Robotics 1

Inverse differential kinematics

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

$$v = J(q)\dot{q} \xrightarrow[]{J \text{ square and non-singular at } q} \dot{q} = J^{-1}(q)v$$

e.g., for a twist $V = \begin{pmatrix} \mathrm{v} \\ \omega \end{pmatrix} \in \mathbb{R}^6$, with velocity $\mathrm{v} \in \mathbb{R}^3$ and angular velocity $\omega \in \mathbb{R}^3$, the matrix $J(q)$ is the geometric Jacobian

- 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

$$q$$

$$\downarrow$$

current next

$$r \longrightarrow \hat{r} = r + dr$$

next

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

first, increment the desired task variables

then, solve the inverse kinematics problem (possibly, with a numerical method from the current configuration)

next

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

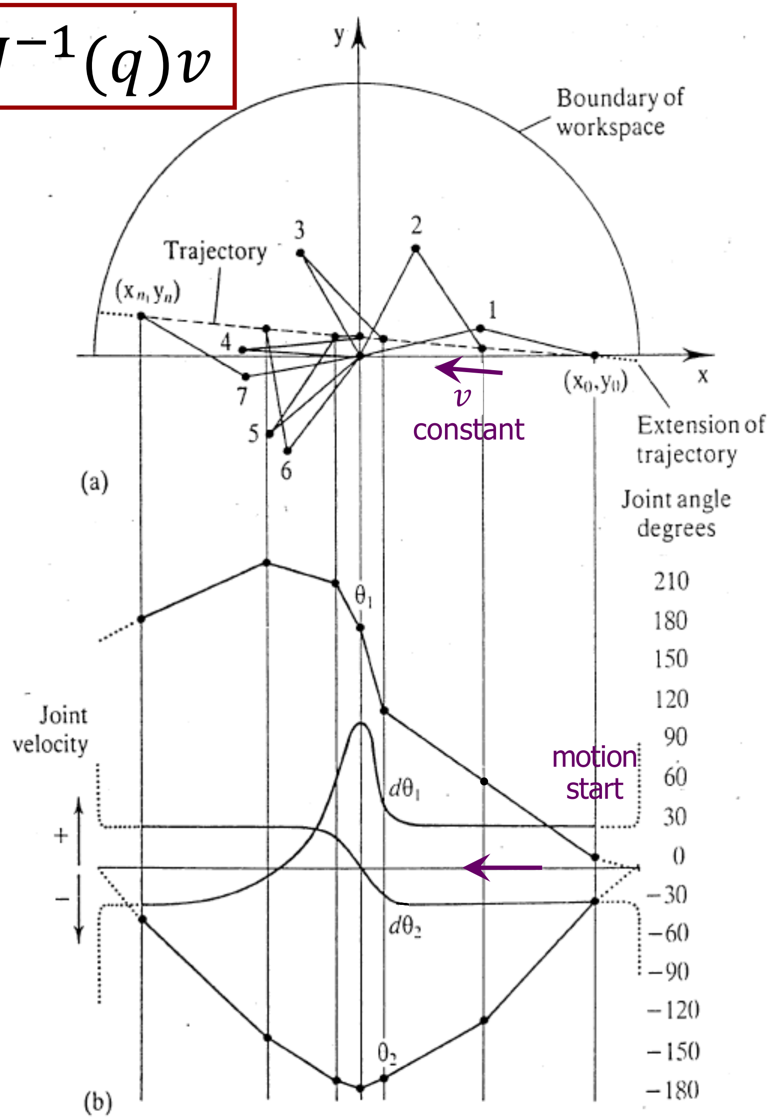
first, solve the inverse differential kinematics problem

then, increment the original joint variables

industrial robots are programmed to work in one (or both) of these ways!

Behavior close to a singularity

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



- 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

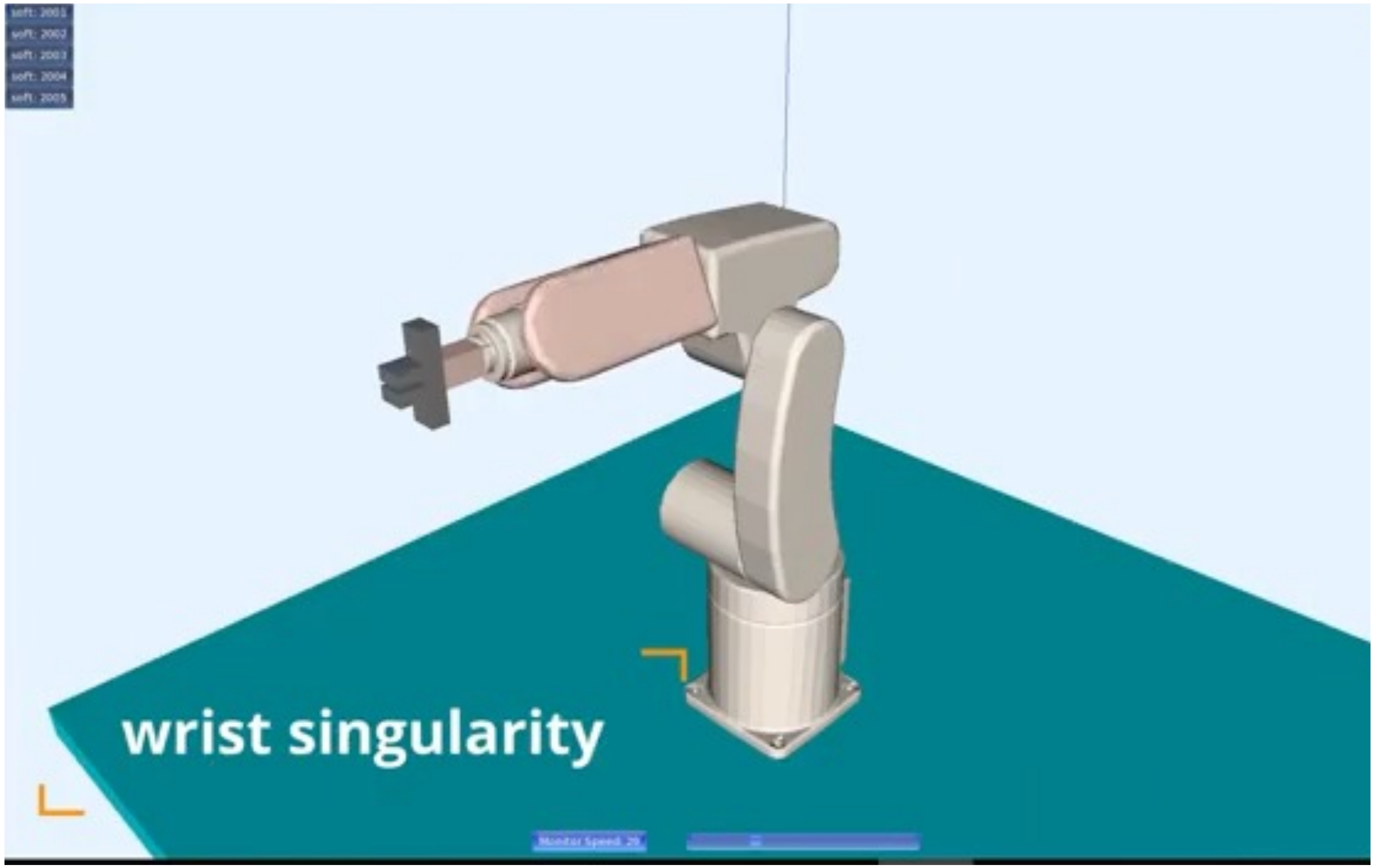
Moving close to a singularity
in inverse (differential) kinematics problems

- on-line inversion of velocities or incremental inverse kinematics
- singular configurations for a 6R robot with spherical wrist

wrist
joint axes
4 & 6 aligned

elbow
arm stretched
(or folded)

shoulder
wrist center on
first joint axis



video

Moving close to a singularity

6R KUKA Agile (with spherical wrist)

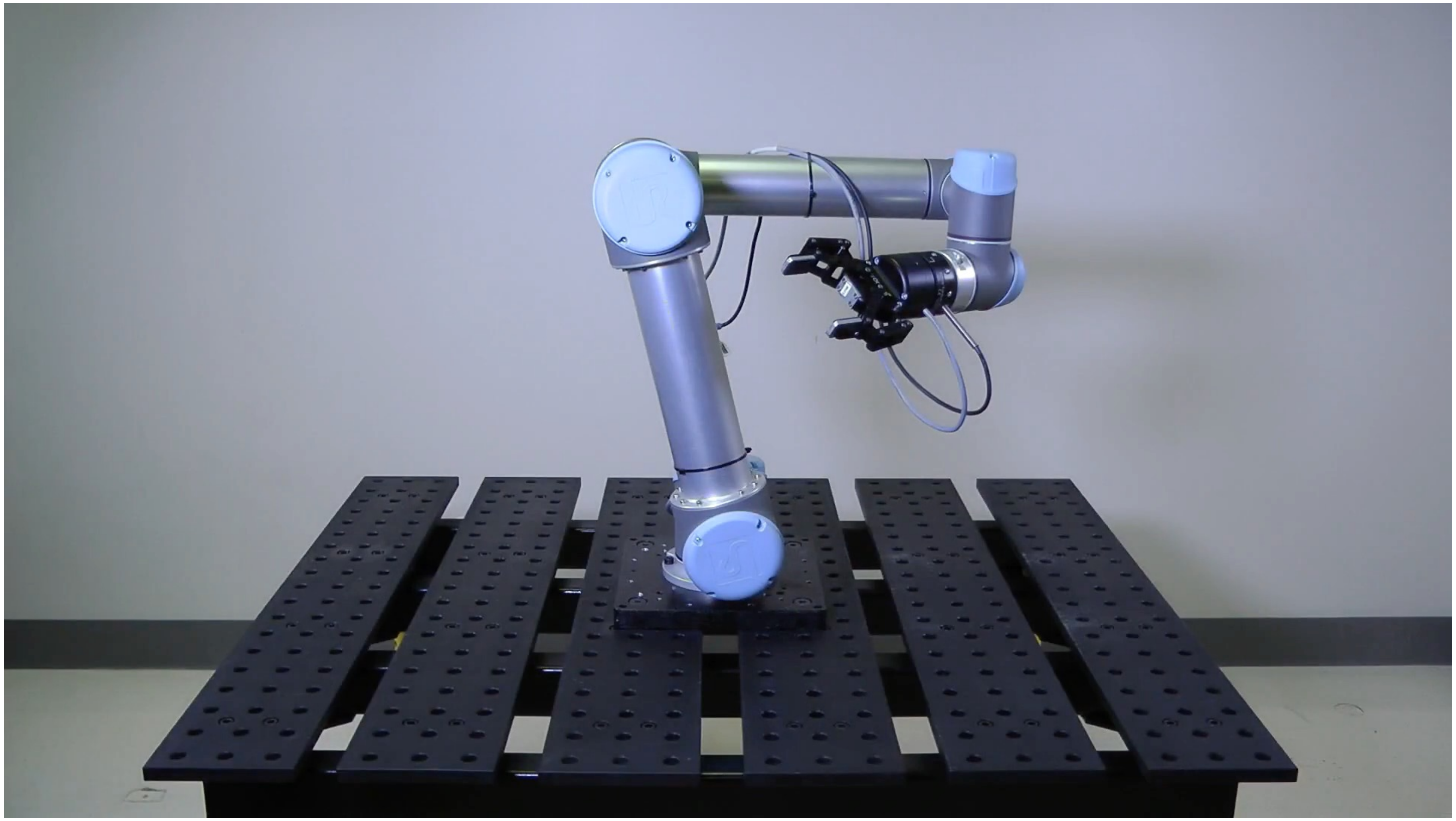
- wrist, shoulder and elbow singularities: feasible joint motions versus end-effector (linear) paths crossing/coming close to critical points

