MEAM 520
Lecture 7: Inverse Kinematics

Cynthia Sung, Ph.D.
Mechanical Engineering & Applied Mechanics
University of Pennsylvania

Announcements

Lab 1 due tomorrow 9/19, 11:59 p.m.

- Feedback on pre-labs went out this morning

Lab 2 will be posted tomorrow

First paper reading posted

- Answer questions by **9/28**

Last Time

The **Denavit-Hartenberg transform** results from successive rotations and translations via the four DH parameters

a parameterization for homogeneous transformations

The transform from i to i-1 is

$$A_i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

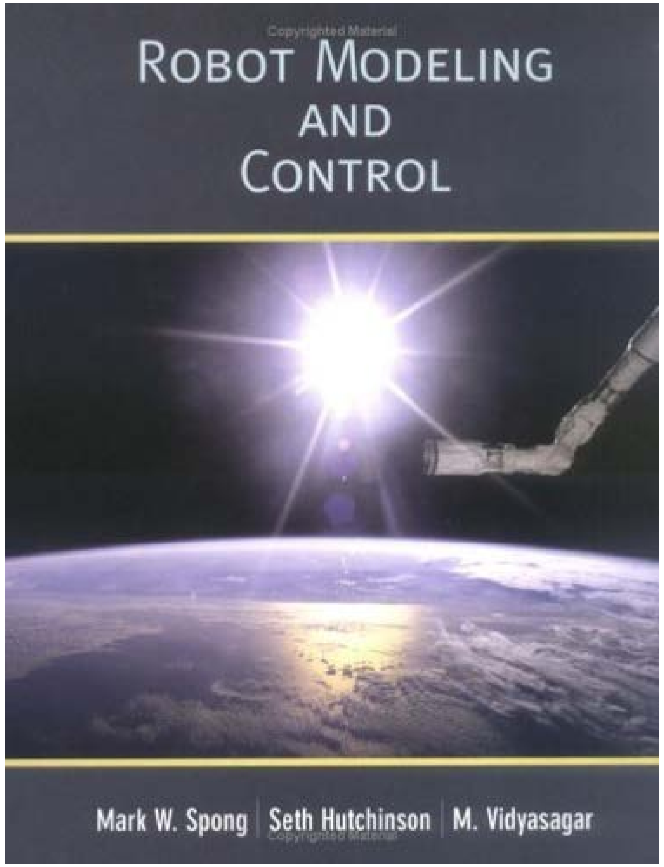
Three DH parameters will be **constant** for each joint's transformation, and one will **vary**.

Plug DH parameters into the above formula to find each joint's transformation matrix.

The final transformation matrix from tip to base is

$$\mathbf{T}_n^0 = A_1(q_1)\cdots A_n(q_n)$$

Today: Inverse Kinematics



Chapter 3: Forward and Inverse Kinematics

- Read Sec. 3.3 – 3.4

Inverse Kinematics

Given $\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$

and a certain manipulator with *n* joints

find q_1,…,q_n such that $\mathbf{T}_n^0(q_1, \dots, q_n) = \mathbf{H}$

Inverse Kinematics

Given $\mathbf{H} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & o_x \\ r_{21} & r_{22} & r_{23} & o_y \\ r_{31} & r_{32} & r_{33} & o_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$

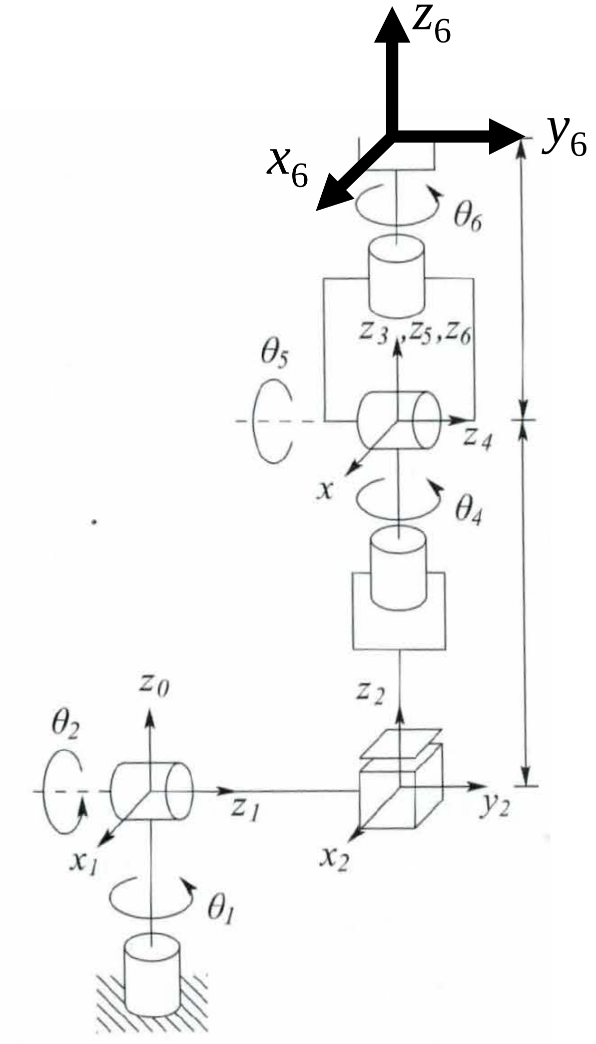
all functions of q

This yields 12 nonlinear equations in n unknown variables

and a certain manipulator with n joints

find $q_1,\ldots,q_n$ such that $\mathbf{T}_n^0(q_1, \ldots, q_n) = \mathbf{H}$

Inverse Kinematics for the Serial Manipulator with a Spherical Wrist



$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix} = T_6^0 = A_1 \cdots A_6 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

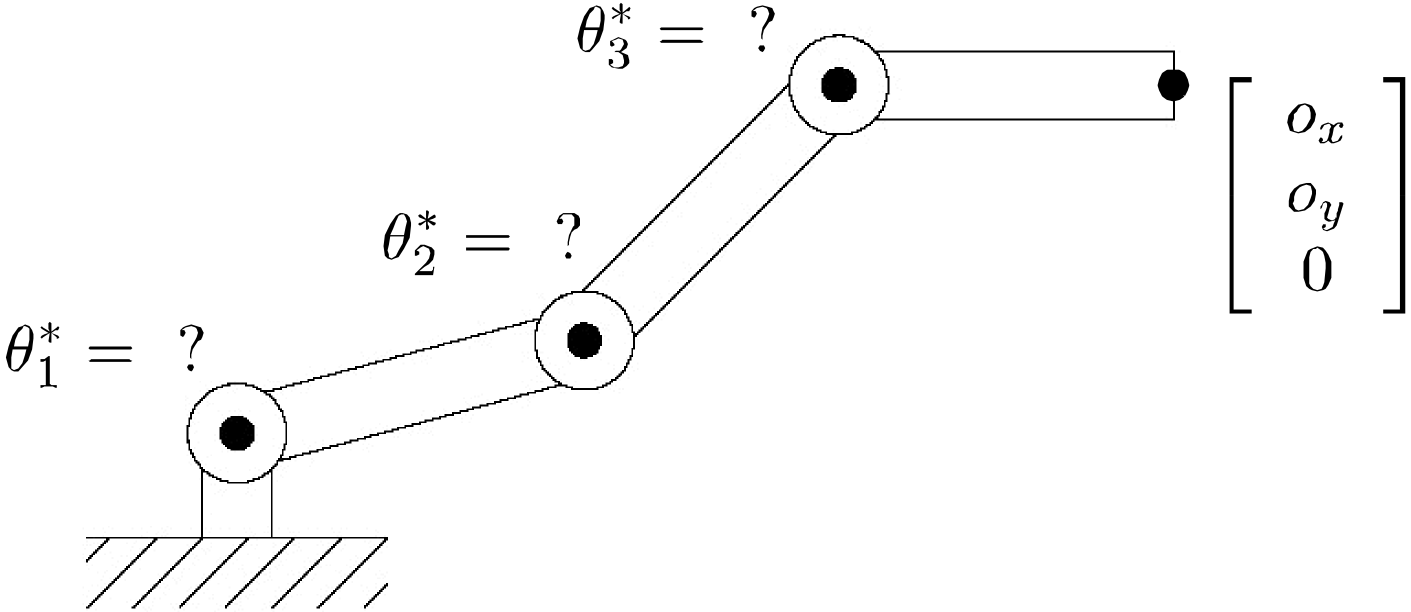
in which

$$\begin{aligned}
r_{11} &= c_1[c_2(c_4c_5c_6 - s_4s_6) - s_2s_5c_6] - d_2(s_4c_5c_6 + c_4s_6) \\
r_{21} &= s_1[c_2(c_4c_5c_6 - s_4s_6) - s_2s_5c_6] + c_1(s_4c_5c_6 + c_4s_6) \\
r_{31} &= -s_2(c_4c_5c_6 - s_4s_6) - c_2s_5c_6 \\
r_{12} &= c_1[-c_2(c_4c_5s_6 + s_4c_6) + s_2s_5s_6] - s_1(-s_4c_5s_6 + c_4c_6) \\
r_{22} &= -s_1[-c_2(c_4c_5s_6 + s_4c_6) + s_2s_5s_6] + c_1(-s_4c_5s_6 + c_4c_6) \\
r_{32} &= s_2(c_4c_5s_6 + s_4c_6) + c_2s_5s_6 \\
r_{13} &= c_1(c_2c_4s_5 + s_2c_5) - s_1s_4s_5 \\
r_{23} &= s_1(c_2c_4s_5 + s_2c_5) + c_1s_4s_5 \\
r_{33} &= -s_2c_4s_5 + c_2c_5 \\
d_x &= c_1s_2d_3 - s_1d_2 + +d_6(c_1c_2c_4s_5 + c_1c_5s_2 - s_1s_4s_5) \\
d_y &= s_1s_2d_3 + c_1d_2 + d_6(c_1s_4s_5 + c_2c_4s_1s_5 + c_5s_1s_2) \\
d_z &= c_2d_3 + d_6(c_2c_5 - c_4s_2s_5)
\end{aligned}$$

Q: Will there always be a solution?

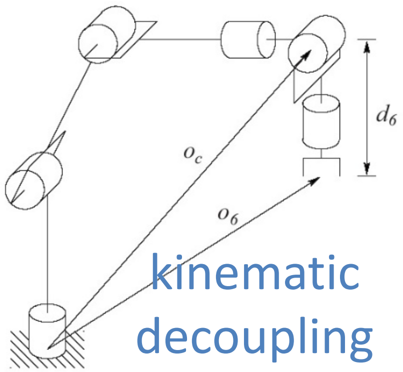
Q: If there is a solution, will it always be unique?

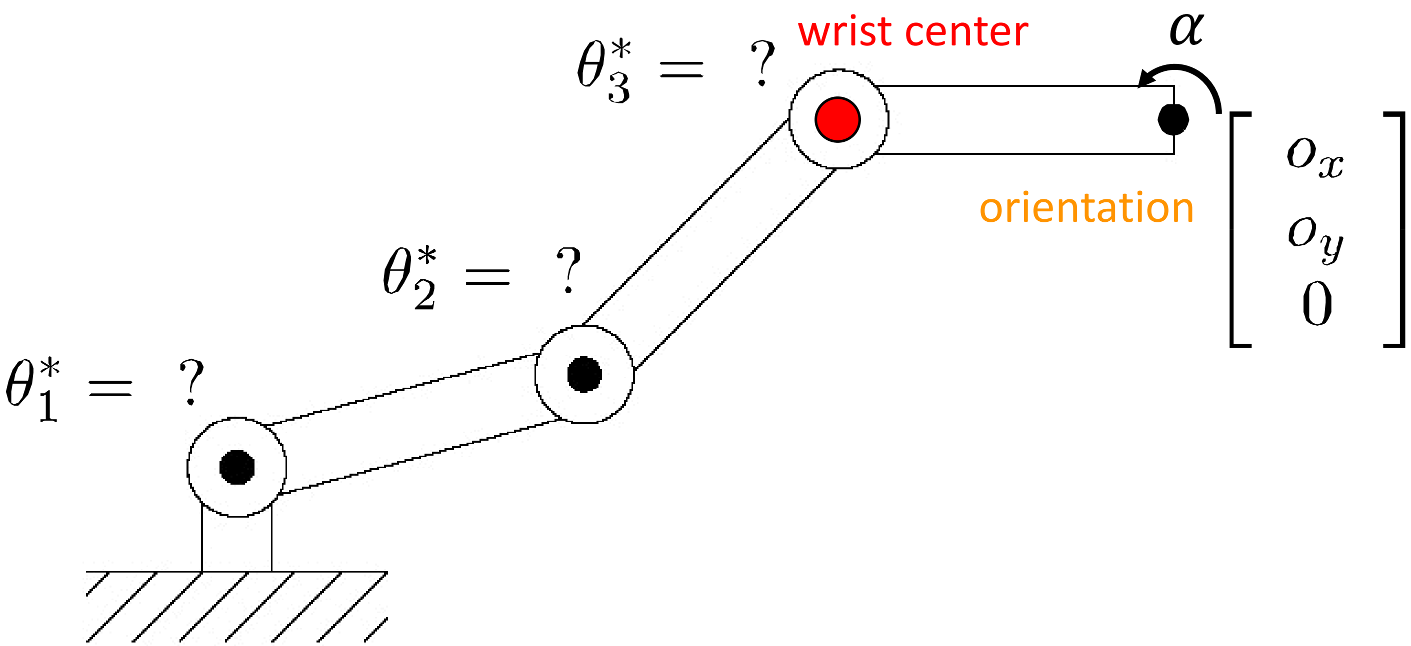
Planar RRR Robot



Q: Given a desired position of the end effector, how many solutions are there to the IK?

