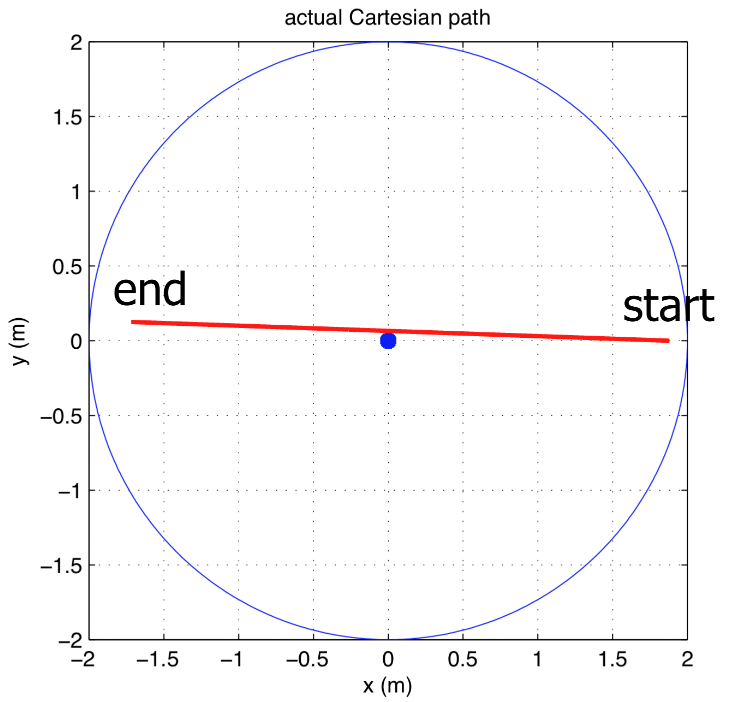
error due only to numerical integration (10^{-10})

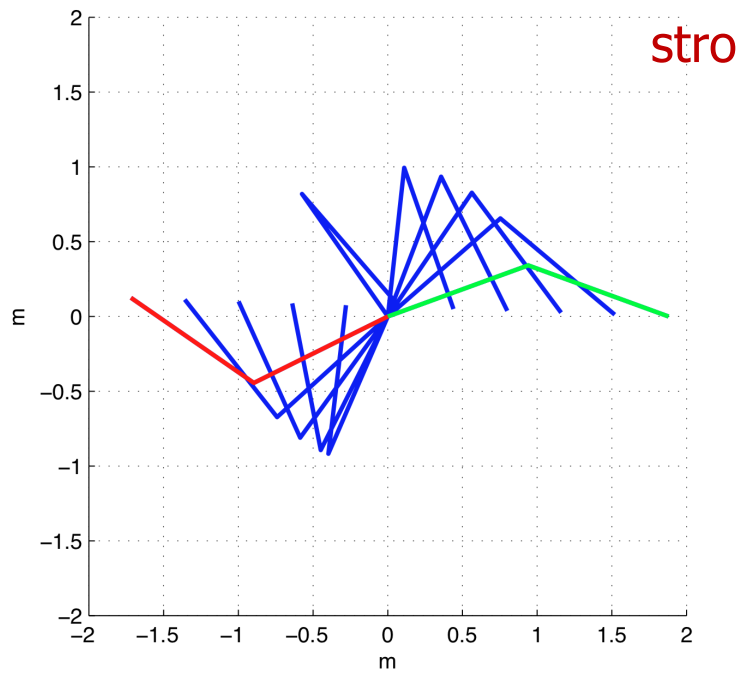
Simulation results

planar 2R robot in straight line Cartesian motion

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

close to singular case





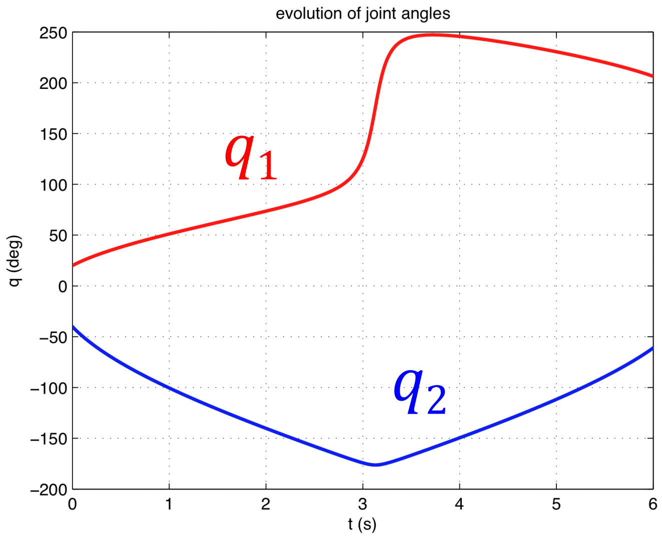
stroboscopic view

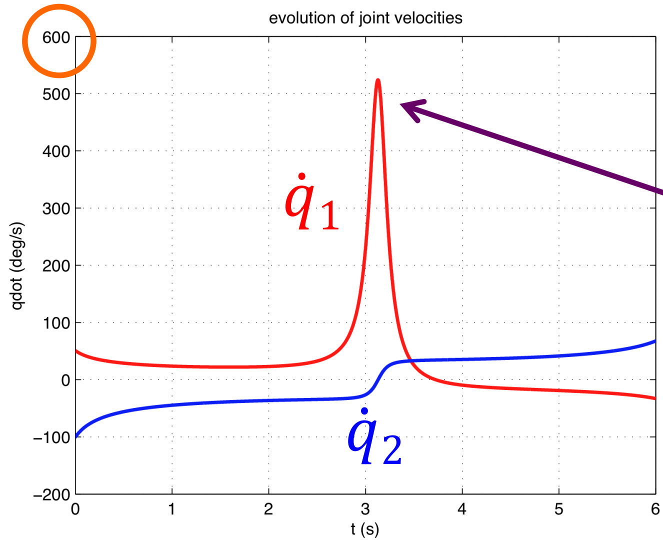
a line from right to left, at $\alpha = 178°$ angle with x-axis, executed at constant speed $v = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

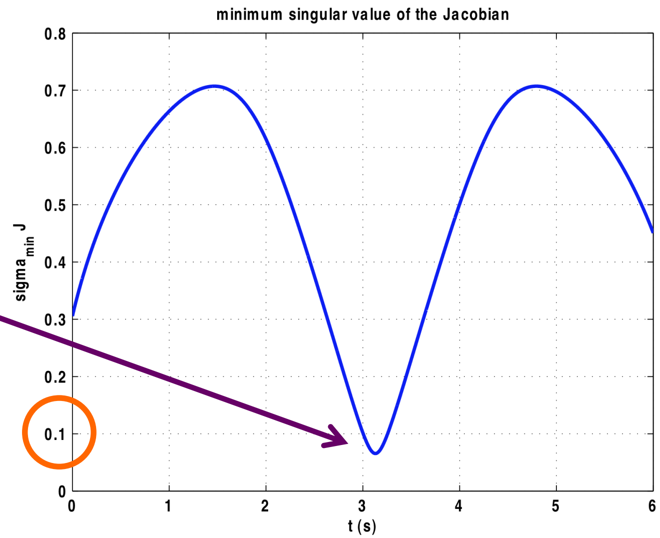
path at $\alpha = 178°$

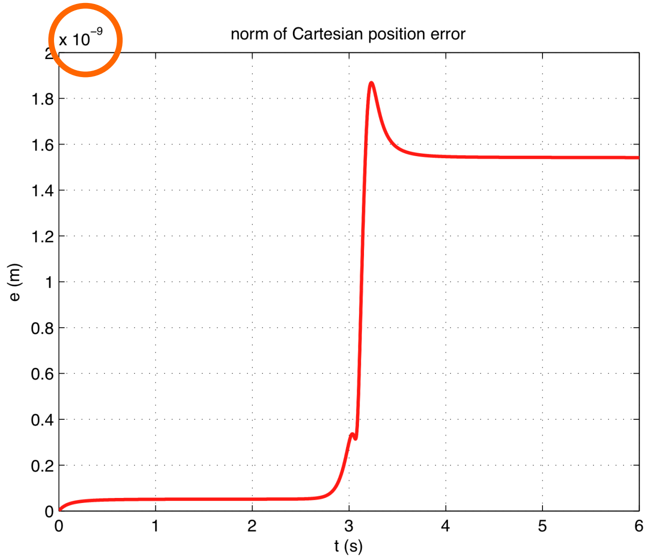




large peak of $\dot{q}_1$

close to singular case





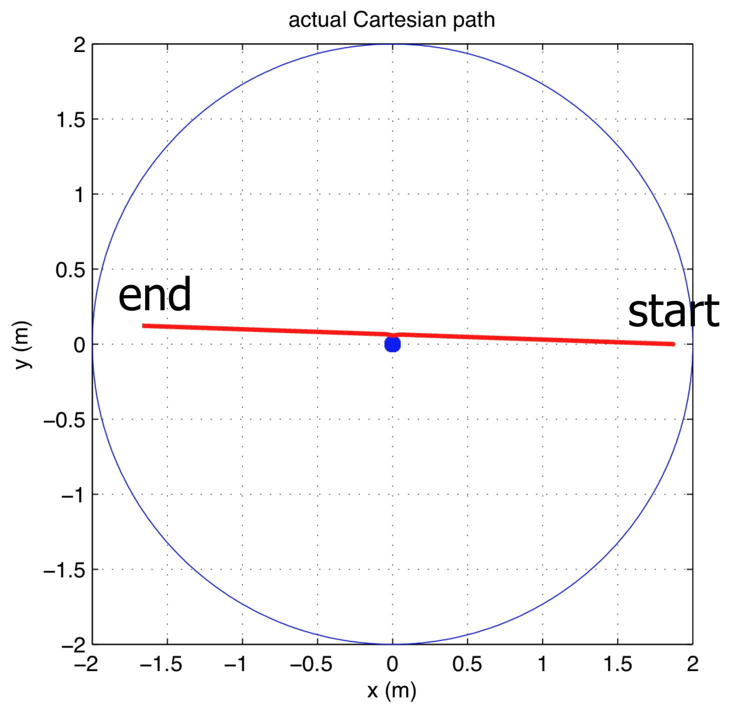
still very small, but increased numerical integration error ($2 \cdot 10^{-9}$)