video Ecole de Technologie Supérieure, CoR Lab, Montreal

Moving close to a singularity

6R Universal Robots UR5 (no spherical wrist)

- same 'wrist', shoulder, and elbow singularities, though with slightly different configurations and full rotation of joints 4 & 6 in first case



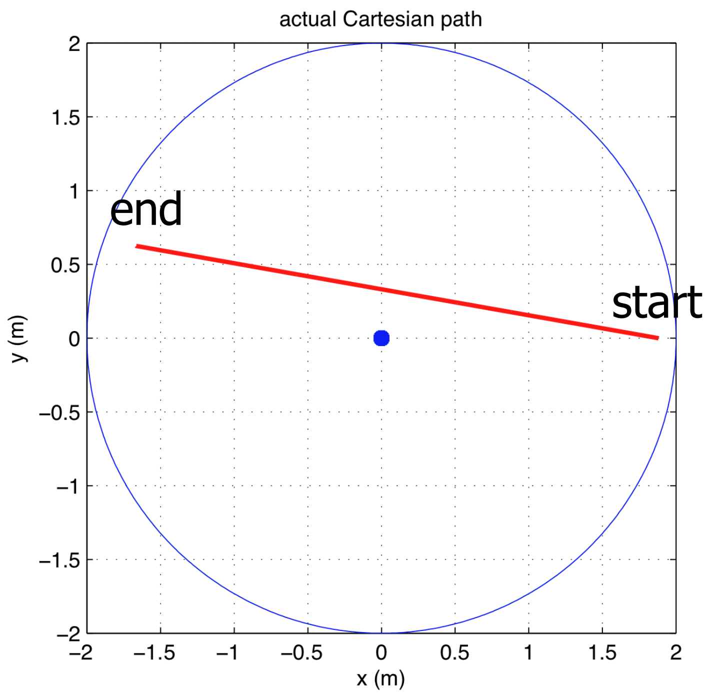
video Ecole de Technologie Supérieure, CoR Lab, Montreal all done in MATLAB

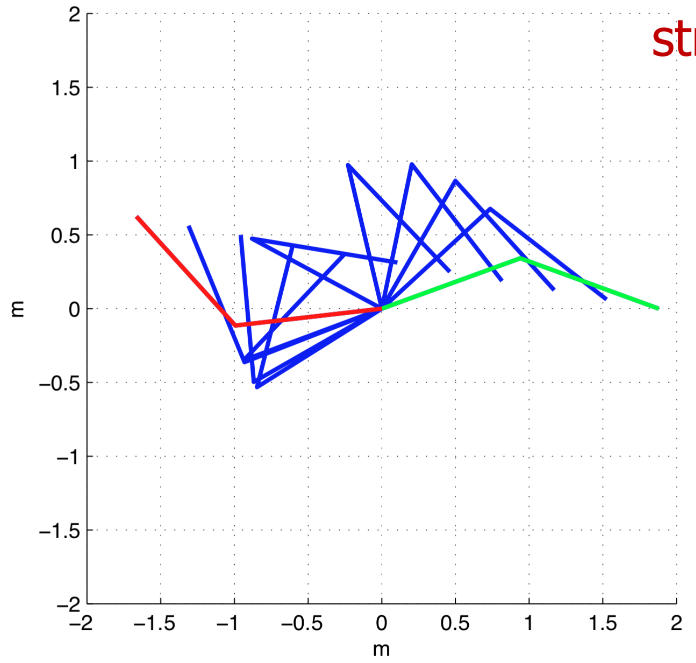
Simulation results

planar 2R robot in straight line Cartesian motion

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

regular case





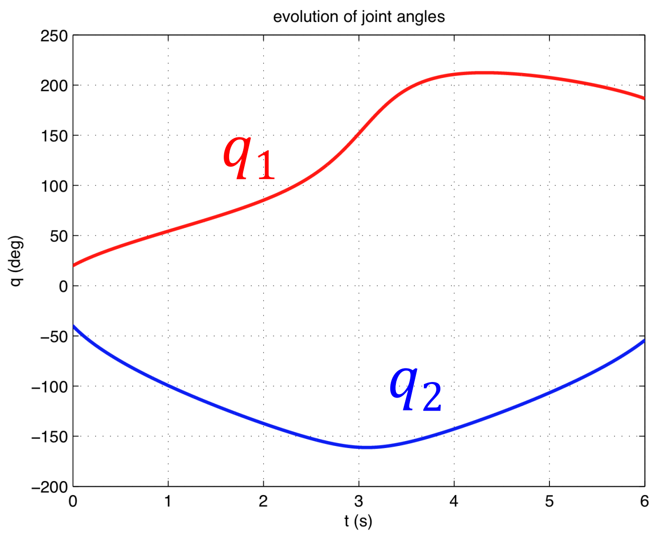
stroboscopic view

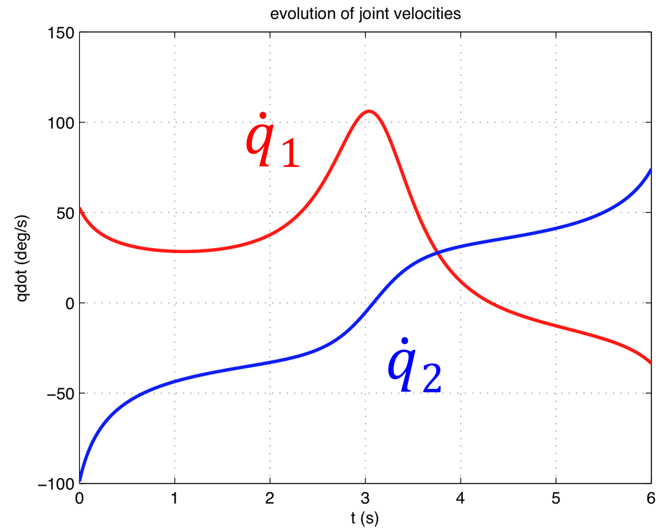
a line from right to left, at $\alpha = 170°$ angle with x-axis, executed at constant speed $\|v\| = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