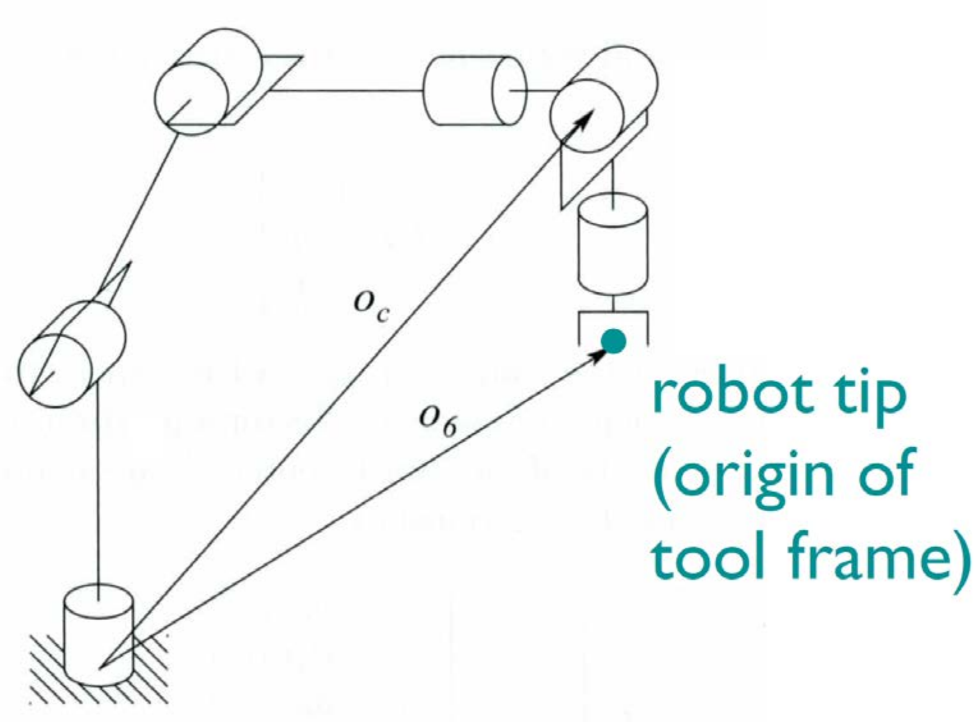
Planar RRR Robot





Q: If the orientation of the end effector is also specified, how many IK solutions are there?

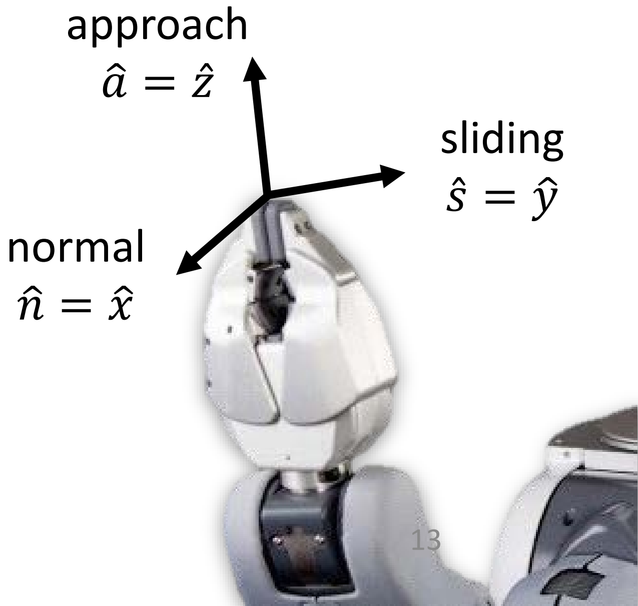
Trick: Exploit the kinematic structure of the manipulator.
If the robot has a spherical wrist, use **Kinematic Decoupling**.



$$\text{given } \mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$$

The convention for placing the end-effector frame:
The z axis points along the approach direction,
coaxial with the final revolute joint.

$$\begin{array}{cc} & \begin{array}{cccc} & \mathbf{R} & & \mathbf{d} \end{array} \\ \mathbf{H} = & \begin{bmatrix} n_x & s_x & a_x & d_x \\ n_y & s_y & a_y & d_y \\ n_z & s_z & a_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ & \begin{array}{cc} \text{three zeros} & \text{one one} \end{array} \end{array}$$

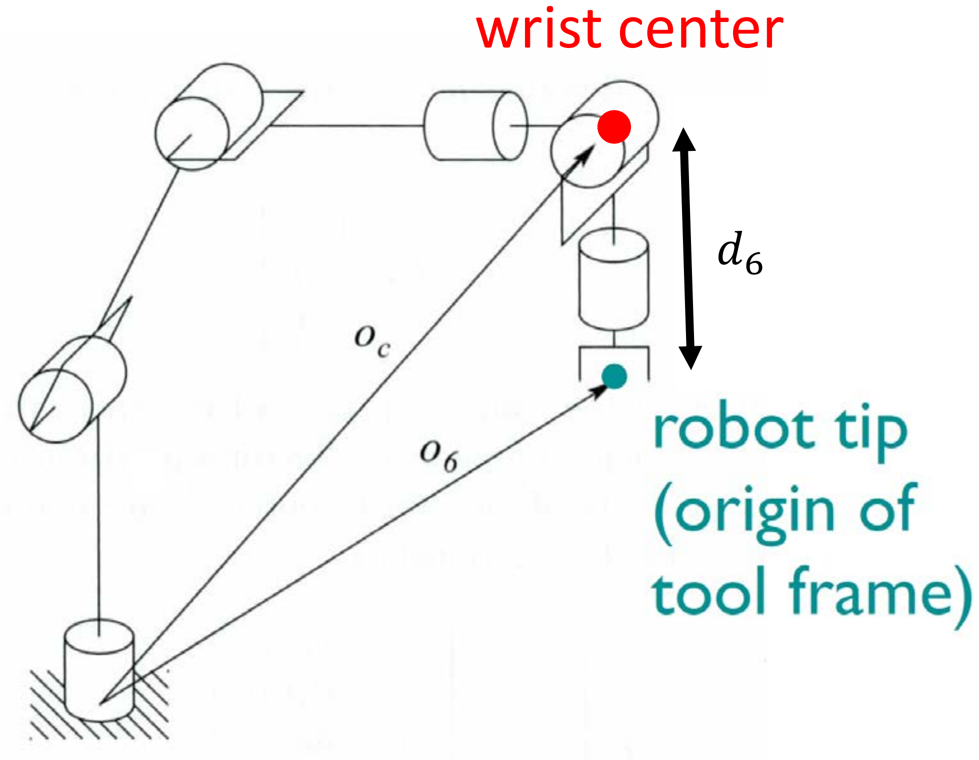


Trick: Exploit the kinematic structure of the manipulator.
If the robot has a spherical wrist, use **Kinematic Decoupling**.

1) Inverse Position

$$o = o_6^0 = o_c^0 + d_6 \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$o_c^0 = o - d_6 \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$



given $\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

position

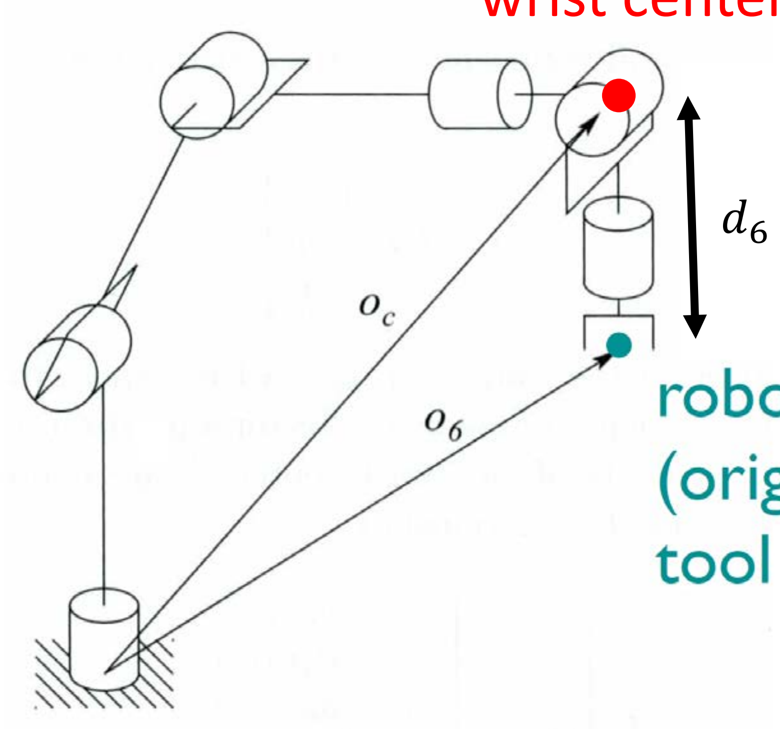
Solve for the joint variables that will put the wrist center in the correct position.

Only joints 1, 2, and 3!

Trick: Exploit the kinematic structure of the manipulator.
If the robot has a spherical wrist, use **Kinematic Decoupling**.

2) Inverse Orientation

wrist center



robot tip (origin of tool frame)

given $\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$

Solve for the joint variables that will put the end-effector at the correct orientation.

All joints (1-6) affect orientation.

BUT we've already solved for joints 1, 2, and 3!

orientation from joints 1,2,3

$$\mathbf{R} = \mathbf{R}_3^0 \mathbf{R}_6^3$$

desired orientation

orientation from joints 4,5,6

$$\mathbf{R}_6^3 = (\mathbf{R}_3^0)^{-1}\mathbf{R} = (\mathbf{R}_3^0)^{\mathrm{T}}\mathbf{R}$$

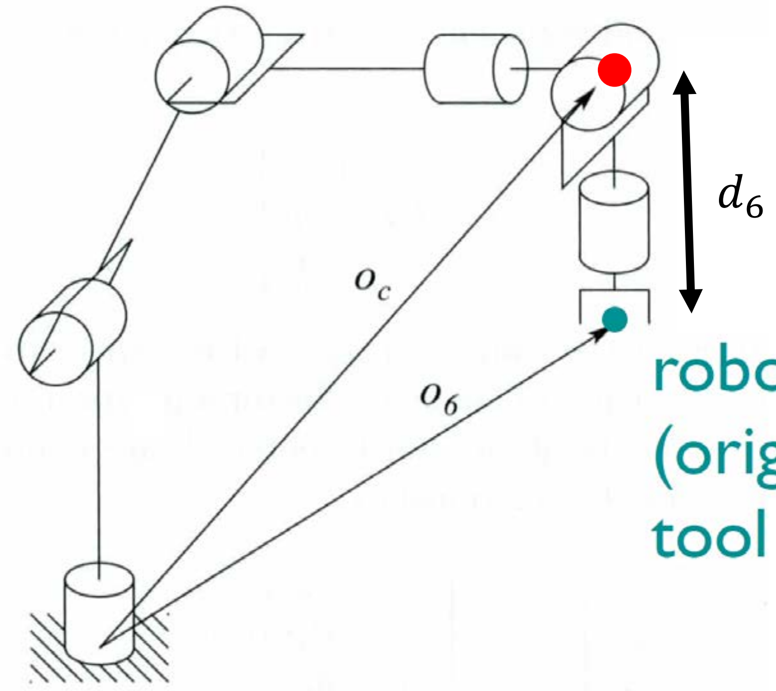
orientation

Given $\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$ and a certain manipulator with n joints,
find $q_1,\ldots,q_n$ such that $\mathbf{T}_n^0(q_1, \ldots, q_n) = \mathbf{H}$

UGLY!

