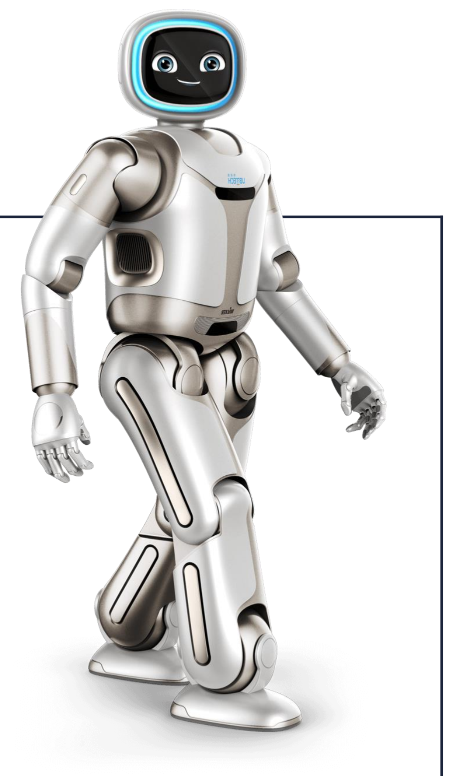
Advanced Robotics

ENGG5402 Spring 2023

Fei Chen



Topics:

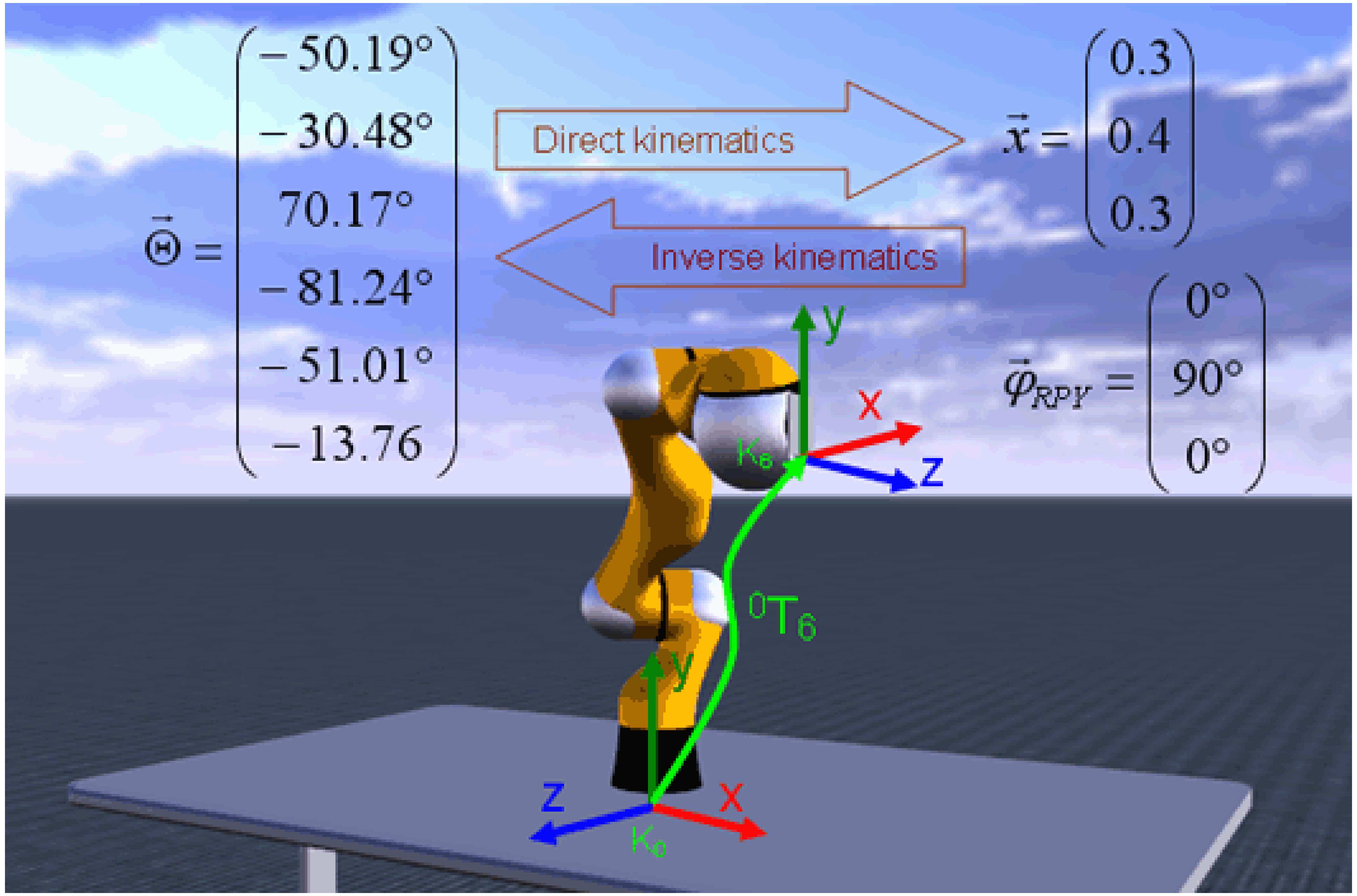
- Inverse kinematics

Readings:

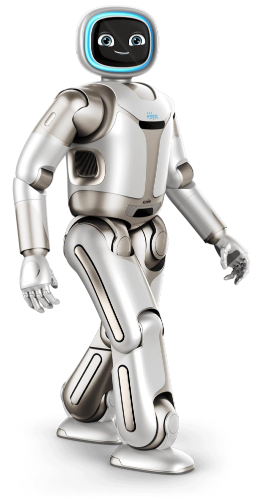
- Siciliano: Sec. 2.12, 3.7

Inverse kinematics

Inverse kinematics (what are we looking for?)



direct kinematics is always unique;
how about inverse kinematics for this 6R robot?



Inverse kinematics problem

- given a desired end-effector pose (position + orientation), find the values of the joint variables q that will realize it
- a synthesis problem, with input data in the form

$$T = \begin{bmatrix} R & p \\ 0^T & 1 \end{bmatrix} = {}^0A_n(q)$$

classical formulation:
inverse kinematics for a given end-effector pose T

$r = f_r(q)$, for a task function

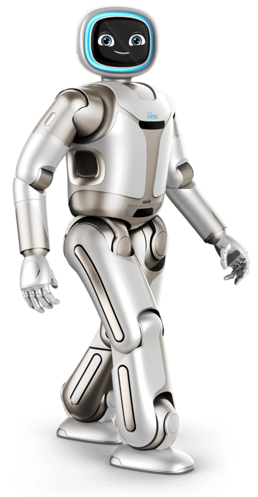
generalized formulation:
inverse kinematics for a given value r of task variables

- a typical nonlinear problem
  - existence of a solution (workspace definition)
  - uniqueness/multiplicity of solutions ($r \in \mathbb{R}^m$, $q \in \mathbb{R}^n$)
  - solution methods

Solvability and Robot Workspace

Solvability and robot workspace (for tasks related to a desired end-effector Cartesian pose

- primary workspace WS_1: set of all positions p that can be reached with at least one orientation (ϕ or R)
  - out of WS_1 there is no solution to the problem
  - if $p \in WS_1$, there is a suitable ϕ (or R) for which a solution exists
- secondary (or dexterous) workspace WS_2: set of positions p that can be reached with any orientation (among those feasible for the robot direct kinematics)
  - if $p \in WS_2$, there exists a solution for any feasible ϕ (or R)
- $WS_2 \subseteq WS_1$