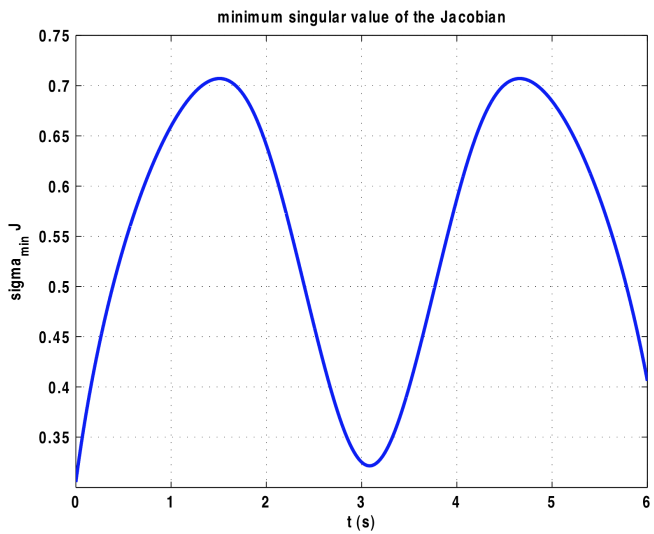
path at
$\alpha = 170°$

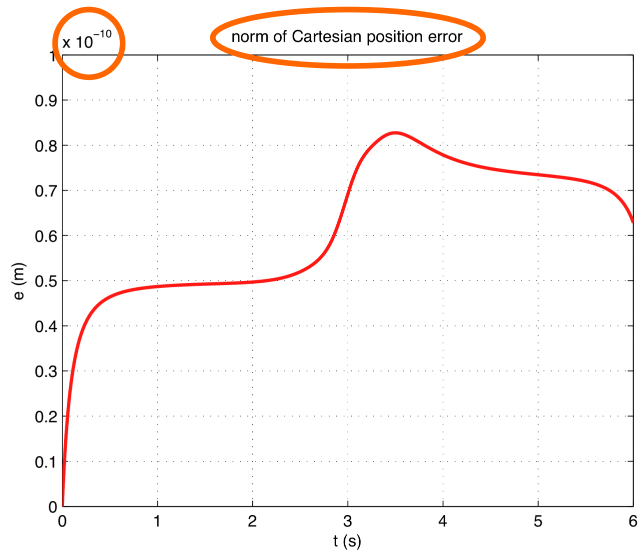




regular
case

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





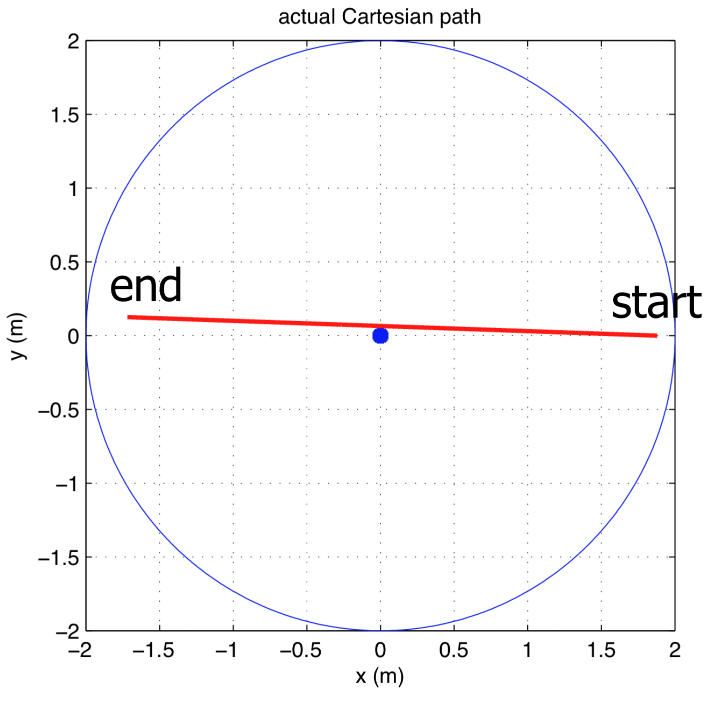
error due only to numerical integration (10^{-10})

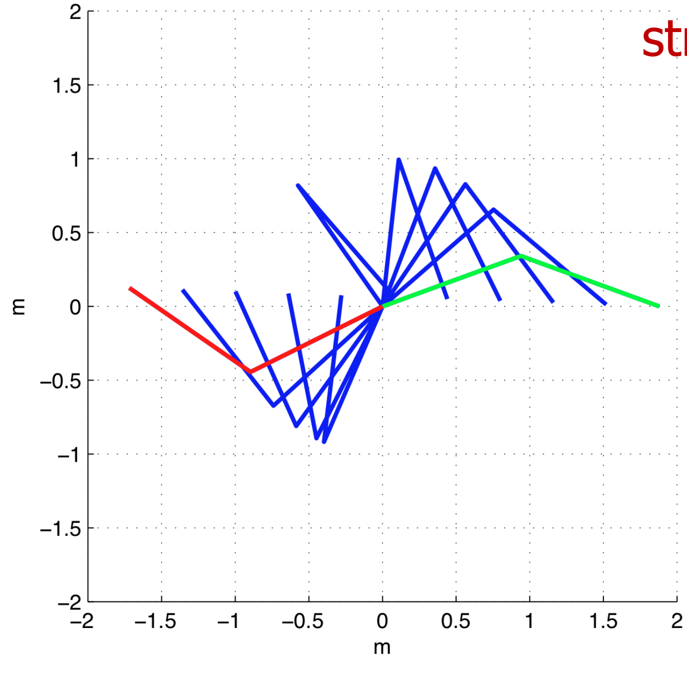
Simulation results

planar 2R robot in straight line Cartesian motion

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

close to singular case





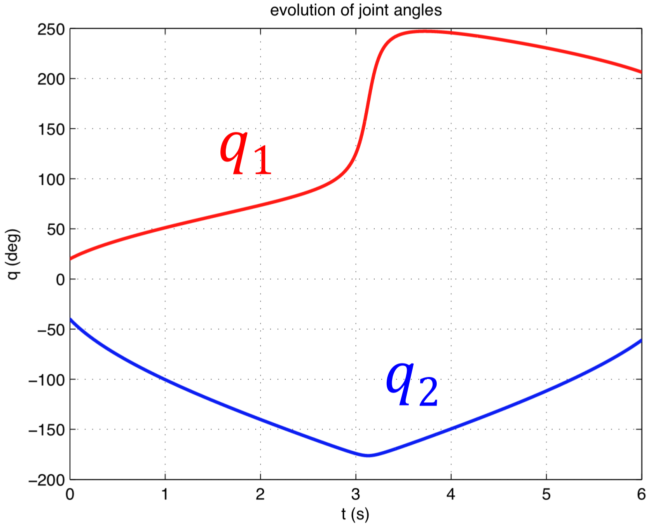
stroboscopic view

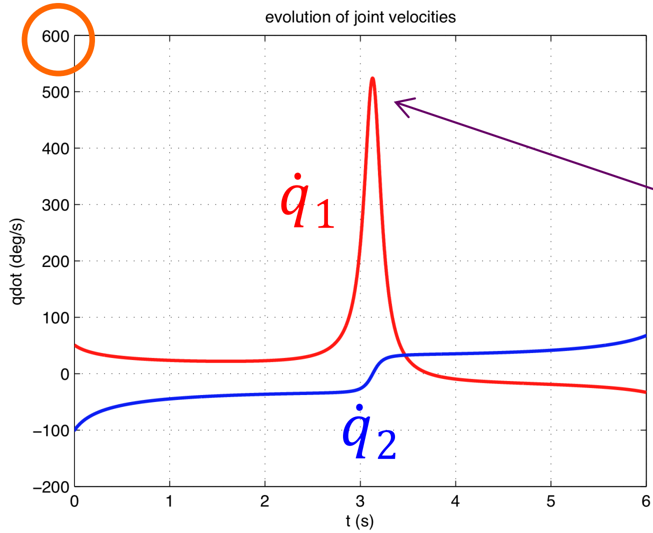
a line from right to left, at $\alpha = 178°$ angle with x-axis, executed at constant speed $\|v\| = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

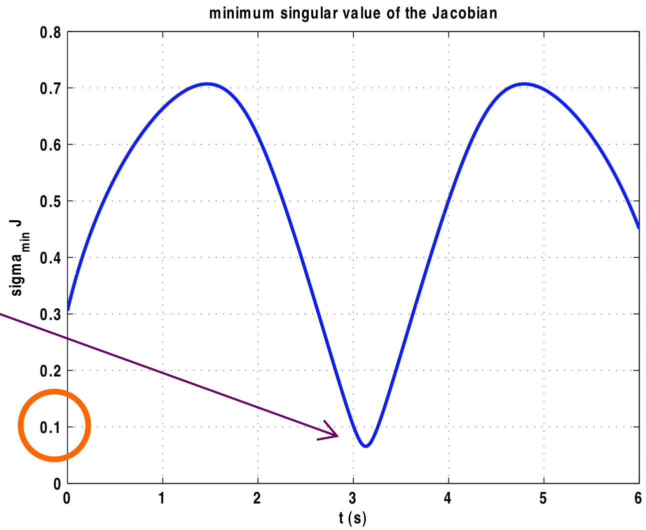
path at $\alpha = 178°$

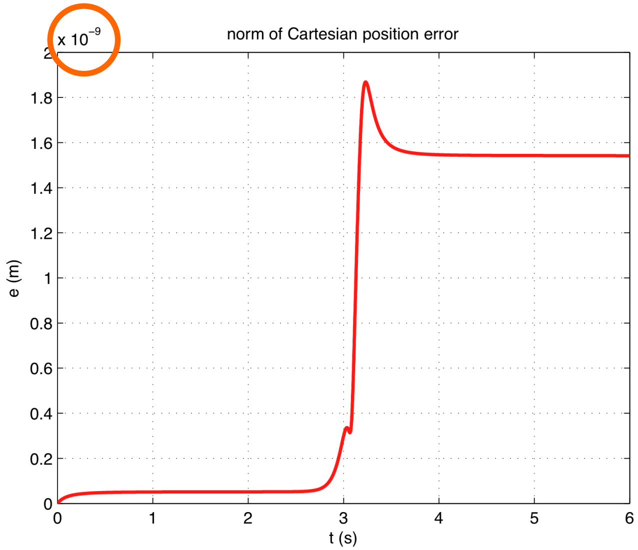




large peak of $\dot{q}_1$

close to singular case





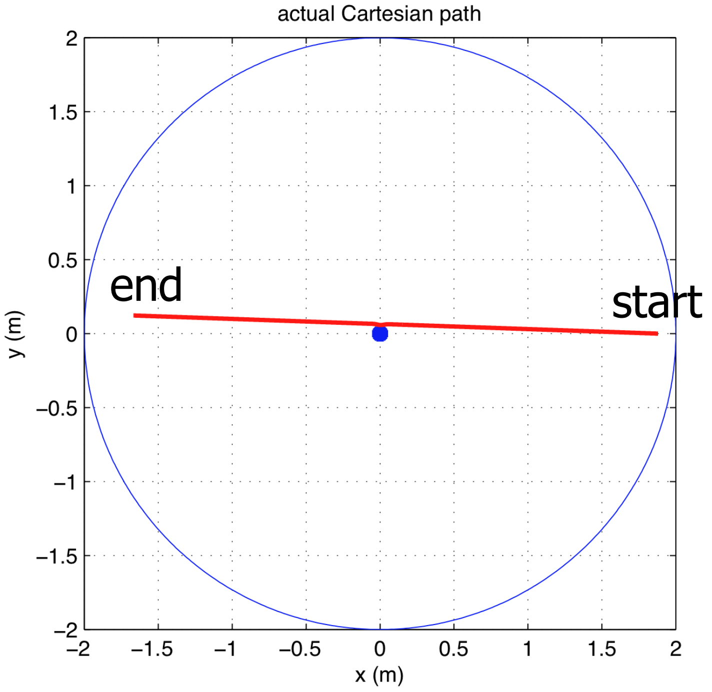
still very small, but increased numerical integration error $(2 \cdot 10^{-9})$