Kinematic Decoupling

wrist center



TODAY

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

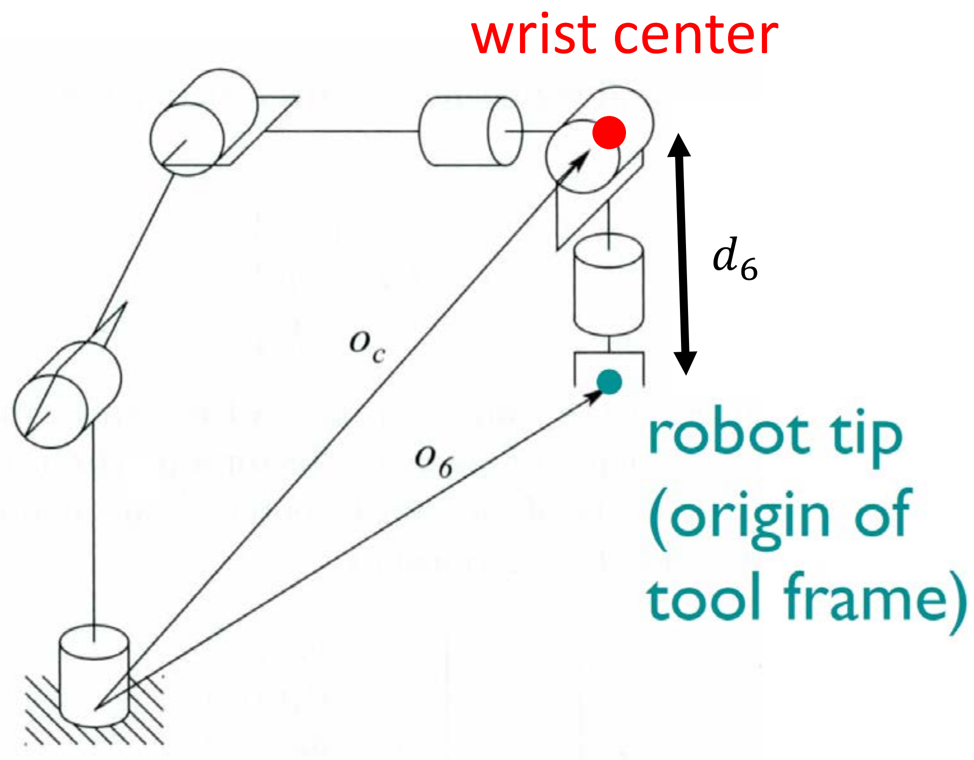
position

NEXT TIME

$$\mathbf{R}_6^3 = (\mathbf{R}_3^0)^{-1}\mathbf{R} = (\mathbf{R}_3^0)^{\mathrm{T}}\mathbf{R}$$

orientation

Inverse Position



given $\mathbf{H} = \begin{bmatrix} \mathbf{R} & o \\ 0 & 1 \end{bmatrix}$

tip of robot

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix}$$

wrist center

vector between them

We want closed-form solutions (explicit equations):

$$q_k = f_k(h_{11}, \dots, h_{34}), \qquad k = 1, \dots, n$$

What is the alternative?

Numerical solution, where an algorithm iteratively looks for a solution to the system of equations.

Why do we want closed-form solutions?

1. IK must often be solved in real-time (e.g., every 20ms). Closed-form solutions evaluate **faster** and are **guaranteed** to yield an answer.
2. There are generally **multiple solutions** to IK. Closed-form solutions allow you to make rules for choosing the best response.

Technique 1: Algebraic Decomposition

Given forward transformation matrix for a manipulator

$$\mathbf{T}_n^0 = \begin{bmatrix} \left[\mathbf{R}_n^0(\mathbf{q})\right]_{3\times 3} & \left[\mathbf{d}_n^0(\mathbf{q})\right]_{3\times 1} \\ [\mathbf{0}]_{1\times 3} & 1 \end{bmatrix}$$

solve the system of 3 equations from the displacement vector

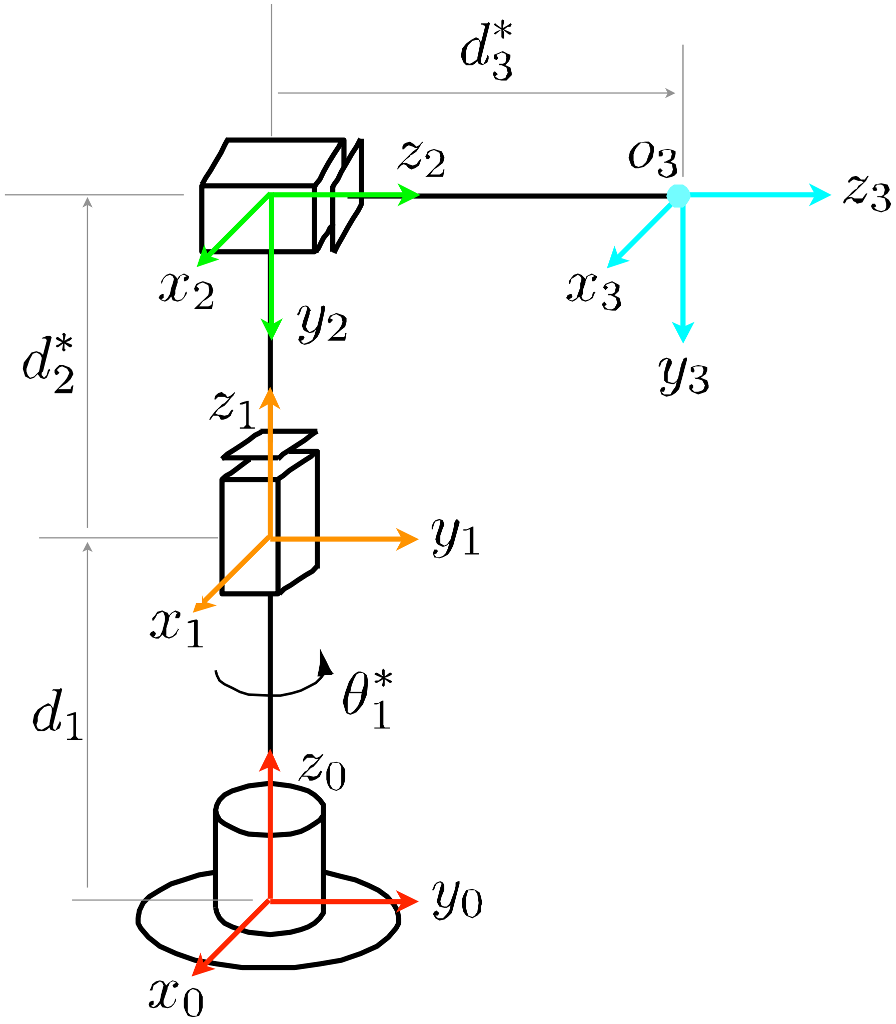
$$\begin{aligned} d_x &= \left[\mathbf{d}_n^0(\mathbf{q})\right]_1 \\ d_y &= \left[\mathbf{d}_n^0(\mathbf{q})\right]_2 \\ d_z &= \left[\mathbf{d}_n^0(\mathbf{q})\right]_3 \end{aligned}$$

to find the joint variables in terms of the end-effector position

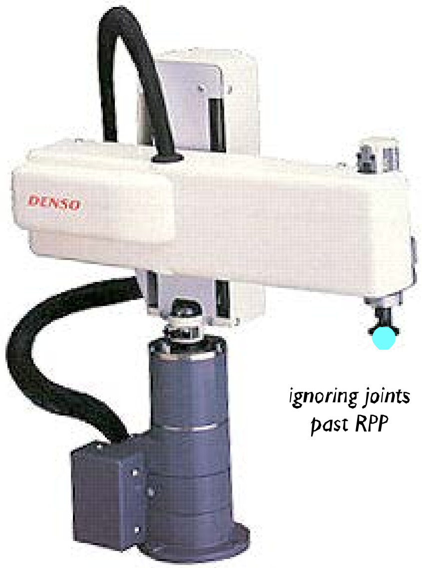
$$\mathbf{q} = \begin{bmatrix} q_1(d_x, d_y, d_z) \\ q_2(d_x, d_y, d_z) \\ \vdots \\ q_n(d_x, d_y, d_z) \end{bmatrix}$$

The RPP Cylindrical Robot

Let's solve it together.



$$o_3^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



	x-step		z-step	
i	a	α	d	θ
1	0	0°	d_1	θ_1^*
2	0	-90°	d_2^*	0°
3	0	0°	d_3^*	0°

$$\theta_1^* = ?$$

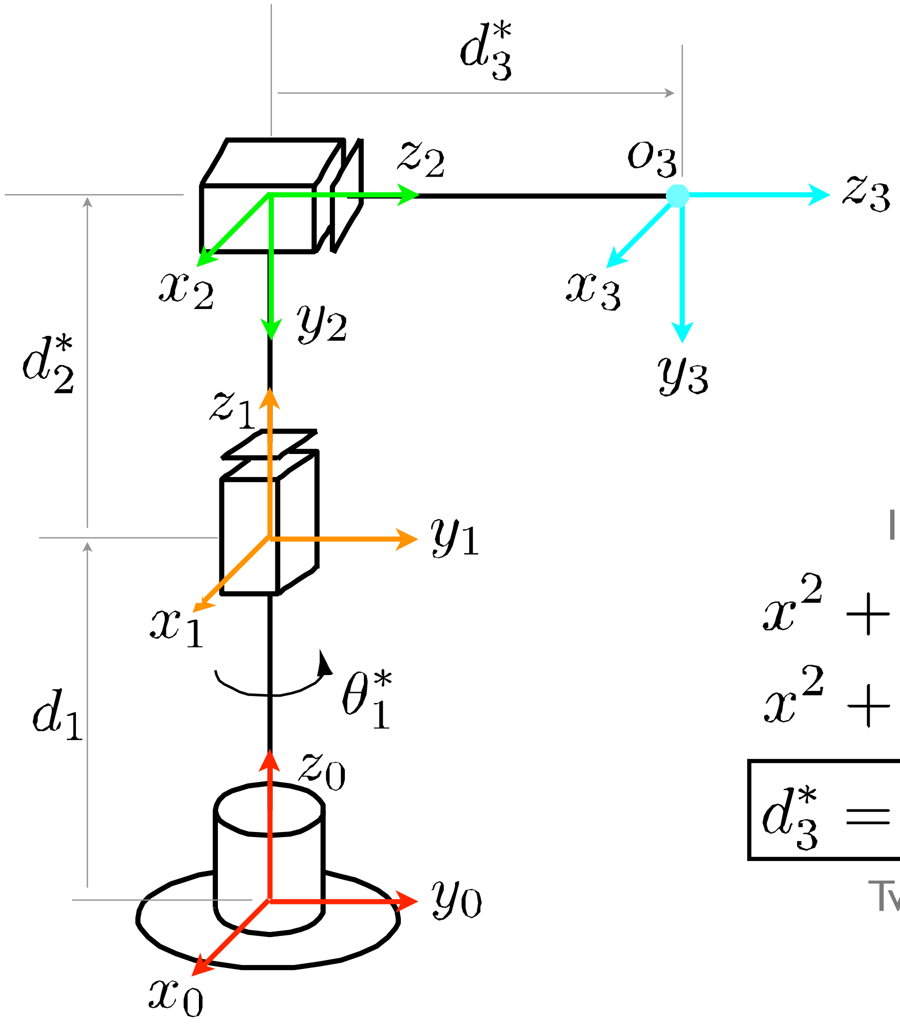
$$d_2^* = ?$$

$$d_3^* = ?$$

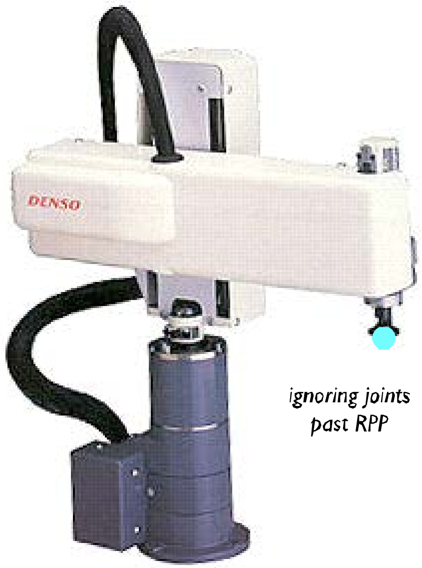
$$\mathbf{T}_3^0 = \begin{bmatrix} c_1^* & 0 & -s_1^* & -d_3^* s_1^* \\ s_1^* & 0 & c_1^* & d_3^* c_1^* \\ 0 & -1 & 0 & d_1 + d_2^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The RPP Cylindrical Robot

Let's solve it together.



$$o_3^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



ignoring joints past RPP

$$\begin{cases} x = -d_3^* \sin(\theta_1^*) \\ y = d_3^* \cos(\theta_1^*) \end{cases}$$

$$z = d_1 + d_2^* \longrightarrow \boxed{d_2^* = z - d_1}$$

1. Square both x and y equations and add them.

$$x^2 + y^2 = (d_3^*)^2 \sin^2(\theta_1^*) + (d_3^*)^2 \cos^2(\theta_1^*)$$

$$x^2 + y^2 = (d_3^*)^2$$

$$\boxed{d_3^* = \pm\sqrt{x^2 + y^2}}$$

Two solutions!