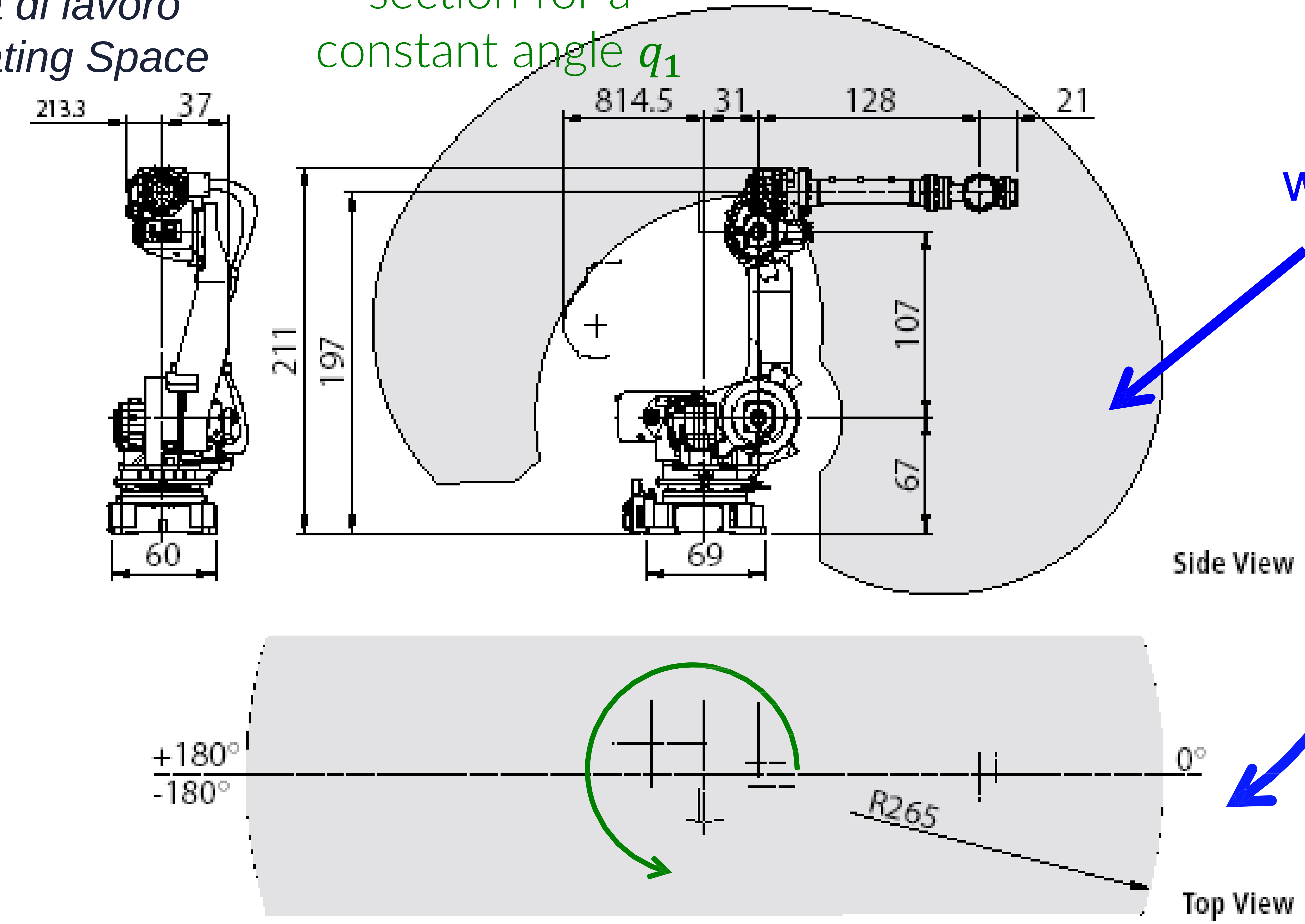


Workspace

Workspace of Fanuc R-2000i/165F

Area di lavoro
Operating Space

section for a constant angle q_1

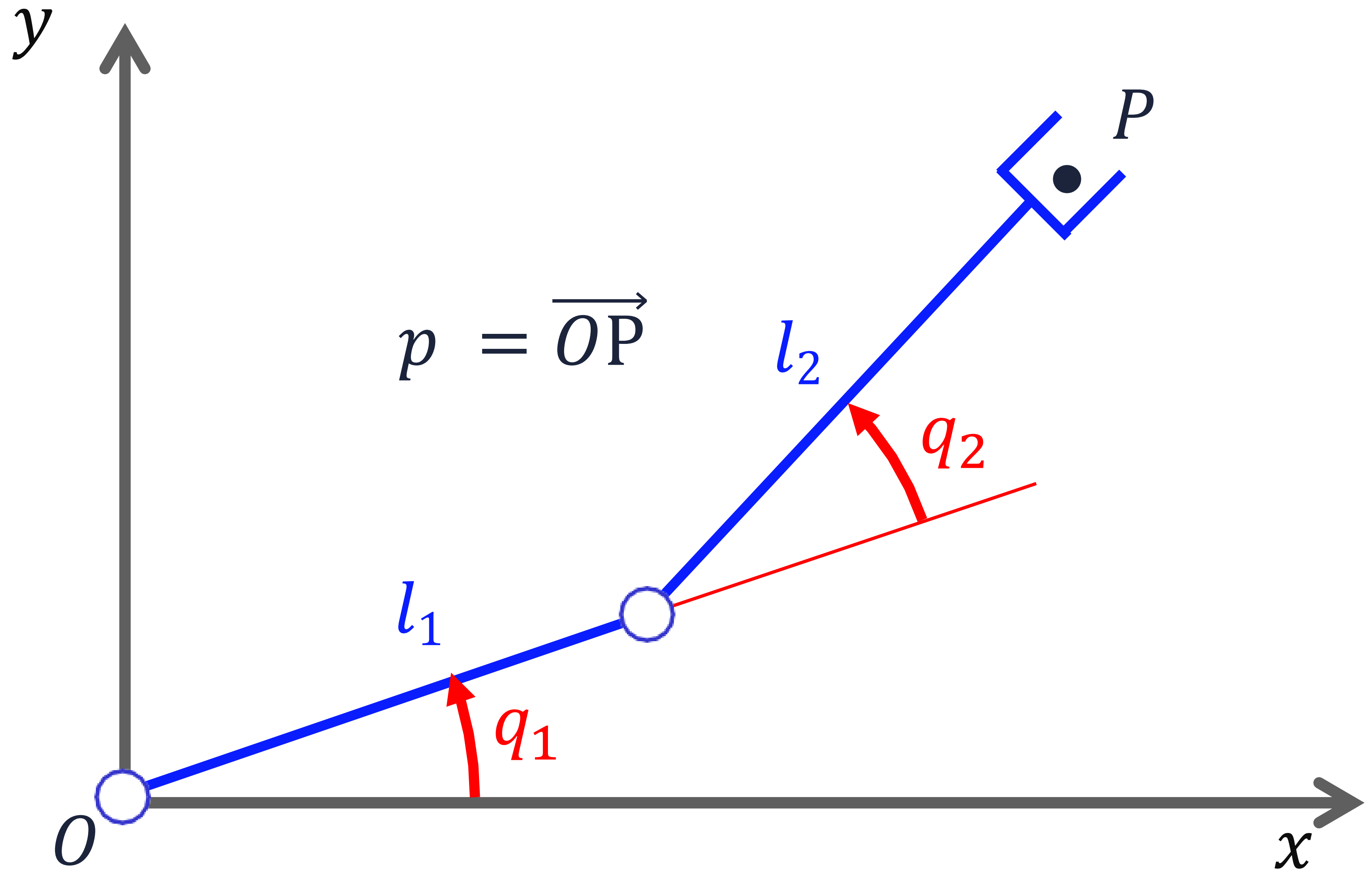


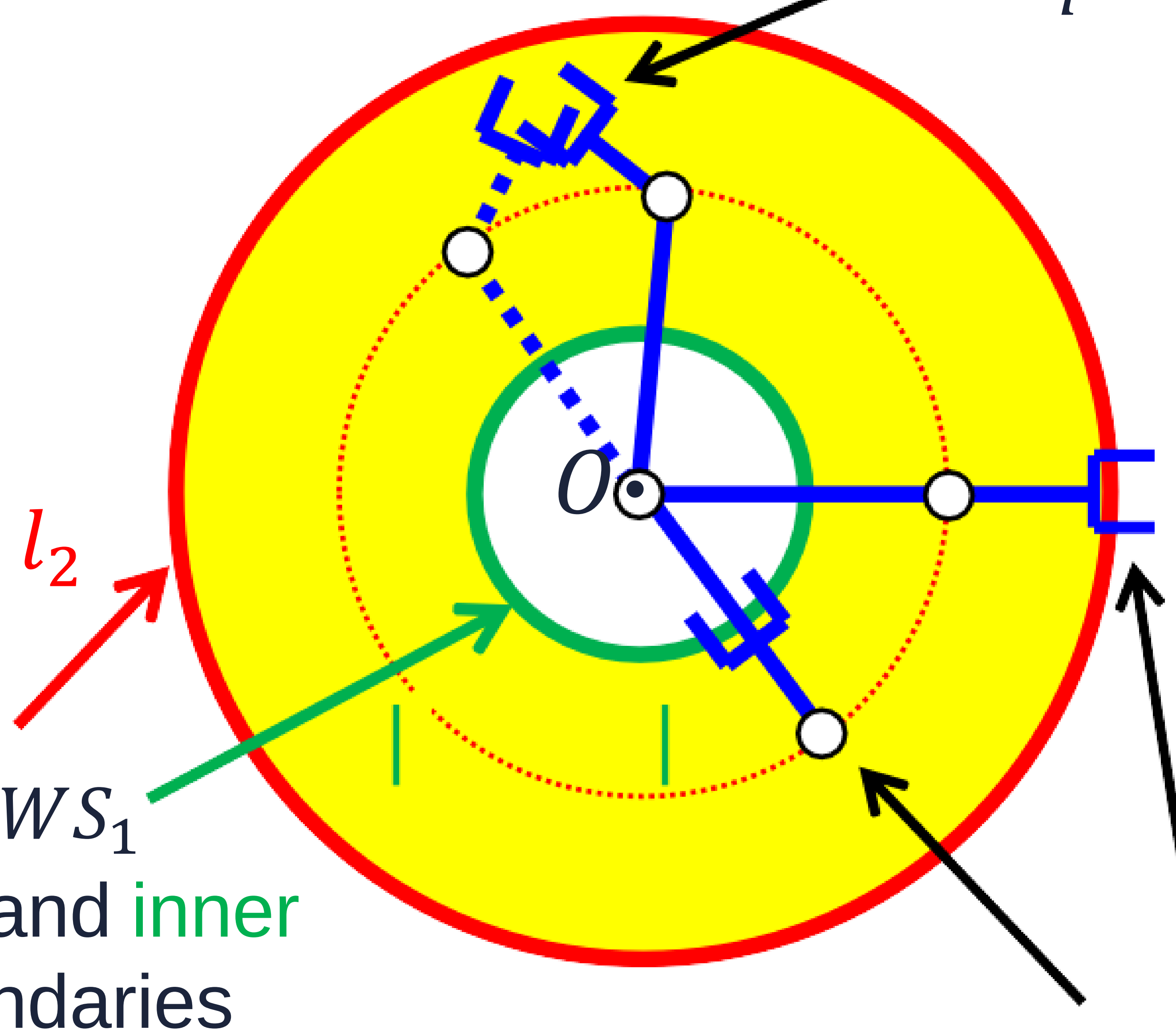
$WS_1 \subset \mathbb{R}^3$
($\approx WS_2$ for spherical wrist without joint limits)

rotating the base joint angle q_1

Workspace

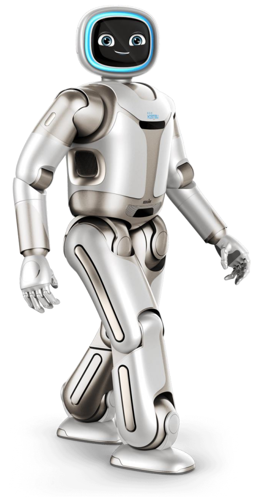
Workspace of a planar 2R arm





- if $l_1 \neq l_2$
  $WS_1 = \{p \in \mathbb{R}^2 : |l_1 - l_2| \leq \|p\| \leq l_1 + l_2\} \subset \mathbb{R}^2$
  $WS_2 = \emptyset$

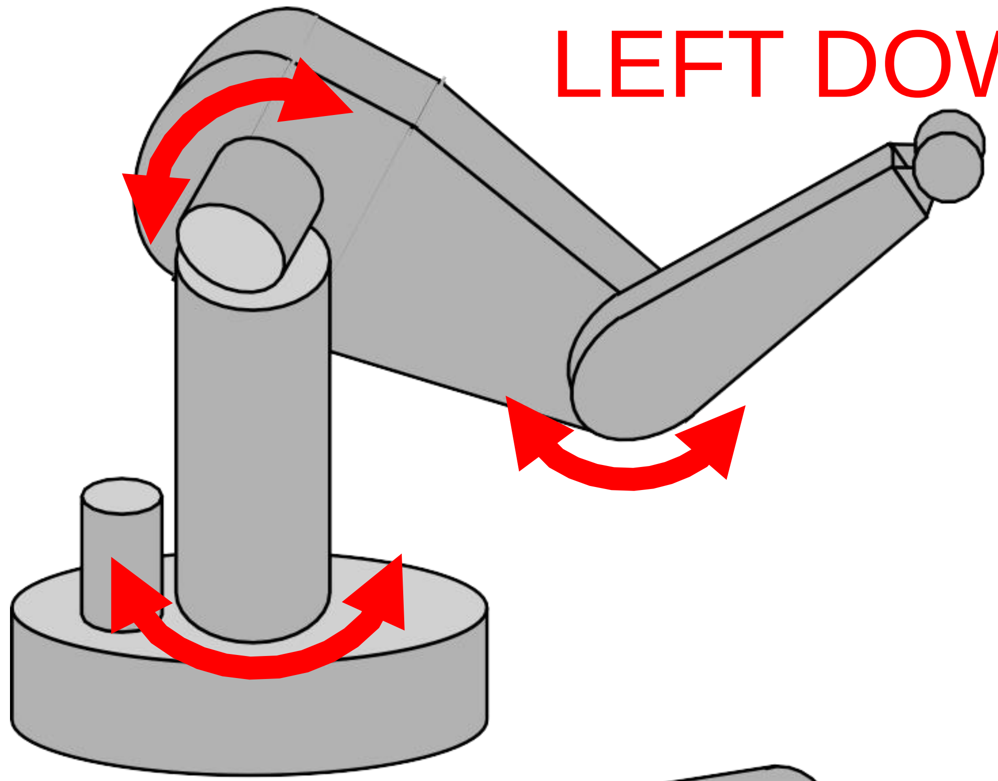
- if $l_1 = l_2 = l$
  $WS_1 = \{p \in \mathbb{R}^2 : \|p\| \leq 2l\} \subset \mathbb{R}^2$
  $WS_2 = \{p = 0\}$ (all feasible orientations at the origin!... an infinite number)



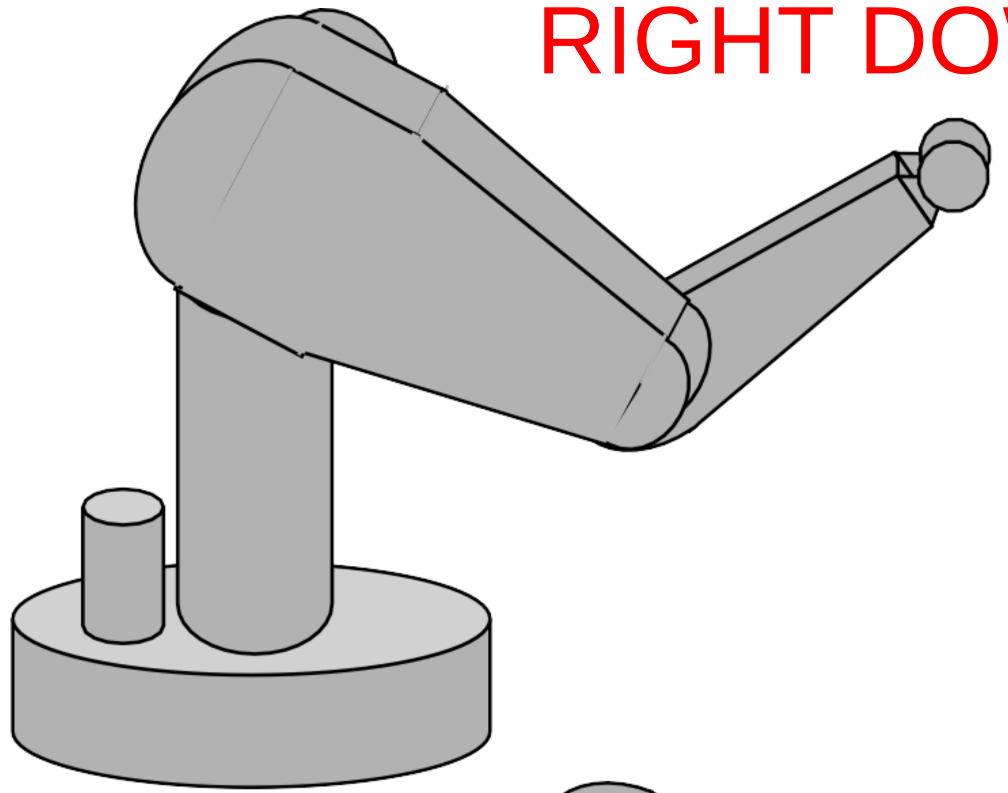
Workspace

Wrist position and E-E pose
(inverse solution for an articulated 6R robot)