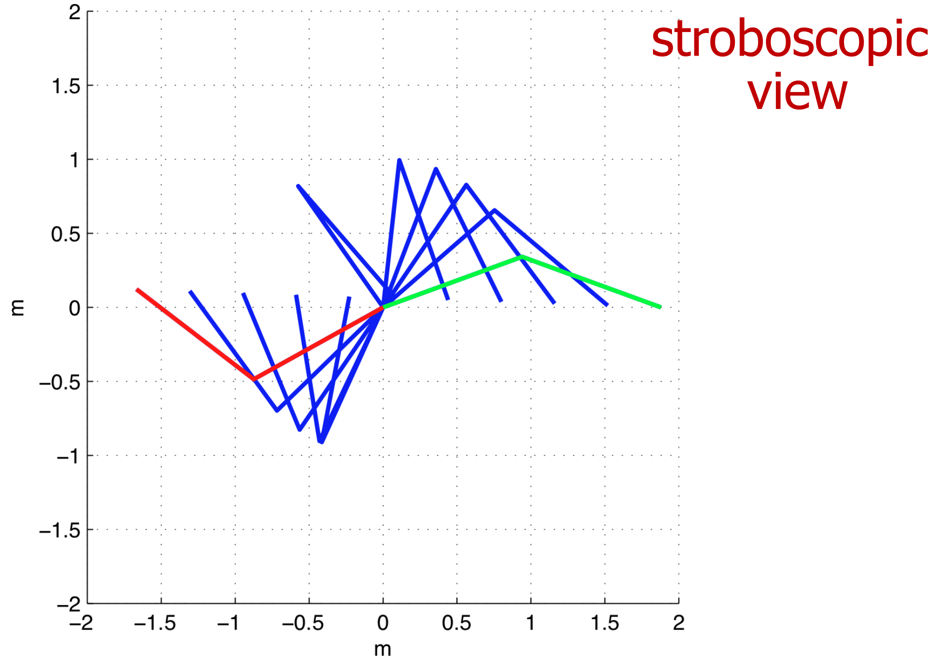
Simulation results

planar 2R robot in straight line Cartesian motion

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

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



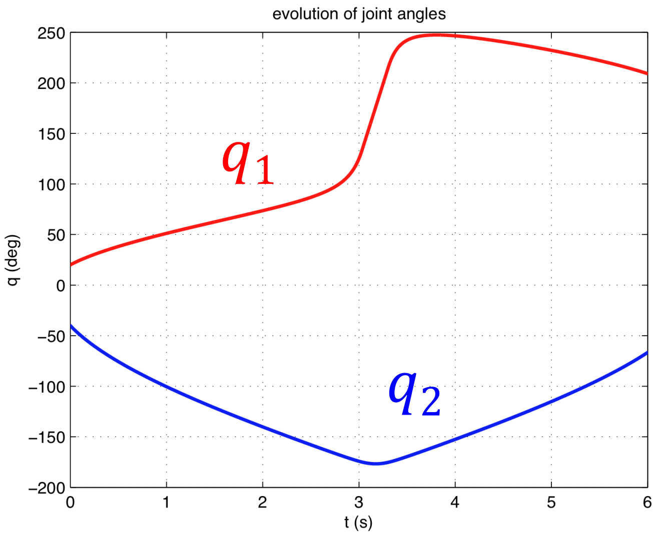


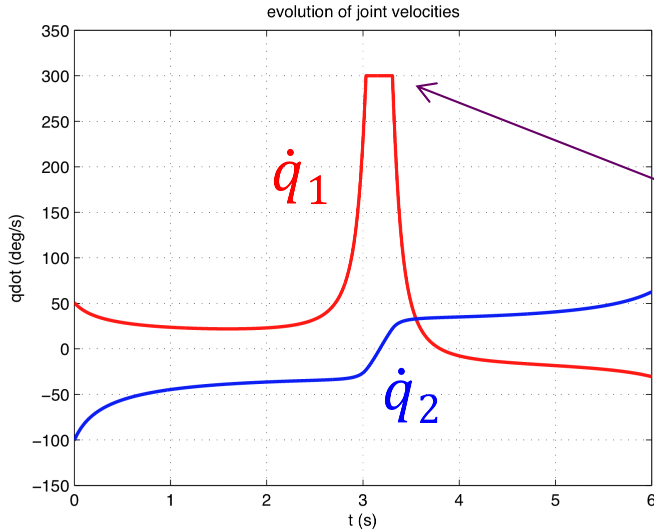
a line from right to left, at $\alpha = 178°$ angle with x-axis,
executed at constant speed $\|v\| = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

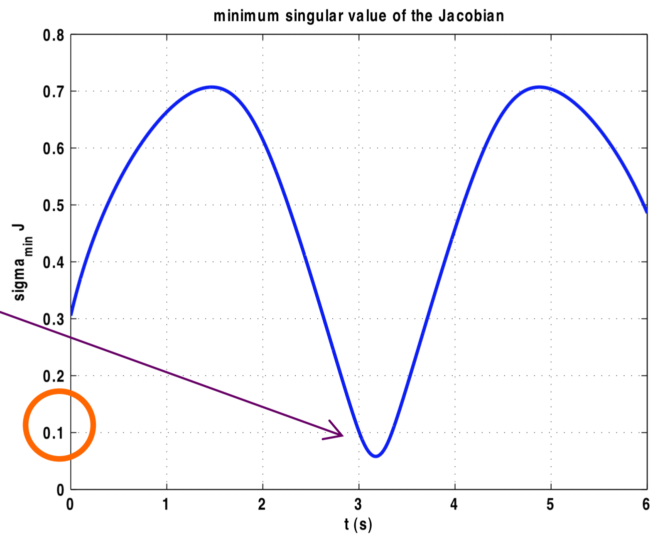
path at
$\alpha = 178°$

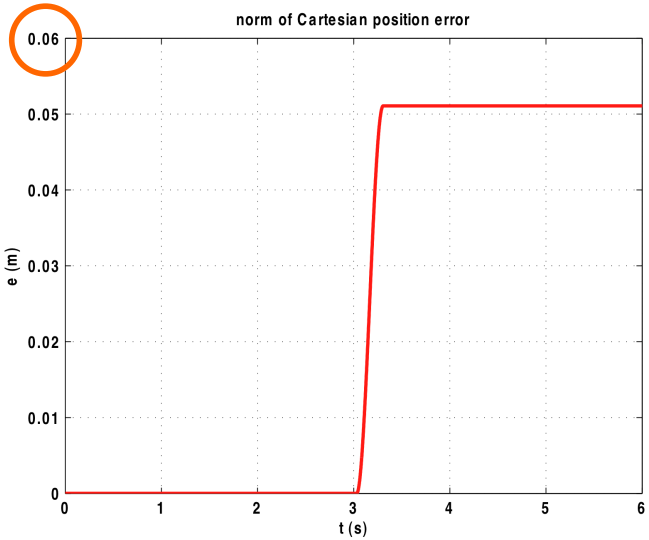




saturated
value
of $\dot{q}_1$

close to
singular
case





actual
position
error!!
(6 cm)

to be recovered
using an
error feedback
control action!

Damped Least Squares (DLS) 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!

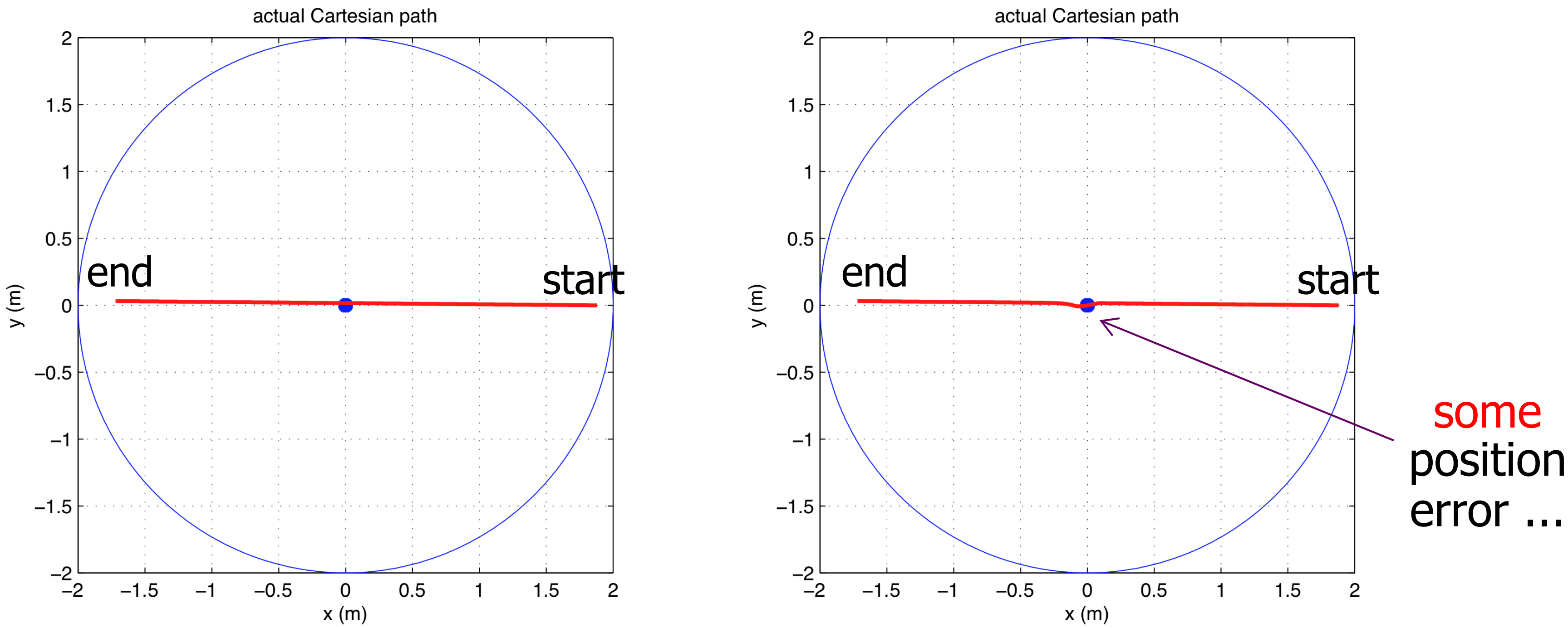
- inversion of differential kinematics as unconstrained optimization problem
- function $H =$ weighted sum of two objectives (norm of joint velocity and error norm on achieved end-effector velocity) to be minimized
- J_{DLS} can be used for both cases: $m = n$ (square) and $m < n$ (redundant)
- $\lambda = 0$ 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 (removing joint velocity saturation)