2. Solve for sin and cos of theta1 and take atan2

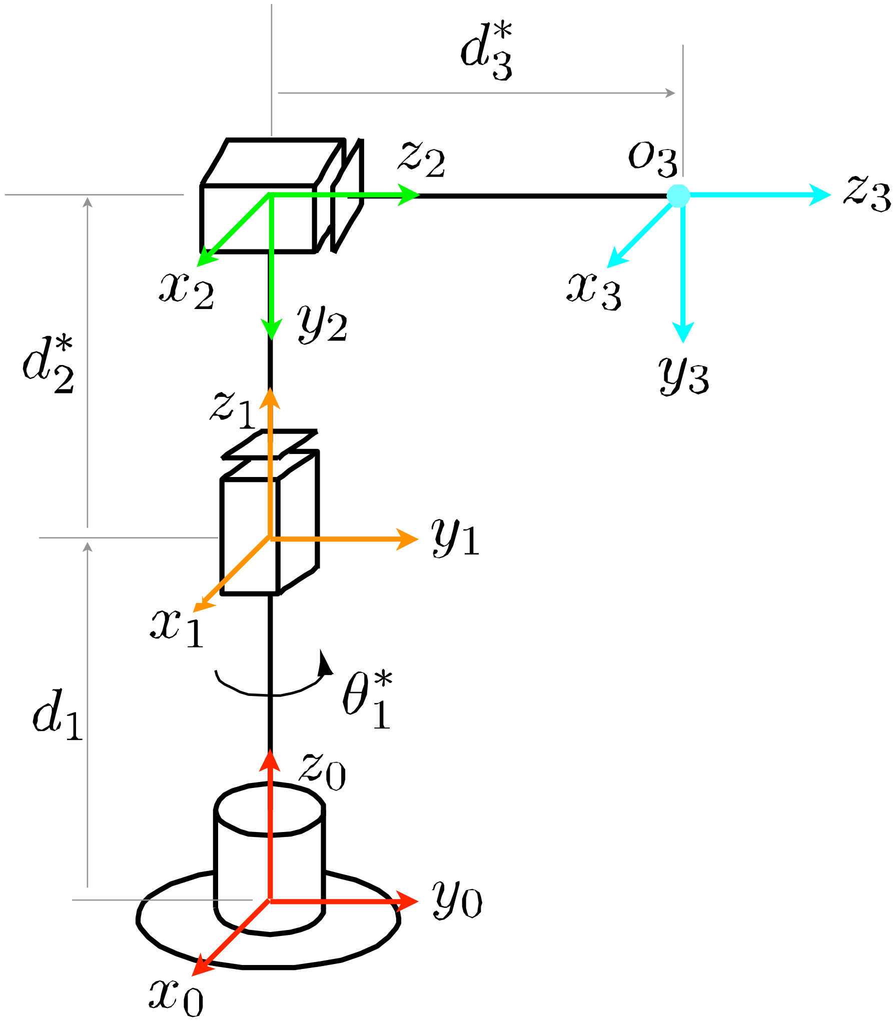
$$\sin(\theta_1^*) = \frac{-x}{d_3^*} \qquad \cos(\theta_1^*) = \frac{y}{d_3^*}$$

One solution for each choice of d3*

$$\boxed{\theta_1^* = \operatorname{atan2}\left(\frac{-x/d_3^*}{y/d_3^*}\right)}$$

The RPP Cylindrical Robot

Let's solve it together.



$$o_3^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

ignoring joints past RPP

$$\boxed{\theta_1^* = \operatorname{atan2}\left(\frac{-x/d_3^*}{y/d_3^*}\right)}$$

$$\boxed{d_2^* = z - d_1}$$

$$\boxed{d_3^* = \pm\sqrt{x^2 + y^2}}$$

Q: Is there always a solution?

Technique 2: Geometric Analysis

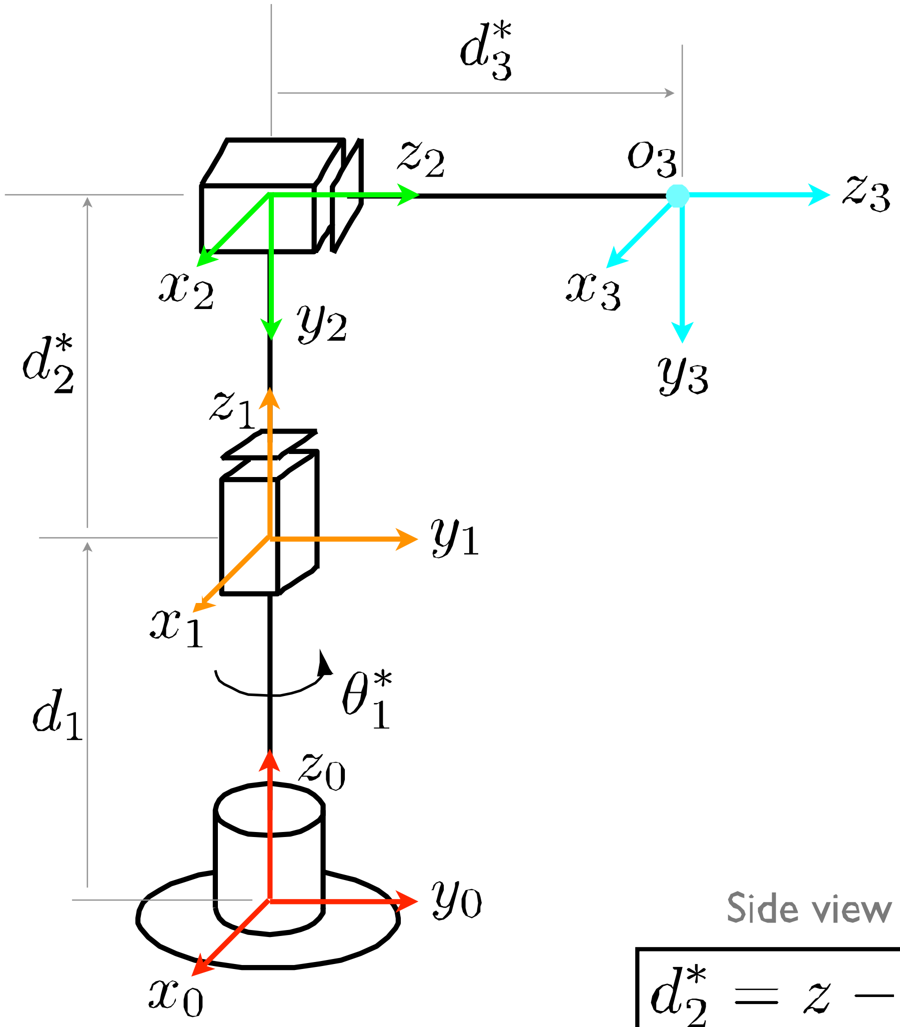
For most simple manipulators, it is easier to use geometry to solve for closed-form solutions to IK.

Solve for each joint angle by projecting the manipulator onto the $x_{i\text{-}1},y_{i\text{-}1}$ plane

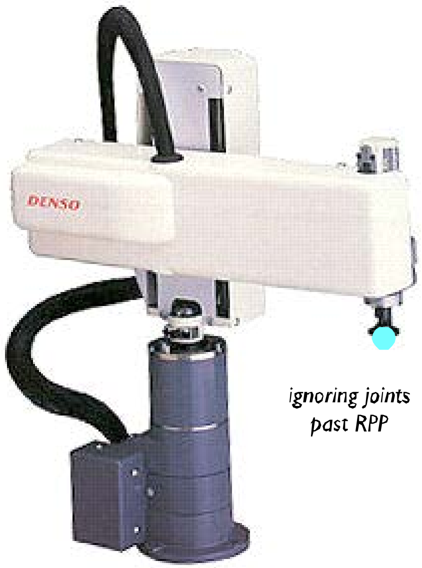
Solve for each joint displacement by projecting the manipulator onto the $x_{i\text{-}1},z_{i\text{-}1}$ or $y_{i\text{-}1},z_{i\text{-}1}$ plane

Closed-form IK solutions are not always possible, and if it is solvable, there are often multiple solutions

The RPP Cylindrical Robot

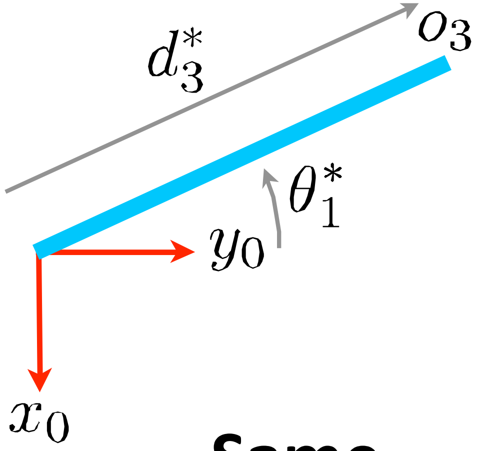


$$o_3^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



ignoring joints past RPP

Top view (looking down along z0).
Draw for a small positive angle theta1.



$$\theta_1^* = ?$$

$$d_2^* = ?$$

$$d_3^* = ?$$

$$\boxed{d_3^* = \pm\sqrt{x^2 + y^2}}$$

Side view

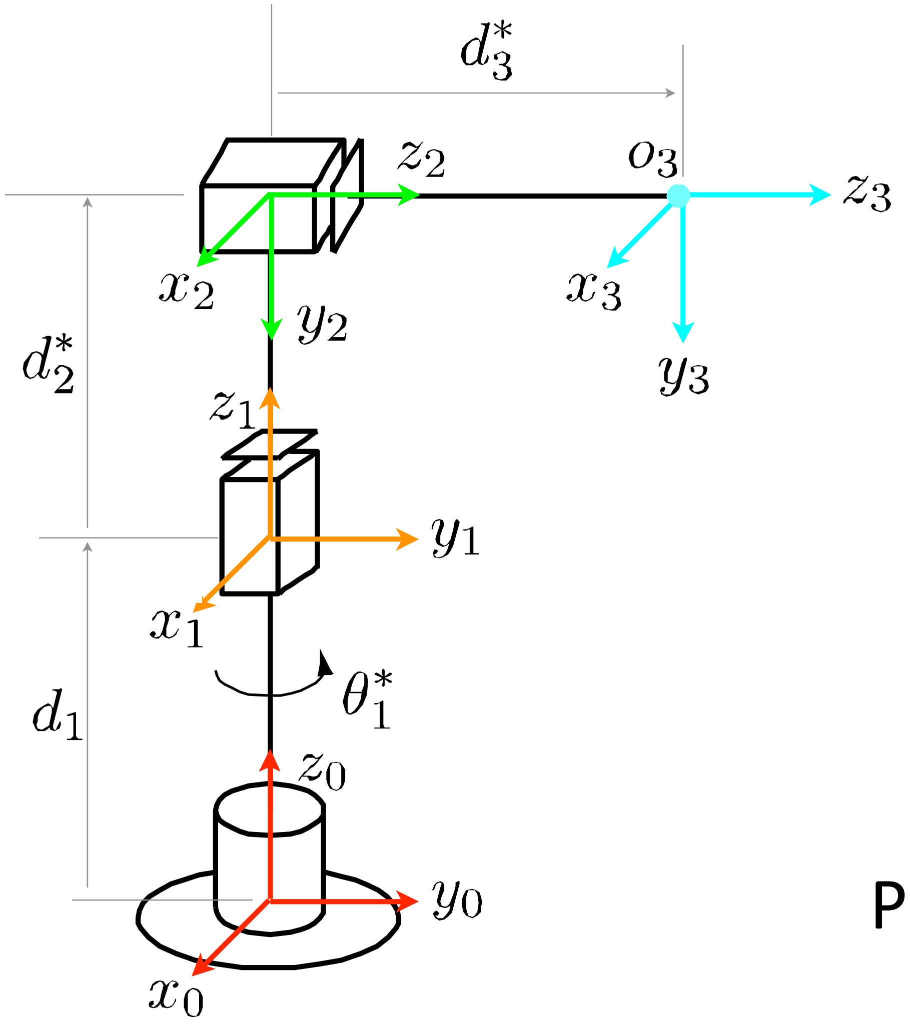
$$\boxed{d_2^* = z - d_1}$$

Same answers!