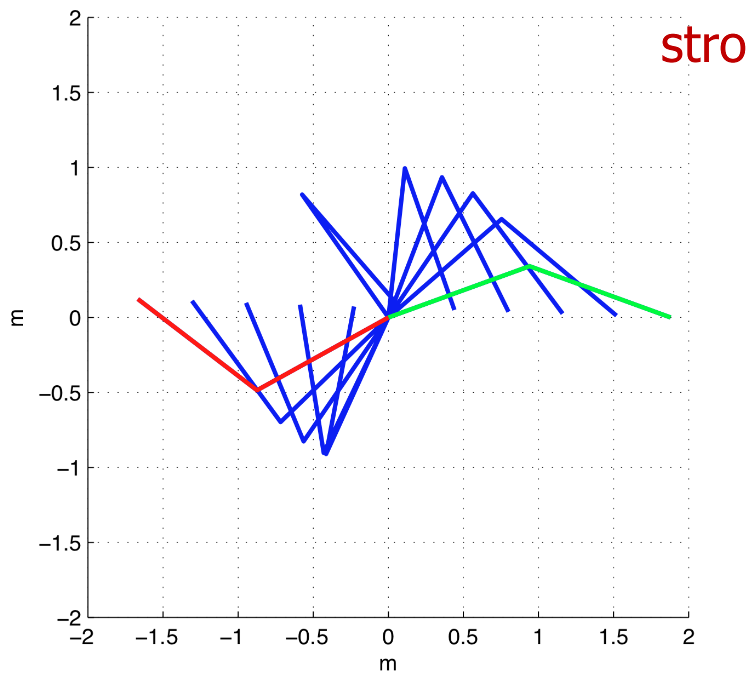
Simulation results

planar 2R robot in straight line Cartesian motion

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

close to singular case
with joint velocity saturation at $V_i = 300°/s$





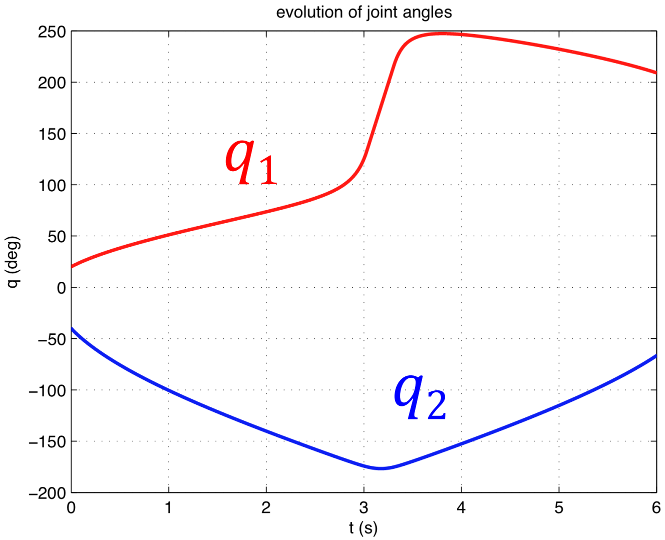
stroboscopic view

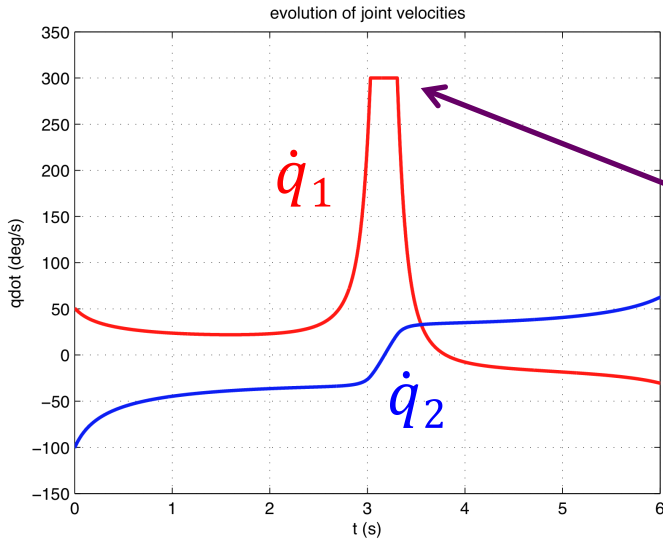
a line from right to left, at $\alpha = 178°$ angle with x-axis,
executed at constant speed $v = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

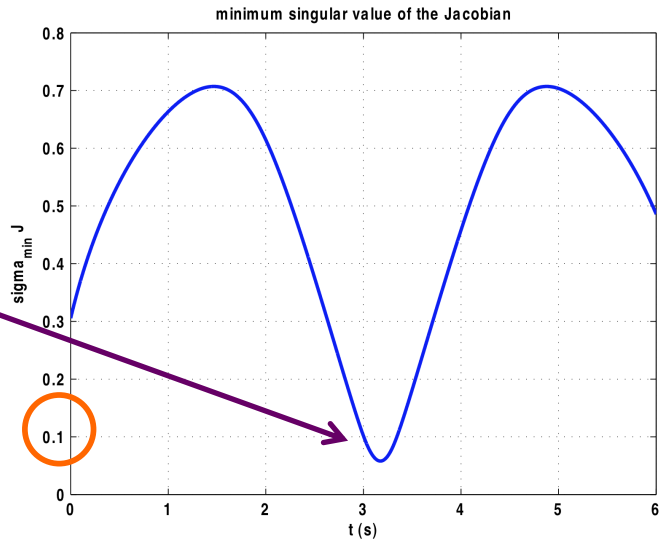
path at
$\alpha = 178°$

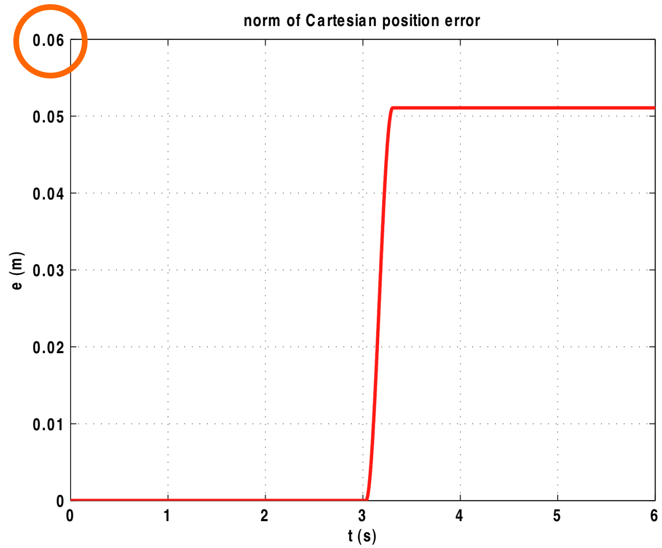




saturated value of $\dot{q}_1$

close to singular case





actual position error!! (6 cm)

to be recovered using an error feedback control action!

Damped Least Squares method

$$\min_{\dot{q}} H = \frac{\lambda}{2}\|\dot{q}\|^2 + \frac{1}{2}\|J\dot{q} - v\|^2, \qquad \lambda \geq 0$$

prove it!

prove it!

$$\dot{q} = (\lambda I_n + J^T J)^{-1} J^T v = J^T (\lambda I_m + J\,J^T)^{-1} v = J_{DLS}\, v$$

two equivalent expressions, but the second is more convenient in redundant robots!

- inversion of differential kinematics as unconstrained optimization problem
- function $H =$ weighted sum of two objectives (norm of joint velocity and error norm on achieved end-effector velocity) to be minimized
- J_{DLS} can be used for both cases: $m = n$ (square) and $m < n$ (redundant)
- $\lambda = 0$ when "far enough" from singularities: $J_{DLS} = J^T (J\,J^T)^{-1} = J^{-1}$ or $J^{\#}$
- with $\lambda > 0$, there is a (vector) error ϵ $(= v - J\dot{q})$ in executing the desired end-effector velocity v (check that $\epsilon = \lambda(\lambda I_m + J\,J^T)^{-1} v$), but the joint velocities are always reduced ("damped")

Simulation results

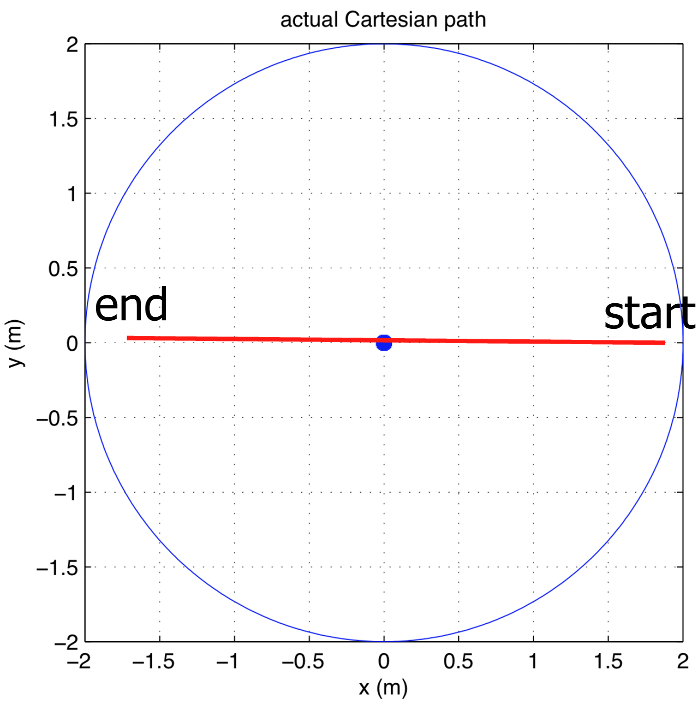
planar 2R robot in straight line Cartesian motion

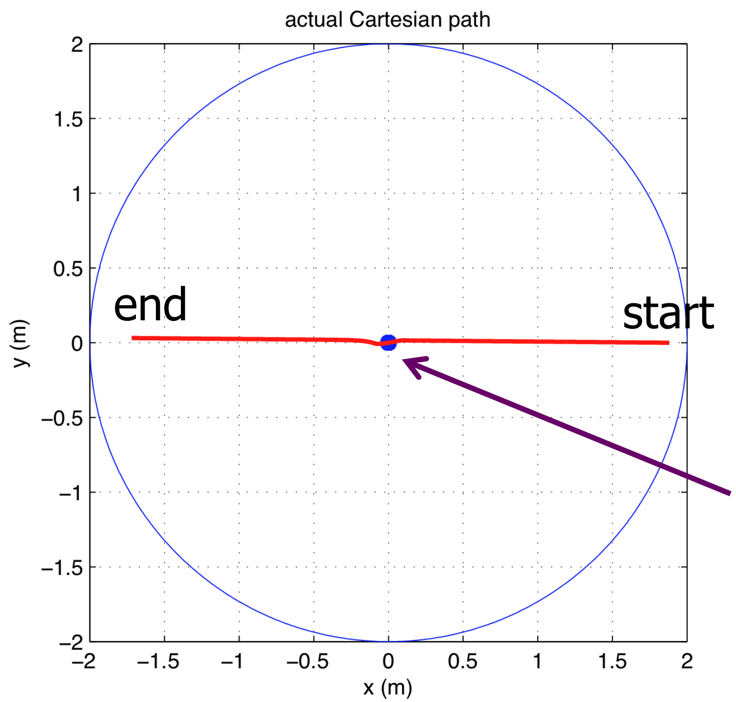
a comparison of inverse and damped inverse Jacobian methods

even closer to singular case

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

$$\dot{q} = J_{DLS}(q)v$$





some position error ...

a line from right to left, at $\alpha = 179.5°$ angle with x-axis, executed at constant speed $v = 0.6$ m/s for $T = 6$ s

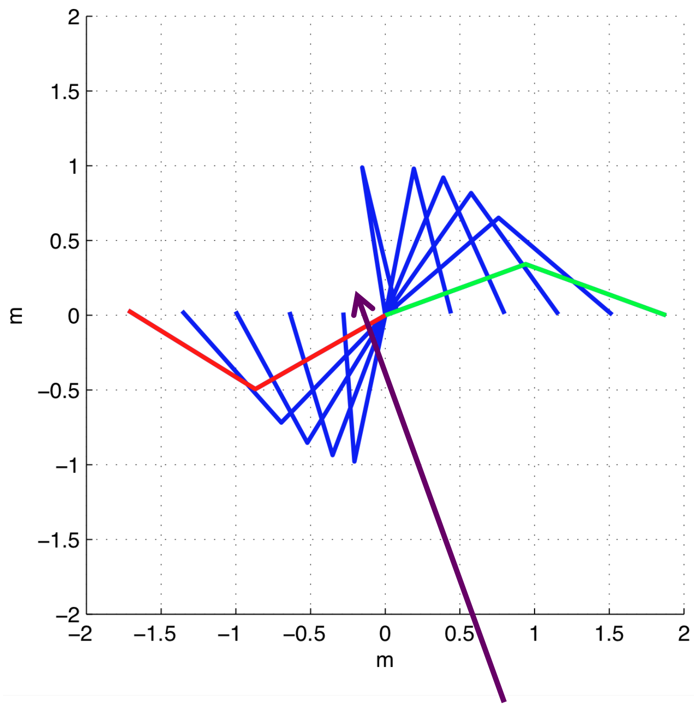
Simulation results

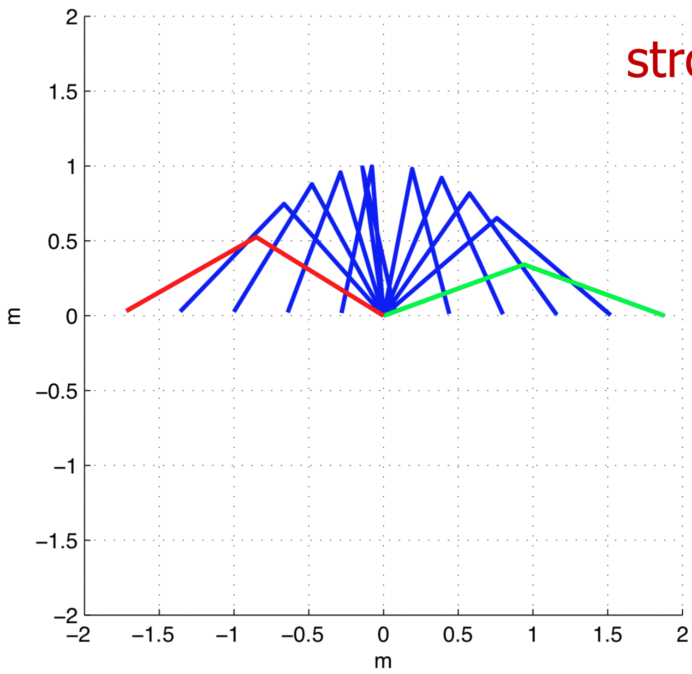
planar 2R robot in straight line Cartesian motion

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

path at
$\alpha = 179.5°$

$$\dot{q} = J_{DLS}(q)v$$





stroboscopic views

here, a very fast reconfiguration of first joint ...