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

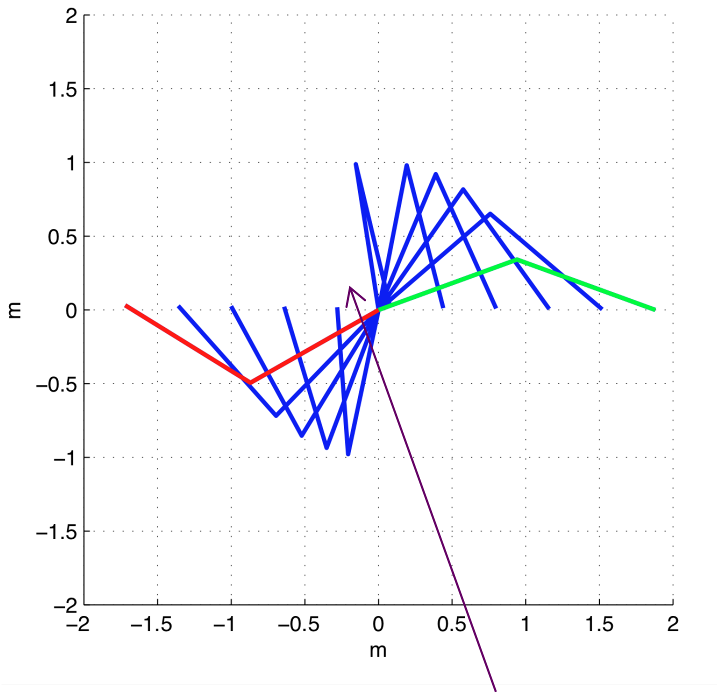


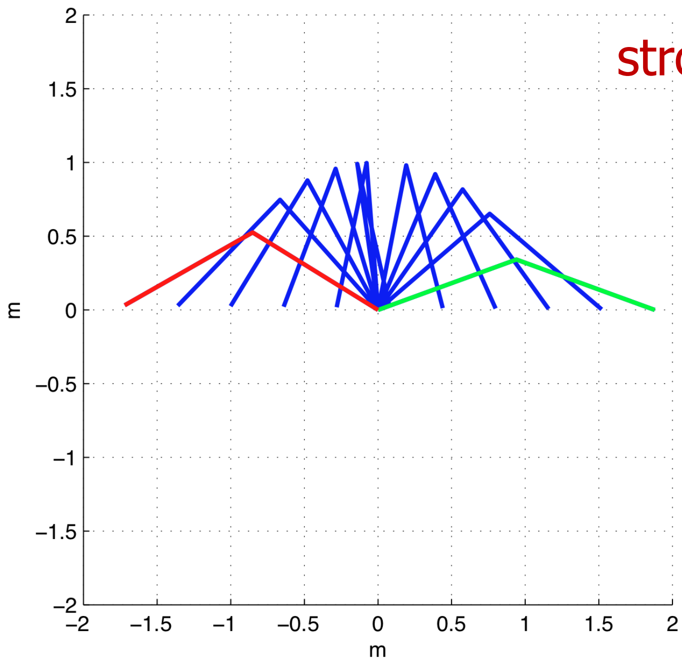
a line from right to left, at $\alpha = 179.5°$ angle with x-axis,
executed at constant speed $\|v\| = 0.6$ m/s for $T = 6$ s

Simulation results

planar 2R robot in straight line Cartesian motion

$\boxed{\dot{q} = J^{-1}(q)v}$ path at $\alpha = 179.5°$ $\boxed{\dot{q} = J_{DLS}(q)v}$





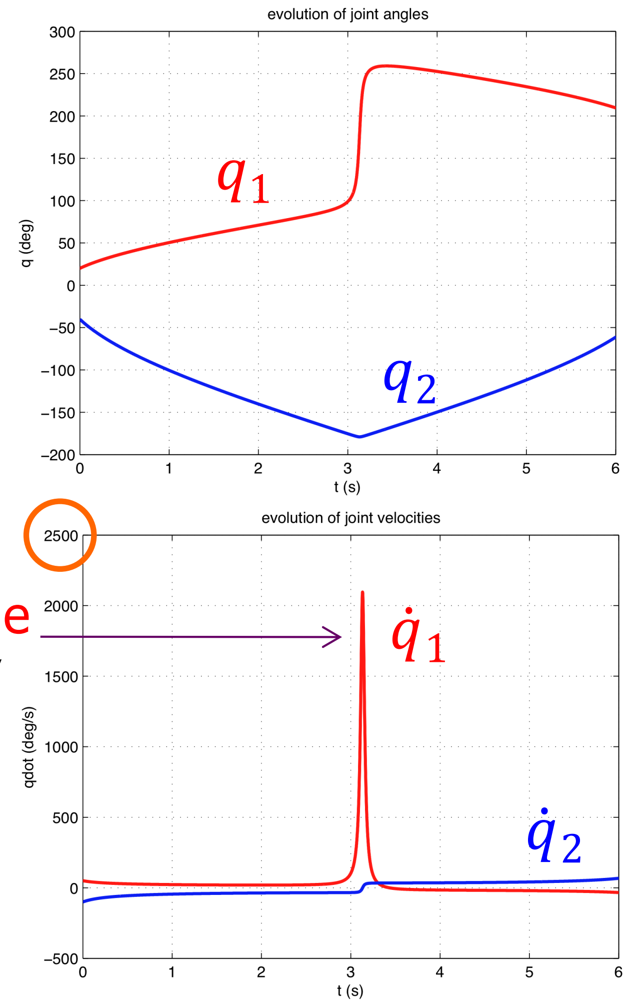
stroboscopic views

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

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

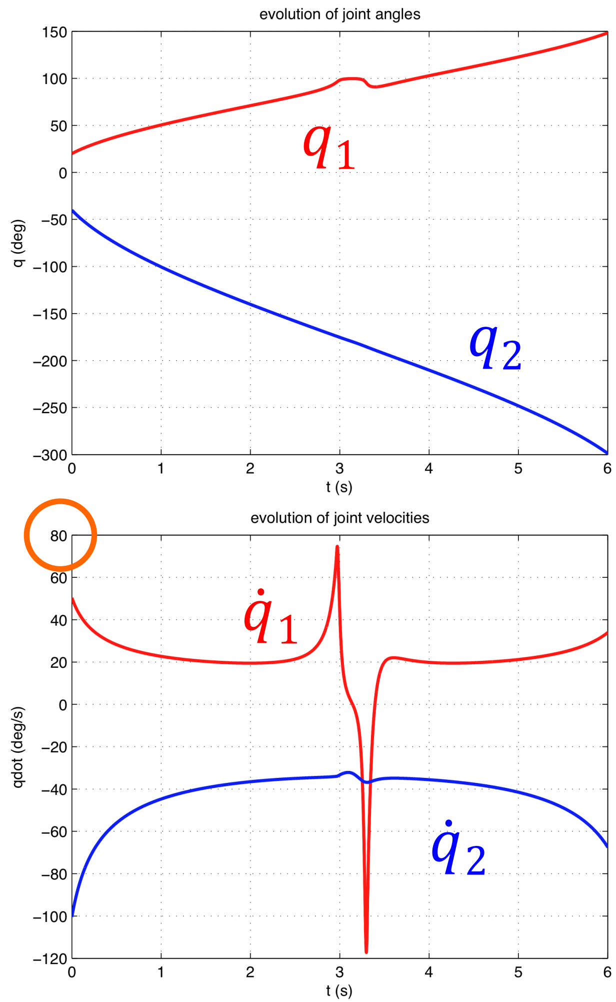
Simulation results
planar 2R robot in straight line Cartesian motion

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



extremely large peak velocity of first joint!!

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

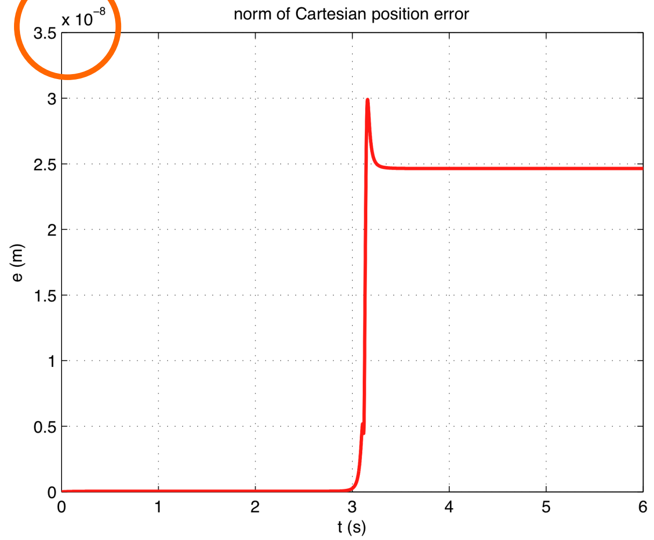


smoother joint motion with limited joint velocities!