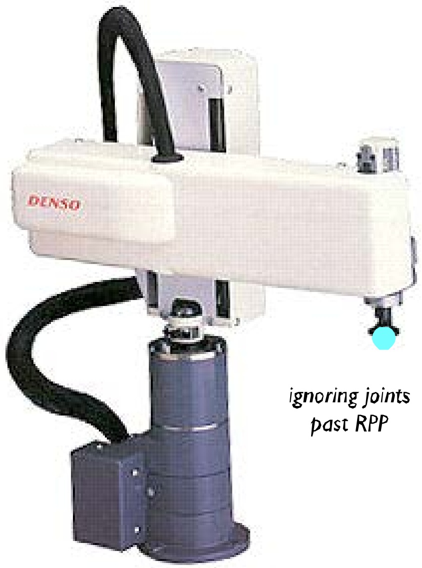
$$\boxed{\theta_1^* = \operatorname{atan2}\left(\frac{-x/d_3^*}{y/d_3^*}\right)}$$

What is the geometric meaning of the second theta1 solution?

The RPP Cylindrical Robot



$$o_3^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



ignoring joints past RPP

$$\boxed{\theta_1^* = \text{atan2}\left(\frac{-x/d_3^*}{y/d_3^*}\right)}$$

$$\boxed{d_2^* = z - d_1}$$

$$\boxed{d_3^* = \pm\sqrt{x^2 + y^2}}$$

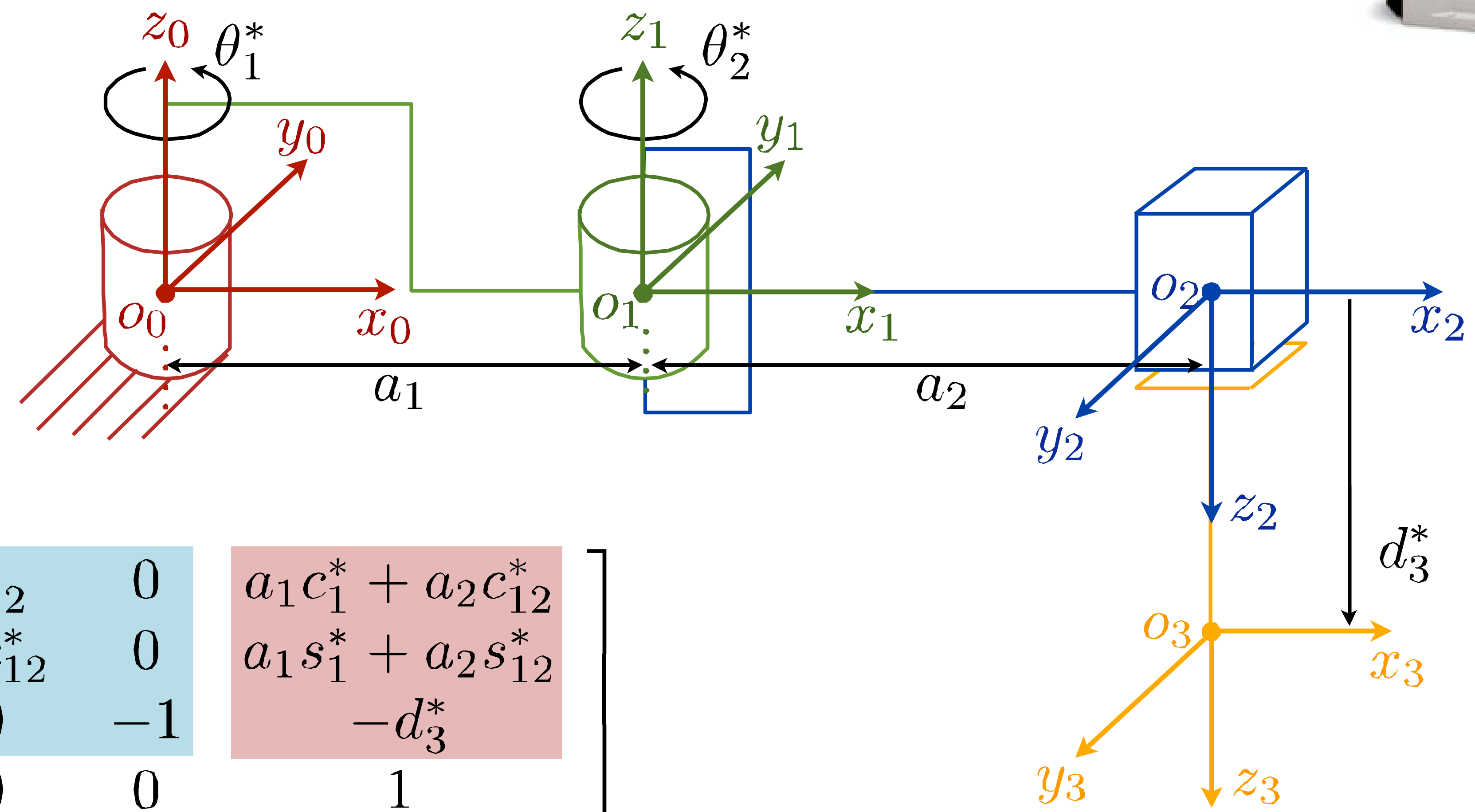
Q: How do I check my answers?

Plug into FK!

$$\mathbf{T}_3^0 = \begin{bmatrix} c_1^* & 0 & -s_1^* & -d_3^* s_1^* \\ s_1^* & 0 & c_1^* & d_3^* c_1^* \\ 0 & -1 & 0 & d_1 + d_2^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3D example: SCARA Robot

What joint angles should we choose to put the end effector at the desired position?



$$T_3^0 = \begin{bmatrix} c_{12}^* & s_{12}^* & 0 & a_1 c_1^* + a_2 c_{12}^* \\ s_{12}^* & -c_{12}^* & 0 & a_1 s_1^* + a_2 s_{12}^* \\ 0 & 0 & -1 & -d_3^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3D example: SCARA Robot

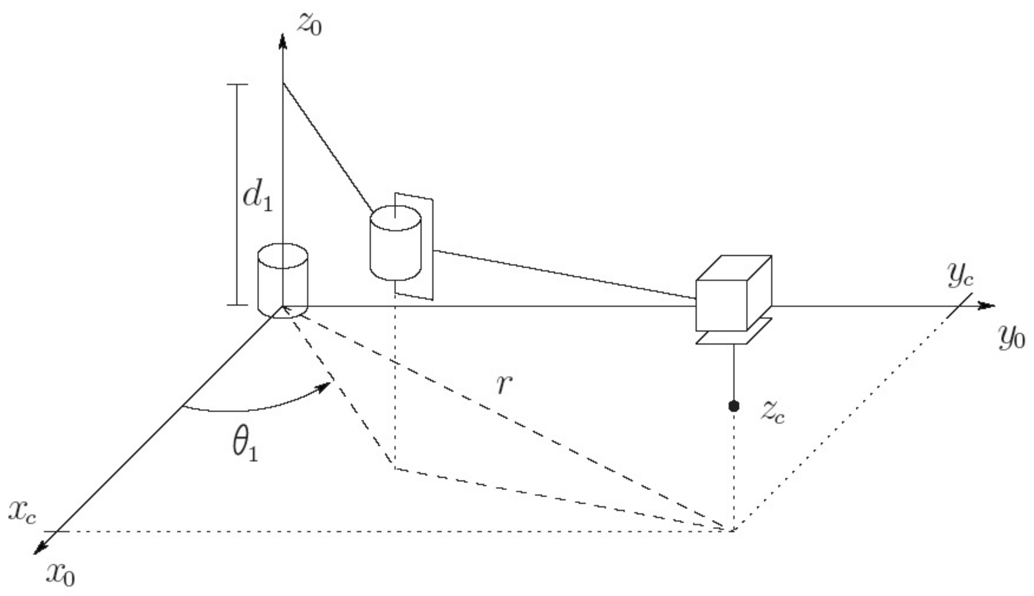


Figure 3.22: SCARA manipulator.

$$z = -d_3^* \longrightarrow \boxed{d_3^* = -z}$$

Use the geometric IK method to find theta1 and theta2.

$$T_3^0 = \begin{bmatrix} c_{12}^* & s_{12}^* & 0 & a_1 c_1^* + a_2 c_{12}^* \\ s_{12}^* & -c_{12}^* & 0 & a_1 s_1^* + a_2 s_{12}^* \\ 0 & 0 & -1 & -d_3^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3D example: SCARA Robot

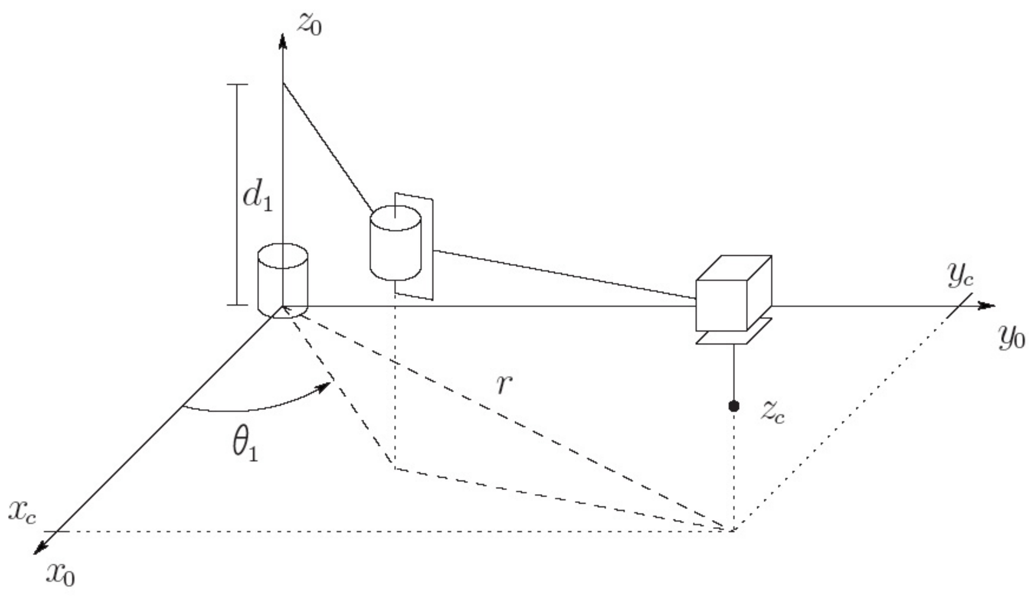
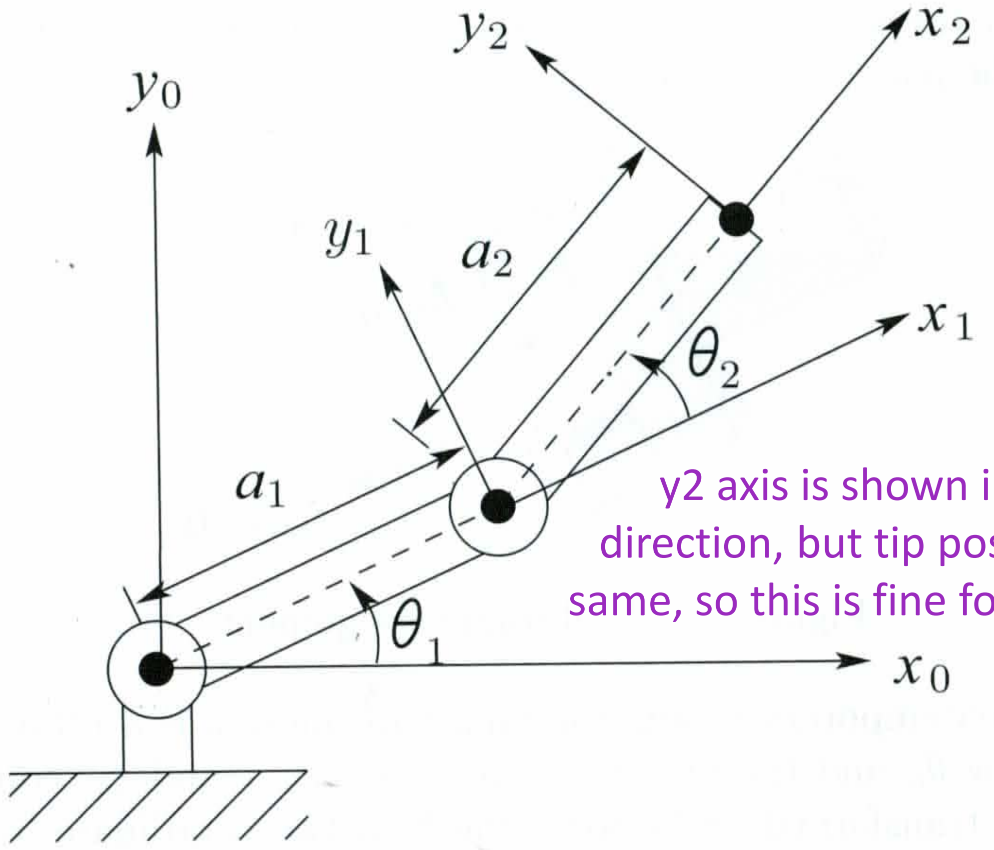