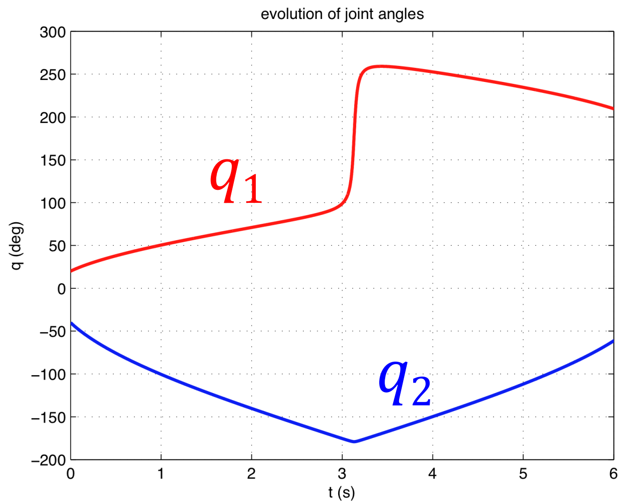
a completely different inverse solution, around/after crossing the region close to the folded singularity

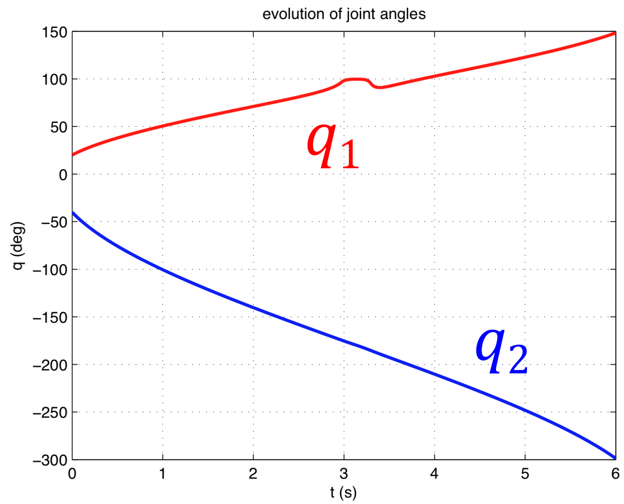
Simulation results

planar 2R robot in straight line Cartesian motion

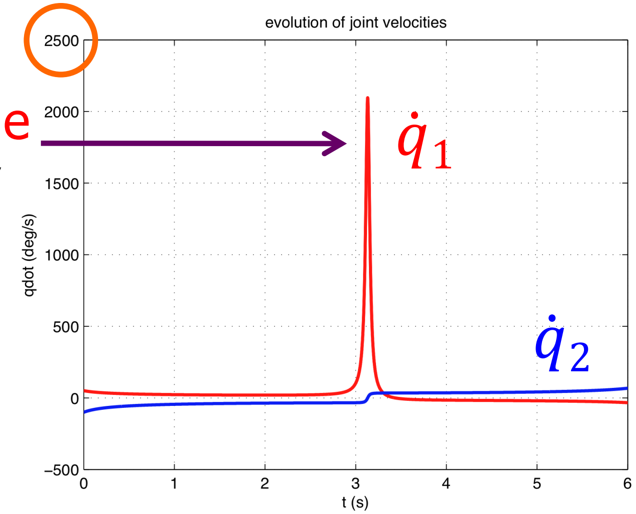
$$\dot{q} = J^{-1}(q)v$$

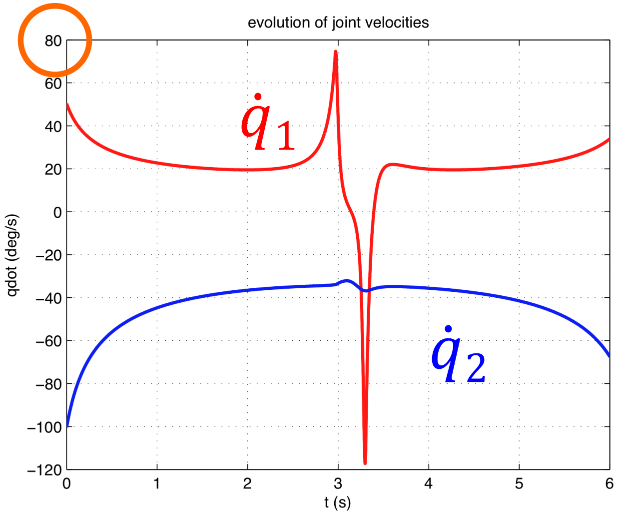
$$\dot{q} = J_{DLS}(q)v$$





extremely large peak velocity of first joint!!



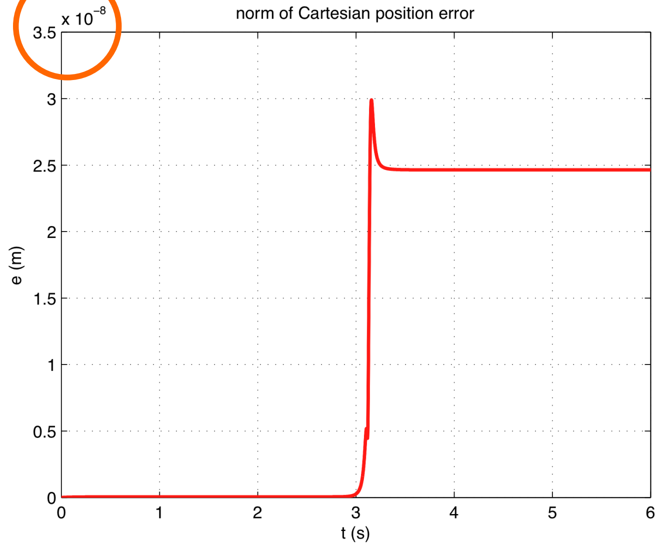


smoother joint motion with limited joint velocities!

Simulation results

planar 2R robot in straight line Cartesian motion

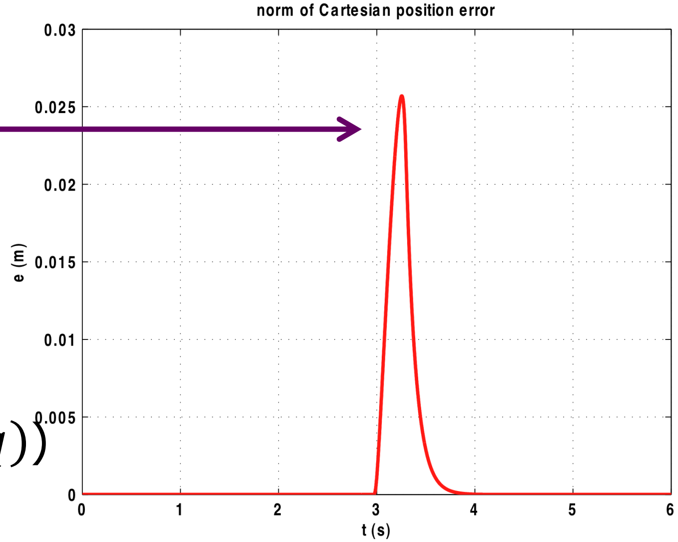
$\dot{q} = J^{-1}(q)v$



increased numerical integration error $(3 \cdot 10^{-8})$

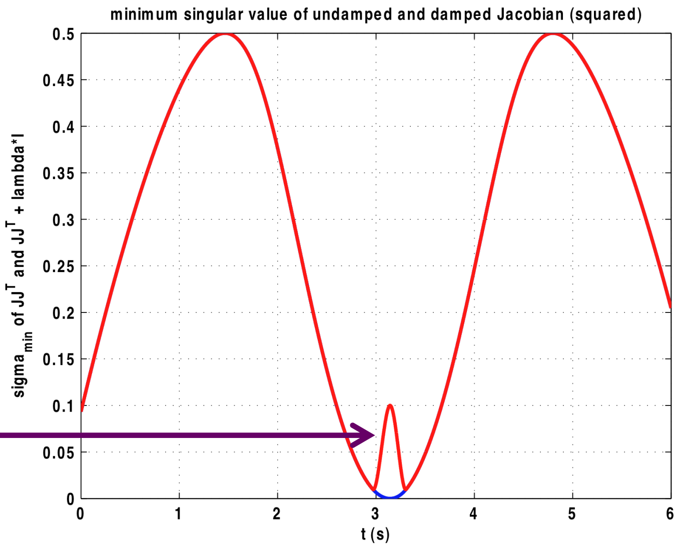
$\dot{q} = J_{DLS}(q)v$

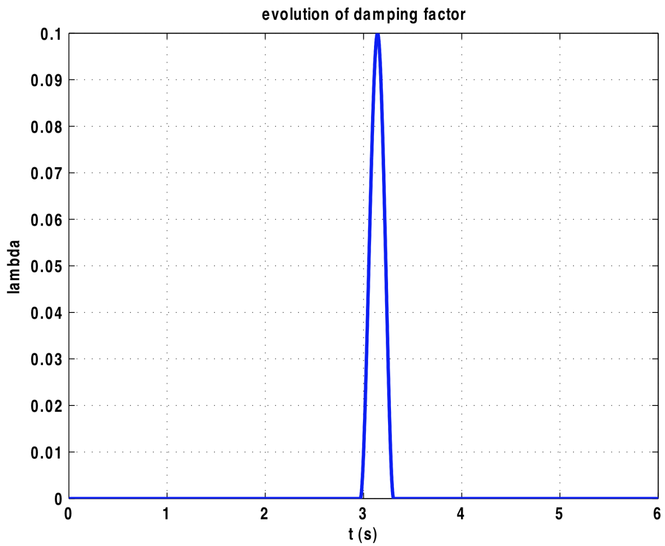
error (25 mm) when crossing the singularity, later recovered by a feedback action ($v \Rightarrow v + K_p e_p$ with $e_p = p_d - p(q)$)



minimum singular value of JJ^T and $\lambda I + JJ^T$

they differ only when damping factor is non-zero





damping factor λ is chosen non-zero only close to singularity!

Pseudoinverse method

a constrained optimization (minimum norm) problem

$$\min_{\dot q} H = \frac{1}{2}\|\dot q\|^2 \quad \text{such that} \quad J\dot q = v$$

$\Leftrightarrow$

$$\min_{\dot q \in S} H = \frac{1}{2}\|\dot q\|^2 \qquad S = \left\{ \begin{array}{c} \dot q \in R^n : \\ \|J\dot q - v\| \text{ is minimum} \end{array} \right\}$$

solution $\quad \dot q = J^{\#} v$

pseudoinverse of J

- if $v \in \mathcal{R}(J)$, the differential constraint is satisfied (v is feasible)
- else, $J\dot q = J\,J^{\#} v = v^{\perp}$, where $v^{\perp}$ minimizes the error $\|J\dot q - v\|$

orthogonal projection of v on $\mathcal{R}(J)$

Definition of the pseudoinverse

given J, is the unique matrix $J^{\#}$ satisfying the four relationships

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

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

- explicit expressions for full rank cases
  - if $\rho(J) = m = n$: $J^{\#} = J^{-1}$
  - if $\rho(J) = m < n$: $J^{\#} = J^T(J\,J^T)^{-1}$
  - if $\rho(J) = n < m$: $J^{\#} = (J^T J)^{-1}\,J^T$
- $J^{\#}$ always exists and is computed in general numerically using the SVD = Singular Value Decomposition of J
  - e.g., with the MATLAB function **pinv** (which uses in turn **svd**)