Simulation results

planar 2R robot in straight line Cartesian motion

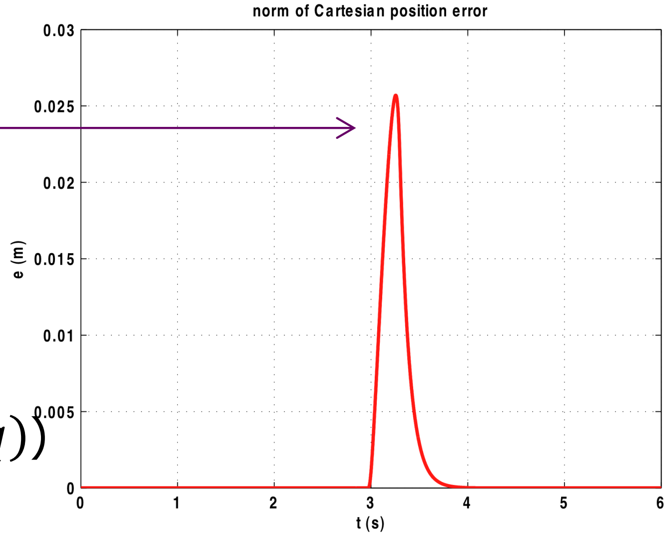
$$\boxed{\dot{q} = J^{-1}(q)v}$$



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

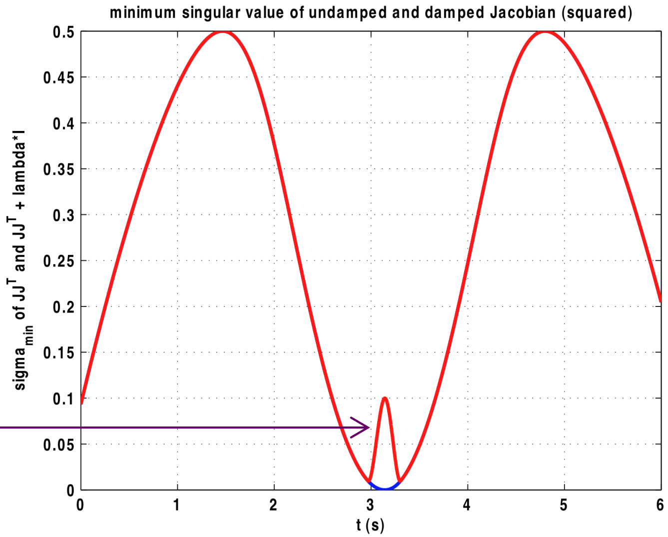
$$\boxed{\dot{q} = J_{DLS}(q)v}$$

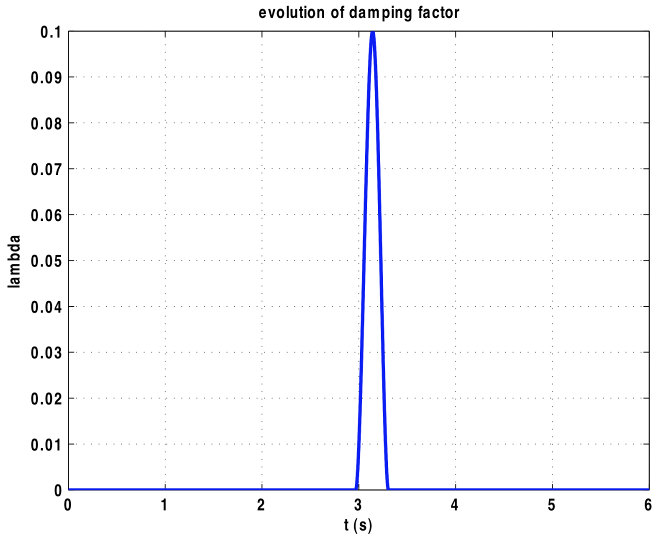
error (25 mm) when crossing the singularity, later recovered by a feedback action ($v \Rightarrow v + K_p e_p$ with $e_p = p_d - p(q)$)



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

$$\boxed{\min_{\dot q} H = \frac{1}{2}\|\dot q\|^2 \quad \text{such that} \quad J\dot q = v} \Leftrightarrow \boxed{\begin{gathered}\min_{\dot q \in S} H = \frac{1}{2}\|\dot q\|^2 \\ S = \left\{ \begin{matrix} \dot q \in R^n : \\ \|J\dot q - v\| \text{ is minimum} \end{matrix} \right\}\end{gathered}}$$

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

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

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

- 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 Singular Value Decomposition* (SVD) of $J = U\Sigma V^{T}$
  - e.g., with the MATLAB function **pinv** (which uses in turn **svd**)

✦ = see Appendix A.7 of new textbook * = see Appendix A.8 of new textbook

Numerical example

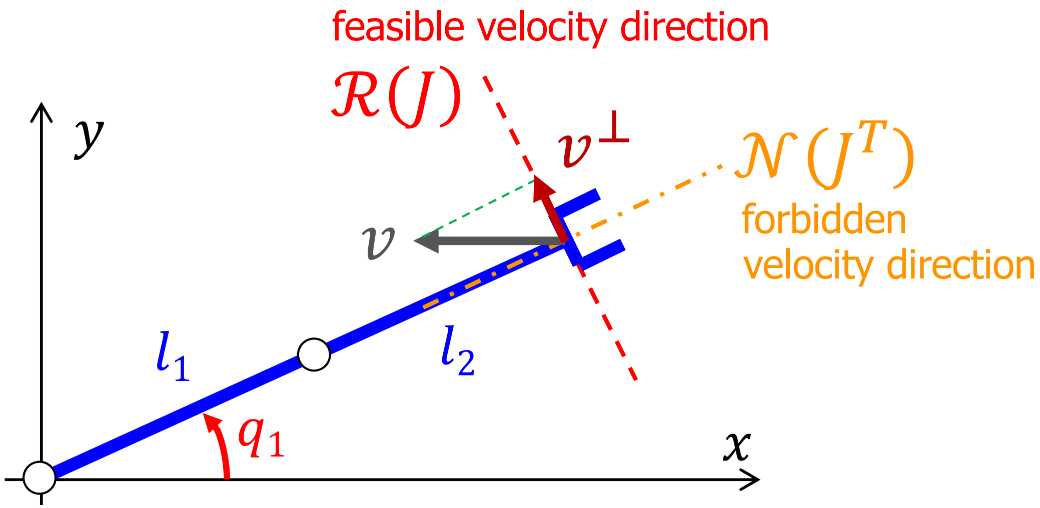
Jacobian of 2R robot with $l_1 = l_2 = 1$ at $q_2 = 0$ (rank $\rho(J) = 1$)

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

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

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

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

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

any joint velocity...

(orthogonal) projection matrix P in the null space $\mathcal{N}(J)$ $\begin{cases} P^T = P \\ P^2 = P \end{cases}$

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

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

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

verification of the actual task velocity that is being obtained

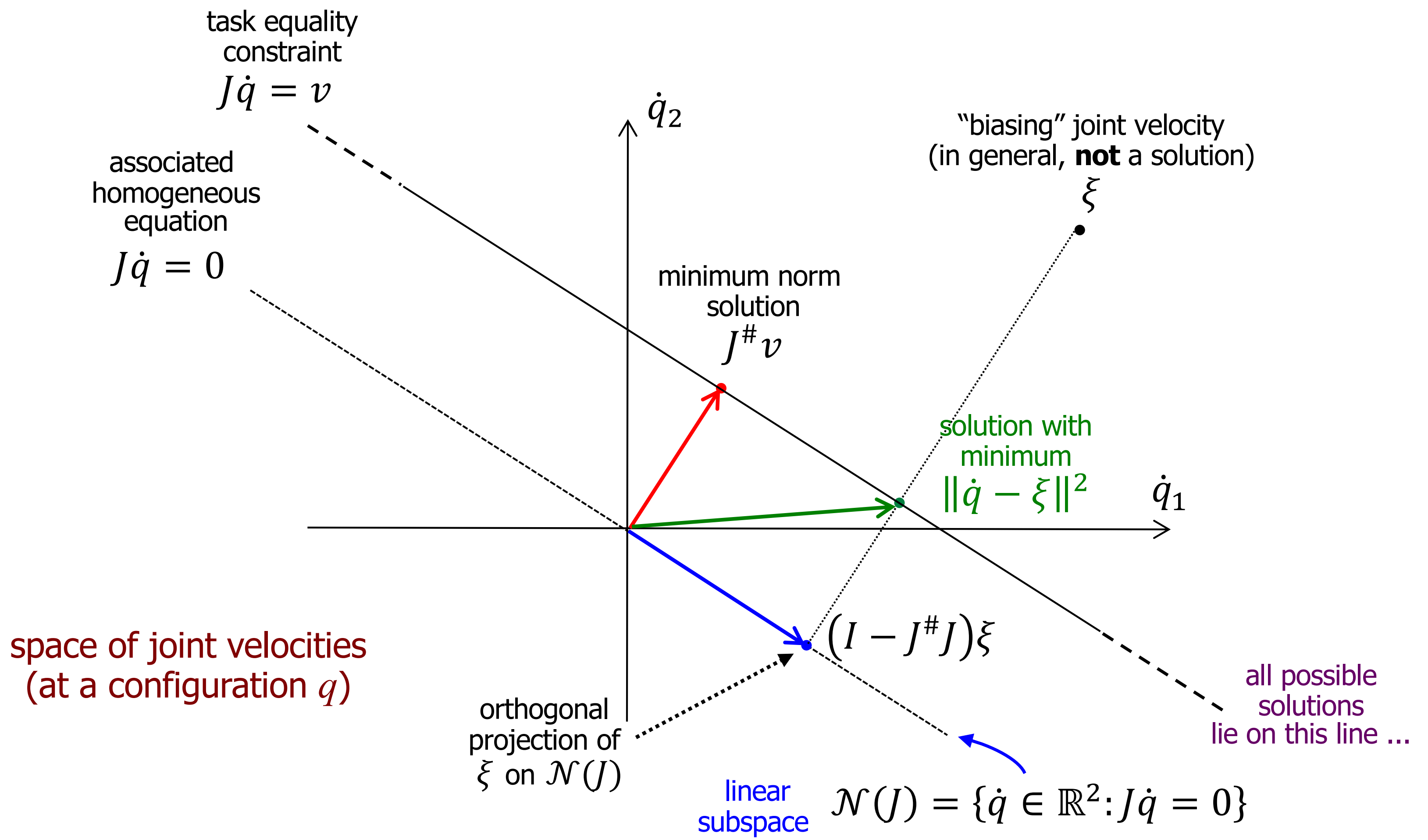
$$v_{actual} = J\dot{q} = J\left(J^{\#}v + \left(I - J^{\#}J\right)\xi\right) = J\,J^{\#}v + \cancel{J\left(I - J^{\#}J\right)\xi} = J\,J^{\#}(Jw) = Jw = v$$