Figure 3.22: SCARA manipulator.

$$T_3^0 = \begin{bmatrix} c_{12}^* & s_{12}^* & 0 & a_1 c_1^* + a_2 c_{12}^* \\ s_{12}^* & -c_{12}^* & 0 & a_1 s_1^* + a_2 s_{12}^* \\ 0 & 0 & -1 & -d_3^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

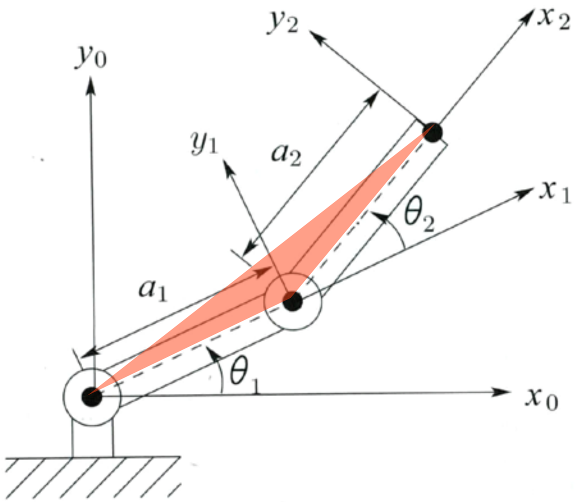
Top view of SCARA, looking along z0 and z1

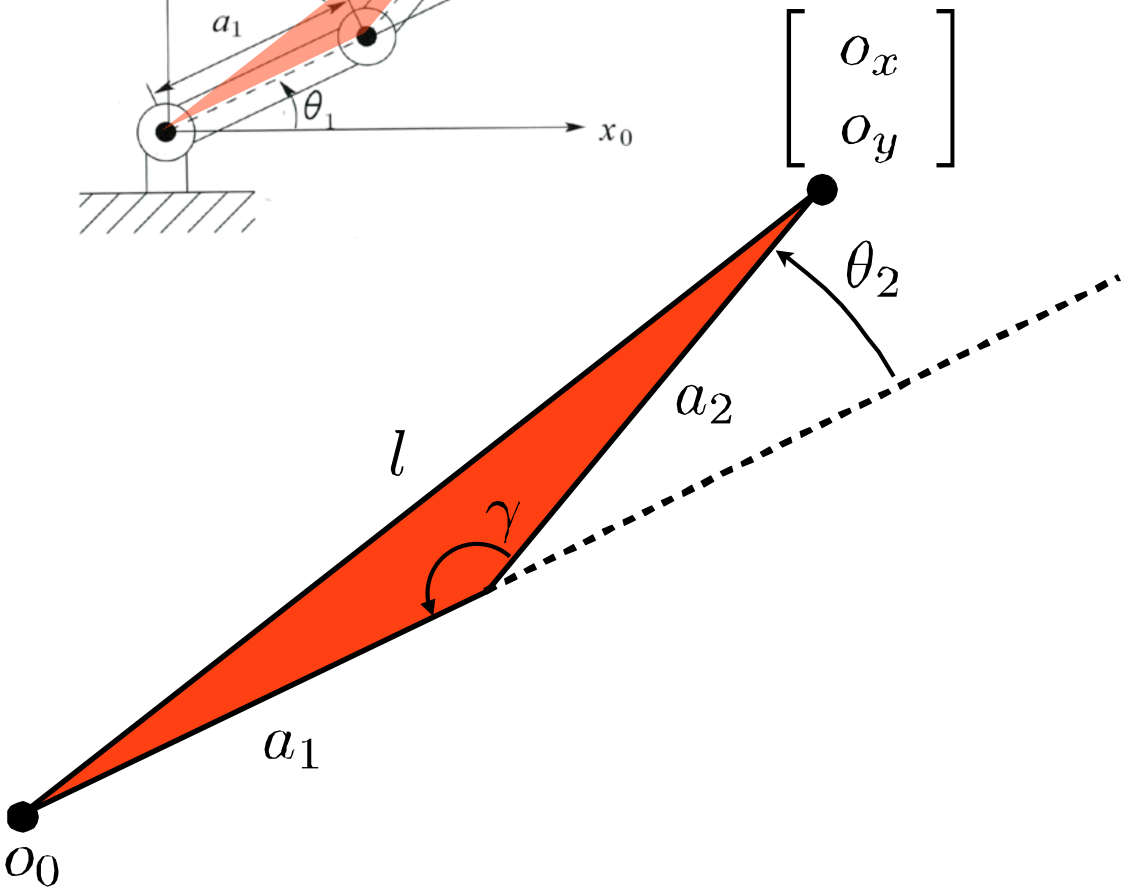
Same as a planar RR!



y2 axis is shown in wrong direction, but tip position is the same, so this is fine for position IK.

Planar RR





$$\boxed{l^2 = o_x^2 + o_y^2}$$

Law of cosines:

$$c^2 = a^2 + b^2 - 2ab\cos\gamma$$

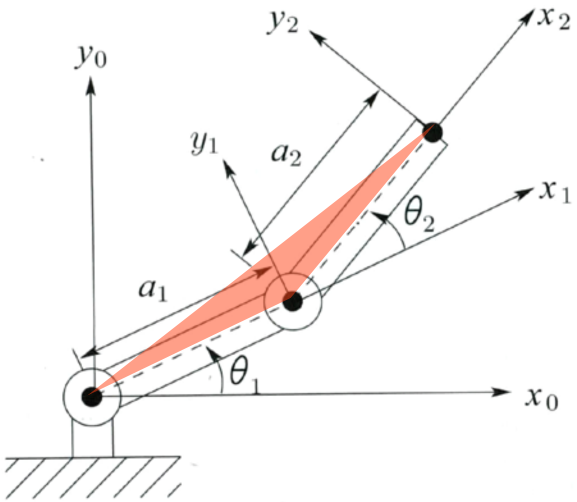
$$\boxed{l^2} = a_1^2 + a_2^2 - 2a_1 a_2 \cos(\pi - \theta_2)$$

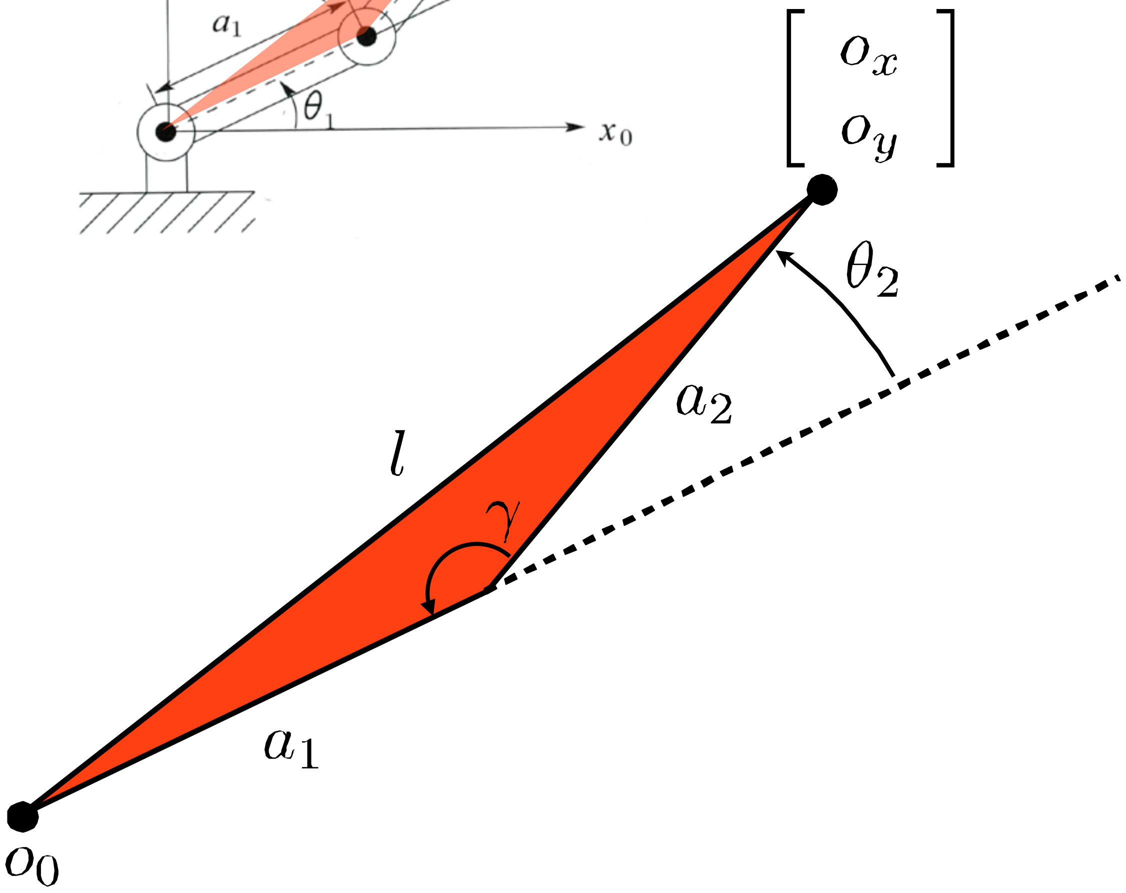
NOTE: Law of Cosines is wrong in SHV Appendix A.

SHV: $c^2 = a^2 + cb^2 - 2ab\cos\gamma$

should be: $c^2 = a^2 + b^2 - 2ab\cos\gamma$

Planar RR





$$l^2 = o_x^2 + o_y^2$$

Law of cosines:

$$c^2 = a^2 + b^2 - 2ab\cos\gamma$$

$$l^2 = a_1^2 + a_2^2 - 2a_1a_2\cos(\pi - \theta_2)$$

$$o_x^2 + o_y^2 = a_1^2 + a_2^2 - 2a_1a_2\cos(\pi - \theta_2)$$

Cosine angle difference identity:

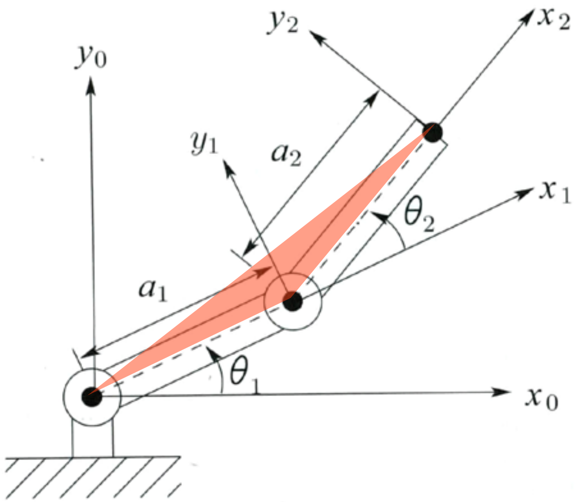
$$\cos(\alpha - \beta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta$$