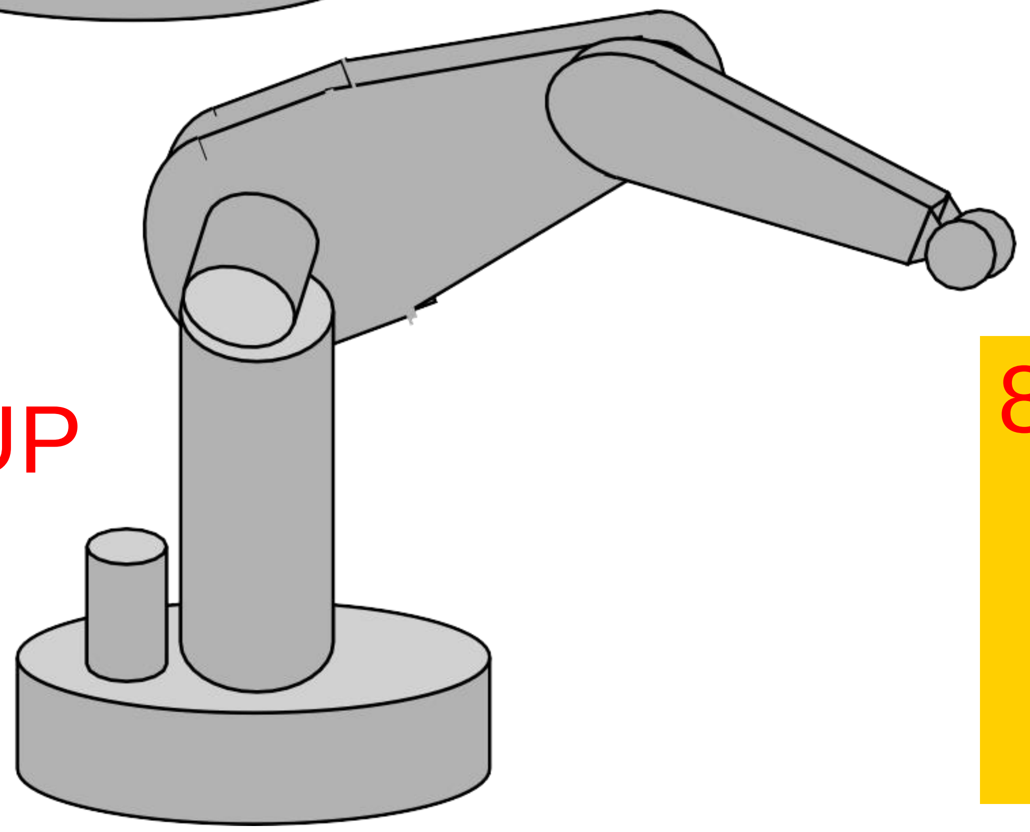
LEFT DOWN



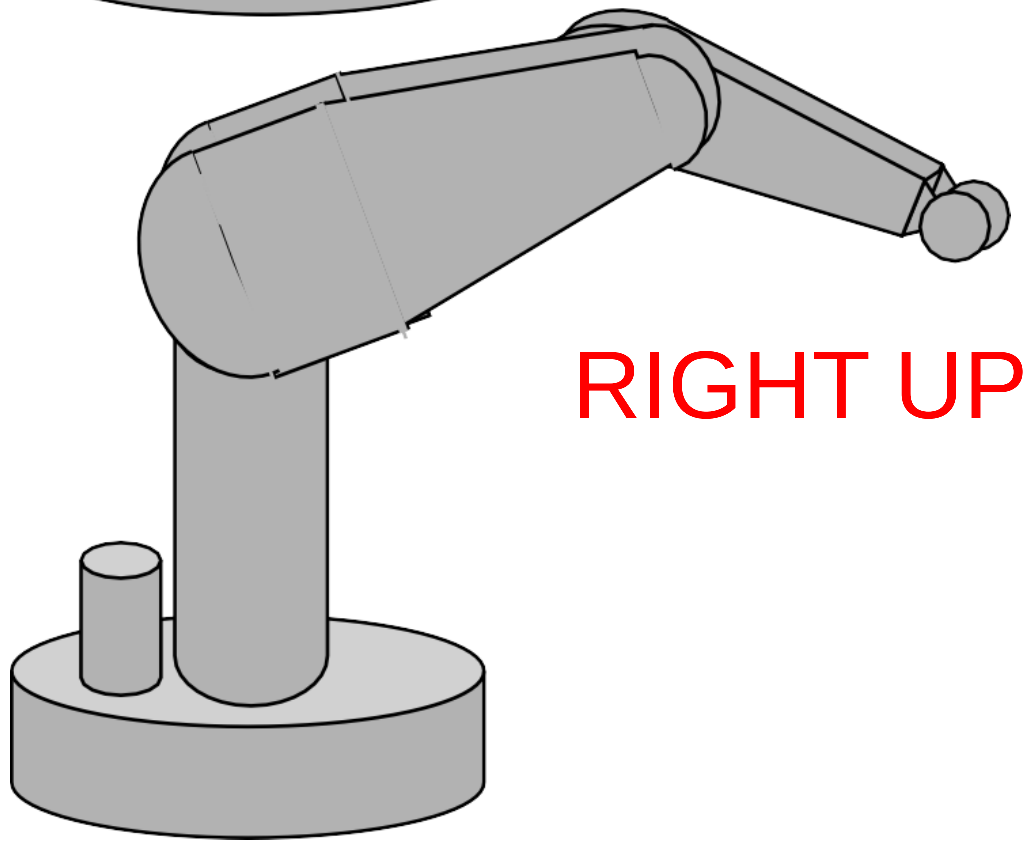
RIGHT DOWN



4 inverse solutions
out of singularities
(for the position of
the wrist center only)

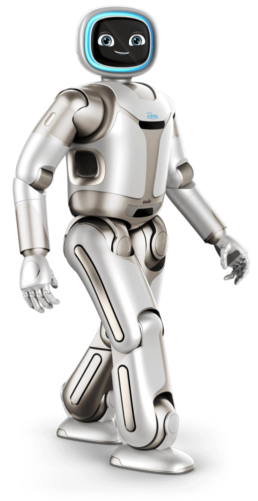


LEFT UP



RIGHT UP

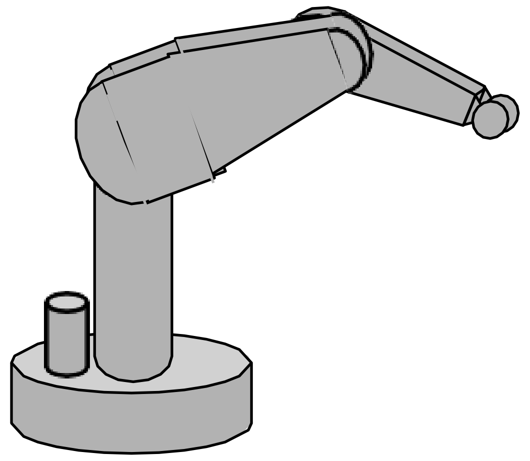
8 inverse solutions considering
the complete E-E pose
(spherical wrist: 2 alternative
solutions for the last 3 joints)



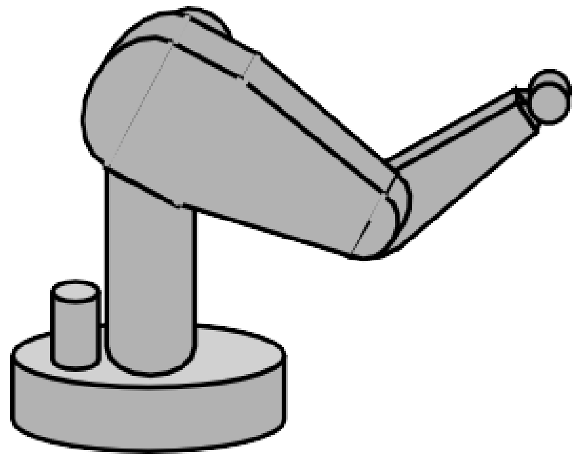
Workspace

Counting/visualizing the 8 solutions
(of the inverse kinematics for a Unimation Puma 560)

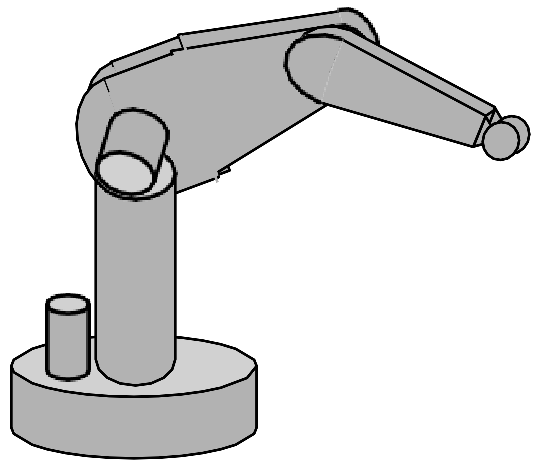
RIGHT UP



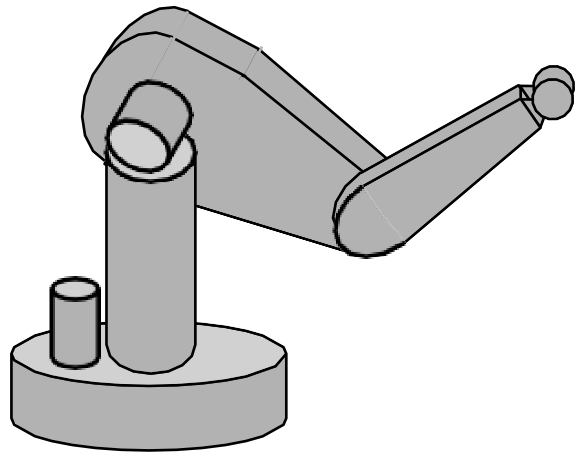
RIGHT DOWN

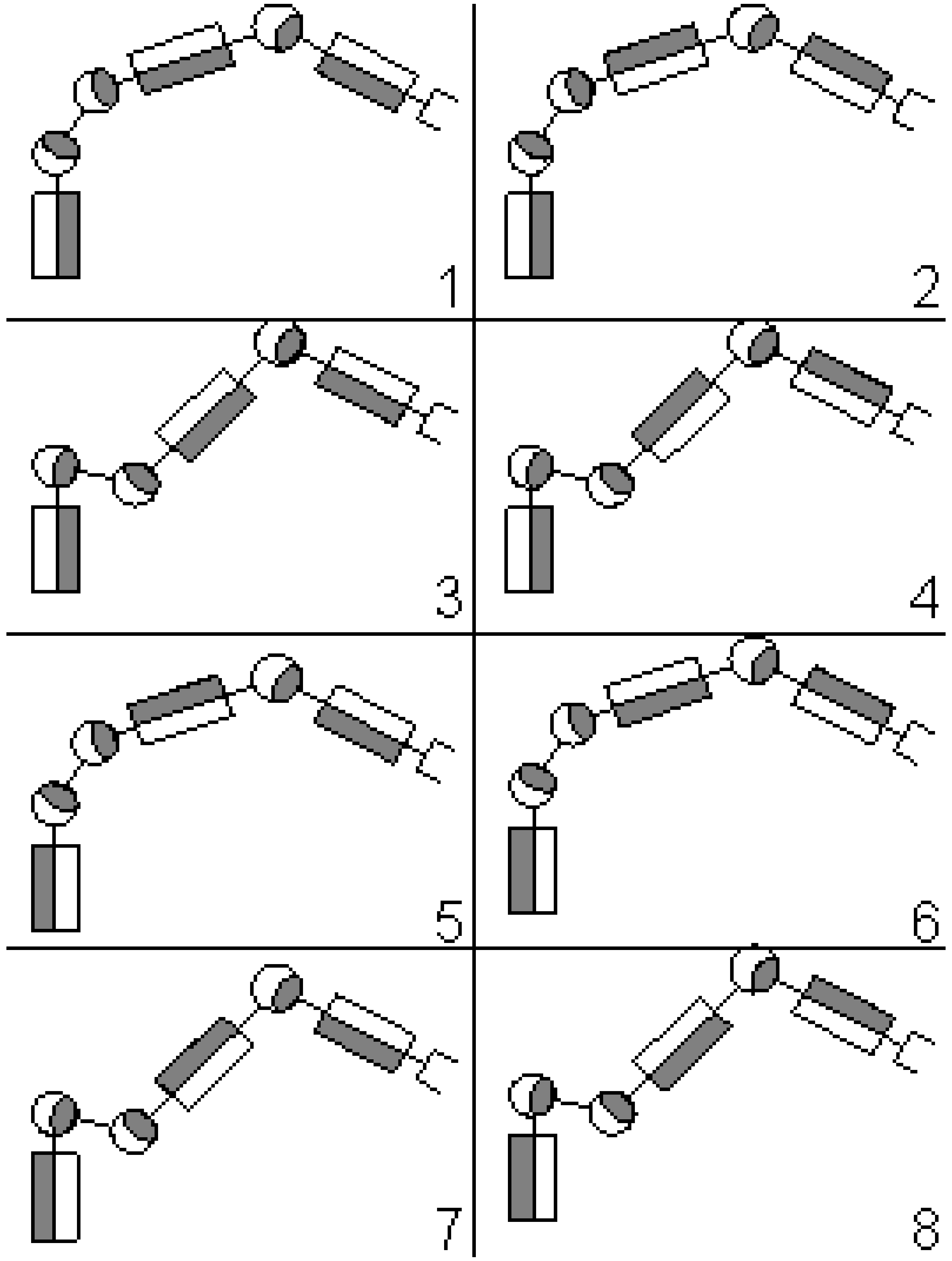


LEFT UP



LEFT DOWN





Kinematic Solutions

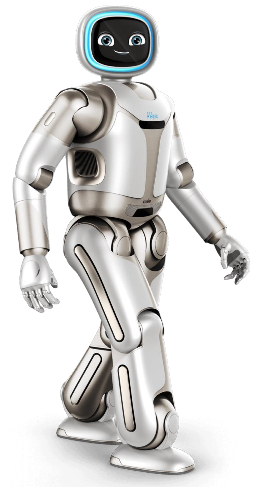
kinematic solutions of UR10
(6-dof Universal Robot UR10, with non-spherical wrist)

desired pose

$$p = \begin{pmatrix} -0.2373 \\ -0.0832 \\ 1.3224 \end{pmatrix} [m] \qquad R = \begin{pmatrix} \sqrt{3}/2 & 0.5 & 0 \\ -0.5 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

home configuration at start

$$q = (0 \quad -\pi/2 \quad 0 \quad -\pi/2 \quad 0 \quad 0)^T [rad]$$



Kinematic Solutions

8 inverse kinematic solutions of UR10

shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 1.0472 \\ -1.2388 \\ -0.7376 \\ -2.6951 \\ -1.5708 \\ 3.1416 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

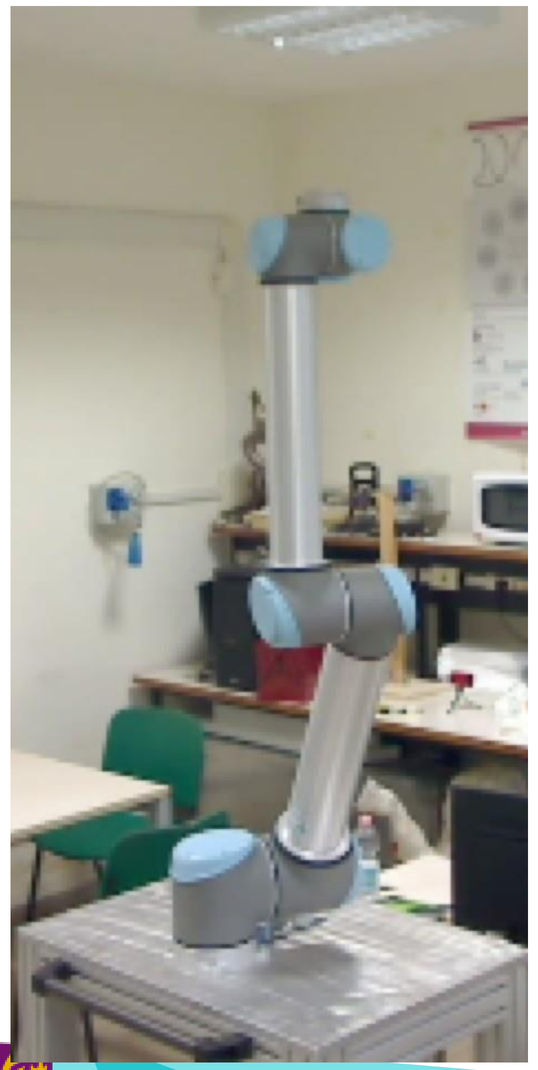
$$q = \begin{pmatrix} 1.0472 \\ -1.9941 \\ 0.7376 \\ 2.8273 \\ -1.5708 \\ 3.1416 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 1.0472 \\ -1.5894 \\ -0.5236 \\ 0.5422 \\ 1.5708 \\ 0 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 1.0472 \\ -2.0944 \\ 0.5236 \\ 0 \\ 1.5708 \\ 0 \end{pmatrix}$$



shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 2.7686 \\ -1.0472 \\ -0.5236 \\ 3.1416 \\ -1.5708 \\ 1.4202 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