Numerical example

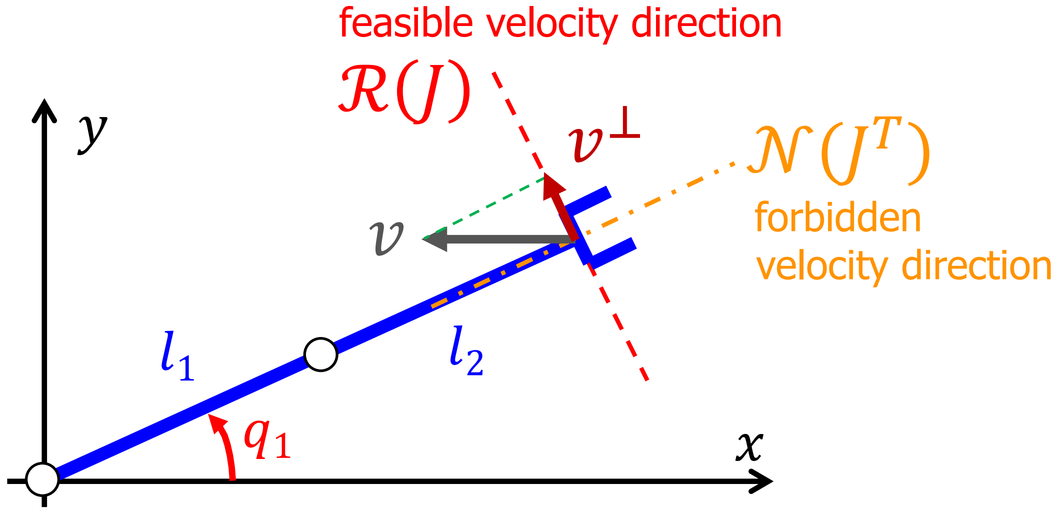
Jacobian of 2R robot with $l_1 = l_2 = 1$ at $q_2 = 0$ (rank $\rho(J) = 1$)

$$J = \begin{pmatrix} -2s_1 & -s_1 \\ 2c_1 & c_1 \end{pmatrix}$$

$$J^\# = \frac{1}{5}\begin{pmatrix} -2s_1 & 2c_1 \\ -s_1 & c_1 \end{pmatrix}$$

$$J\,J^\# = \begin{pmatrix} s_1^2 & -s_1c_1 \\ -s_1c_1 & c_1^2 \end{pmatrix} \qquad J^\# J = \begin{pmatrix} 0.8 & 0.4 \\ 0.4 & 0.2 \end{pmatrix}$$

both symmetric …



$\dot{q} = J^\# v$ is the minimum norm joint velocity vector that realizes exactly $v^\perp$

- at $q_1 = \pi/6$: for $v = \begin{pmatrix} -0.5 \\ 0 \end{pmatrix}$ [m/s], $\dot{q} = J^\# v = \begin{pmatrix} 0.1 \\ 0.05 \end{pmatrix}$ [rad/s] $\Rightarrow v^\perp = J\,J^\# v = \begin{pmatrix} -1/8 \\ \sqrt{3}/8 \end{pmatrix}$ [m/s]
- at $q_1 = \pi/2$: $J = \begin{pmatrix} -2 & 1 \\ 0 & 0 \end{pmatrix} \Rightarrow J^\# = \begin{pmatrix} -0.4 & 0 \\ -0.2 & 0 \end{pmatrix}$; now the same $v \in \mathcal{R}(J)$, $\dot{q} = \begin{pmatrix} 0.2 \\ 0.1 \end{pmatrix} \Rightarrow v^\perp = v$ (no error!)

General solution for $m < n$

ALL solutions of the inverse differential kinematics problem can be written as

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

any joint velocity...

projection matrix in the null space $\mathcal{N}(J)$

this is the solution of a slightly modified constrained optimization problem ("biased" toward the joint velocity ξ, chosen to avoid obstacles, joint limits, etc.)

$$\min_{\dot{q}} H = \frac{1}{2} \|\dot{q} - \xi\|^2 \text{ such that } J\dot{q} = v \quad \Leftrightarrow \quad \min_{\dot{q} \in S} H = \frac{1}{2} \|\dot{q} - \xi\|^2$$

$$S = \left\{ \begin{matrix} \dot{q} \in R^n : \\ \|J\dot{q} - v\| \text{ is minimum} \end{matrix} \right\}$$

verification of the actual task velocity that is being obtained

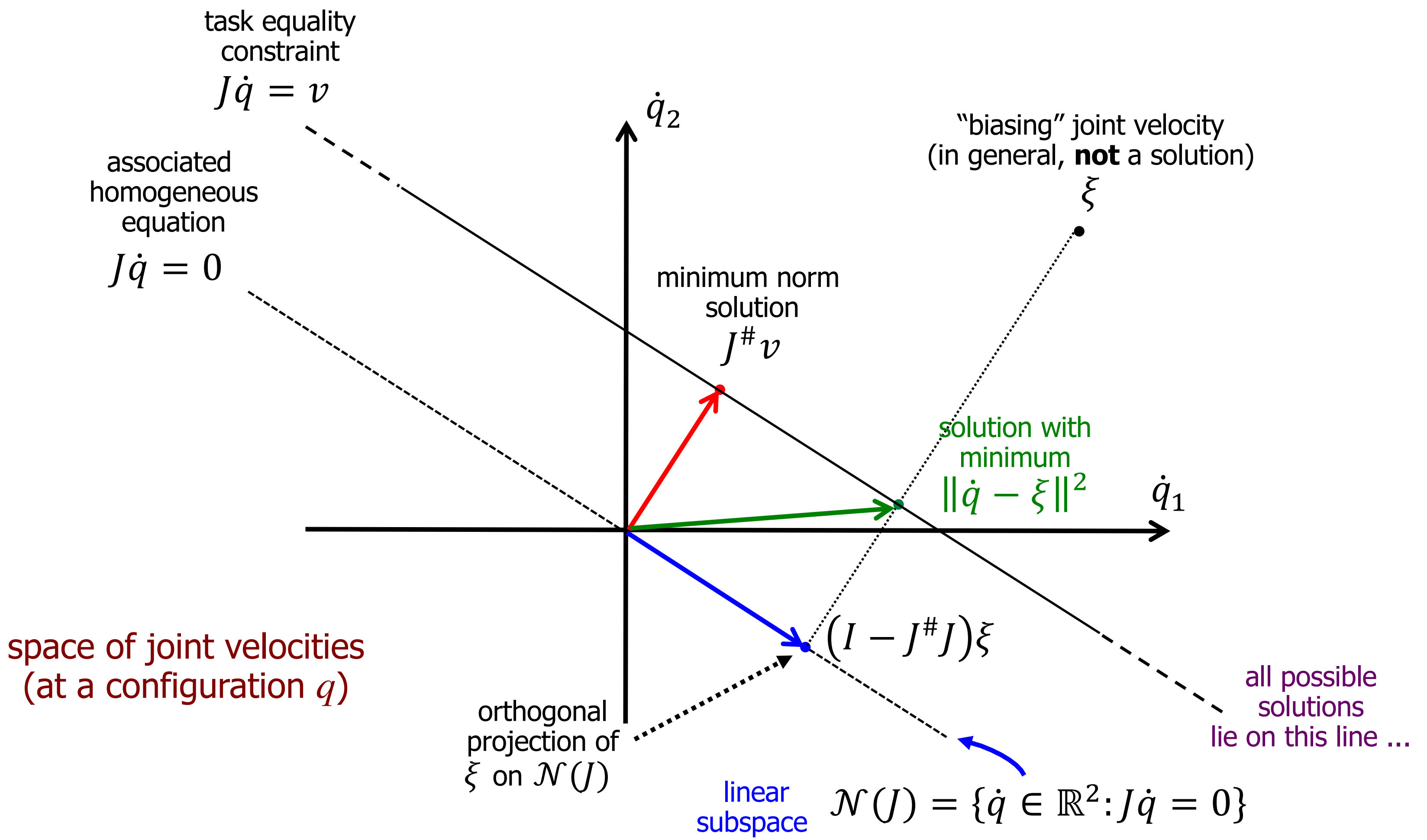
$$v_{actual} = J\dot{q} = J\left(J^{\#} v + \left(I - J^{\#} J\right)\xi\right) = J\, J^{\#} v + J\left(I - J^{\#} J\right)\xi = J\, J^{\#}(Jw) = Jw = v$$

if $v \in \mathcal{R}(J) \Rightarrow v = Jw$ for some $w \in \mathbb{R}^n$

Geometric interpretation for $m < n$

a simple case with $n = 2,\ m = 1$
at a given configuration

$$J\dot{q} = [\,j_1 \quad j_2\,] \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = v \in \mathbb{R}$$



Higher-order differential inversion

- inversion of motion from task to joint space can be performed also at a higher differential level
- acceleration-level: given $q, \dot{q}$

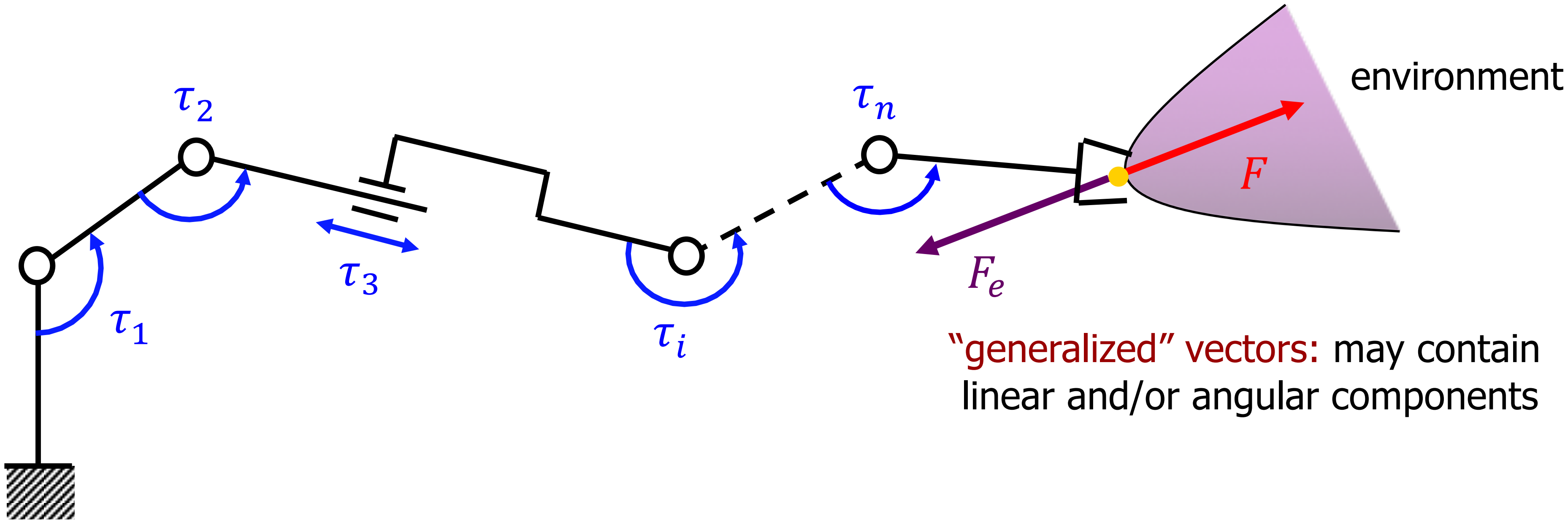
$$\ddot{q} = J_r^{-1}(q)\left(\ddot{r} - \dot{J}_r(q)\dot{q}\right)$$

- jerk-level: given $q, \dot{q}, \ddot{q}$

$$\dddot{q} = J_r^{-1}(q)\left(\dddot{r} - \dot{J}_r(q)\ddot{q} - 2\ddot{J}_r(q)\dot{q}\right)$$

- (pseudo-)inverse of the Jacobian is always the leading term
- smoother joint motions are expected (at least, due to the existence of higher-order time derivatives $\ddot{r}, \dddot{r}, \ldots$)