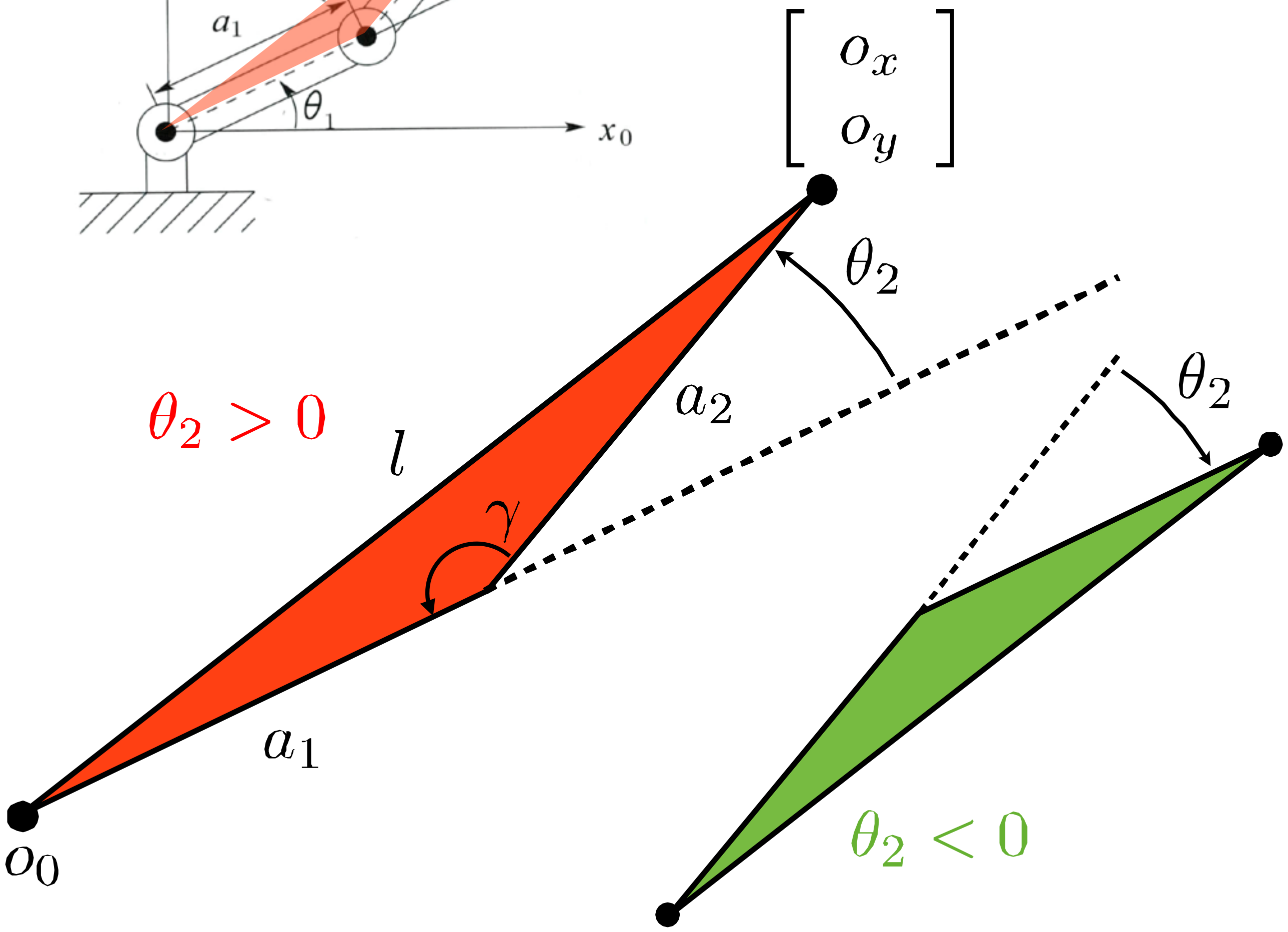
$$\cos\pi = -1 \qquad \sin\pi = 0$$

$$\cos(\pi - \theta_2) = -\cos\theta_2$$

$$o_x^2 + o_y^2 = a_1^2 + a_2^2 + 2a_1a_2\cos\theta_2$$

Planar RR





$$o_x^2 + o_y^2 = a_1^2 + a_2^2 + 2a_1a_2\cos\theta_2$$

$$\cos\theta_2 = \frac{o_x^2 + o_y^2 - a_1^2 - a_2^2}{2a_1a_2}$$

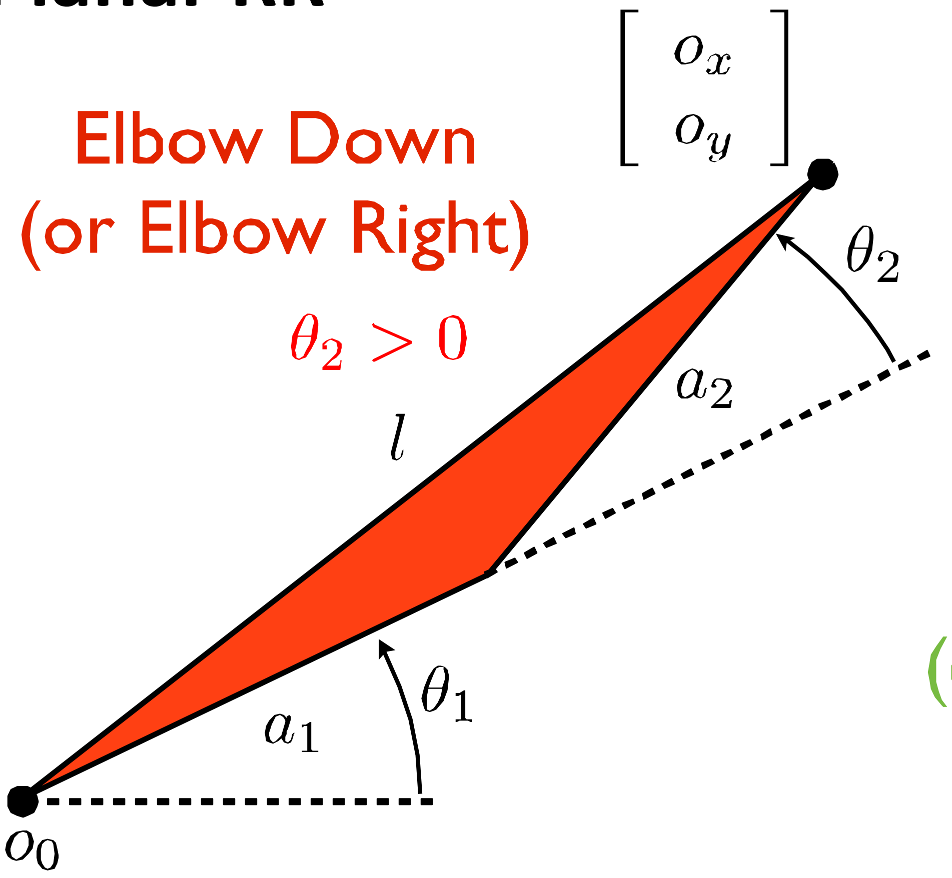
$$\boxed{\theta_2 = \cos^{-1}\left(\frac{o_x^2 + o_y^2 - a_1^2 - a_2^2}{2a_1a_2}\right)}$$

How many solutions are there?

Two: one positive, one negative

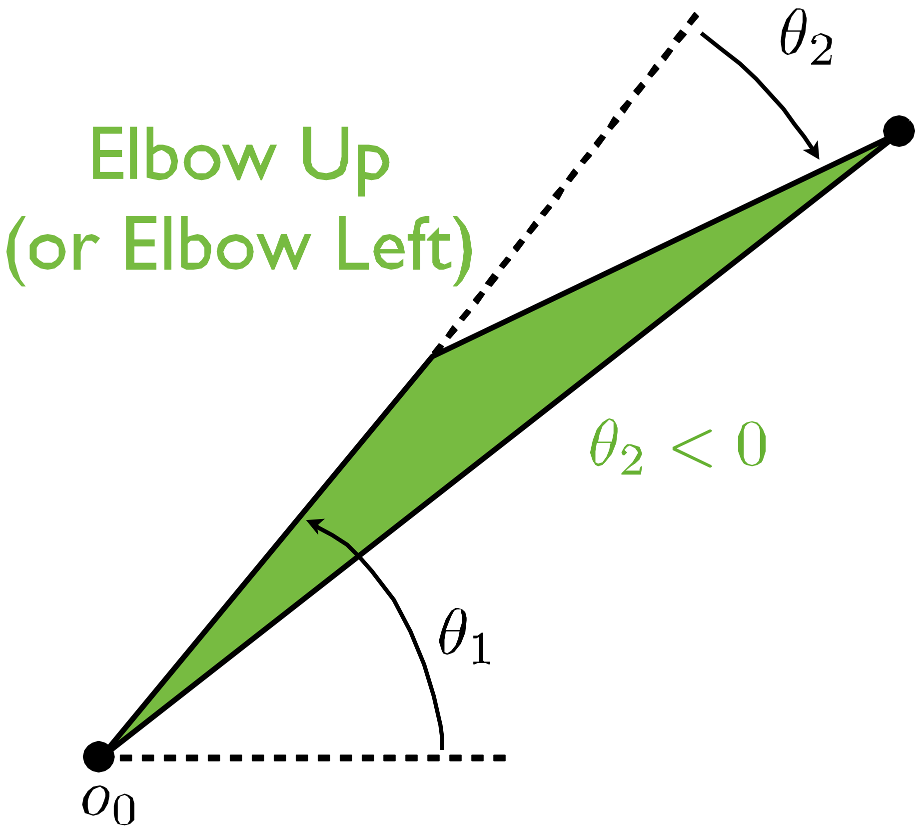
These two solutions are commonly called “elbow down” and “elbow up”

Planar RR



Q: Should the value of theta1 be the same for these 2 solutions?

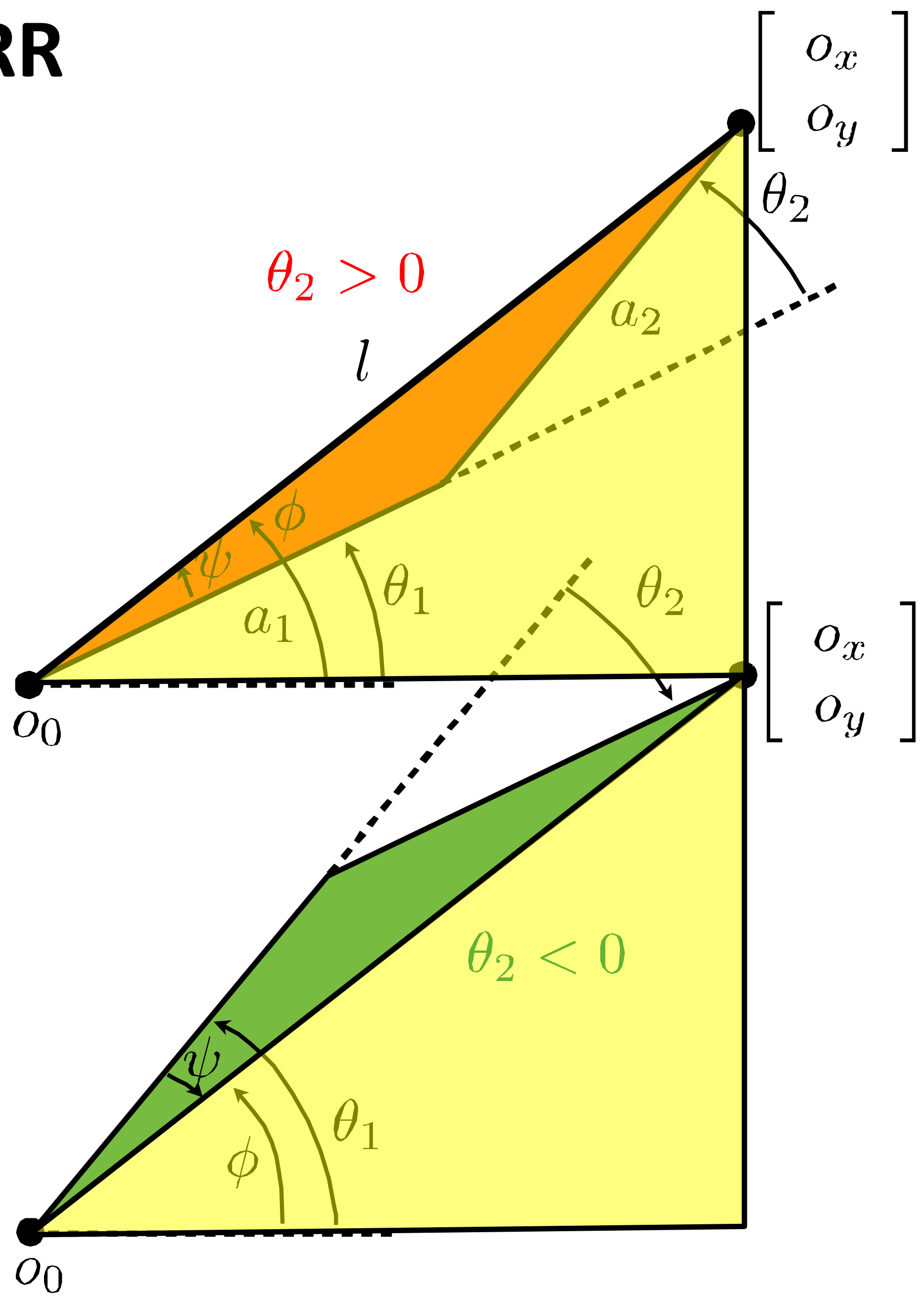
No. It depends on theta2.