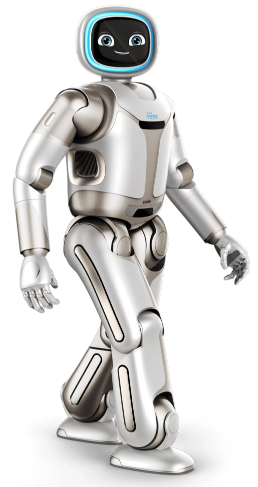
$$q = \begin{pmatrix} 2.7686 \\ -1.5522 \\ 0.5236 \\ 2.5994 \\ -1.5708 \\ 1.4202 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 2.7686 \\ -1.1475 \\ -0.7376 \\ 0.3143 \\ -1.5708 \\ -1.7214 \end{pmatrix}$$

shoulder Right
wrist Down
elbow Up

$$q = \begin{pmatrix} 2.7686 \\ -1.8583 \\ 0.7376 \\ -0.4501 \\ -1.5708 \\ -1.7214 \end{pmatrix}$$



Kinematic Solutions

Multiplicity of solutions (few examples)

- E-E positioning ($m = 2$) of a planar $2R$ robot
  - 2 regular solutions in int(WS_1)
  - 1 solution on ∂WS_1
  - for $l_1 = l_2$: ∞ solutions in WS_1

  } singular solutions

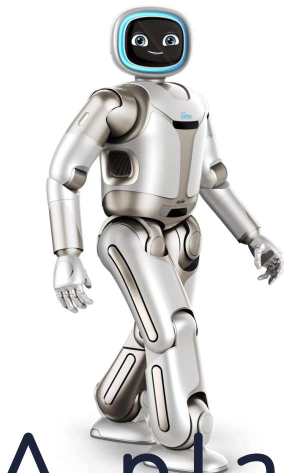
- E-E positioning ($m = 3$) of an elbow-type spatial 3R robot
  - 4 regular solutions in WS_1)with singular cases yet to be investigated ...)
- Spatial $6R$ robot arms
  - ≤ 16 distinct solutions, out of singularities: this "upper bound" of solutions was shown to be attained by a particular instance of "orthogonal" robot, i.e., with twist angles $\alpha_i = 0$ or $\pm\,\pi/2$ $(\forall i)$

Self-reading analysis based on algebraic transformations of robot kinematics

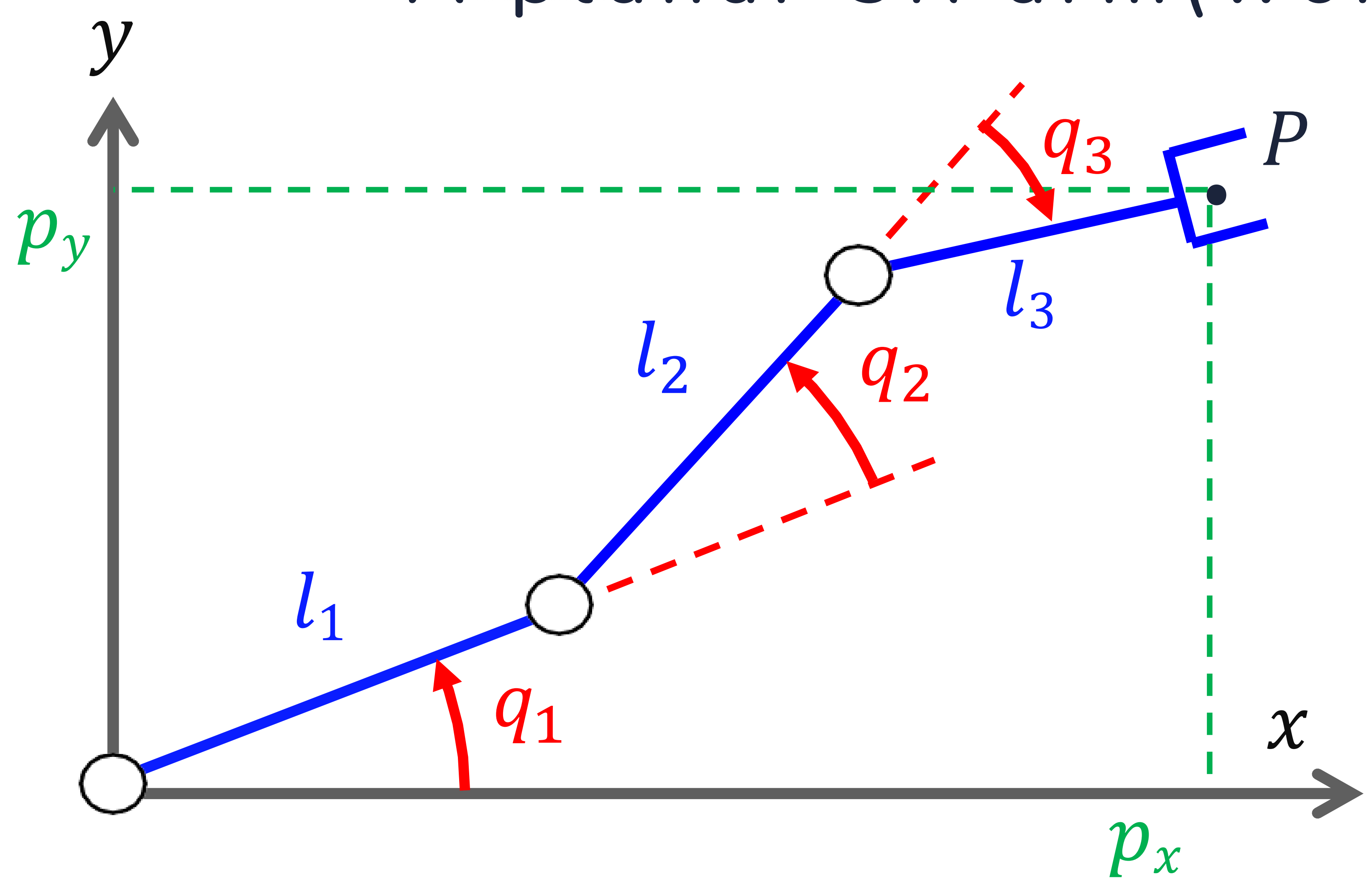
- transcendental equations are transformed into a single polynomial equation in one variable (number of roots = degree of the polynomial)
- seek for a transformed polynomial equation of the least possible degree

https://people.eecs.berkeley.edu/~jfc/papers/94/MCtra94.pdf

A Planar 3R arm



A planar 3R arm(workspace and number/type of inverse solutions)

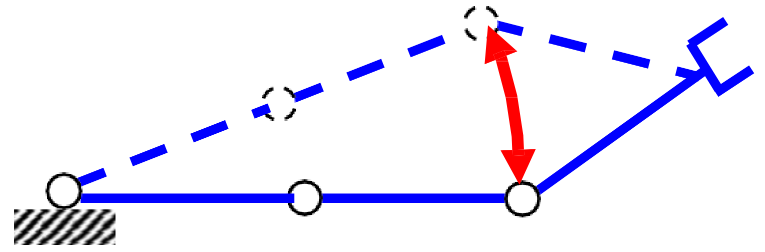


$l_1 = l_2 = l_3 = l \qquad n = 3, m = 2$

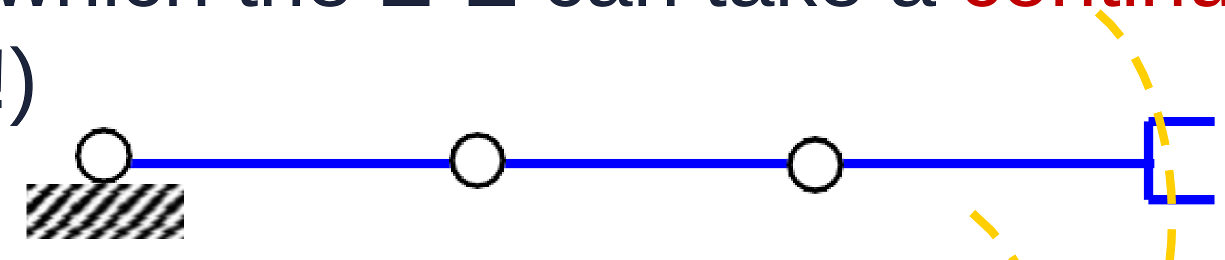
$WS_1 = \{p \in \mathbb{R}^2 : \|p\| \le 3l\} \subset \mathbb{R}^2$

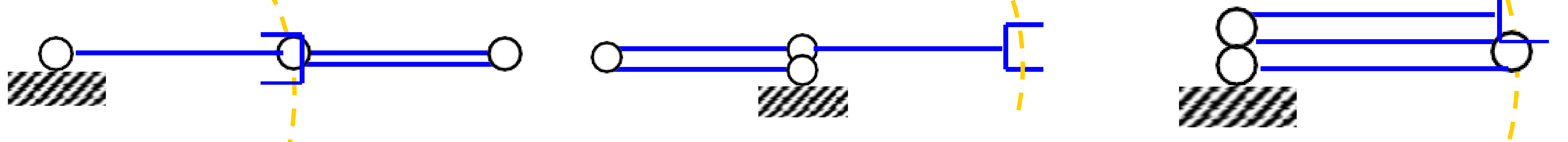
$WS_2 = \{p \in \mathbb{R}^2 : \|p\| \le l\} \subset \mathbb{R}^2$

any planar orientation is feasible in WS_2



1. in int WS_1: ∞^1 regular (except for 3.) solutions, at which the E-E can take a continuum of ∞ orientations (but not all orientations in the plane!)
2. if $\|p\| = 3l$: only 1 solution, singular
3. if $\|p\| = l$: ∞^1 solutions, 3 of which singular





4. if $\|p\| < l$: ∞^1 regular solutions (that are never singular)

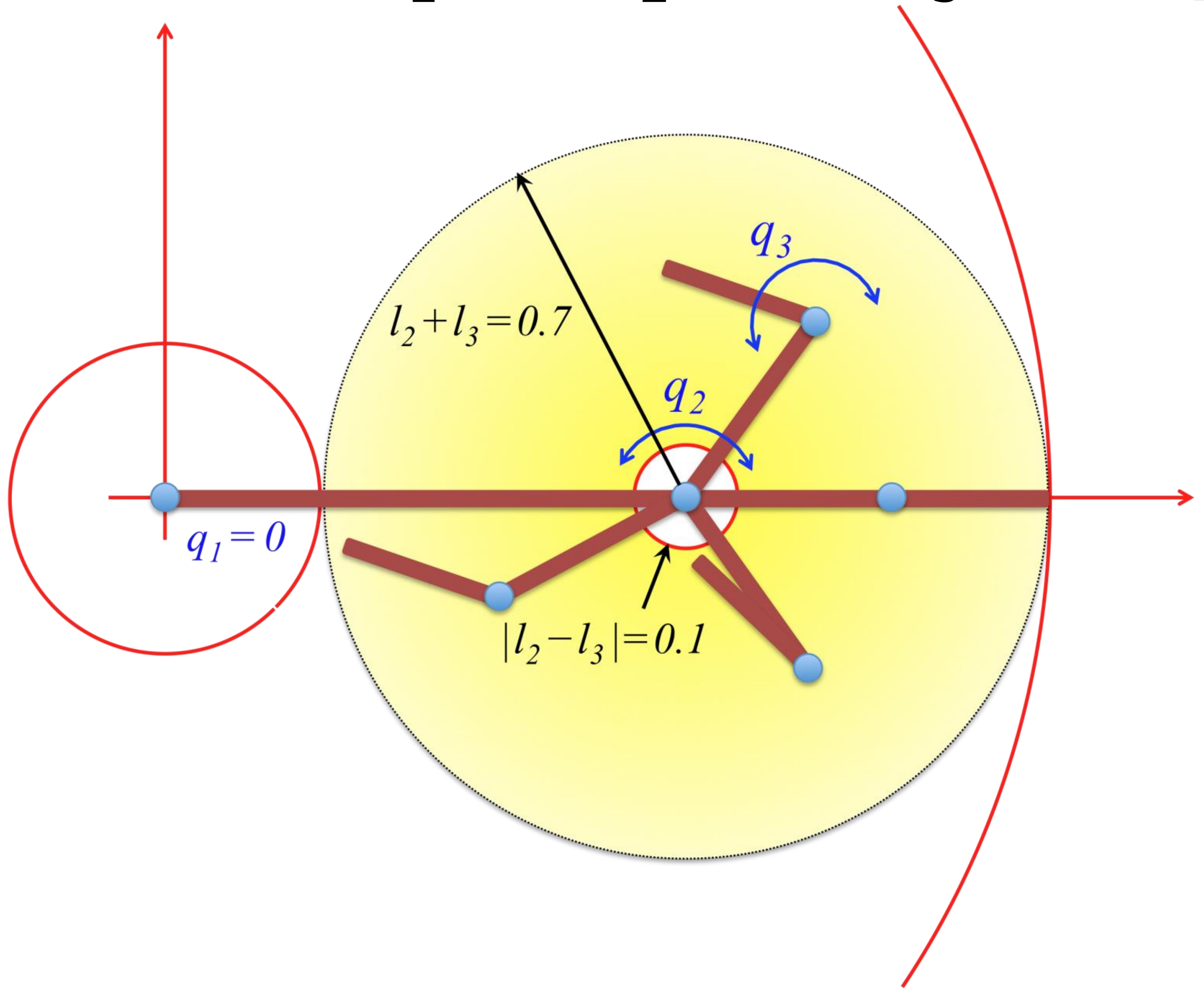
A Planar 3R arm

Workspace of a planar 3R arm(with generic link lengths)