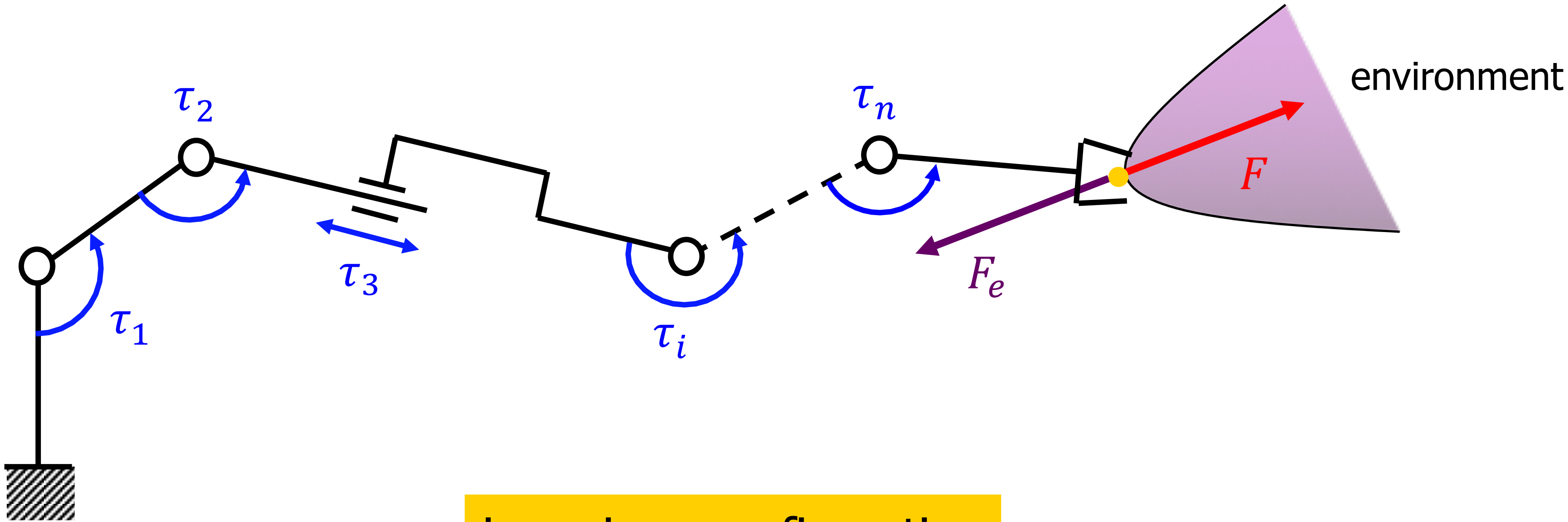
Generalized forces and torques



- τ = forces/torques exerted by the motors at the robot joints
- F = equivalent forces/torques exerted by the robot end-effector
- F_e = forces/torques exerted by the environment at the end-effector
- principle of action and reaction: $F_e = -F$

  reaction from environment is equal and opposite to the robot action on it

Transformation of forces – Statics



in a given configuration

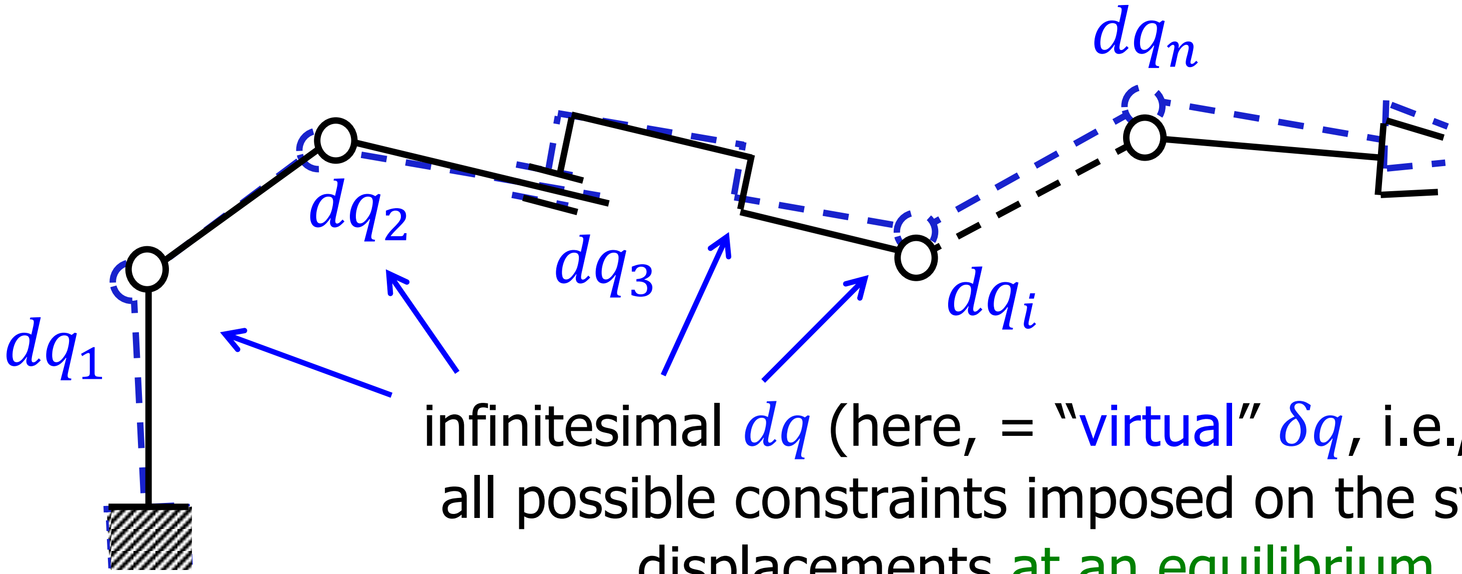
- what is the transformation between F at robot end-effector and τ at joints?

in **static equilibrium** conditions (i.e., **no motion**):

- what F will be exerted on environment by a τ applied at the robot joints?
- what τ at the joints will balance a F_e (= $-F$) exerted by the environment?

all equivalent formulations

Virtual displacements and works



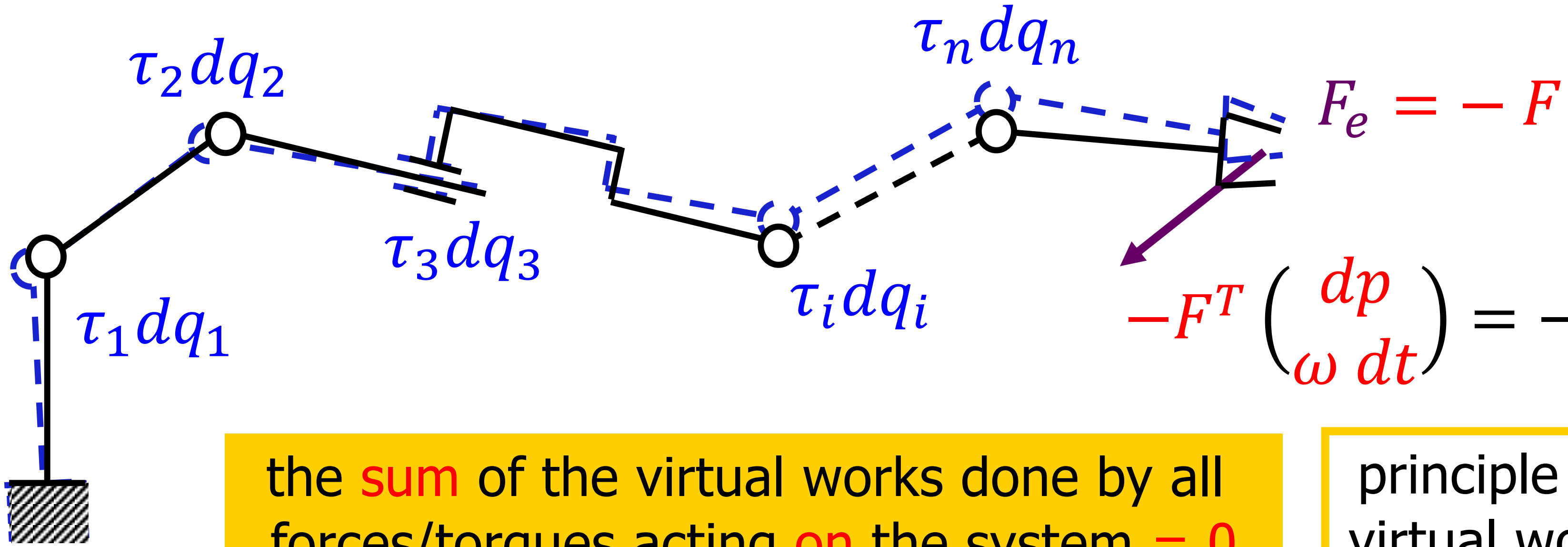
$$\begin{pmatrix} dp \\ \omega\, dt \end{pmatrix} = J\, dq$$

infinitesimal dq (here, = "virtual" δq, i.e., satisfy all possible constraints imposed on the system) displacements at an equilibrium

➡
- without kinetic energy variation (zero acceleration)
- without dissipative effects (zero velocity)

the virtual work is the work done by all forces/torques acting on the system for a given virtual displacement

Principle of virtual work



$$-F^T \begin{pmatrix} dp \\ \omega\, dt \end{pmatrix} = -F^T J\, dq$$

the sum of the virtual works done by all forces/torques acting on the system = 0

principle of virtual work

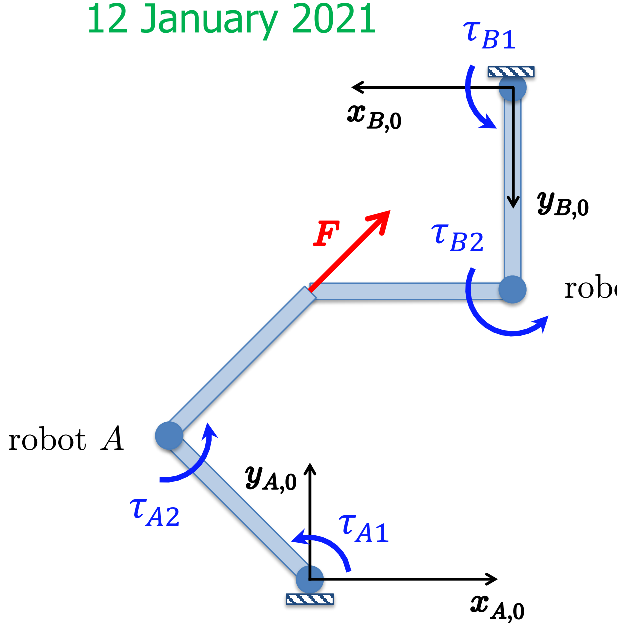
$$\tau^T dq - F^T \begin{pmatrix} dp \\ \omega dt \end{pmatrix} = \tau^T dq - F^T J dq = 0 \qquad \forall dq$$

$$\Rightarrow \quad \boxed{\tau = J^T(q) F}$$

Exercise on static balance

whiteboard …

Q#7 in Robotics 1 exam, 12 January 2021



Two planar 2R robots A and B having unitary link lengths are in their D-H configurations $\boldsymbol{q}_A = (3\pi/4, -\pi/2)$, $\boldsymbol{q}_B = (\pi/2, -\pi/2)$ [rad] w.r.t. their base frames, as in figure (no gravity!).

Robot A pushes against robot B with a force $\boldsymbol{F} \in \mathbb{R}^2$ of norm $\|\boldsymbol{F}\| = 10$ [N], as in figure. Compute the joint torques $\boldsymbol{\tau}_A \in \mathbb{R}^2$ and $\boldsymbol{\tau}_B \in \mathbb{R}^2$ (both in [Nm]) that keep the two robots in equilibrium.

solution

i) evaluate the task Jacobians of the two robots ($\dot{q}_A \to v_A$ and $\dot{q}_B \to v_B$)

$$\boldsymbol{J}_A(\boldsymbol{q}_A) = \begin{pmatrix} -\sin q_1 - \sin(q_1+q_2) & -\sin(q_1+q_2) \\ \cos q_1 + \cos(q_1+q_2) & \cos(q_1+q_2) \end{pmatrix}\Bigg|_{\boldsymbol{q}=\boldsymbol{q}_A} = \begin{pmatrix} -\sqrt{2} & -\frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{2}}{2} \end{pmatrix}$$

$$\boldsymbol{J}_B(\boldsymbol{q}_B) = \begin{pmatrix} -\sin q_1 - \sin(q_1+q_2) & -\sin(q_1+q_2) \\ \cos q_1 + \cos(q_1+q_2) & \cos(q_1+q_2) \end{pmatrix}\Bigg|_{\boldsymbol{q}=\boldsymbol{q}_B} = \begin{pmatrix} -1 & 0 \\ 1 & 1 \end{pmatrix}$$

ii) express the exchanged force in the proper frame(s) …

$${}^A\boldsymbol{F}_A = \|\boldsymbol{F}\| \cdot \begin{pmatrix} \cos(q_1+q_2) \\ \sin(q_1+q_2) \end{pmatrix}\Bigg|_{\boldsymbol{q}=\boldsymbol{q}_A} = 10 \begin{pmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix} \text{ [N]}$$

$${}^B\boldsymbol{F}_B = {}^B\boldsymbol{R}_A\, {}^A\boldsymbol{F}_B = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} (-\,{}^A\boldsymbol{F}_A) = {}^A\boldsymbol{F}_A = 10 \begin{pmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix} \text{ [N]}$$

planar rotation matrix $\in SO(2)$

iii) … and compute the torque for each robot by the virtual work principle

$$\boldsymbol{\tau}_A = \boldsymbol{J}_A^T(\boldsymbol{q}_A)\, {}^A\boldsymbol{F}_A = \begin{pmatrix} -10 \\ 0 \end{pmatrix} \text{ [Nm]}$$

$$\boldsymbol{\tau}_B = \boldsymbol{J}_B^T(\boldsymbol{q}_B)\, {}^B\boldsymbol{F}_B = \begin{pmatrix} 0 \\ 5\sqrt{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 7.0711 \end{pmatrix} \text{ [Nm].}$$

Duality between velocity and force

$J(q)$

velocity $\dot{q}$
(or displacement dq)
in the joint space

generalized velocity v
(or e-e displacement $\begin{pmatrix} dp \\ \omega\, dt \end{pmatrix}$)
in the Cartesian space

forces/torques τ
at the joints

generalized forces F
at the Cartesian e-e

$J^T(q)$

the singular configurations
for the velocity map are the same
as those for the force map