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

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



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

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

task velocity manipulability ellipsoid

if $\rho(J) = m$

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

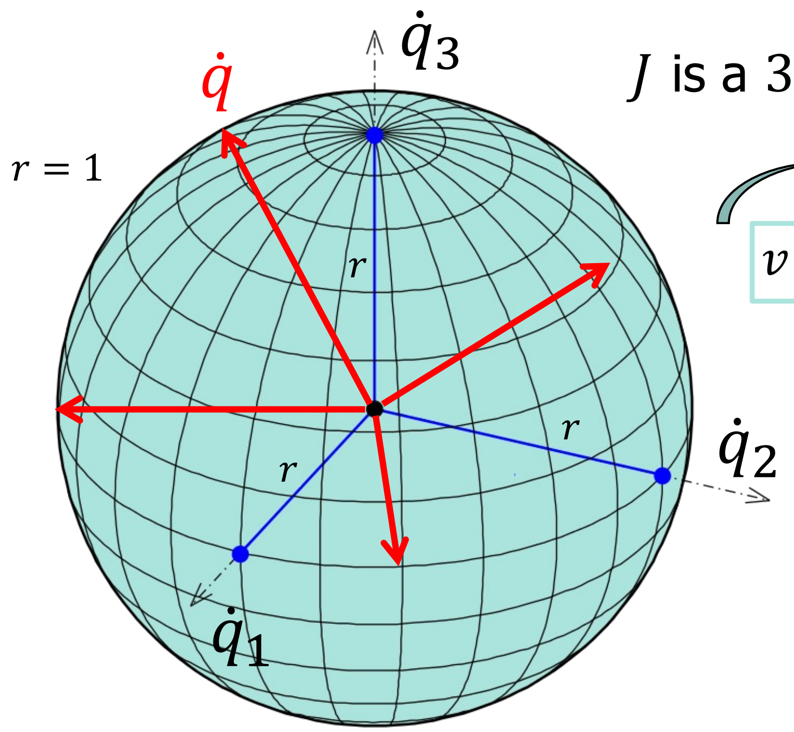
note: the "core" matrix of an ellipsoid equation $v^T A^{-1} v = 1$ is the matrix A!

(Hyper-)Spheres and Ellipsoids

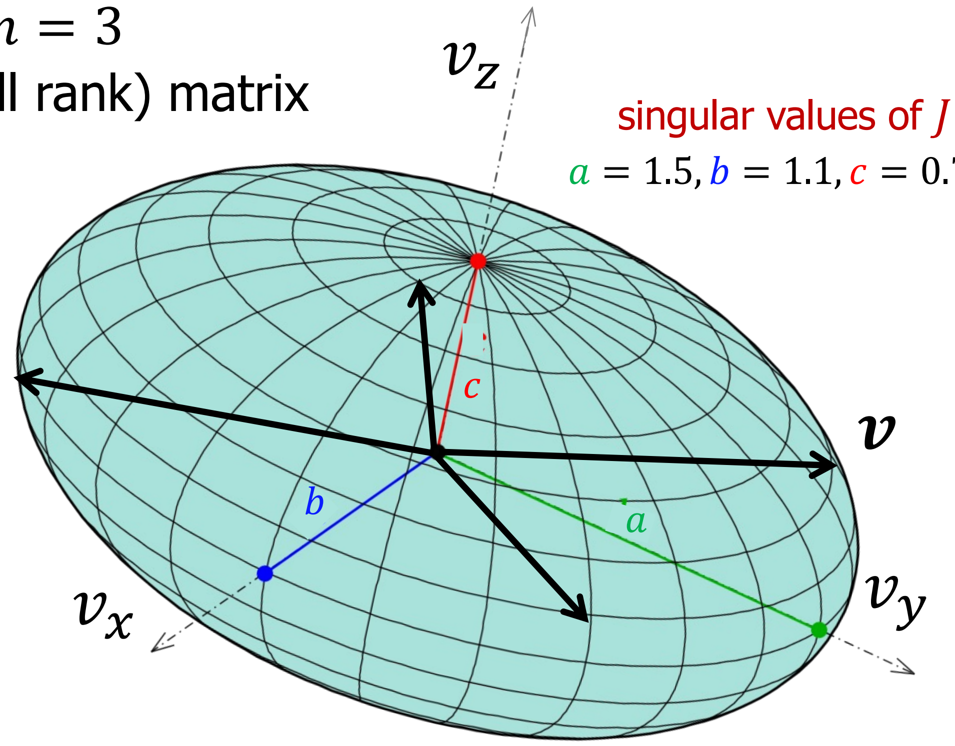
whiteboard …

$m = n = 3$

J is a 3×3 (full rank) matrix



$v = J\dot{q}$



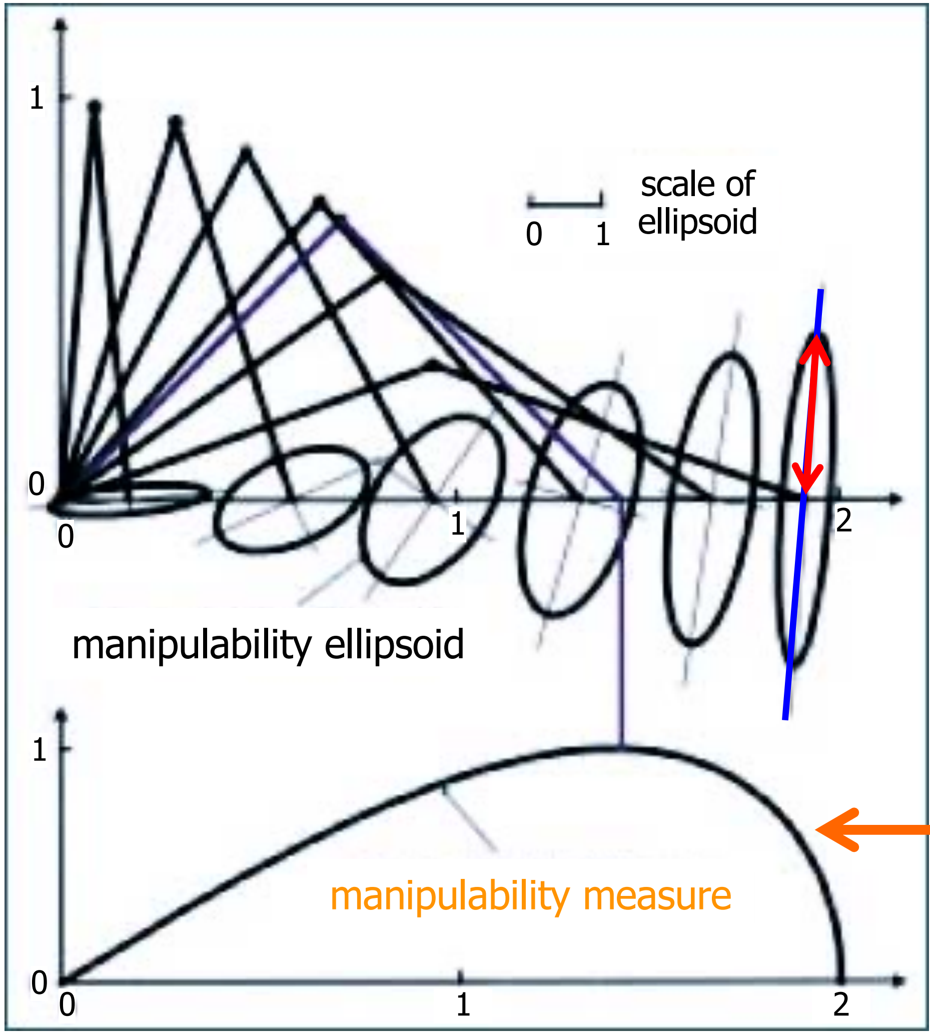
$$\dot{q}_1^2 + \dot{q}_2^2 + \dot{q}_3^2 = \dot{q}^T \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} \dot{q} = 1$$