$l_{max} = \max\{l_i, i = 1,2,3\}$ $\quad R_{out} = l_{min} + l_{med} + l_{max} = l_1 + l_2 + l_3$

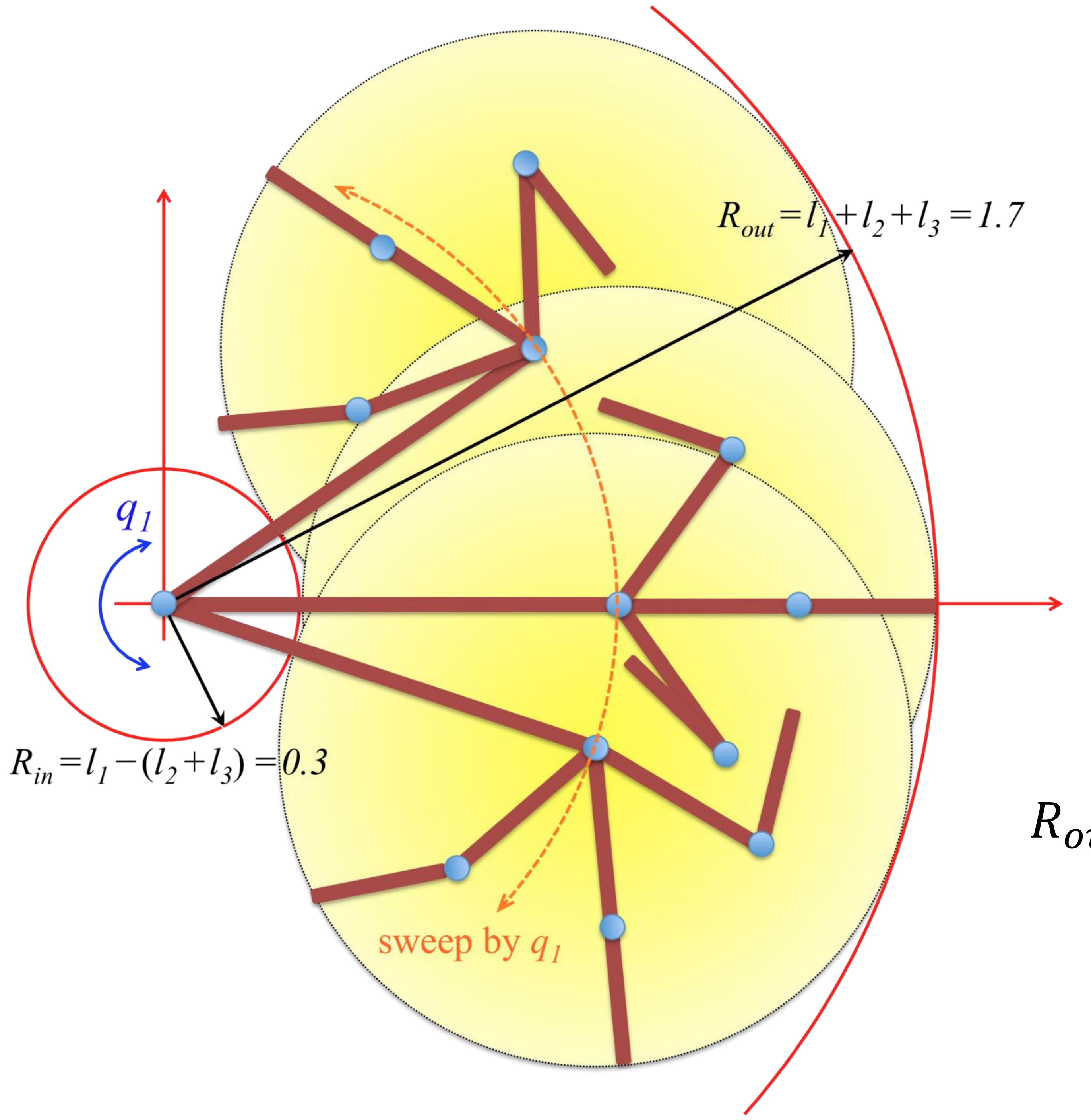
$l_{min} = \min\{l_i, i = 1,2,3\}$ ➡ $R_{in} = \max\{0, l_{max} - (l_{med} + l_{min})\}$

$l_1 = 1, l_2 = 0.4, l_3 = 0.3\ [m]$ $\Rightarrow$ $l_{max} = l_1 = 1,\ l_{med} = l_2 = 0.4,\ l_{min} = l_3 = 0.3$



➡

$R_{in} = 0.3$



$R_{out} = 1.7$

$l_1 = 0.5, l_2 = 0.7, l_3 = 0.5 [m]$ $\Rightarrow$ $l_{max} = l_2 = 0.7,\ l_{med} = l_{min} = l_1 (\text{or } l_3) = 0.5$

➡ $R_{in} = 0,\ R_{out} = 1.7$

Multiplicity of Solutions

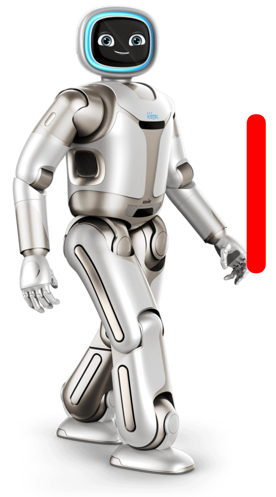
Multiplicity of solutions (summary of the general cases)

- if $m = n$
  - $\nexists$ solutions
  - a finite number of solutions (regular/generic case)
  - "degenerate" solutions: infinite or finite set, but anyway different in number from the generic case (singularity)
- if $m < n$ (robot is kinematically redundant for the task)
  - $\nexists$ solutions
  - ∞^{n-m} solutions (regular/generic case)
  - a finite or infinite number of singular solutions
- use of the term singularity will become clearer when dealing with differential kinematics
  - instantaneous velocity mapping from joint to task velocity
  - lack of full rank of the associated $m \times n$ Jacobian matrix $J(q)$

Dexter 8R Robot Arm

Dexter 8R robot arm

- $m = 6$ (position and orientation of E-E)
- $n = 8$ (all revolute joints)
- ∞^2 inverse kinematic solutions (redundancy degree $= n - m = 2$)



Inverse Kinematics Solution Methods

ANALYTICAL solution
(in closed form)

- preferred, if it can be found*
- use ad-hoc geometric inspection
- algebraic methods (solution of polynomial equations)
- systematic ways for generating a
  - reduced set of equations to be solved

* sufficient conditions for 6-dof arms

- 3 consecutive rotational joint axes are incident (e.g., spherical wrist), or
- 3 consecutive rotational joint axes are parallel

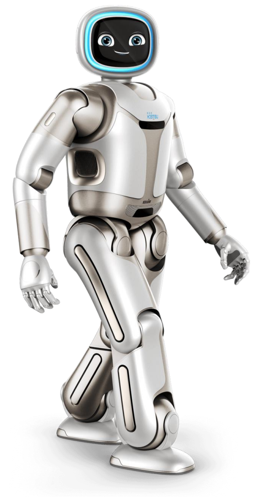
D. Pieper, PhD thesis, Stanford University, 1968

NUMERICAL solution
(in iterative form)

- certainly needed if $n > m$ (redundant case) or at/close to singularities
- slower, but easier to be set up
- in its basic form, it uses the
  - (analytical) Jacobian matrix of the direct kinematics map

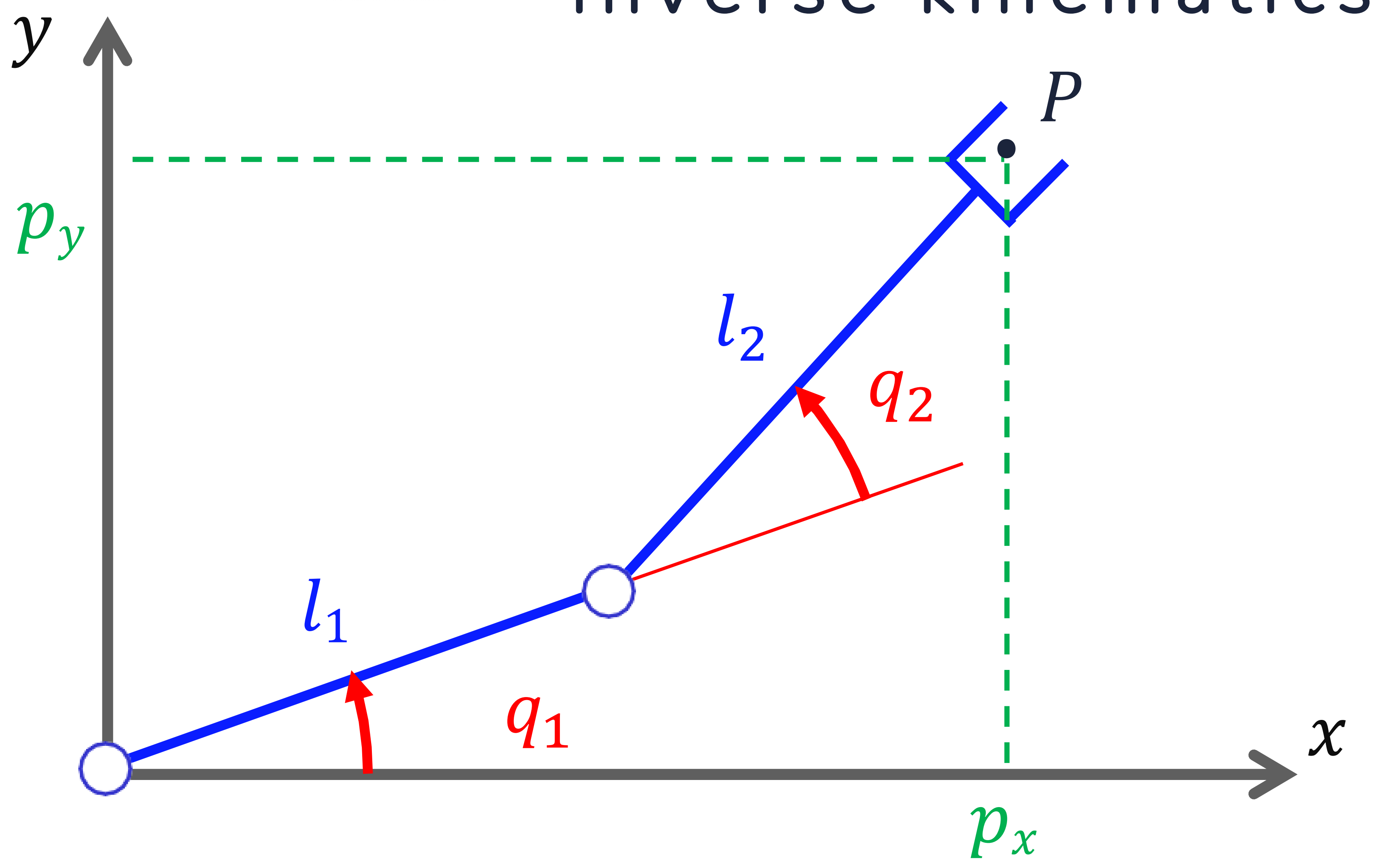
$$J_r(q) = \frac{\partial f_r(q)}{\partial q}$$

- Newton method, Gradient method, and so on…



Inverse Kinematics (2R)

Inverse kinematics of planar 2R arm – computer q_2



direct kinematics

$$p_x = l_1 c_1 + l_2 c_{12}$$
$$p_y = l_1 s_1 + l_2 s_{12}$$

data (p_x, p_y)

q_1, q_2 unknowns

"squaring and summing" the equations of the direct kinematics

$$p_x^2 + p_y^2 - (l_1^2 + l_2^2) = 2 l_1 l_2 (c_1 c_{12} + s_1 s_{12}) = 2 l_1 l_2 c_2$$

and from this

$$c_2 = \left(p_x^2 + p_y^2 - (l_1^2 + l_2^2)\right) / 2 l_1 l_2 \quad , s_2 = \pm\sqrt{1 - c_2^2}$$