$\rho(J) = \rho(J^T)$

Dual subspaces of velocity and force

summary of definitions

$$\mathcal{R}(J) = \{v \in \mathbb{R}^m : \exists \dot{q} \in \mathbb{R}^n, J\dot{q} = v\}$$

$$\mathcal{N}(J^T) = \{F \in \mathbb{R}^m : J^T F = 0\}$$

$$\mathcal{R}(J) \bigoplus \mathcal{N}(J^T) = \mathbb{R}^m$$

$$\mathcal{R}(J^T) = \{\tau \in \mathbb{R}^n : \exists F \in \mathbb{R}^m, J^T F = \tau\}$$

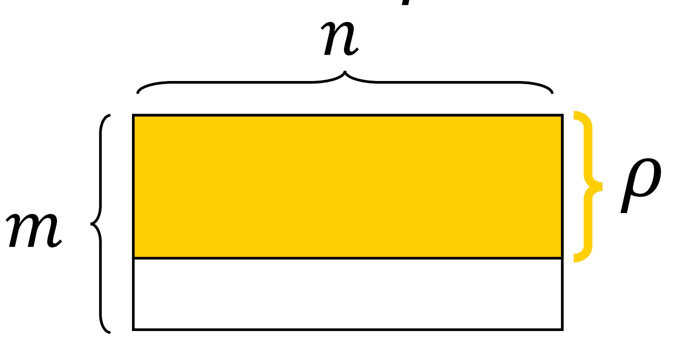
$$\mathcal{N}(J) = \{\dot{q} \in \mathbb{R}^n : J\dot{q} = 0\}$$

$$\mathcal{R}(J^T) \bigoplus \mathcal{N}(J) = \mathbb{R}^n$$

Velocity and force singularities

list of possible cases

$$\rho = \mathrm{rank}(J) = \mathrm{rank}(J^T) \leq \min(m, n)$$



1. $\rho = m$

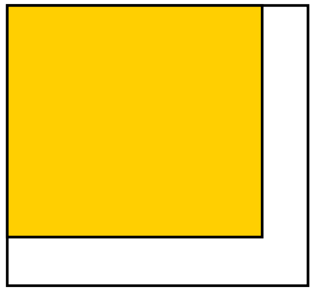
$$\exists \dot{q} \neq 0 : \; J\dot{q} = 0$$

$$\mathcal{N}(J^T) = \{0\}$$

2. $\rho < m$

$$\exists \dot{q} \neq 0 : \; J\dot{q} = 0$$

$$\exists F \neq 0 : \; J^T F = 0$$



1. $\det J \neq 0$

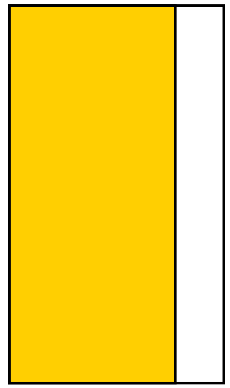
$$\mathcal{N}(J) = \{0\}$$

$$\mathcal{N}(J^T) = \{0\}$$

2. $\det J = 0$

$$\exists \dot{q} \neq 0 : \; J\dot{q} = 0$$

$$\exists F \neq 0 : \; J^T F = 0$$



1. $\rho = n$

$$\mathcal{N}(J) = \{0\}$$

$$\exists F \neq 0 : \; J^T F = 0$$

2. $\rho < n$

$$\exists \dot{q} \neq 0 : \; J\dot{q} = 0$$

$$\exists F \neq 0 : \; J^T F = 0$$

Singularity analysis

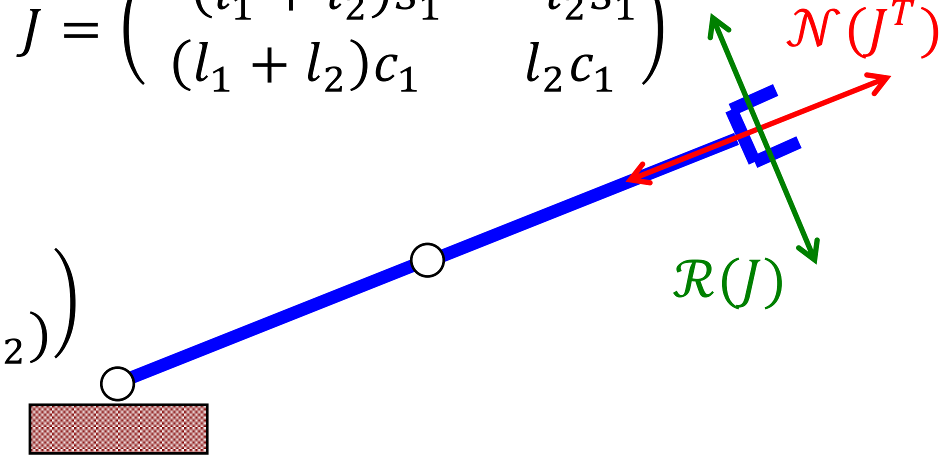
planar 2R arm with
link lengths l_1 and l_2

$$J(q) = \begin{pmatrix} -(l_1 s_1 + l_2 s_{12}) & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \end{pmatrix} \qquad \det J(q) = l_1 l_2 s_2$$

singularity at $q_2 = 0$ (arm straight) ➡ $J = \begin{pmatrix} -(l_1 + l_2) s_1 & -l_2 s_1 \\ (l_1 + l_2) c_1 & l_2 c_1 \end{pmatrix}$

$$\mathcal{R}(J) = \alpha \begin{pmatrix} -s_1 \\ c_1 \end{pmatrix} \qquad \mathcal{N}(J^T) = \alpha \begin{pmatrix} c_1 \\ s_1 \end{pmatrix}$$

$$\mathcal{R}(J^T) = \beta \begin{pmatrix} l_1 + l_2 \\ l_2 \end{pmatrix} \qquad \mathcal{N}(J) = \beta \begin{pmatrix} l_2 \\ -(l_1 + l_2) \end{pmatrix}$$



singularity at $q_2 = \pi$ (arm folded) ➡ $J = \begin{pmatrix} (l_2 - l_1) s_1 & l_2 s_1 \\ -(l_2 - l_1) c_1 & -l_2 c_1 \end{pmatrix}$

$\mathcal{R}(J)$ and $\mathcal{N}(J^T)$ as above

$$\mathcal{R}(J^T) = \beta \begin{pmatrix} l_2 - l_1 \\ l_2 \end{pmatrix} \text{ (for } l_1 = l_2\text{: } \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix}\text{)} \qquad \mathcal{N}(J) = \beta \begin{pmatrix} l_2 \\ -(l_2 - l_1) \end{pmatrix} \text{ (for } l_1 = l_2\text{: } \beta \begin{pmatrix} 1 \\ 0 \end{pmatrix}\text{)}$$

Velocity manipulability

- in a given configuration, evaluate how effective is the transformation between joint and end-effector velocities
  - "how easily" can the end-effector be moved in various directions of the task space
  - equivalently, "how far" is the robot from a singular condition
- we consider all end-effector velocities that can be obtained by choosing joint velocity vectors of unit norm

$$\dot{q}^T \dot{q} = 1 \quad \Rightarrow \quad v^T {J^{\#}}^T J^{\#} v = 1$$

task velocity manipulability ellipsoid

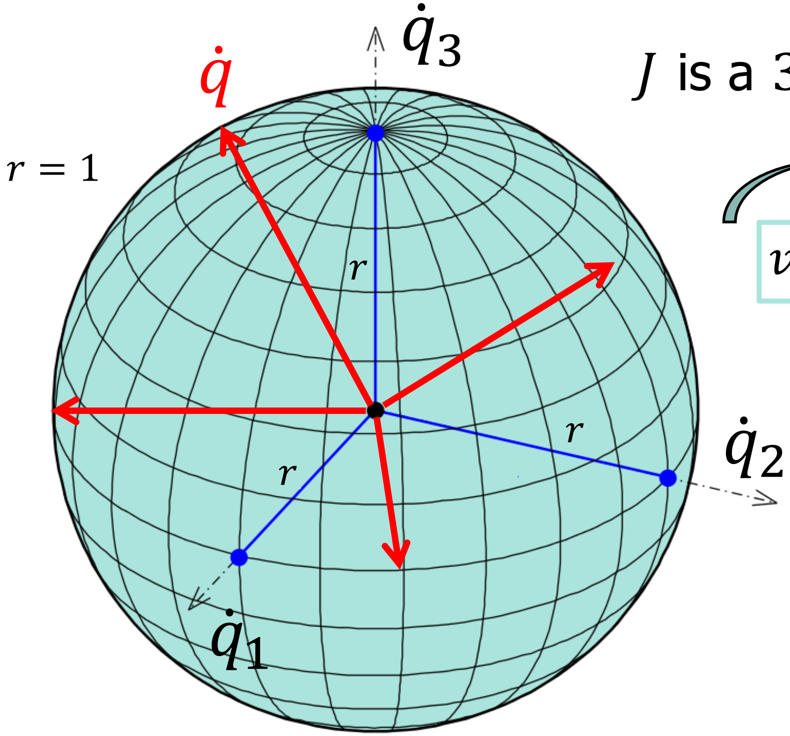
if $\rho(J) = m$

$$(J\,J^T)^{-1}$$

note: the "core" matrix of the ellipsoid equation $v^T A^{-1} v = 1$ is the matrix A!

(Hyper-) Spheres and Ellipsoids

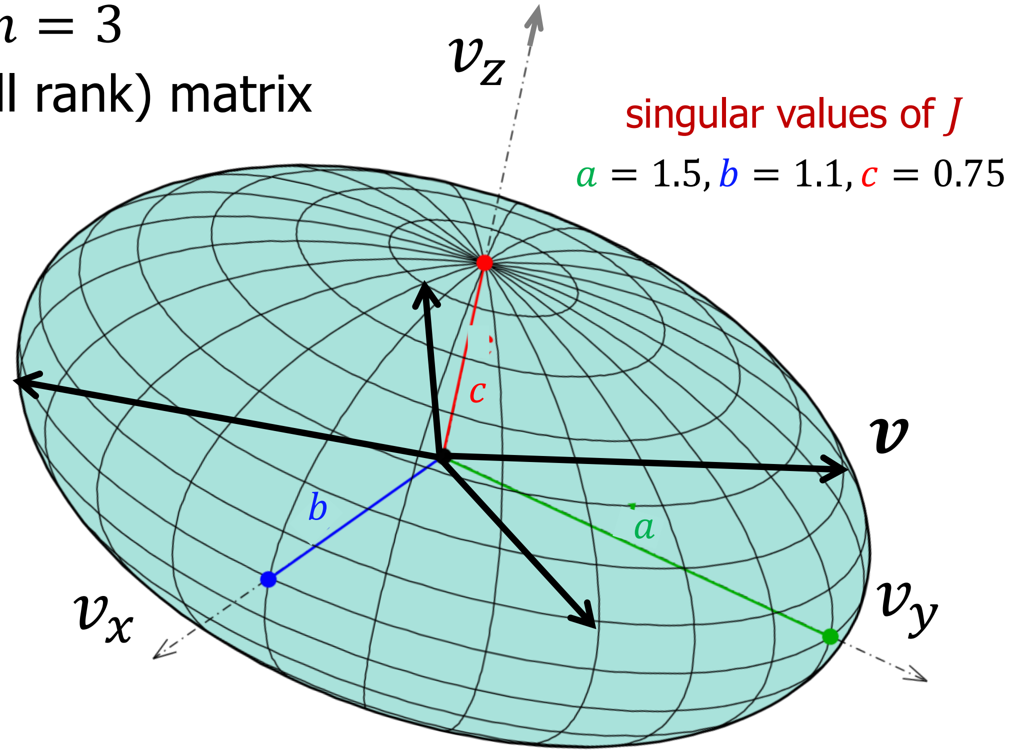
whiteboard ...