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

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

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

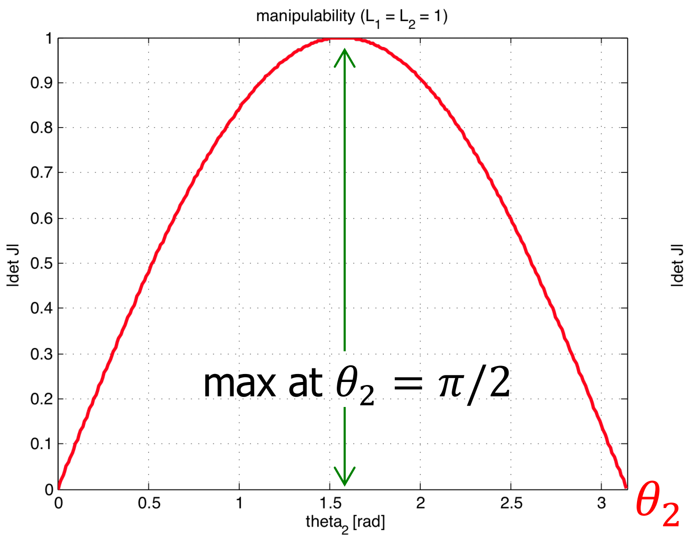
eigenvectors associated to λ_i

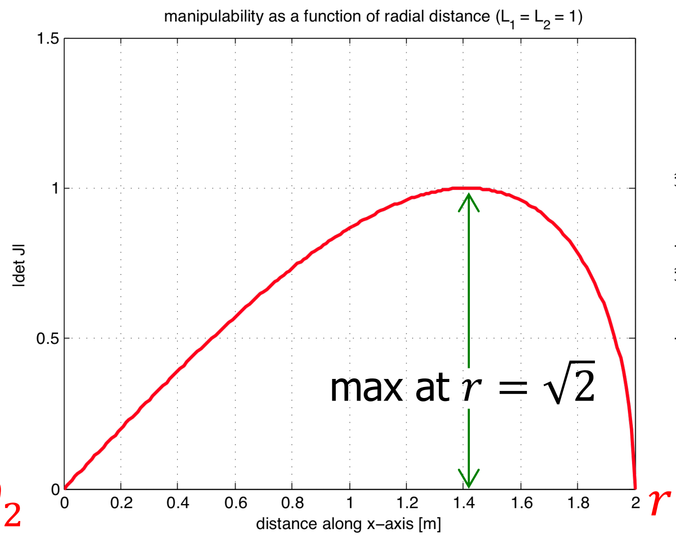
$$w = \sqrt{det\,(J\,J^T)} = \prod_{i=1}^{m} \sigma_i \geq 0$$

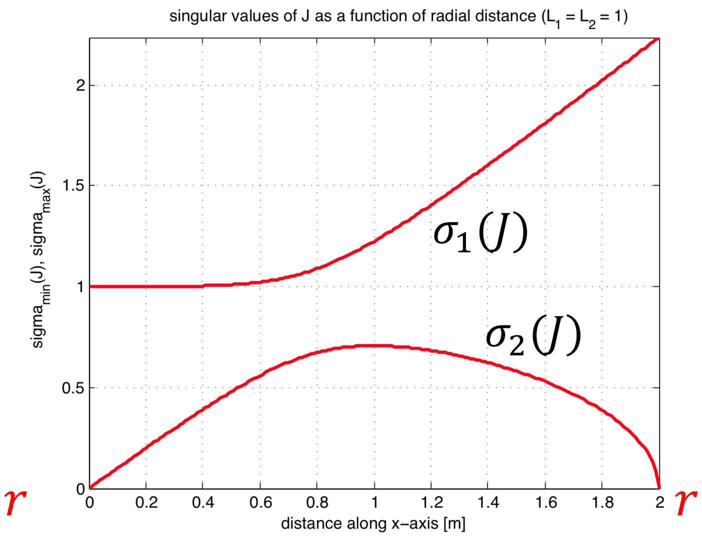
proportional to the volume of the ellipsoid (for $m = 2$, to its area)

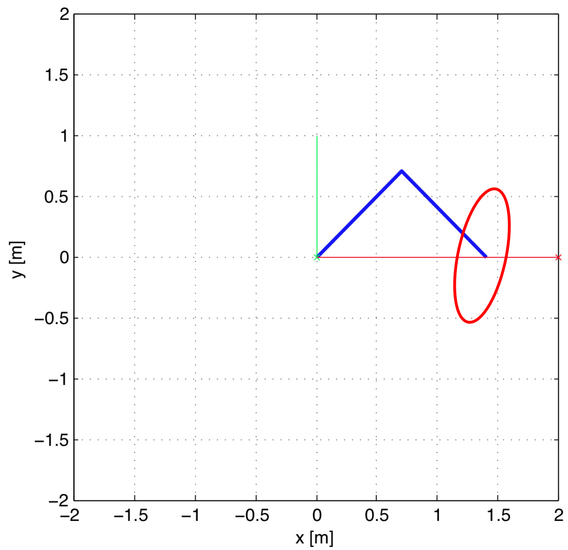
Manipulability measure

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







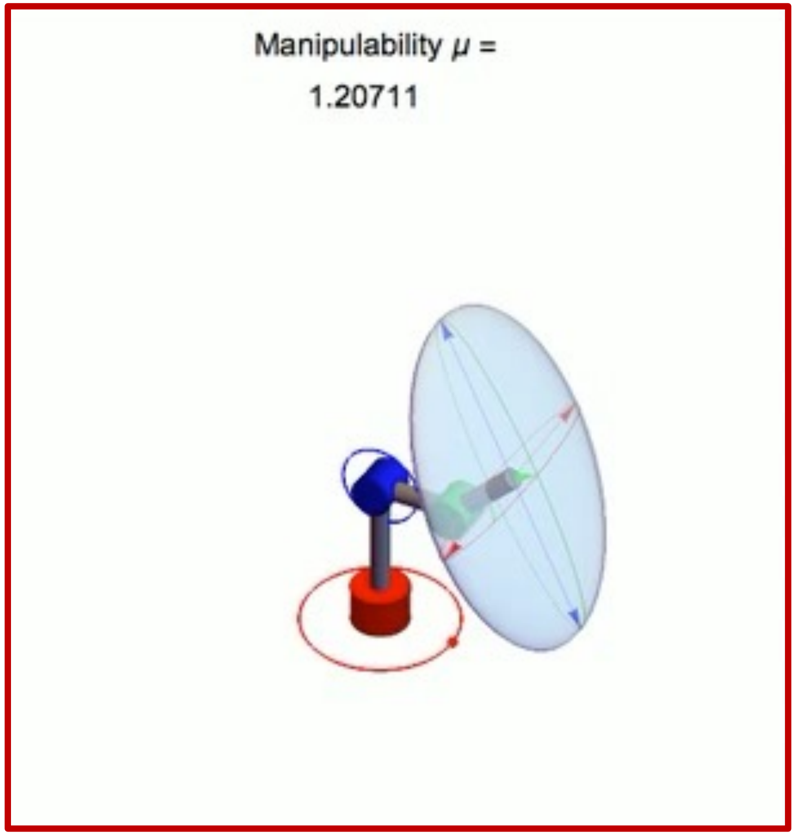


best posture for manipulation (similar to a human arm!)

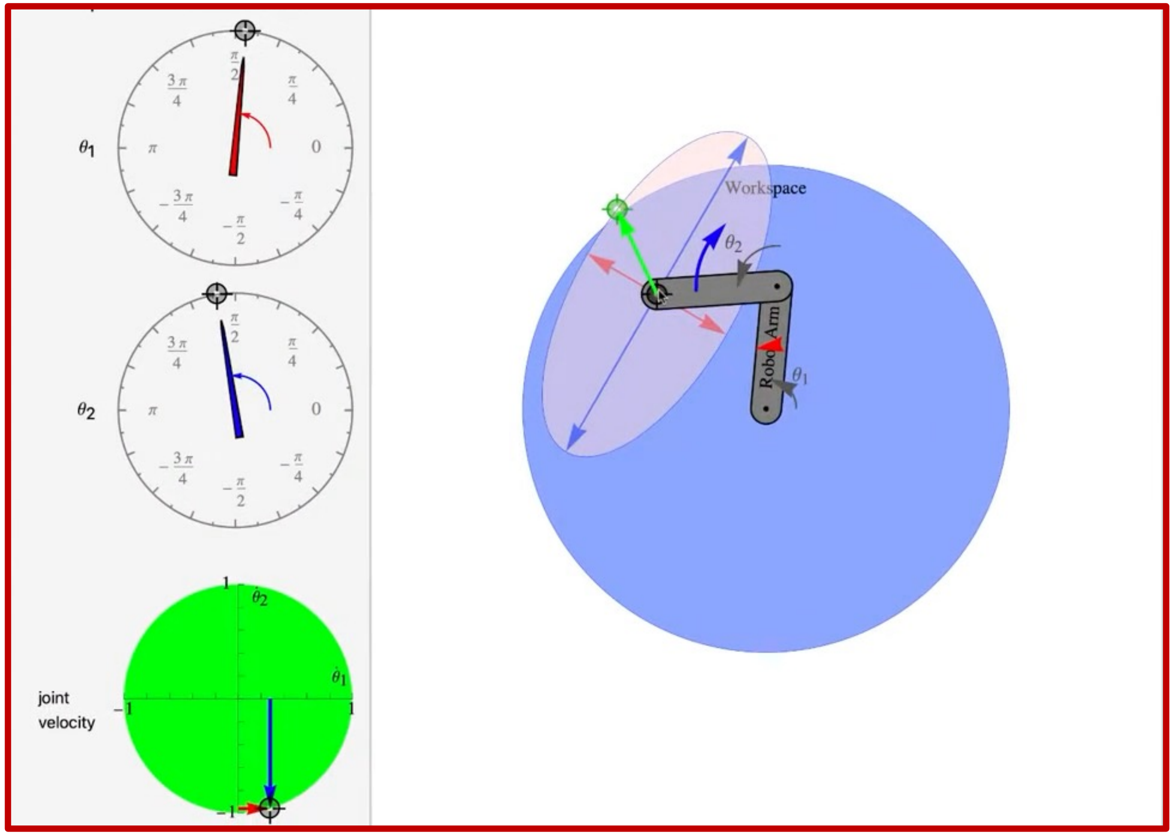
no full isotropy (i.e., a circle) is obtained for this robot since it is never $\sigma_1 = \sigma_2$

Manipulability ellipsoid
in action

spatial 3R robot (elbow-type)



planar 2R robot (any link length) video



https://youtu.be/0HDyR2pu6CY?si=cjPFad91srGiHYKK

credits to Aaron T. Becker (U Houston)

more from Mathematica™ in the Cloud
https://demonstrations.wolfram.com/ManipulabilityEllipsoidOfARobotArm

Higher-order differential inversion

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

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

- jerk level: given $q, \dot{q}, \ddot{q}$, for a desired $\dddot{r}$

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

- (pseudo-)inverse of the Jacobian is always the leading term
- smoother joint motions are expected when commanding joint velocities to the robot by integrating higher-order quantities
  - at least, because of the existence of higher-order time derivatives $\dot{r}, \ddot{r}, \dots$