$$q_2 = \mathrm{atan}\,2\,\{s_2, c_2\}$$

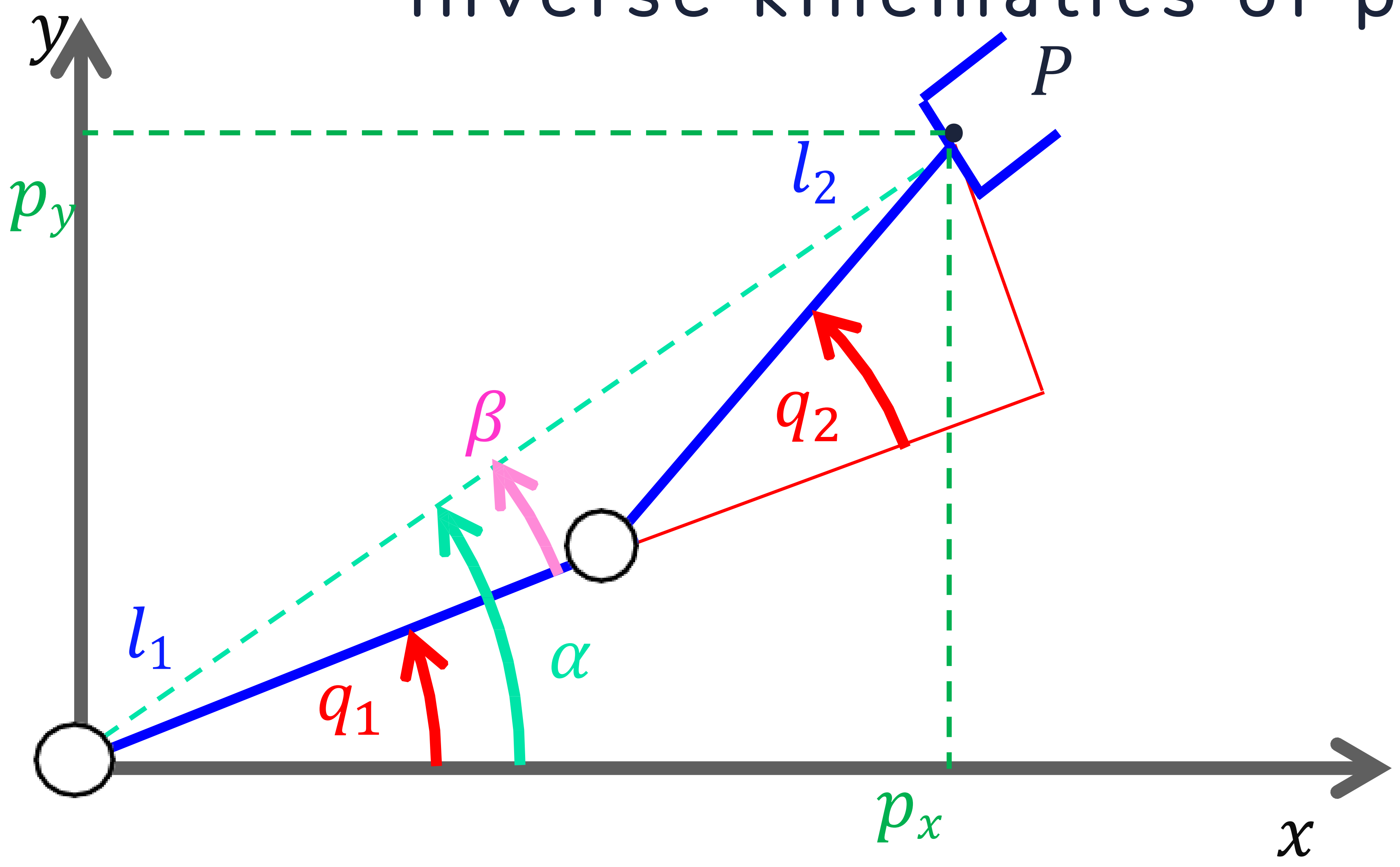
must be in [−1,1] (else, point P is outside robot workspace!)

2 solutions

in analytical form

Inverse Kinematics (2R)

Inverse kinematics of planar 2R arm – compute q_1 (method 1)



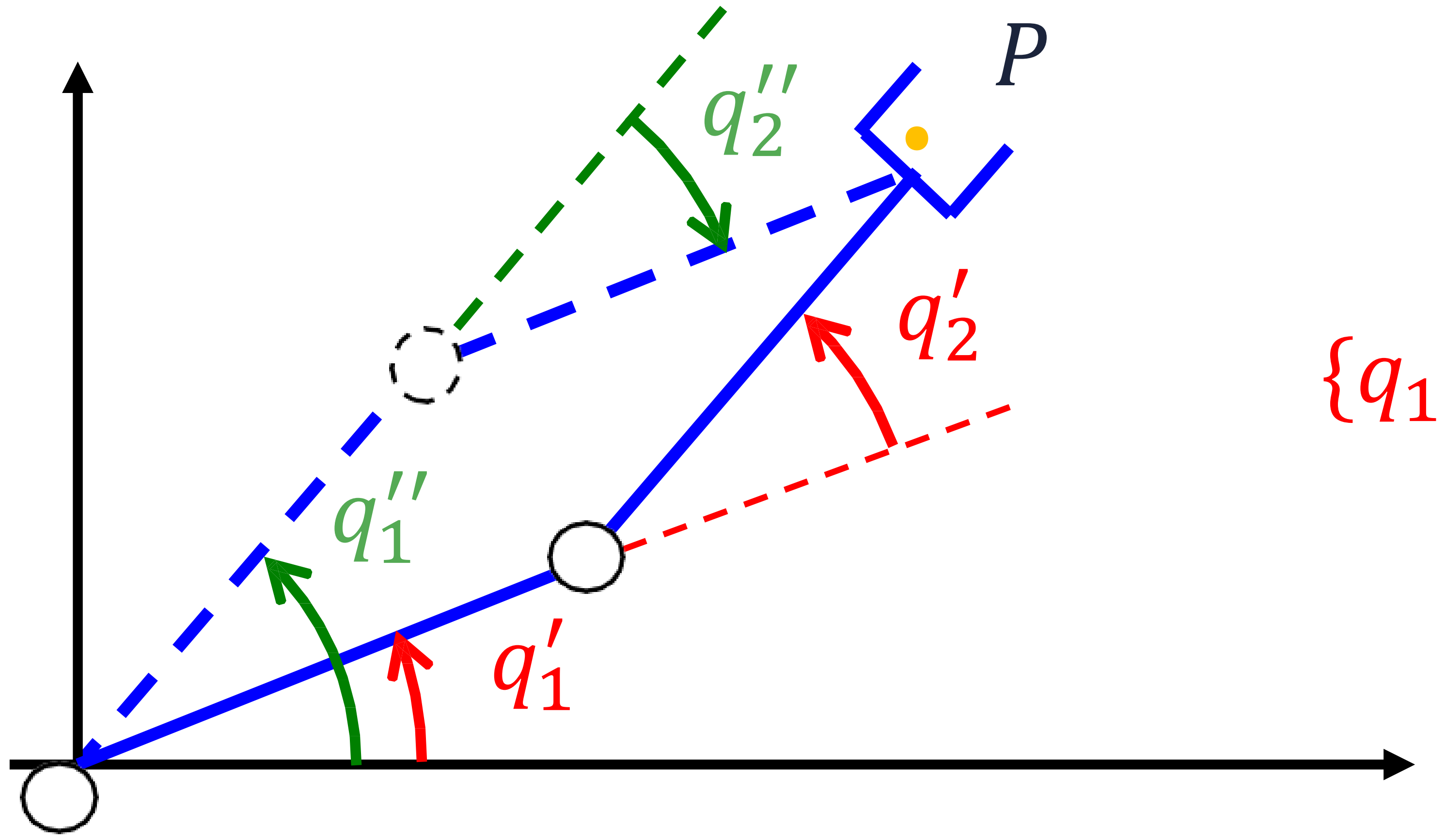
by geometric inspection

$q_1 = \alpha - \beta$

$$q_1 = \text{atan}\,2\left\{p_y, p_x\right\} - \text{atan}\,2\left\{l_2 s_2, l_1 + l_2 c_2\right\}$$

2 solutions

(one for each value of s_2)



note: difference of atan2's needs to be re-expressed in $(-\pi, \pi]$!!

q_2' and q_2'' have same absolute value, but opposite signs

Inverse Kinematics (2R)

Inverse kinematics of planar 2R arm – compute q_1 (method 2)

$$p_x = l_1 c_1 + l_2 c_{12} = l_1 c_1 + l_2 (c_1 c_2 - s_1 s_2)$$
$$p_y = l_1 s_1 + l_2 s_{12} = l_1 s_1 + l_2 (s_1 c_2 + c_1 s_2)$$

linear in s_1 and c_1

$$\begin{bmatrix} l_1 + l_2 c_2 & -l_2 s_2 \\ l_2 s_2 & l_1 + l_2 c_2 \end{bmatrix} \begin{bmatrix} c_1 \\ s_1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \end{bmatrix}$$

$$\det = l_1^2 + l_2^2 + 2 l_1 l_2 c_2 > 0$$

except if $l_1 = l_2$ and $c_2 = -1$
being then q_1 undefined
(singular case: ∞^1 solutions)

$$q_1 = \operatorname{atan2}\{s_1, c_1\} = \operatorname{atan2}\{(p_y (l_1 + l_2 c_2) - p_x l_2 s_2)/\det, (p_x (l_1 + l_2 c_2) + p_y l_2 s_2)/\det\}$$

notes: a) this method provides directly the result in $(-\pi, \pi]$
b) when evaluating atan2, $\det > 0$ can be in fact eliminated from the expressions of s_1 and c_1

Inverse Kinematics Solution Methods

ANALYTICAL solution
(in closed form)

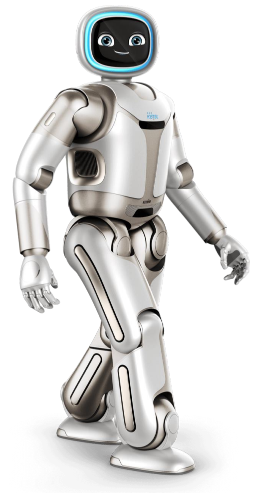
NUMERICAL solution
(in iterative form)

- preferred, if it can be found*
- use ad-hoc geometric inspection
- algebraic methods (solution of polynomial equations)
- systematic ways for generating a
  - reduced set of equations to be solved

- certainly needed if $n > m$ (redundant case) or at/close to singularities
- slower, but easier to be set up
- in its basic form, it uses the
  - (analytical) Jacobian matrix of the direct kinematics map

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

- Newton method, Gradient method, and so on…



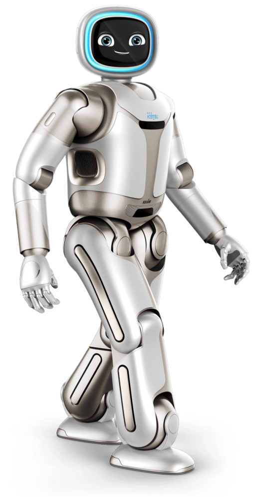
Numerical Solution of IK

- Use when a closed-form solution q to $r_d = f_r(q)$ does not exist or is “too hard” to be found
- All methods are iterative and need the matrix $J_r(q) = \frac{\partial f_r(q)}{\partial q}$ (analytical Jacobian)
- Newton method (here only for $m = n$, at the kth iteration/guess)

$$r_d = f_r(q) = f_r(q^k) + J_r(q^k)(q - q^k) + o(\|q - q^k\|) \leftarrow \text{neglected}$$

$$q^{k+1} = q^k + J_r^{-1}(q^k)[r_d - f_r(q^k)]$$

- ✓ convergence for q^0 (initial guess) close enough to some q^*: $f_r(q^*) = r_d$
- ✓ problems in/near singularities of the Jacobian matrix $J_r(q)$
- ✓ in case of robot redundancy $(m < n)$, use the pseudo-inverse $J_r^{\#}(q)$
- ✓ has quadratic convergence rate when near to a solution (fast!)



Numerical Solution of IK

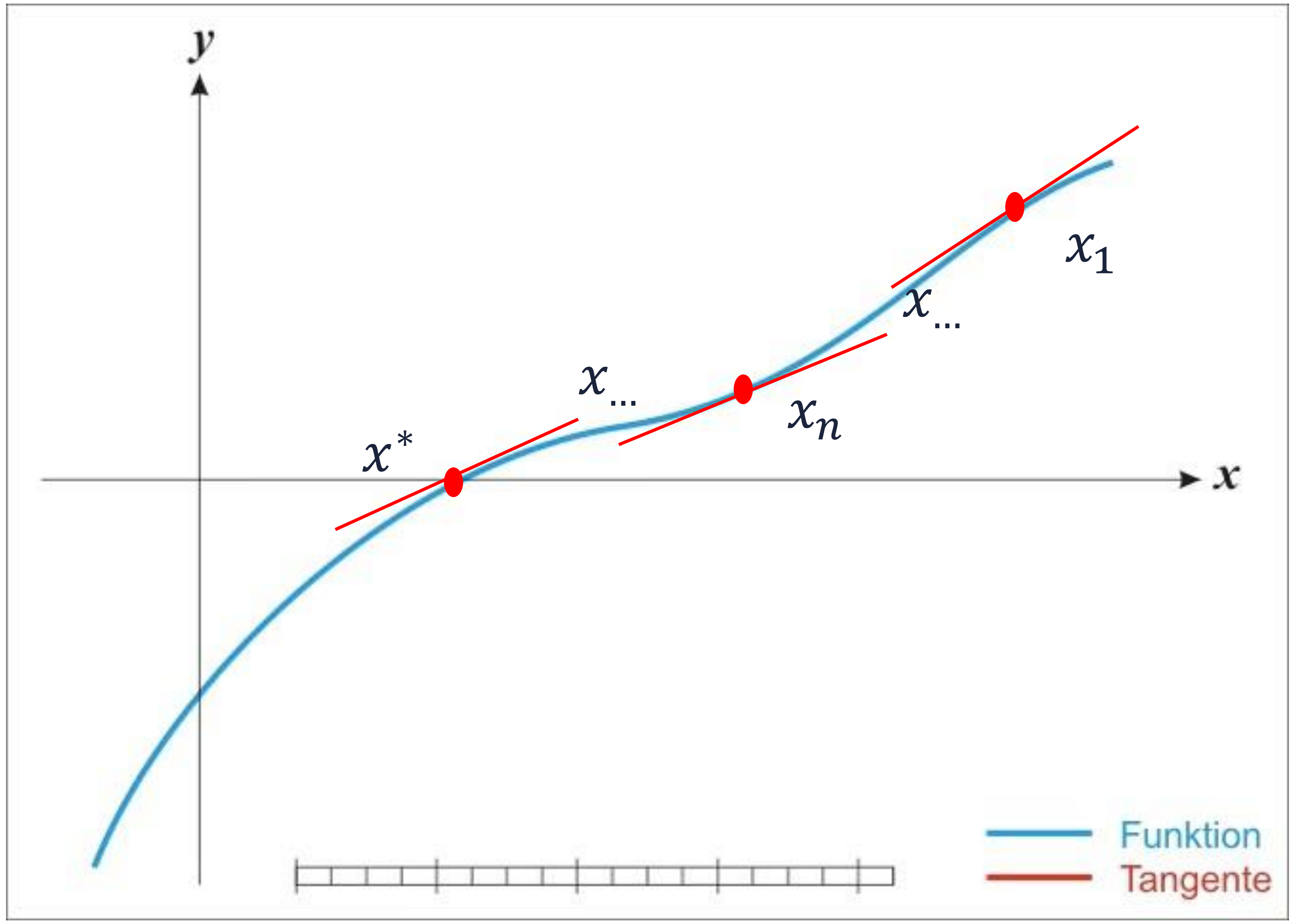
Operation of Newton method

- In the scalar case, also known as “method of the tangent”
- For a differentiable function $f(x)$, find a root of $f(x^*) = 0$ by iterating as