$m = n = 3$

J is a 3×3 (full rank) matrix

$v = J\dot q$

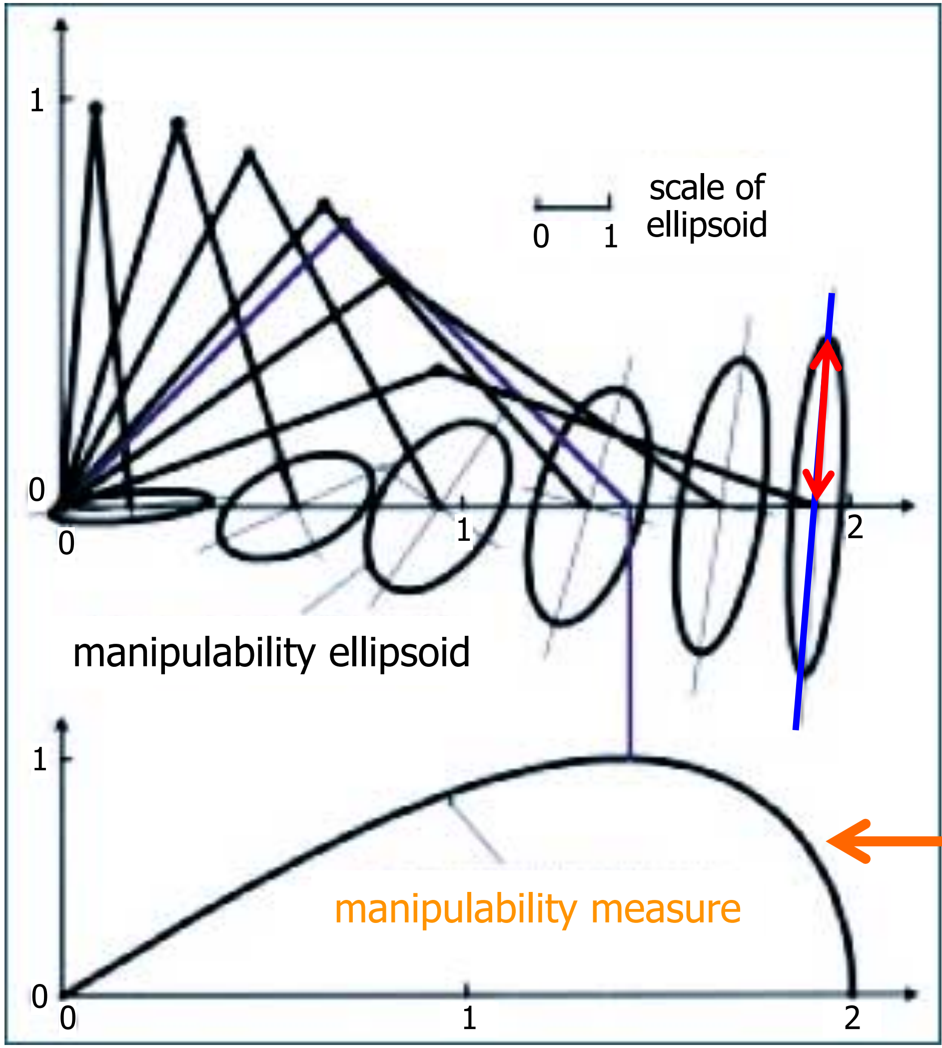


$$\dot q_1^2 + \dot q_2^2 + \dot q_3^2 = \dot q^T \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} \dot q = 1 \qquad \frac{v_x^2}{a^2} + \frac{v_y^2}{b^2} + \frac{v_z^2}{c^2} = \boldsymbol{v}^T \begin{pmatrix} a^2 & & \\ & b^2 & \\ & & c^2 \end{pmatrix}^{-1} \boldsymbol{v} = 1$$

$$\dot{\boldsymbol{q}}^T \dot{\boldsymbol{q}} = 1 \quad \Rightarrow \quad \boxed{\boldsymbol{v}^T \left(J\,J^T\right)^{-1} \boldsymbol{v} = 1}$$

Manipulability ellipsoid
in velocity

planar 2R arm with unitary links



length of principal (semi-)axes

singular values σ_i of J (in its SVD)

$$\sigma_i(J) = \sqrt{\lambda_i(J\,J^T)}$$

in a singularity, the ellipsoid loses a dimension
(for $m = 2$, it becomes a segment)

direction of principal axes

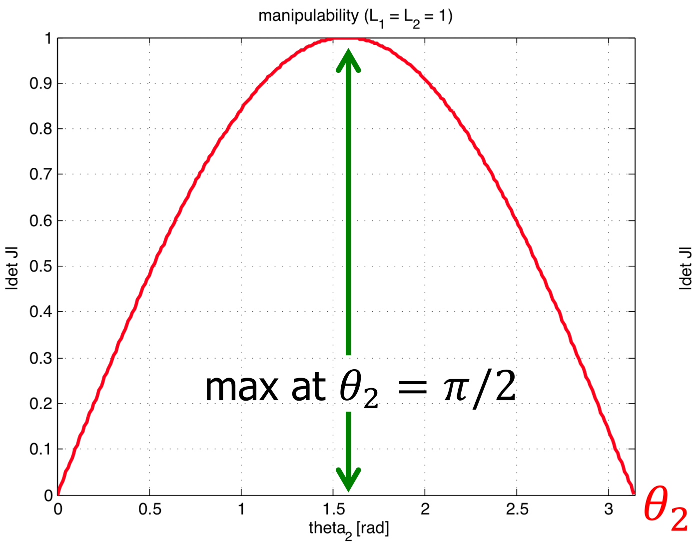
eigenvectors associated to λ_i

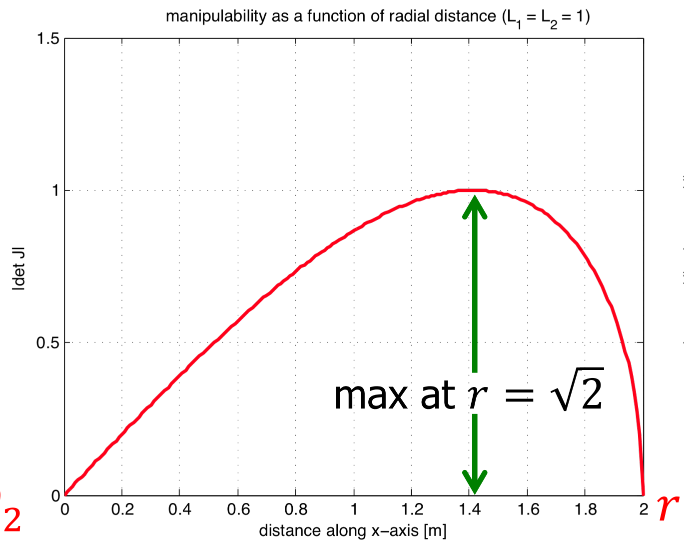
$$w = \sqrt{det\,(J\,J^T)} = \prod_{i=1}^{m} \sigma_i \geq 0$$

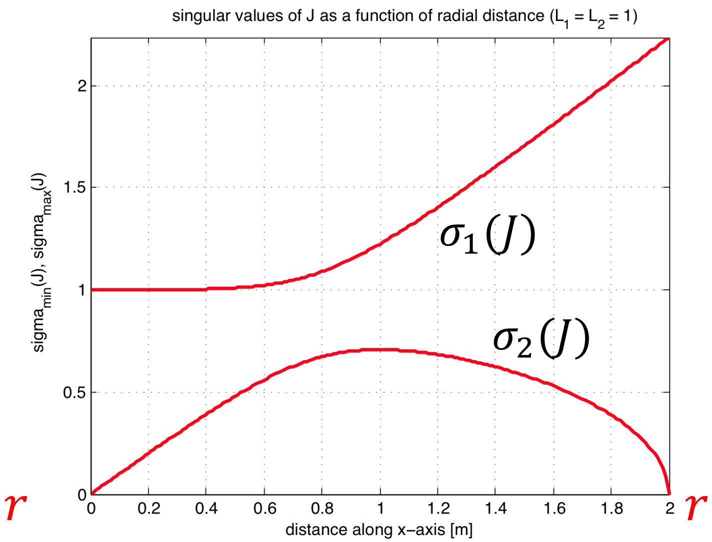
proportional to the volume of the ellipsoid (for $m = 2$, to its area)

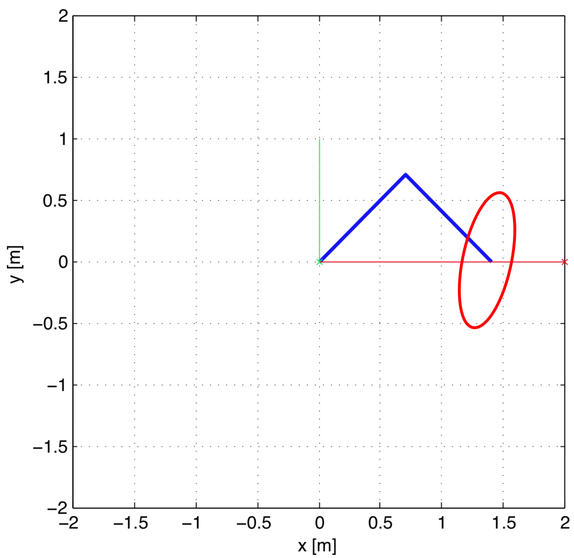
Manipulability measure

planar 2R arm (with $l_1 = l_2 = 1$): $\sqrt{det(J\,J^T)} = \sqrt{det(J) \cdot det(J^T)} = |\det J| = \prod_{i=1}^{2} \sigma_i$









best posture for manipulation (similar to a human arm!)

no full isotropy here, since it is always $\sigma_1 \neq \sigma_2$

Force manipulability

- in a given configuration, evaluate how effective is the transformation between joint torques and end-effector forces
  - "how easily" can the end-effector apply generalized forces (or balance applied ones) in the various directions of the task space
  - in singular configurations, there are directions in the task space where external forces are balanced without the need of any joint torque
- we consider all end-effector forces that can be applied (or balanced) by choosing joint torque vectors of unit norm

$$\tau^T \tau = 1 \quad \Rightarrow \quad \boxed{F^T J J^T F = 1}$$

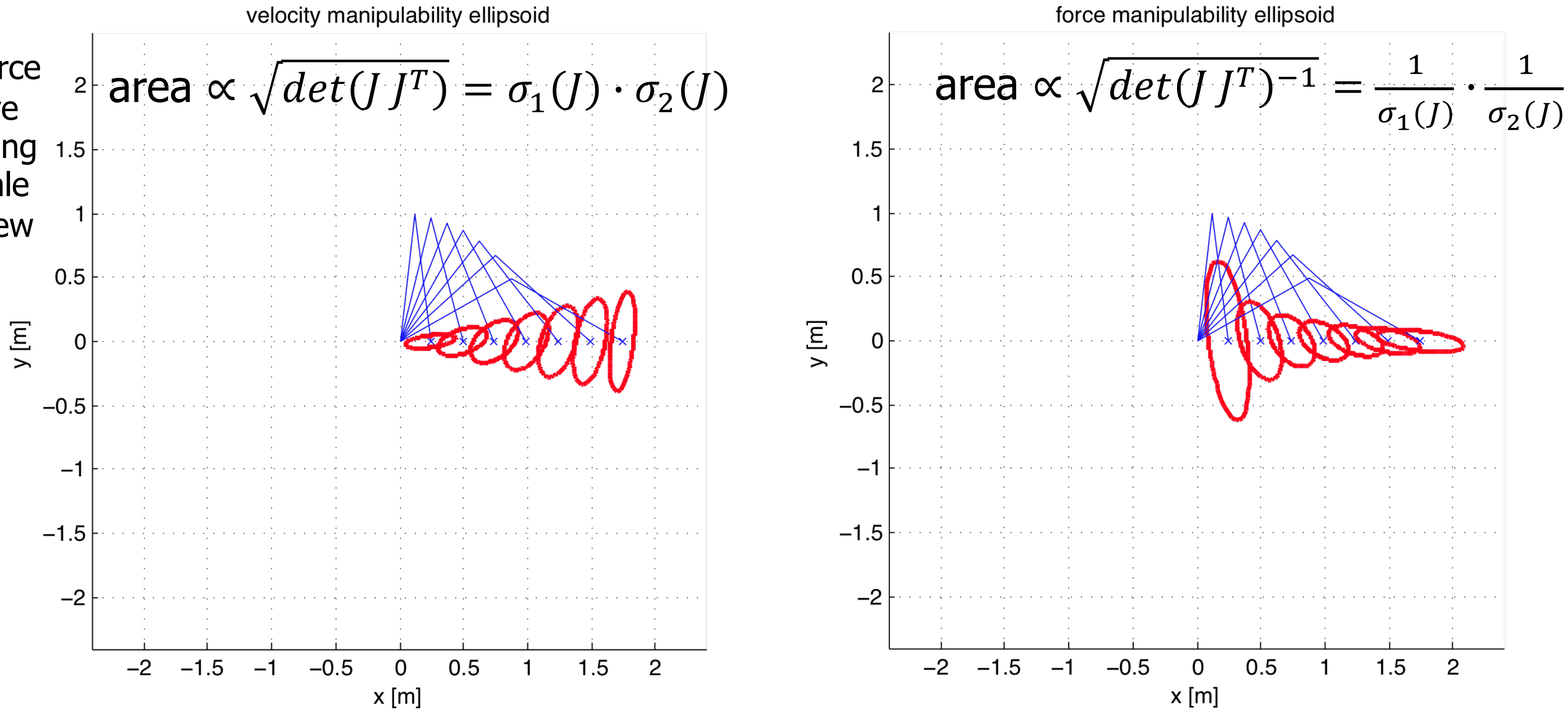
same directions of the principal axes of the velocity ellipsoid, but with semi-axes of inverse lengths → task force manipulability ellipsoid