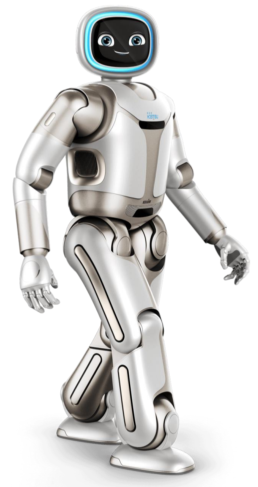
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

an approximating sequence

$$\{x_1, \cdots, x_n, \cdots\} \longrightarrow x^*$$



animation from
http://en.wikipedia.org/wiki/File:NewtonIteration_Ani.gif



Numerical Solution of IK

- Gradient method (max descent)
  - minimize the error function

$$H(q) = \frac{1}{2}\|r_d - f_r(q)\|^2 = \frac{1}{2}(r_d - f_r(q))^T(r_d - f_r(q))$$

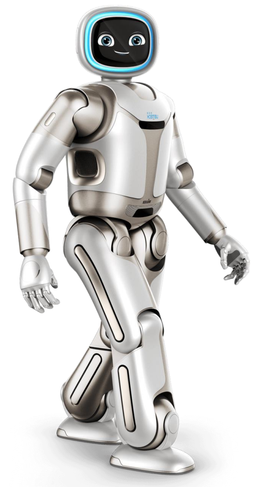
$$q^{k+1} = q^k - \alpha \nabla_q H\left(q^k\right)$$

from $\nabla_q H(q) = (\partial H(q)/\partial q)^T = -\left((r_d - f_r(q))^T(\partial f_r(q)/\partial q)\right)^T = -J_r^T(q)(r_d - f_r(q))$

we get

$$q^{k+1} = q^k + \alpha J_r^T\left(q^k\right)\left(r_d - f_r\left(q^k\right)\right)$$

- the scalar step size $\alpha > 0$ should be chosen so as to guarantee a decrease of the error function at each iteration: too large values for α may lead the method to "miss" the minimum
- when the step size is too small, convergence is extremely slow
- computationally simpler: use the Jacobian transpose, rather than its (pseudo)-inverse
- same use also for robots that are redundant ($n > m$) for the task
- may not converge to a solution, but it never diverges



Numerical Solution of IK

A case study
(analytic expressions of Newton and gradient iterations)

- 2R robot with $l_1 = l_2 = 1$, desired end-effector position $r_d = p_d = (1,1)$
- direct kinematic function and error

$$f_r(q) = \begin{pmatrix} c_1 + c_{12} \\ s_1 + s_{12} \end{pmatrix} \qquad e = p_d - f_r(q) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - f_r(q)$$

- Jacobian matrix

$$J_r(q) = \frac{\partial f_r(q)}{\partial q} = \begin{pmatrix} -(s_1 + s_{12}) & -s_{12} \\ c_1 + c_{12} & c_{12} \end{pmatrix}$$

- Newton versus Gradient iteration

Newton method $J_r^{-1}(q^k)$

$$q^{k+1} = q^k + \left| \begin{array}{l} \dfrac{1}{s_2}\begin{pmatrix} c_{12} & s_{12} \\ -(c_1 + c_{12}) & -(s_1 + s_{12}) \end{pmatrix}_{|q=q^k} \\ \alpha \begin{pmatrix} -(s_1 + s_{12}) & c_1 + c_{12} \\ -s_{12} & c_{12} \end{pmatrix}_{|q=q^k} \end{array} \right| \times \overbrace{\begin{pmatrix} 1 - (c_1 + c_{12}) \\ 1 - (s_1 + s_{12}) \end{pmatrix}_{|q=q^k}}^{e^k}$$

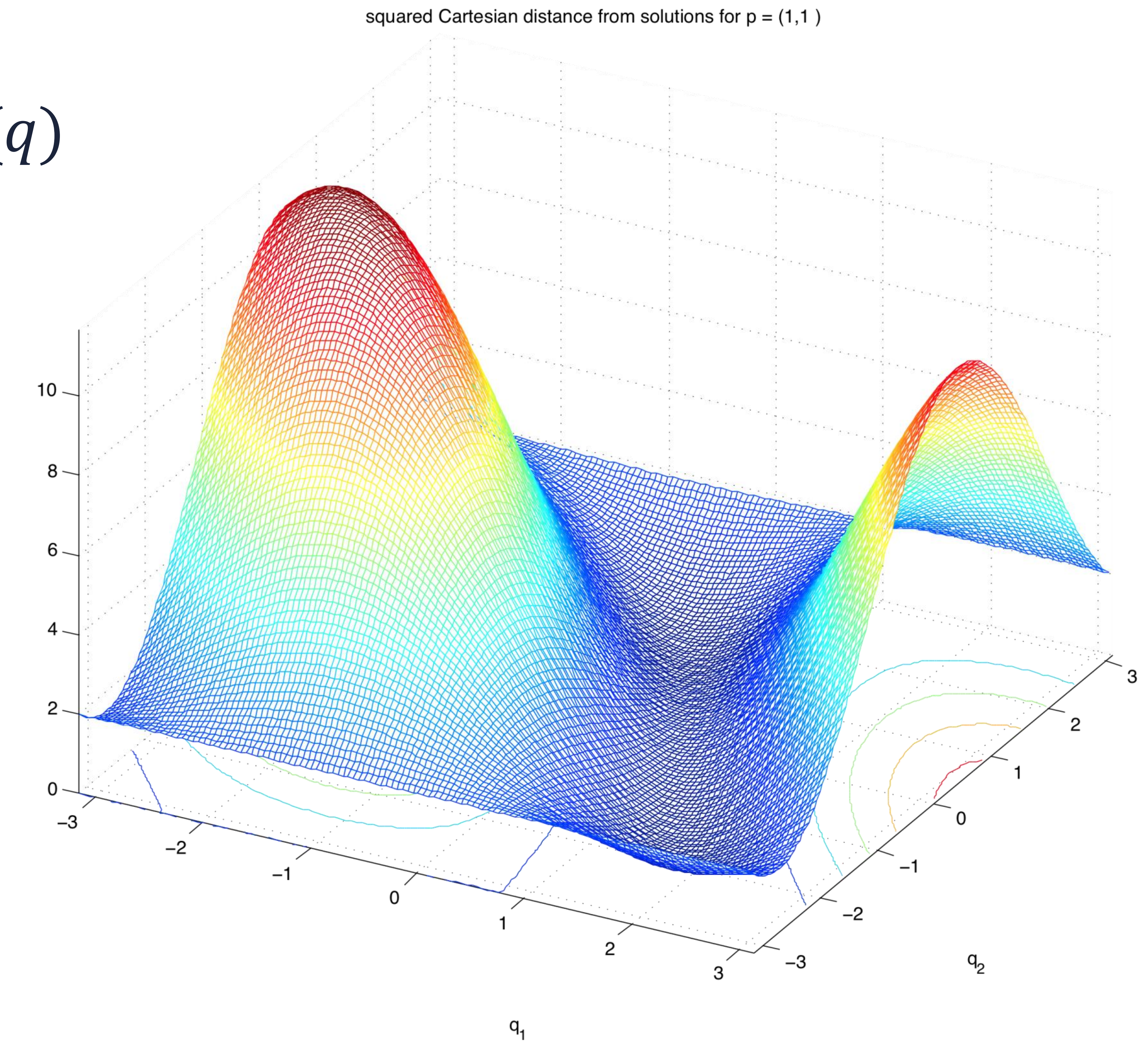
Gradient method $J_r^T(q^k)$

Numerical Solution of IK

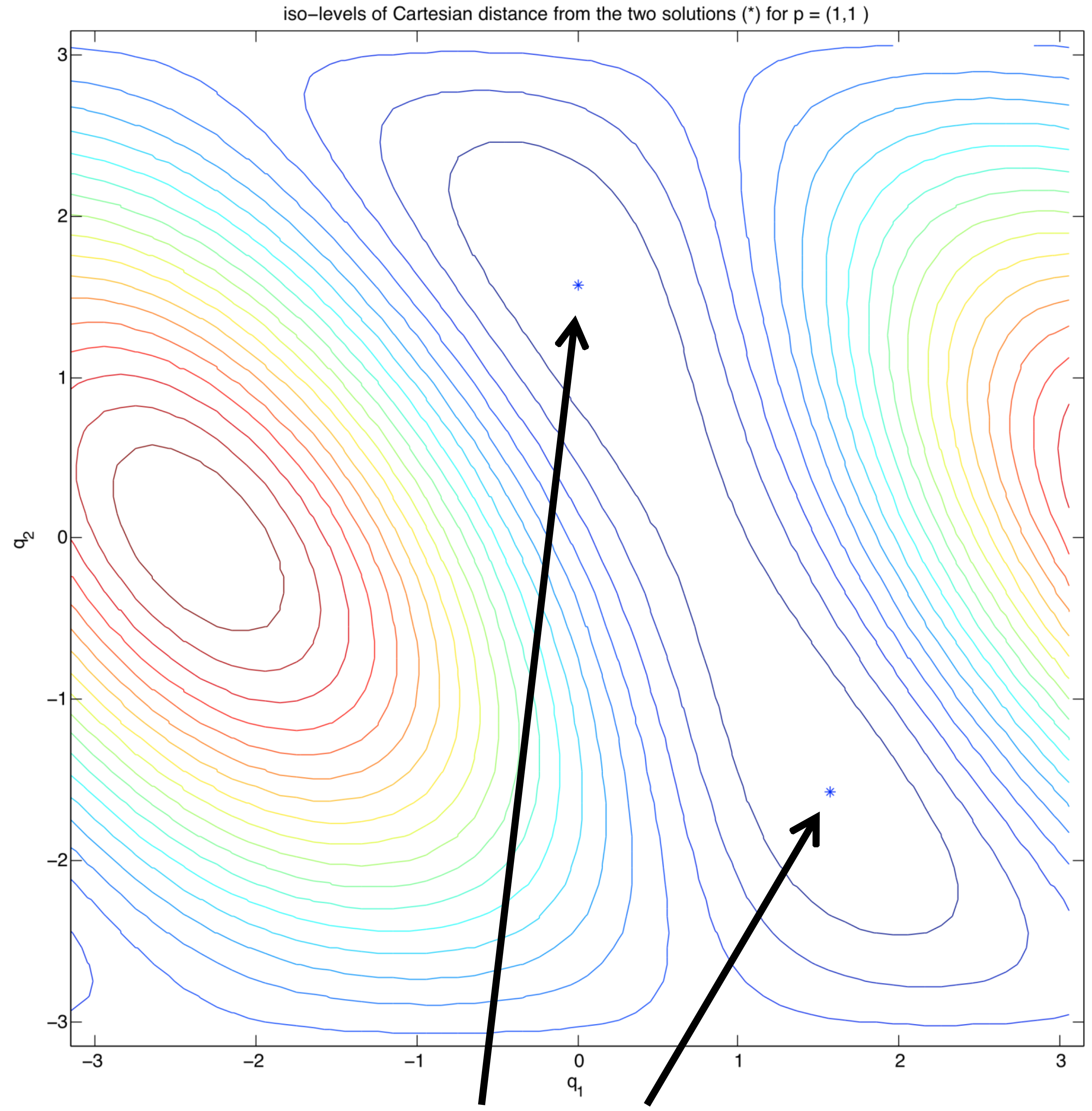
Error function

- 2R robot with $l_1 = l_2 = 1$ and desired end-effector position $p_d = (1,1)$

$$e = p_d - f_r(q)$$



plot of $||e||^2$ as a function of $q = (q_1, q_2)$

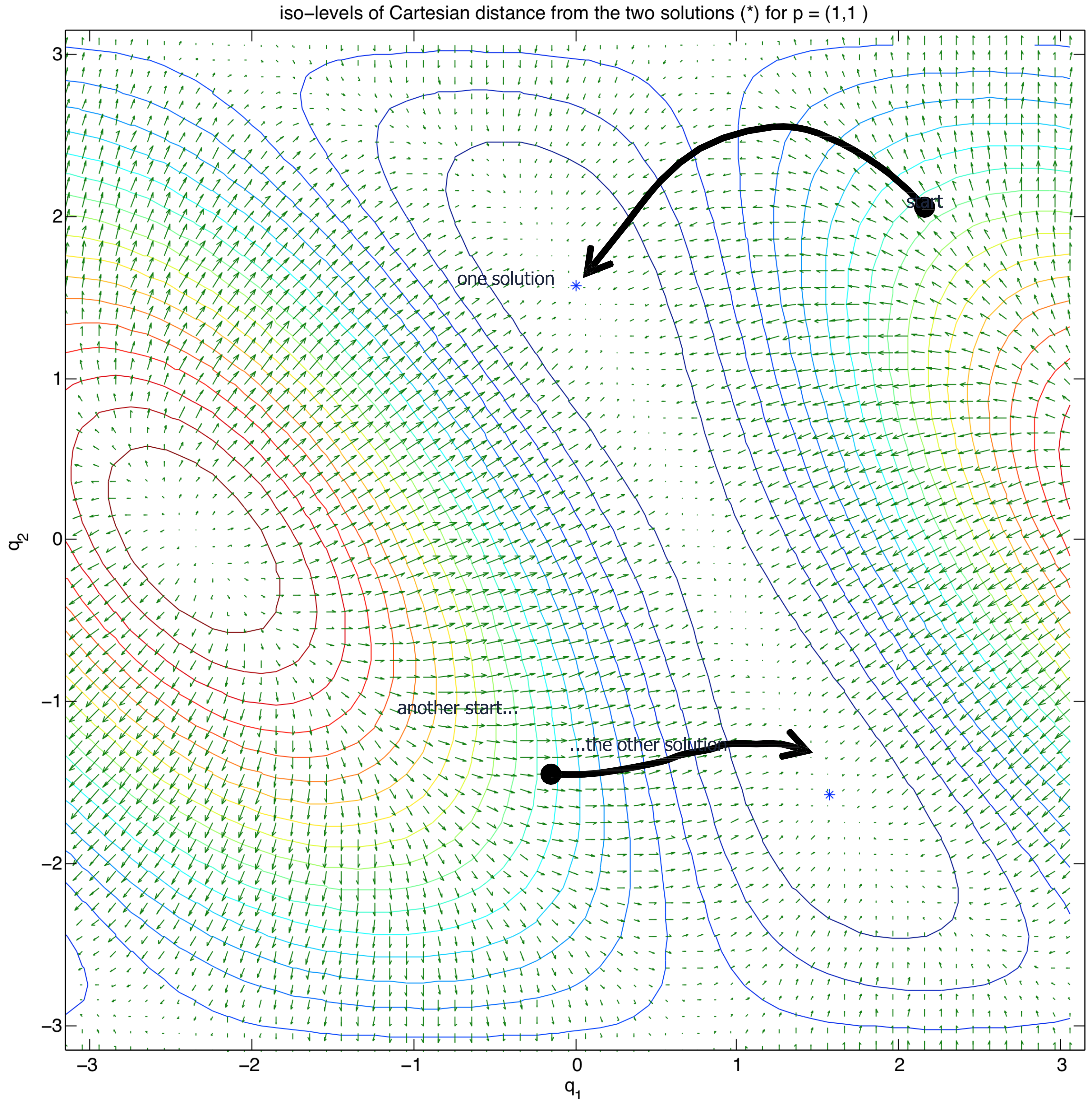


two local minima
(inverse kinematic solutions)

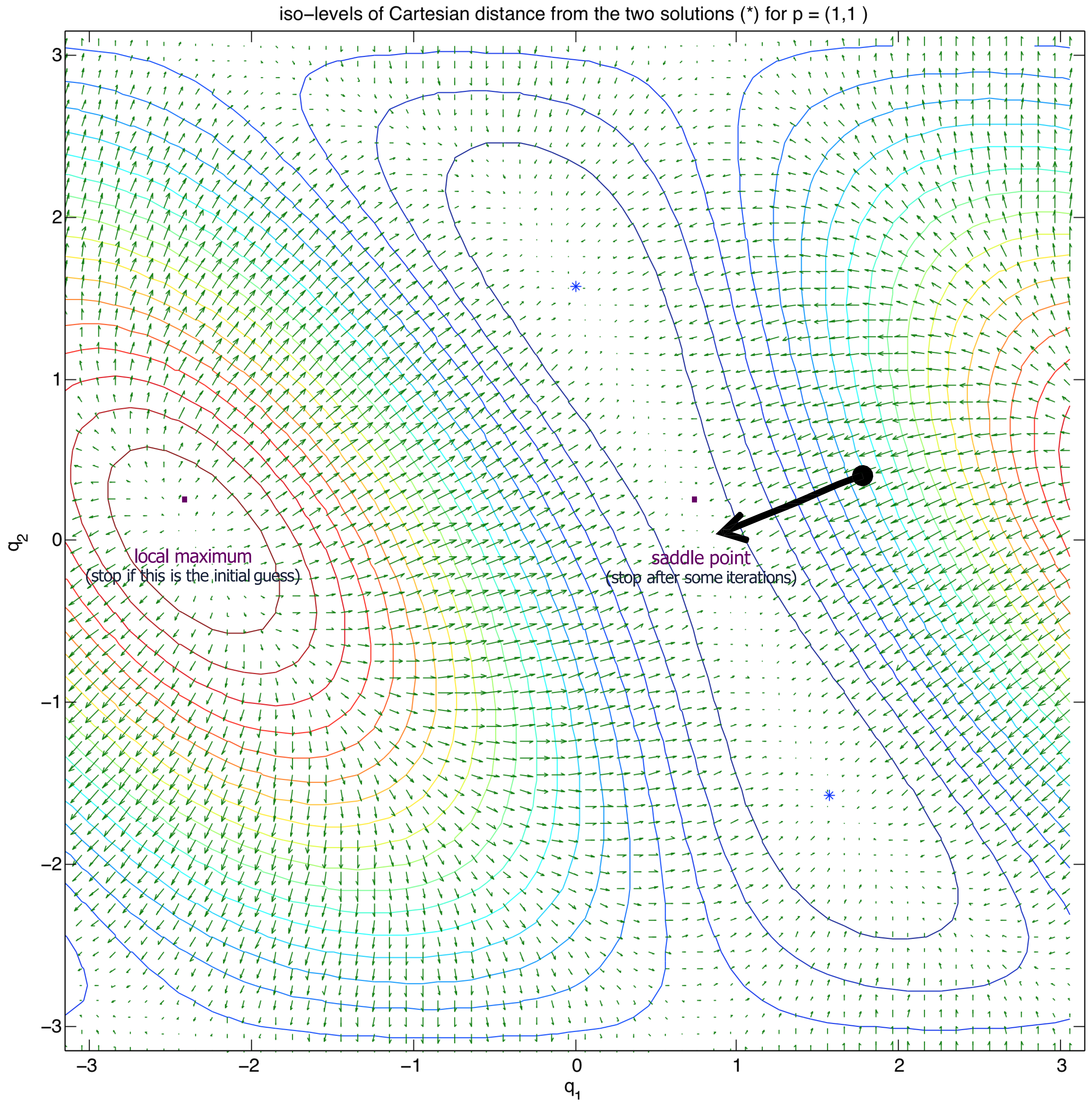
Numerical Solution of IK

Error reduction by Gradient method

- flow of iterations along the negative (or anti-) gradient
- two possible cases: convergence or stuck (at zero gradient)

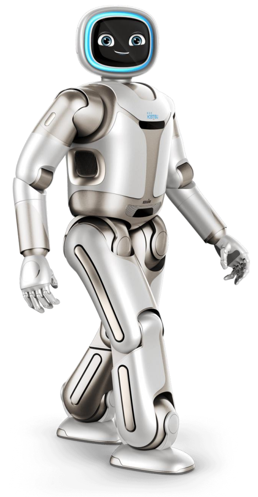


$(q_1, q_2)' = (0, \pi/2)$ $(q_1, q_2)'' = (\pi/2, -\pi/2)$



$(q_1, q_2)_{max} = (-3\pi/4, 0)$ $(q_1, q_2)_{saddle} = (\pi/4, 0)$

$e \in \mathcal{N}(J_r^T(q))$!



6R Robot Arm

Extended Reading

- the most complex inverse kinematics that could be solved in principle in closed form (i.e., analytically) is that of a 6R serial manipulator, with arbitrary DH table
  - ways to systematically generate equations from the direct kinematics that could be easier to solve ⇒ some scalar equations may contain perhaps a single unknown variable!

$${}^0T_6 = {}^0A_1(\theta_1)\,{}^1A_2(\theta_2)\cdots{}^5A_6(\theta_6) = U_0$$

method used for the Unimation PUMA 600 in (*)

$${}^0A_1^{-1}\,{}^0T_6 = U_1(={}^1A_2\cdots{}^5A_6)$$
$${}^1A_2^{-1}\,{}^0A_1^{-1}\,{}^0T_6 = U_2(={}^2A_3\cdots{}^5A_6)^4$$
$$\cdots$$
$${}^4A_5^{-1}\ldots{}^1A_2^{-1}\,{}^0A_1^{-1}\,{}^0T_6 = U_5(={}^5A_6)$$

or also ...

$${}^0T_6\,{}^5A_6^{-1} = V_5(={}^1A_2\cdots{}^4A_5)$$
$${}^0T_6\,{}^5A_6^{-1}\,{}^4A_5^{-1} = V_4(={}^1A_2\cdots{}^3A_4)$$
$$\cdots$$
$${}^0T_6\,{}^5A_6^{-1}\,{}^4A_5^{-1}\ldots{}^1A_2^{-1} = V_1(={}^0A_1)$$

(*) Paul, Shimano, and Mayer: IEEE Transactions on Systems, Man, and Cybernetics, 1981

- generating from the direct kinematics a reduced set of equations to be solved (setting w.l.o.g. $d_1 = d_6 = 0$) ⇒ 4 compact scalar equations in the 4 unknowns $\theta_2, \ldots, \theta_5$

solved analytically or numerically ...

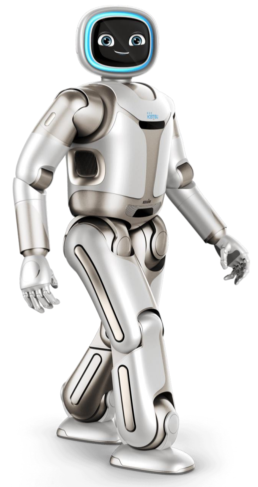
$${}^0T_6 = \begin{bmatrix} n & s & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0A_6(\theta) \qquad z = [0 \quad 0 \quad 1]^T$$

⟶

$$a_z = a^T(\theta)z \qquad \|p\|^2 = p^T(\theta)p(\theta)$$
$$p_z = p^T(\theta)z \qquad p^Ta = p^T(\theta)a(\theta)$$

... then solve easily for the remaining θ_1 and θ_6

Manseur and Doty: International Journal of Robotics Research, 1988



Numerical Solution of IK

Issues in implementation

- initial guess $q^{)}$
  - only one inverse solution is generated for each guess
  - multiple initializations for obtaining other solutions
- optimal step size $\alpha > 0$ in Gradient method
  - a constant step may work good initially, but not close to the solution (or vice versa)
  - an adaptive one-dimensional line search (e.g., Armijo's rule) could be used to choose the best α at each iteration
- stopping criteria

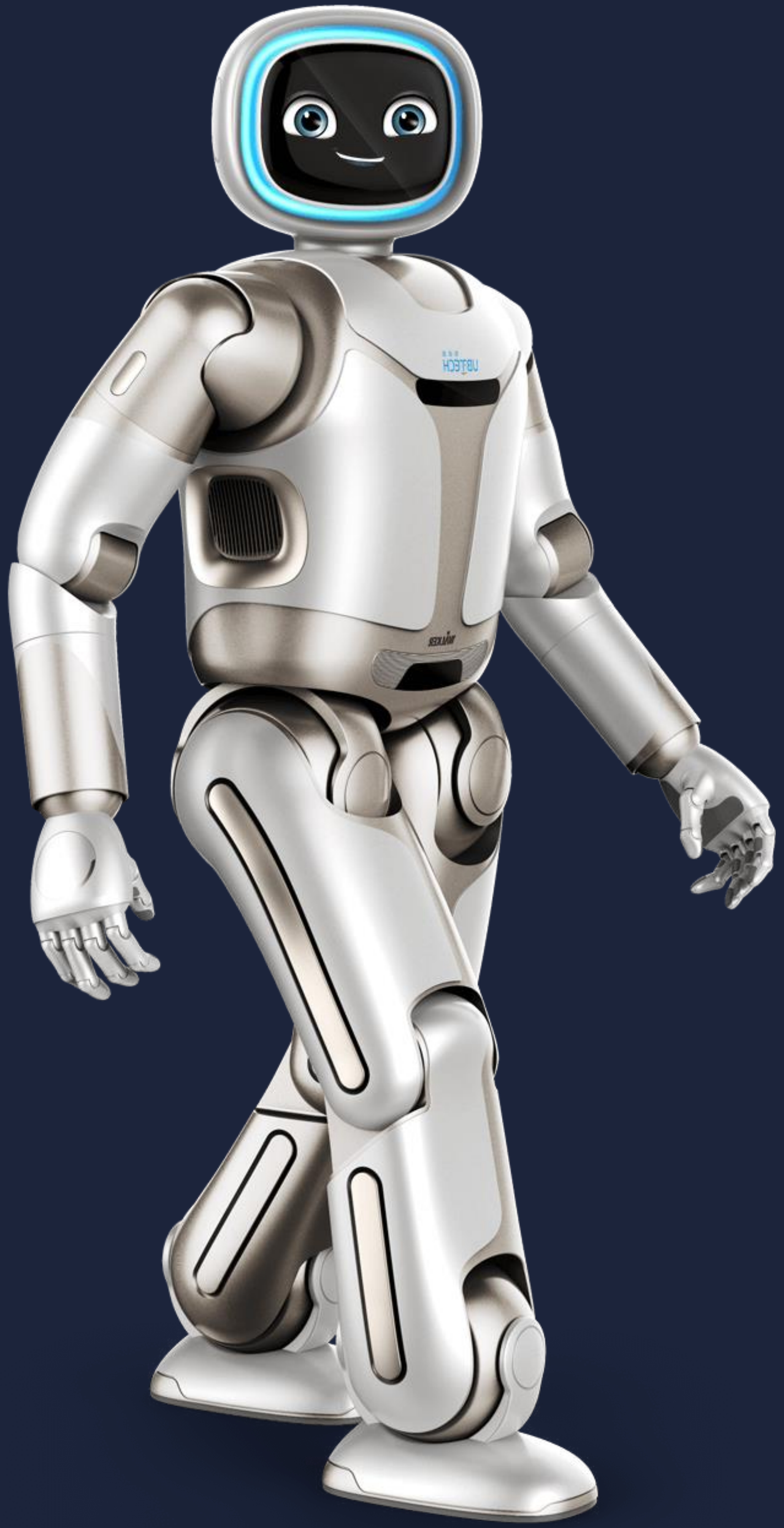
Cartesian error (possibly, separate for position and orientation) $\left\| r_d - f_r(q^k) \right\| \leq \varepsilon$

algorithm increment $\left\| q^{k+1} - q^k \right\| \leq \varepsilon_q$

- understanding closeness to singularities

$$\sigma_{min}\{J_r(q^k)\} > \sigma_0$$

good numerical conditioning of Jacobian matrix (SVD) (or a simpler test on its determinant, for $m = n$)



Q&A