Velocity and force manipulability

dual comparison of actuation vs. control

planar 2R arm with unitary links

note:
velocity and force ellipsoids have been drawn using a different scale for a better view



Cartesian **actuation** task (joint-to-task high transformation ratio):
preferred velocity (or force) directions are those where the ellipsoid stretches

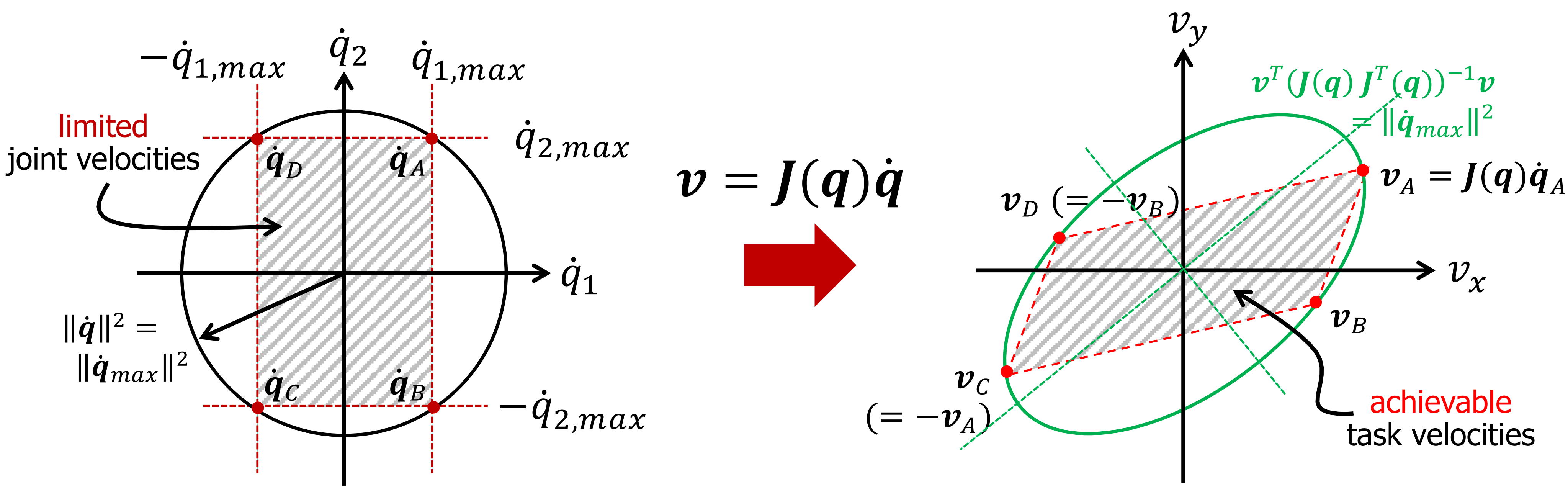
⇕

Cartesian **control** task (low transformation ratio = high resolution):
preferred velocity (or force) directions are those where the ellipsoid shrinks

Ellipsoids and polytopes

manipulability versus task limits due to bounds

- manipulability: instantaneous capability of moving the end-effector (or of resisting to task forces) in different directions
- task limits: maximum velocity (or static balanced force) achievable in different task directions in the presence of joint velocity bounds



$$-\dot{q}_{i,max} \leq \dot{q}_i \leq \dot{q}_{i,max}$$

velocity ellipsoid and polytope at $\boldsymbol{q}$
for a 2R robot with joint velocity bounds

- a polytope is the convex hull of a set of p points in an Euclidean space
- linear maps transform polytopes into polytopes

Velocity and force transformations

- same reasoning made for relating end-effector to joint forces/torques (virtual work principle + static equilibrium) used also for transforming forces and torques applied at different places of a rigid body and/or expressed in different reference frames

transformation among generalized velocities

$$\begin{bmatrix} {}^A v_A \\ {}^A \omega \end{bmatrix} = \begin{bmatrix} {}^A R_B & -{}^A R_B S({}^B r_{BA}) \\ 0 & {}^A R_B \end{bmatrix} \begin{bmatrix} {}^B v_B \\ {}^B \omega \end{bmatrix} = J_{BA} \begin{bmatrix} {}^B v_B \\ {}^B \omega \end{bmatrix}$$

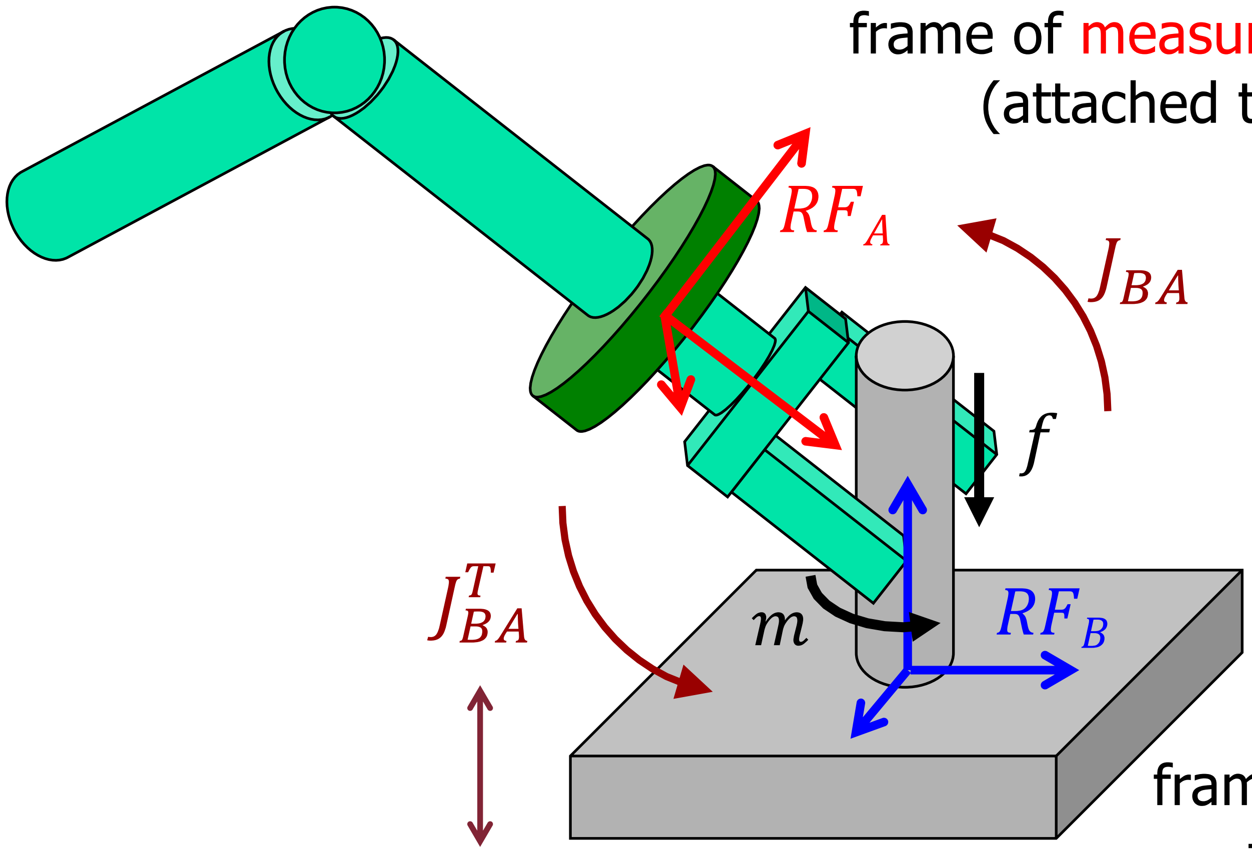
$$\begin{bmatrix} {}^B f_B \\ {}^B m \end{bmatrix} = J_{BA}^T \begin{bmatrix} {}^A f_A \\ {}^A m \end{bmatrix} = \begin{bmatrix} {}^B R_A & 0 \\ S({}^B r_{BA})\, {}^B R_A & {}^B R_A \end{bmatrix} \begin{bmatrix} {}^A f_A \\ {}^A m \end{bmatrix}$$

transformation among generalized forces

for skew-symmetric matrices, it is: $-S^T(r) = S(r)$

Example: 6D force/torque sensor

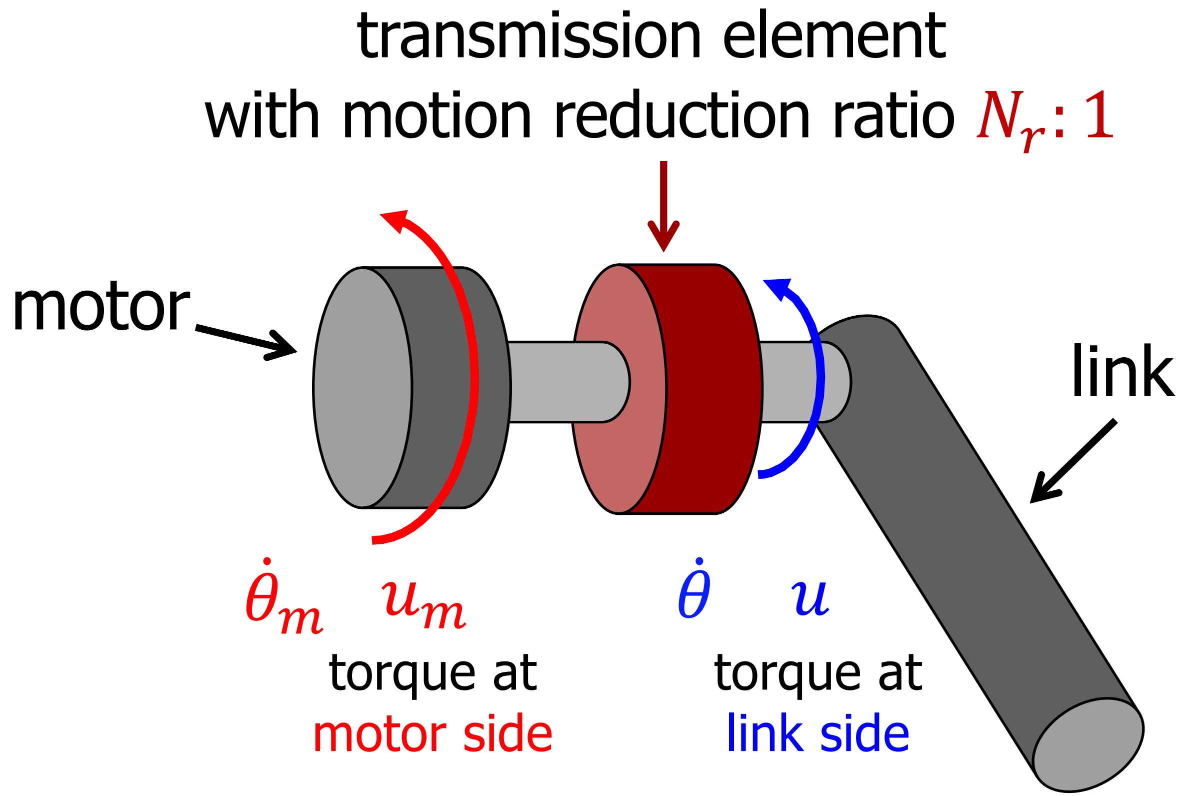
frame of measure for the forces/torques (attached to the wrist sensor)



frame of interest for evaluating forces/torques in a task with environment contact

$$\begin{bmatrix} {}^B f_B \\ {}^B m \end{bmatrix} = \begin{bmatrix} {}^B R_A & 0 \\ S({}^B r_{BA})\,{}^B R_A & {}^B R_A \end{bmatrix} \begin{bmatrix} {}^A f_A \\ {}^A m \end{bmatrix}$$

Example: Gear reduction at joints



one of the simplest applications of the principle of virtual work:

$P_m = u_m \dot{\theta}_m = u\,\dot{\theta} = P$

$$\dot{\theta}_m = N_r \dot{\theta}$$
$$u = N_r u_m$$

here, $J = J^T = N_r$ (a scalar!)