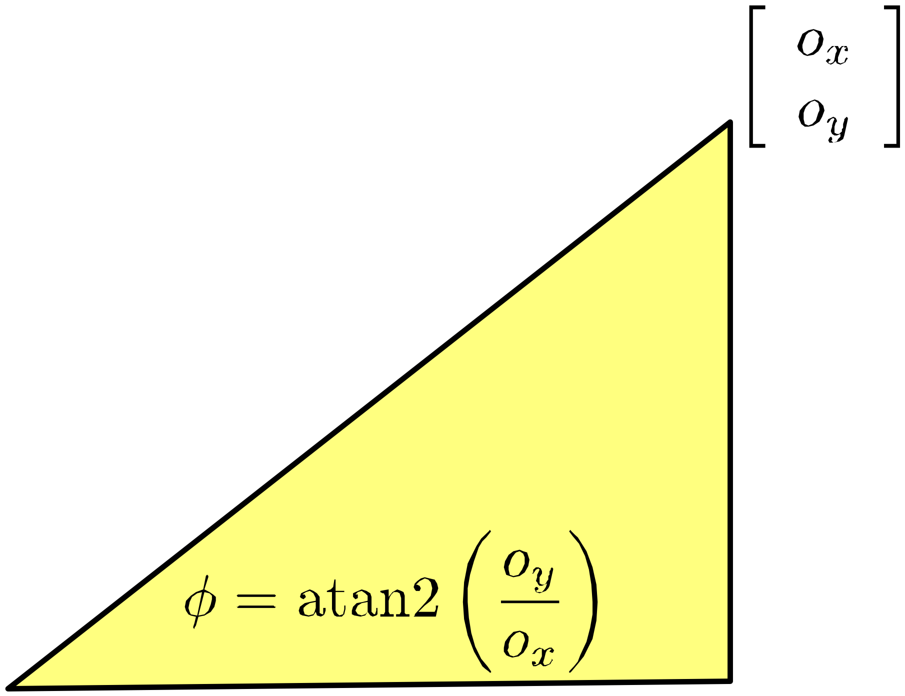


Planar RR

$\theta_1 = \phi - \psi$

$\phi = ?$

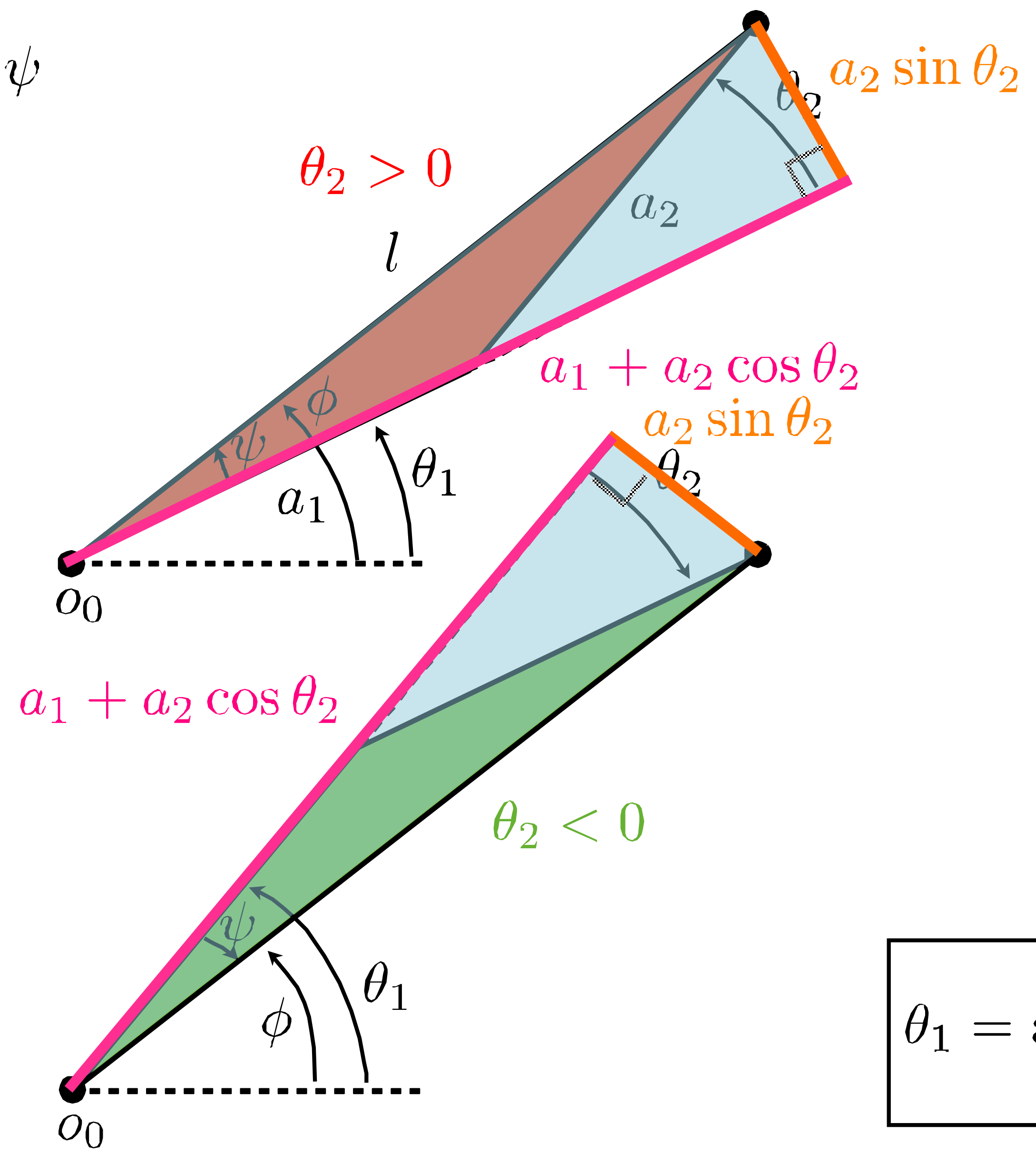


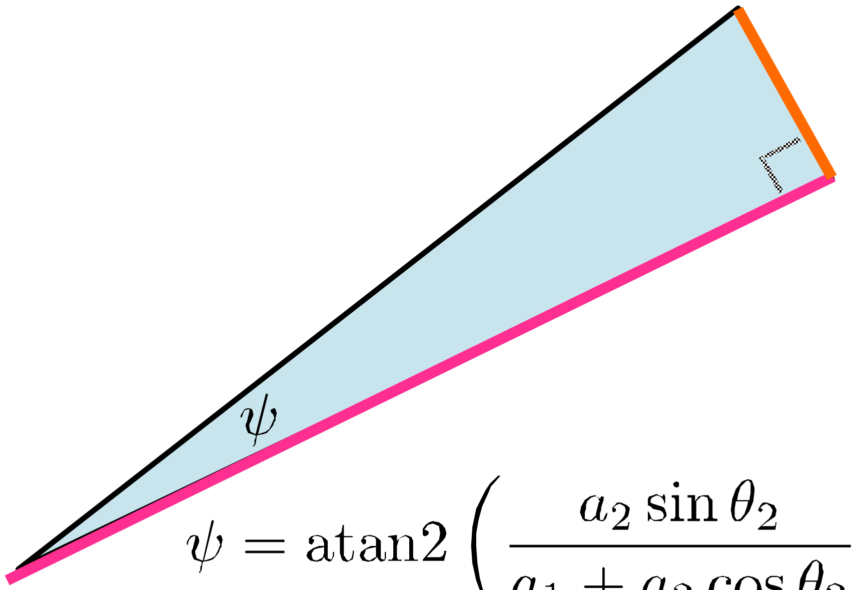


Planar RR

$\theta_1 = \phi - \psi$

$\psi = ?$



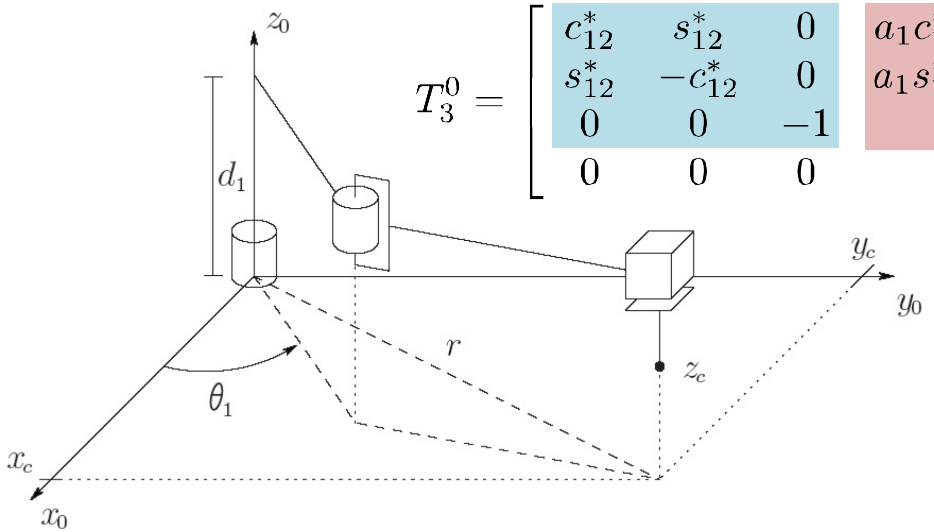


$$\psi = \operatorname{atan2}\left(\frac{a_2\sin\theta_2}{a_1 + a_2\cos\theta_2}\right)$$

$$\theta_1 = \phi - \psi$$

$$\boxed{\theta_1 = \operatorname{atan2}\left(\frac{o_y}{o_x}\right) - \operatorname{atan2}\left(\frac{a_2\sin\theta_2}{a_1 + a_2\cos\theta_2}\right)}$$

3D example: SCARA Robot



$$T_3^0 = \begin{bmatrix} c_{12}^* & s_{12}^* & 0 & a_1 c_1^* + a_2 c_{12}^* \\ s_{12}^* & -c_{12}^* & 0 & a_1 s_1^* + a_2 s_{12}^* \\ 0 & 0 & -1 & -d_3^* \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{d_3^* = -z}$$

$$\boxed{\theta_2 = \cos^{-1}\left(\frac{o_x^2 + o_y^2 - a_1^2 - a_2^2}{2a_1 a_2}\right)}$$

$$\boxed{\theta_1 = \operatorname{atan2}\left(\frac{o_y}{o_x}\right) - \operatorname{atan2}\left(\frac{a_2 \sin\theta_2}{a_1 + a_2\cos\theta_2}\right)}$$

Example 3.10 SCARA Manipulator

As another example, we consider the SCARA manipulator whose forward kinematics is defined by T_4^0 from (3.30). The inverse kinematics solution is then given as the set of solutions of the equation

$$\begin{aligned} T_4^0 &= \begin{bmatrix} R & o \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_{12}c_4 + s_{12}s_4 & s_{12}c_4 - c_{12}s_4 & 0 & a_1c_1 + a_2c_{12} \\ s_{12}c_4 - c_{12}s_4 & -c_{12}c_4 - s_{12}s_4 & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & -1 & -d_3 - d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{3.81}$$

We first note that, since the SCARA has only four degrees-of-freedom, not every possible H from $SE(3)$ allows a solution of (3.81). In fact we can easily see that there is no solution of (3.81) unless R is of the form

$$R = \begin{bmatrix} c_\alpha & s_\alpha & 0 \\ s_\alpha & -c_\alpha & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{3.82}$$

and if this is the case, the sum $\theta_1 + \theta_2 - \theta_4$ is determined by

$$\theta_1 + \theta_2 - \theta_4 = \alpha = \operatorname{atan2}(r_{11}, r_{12}) \tag{3.83}$$

Projecting the manipulator configuration onto the $x_0 - y_0$ plane immediately yields the situation of Figure 3.22. We see from this that

$$\theta_2 = \operatorname{atan2}\left(c_2, \pm\sqrt{1 - c_2}\right) \tag{3.84}$$

where

$$c_2 = \frac{o_x^2 + o_y^2 - a_1^2 - a_2^2}{2a_1a_2} \tag{3.85}$$

$$\theta_1 = \operatorname{atan2}(o_x, o_y) - \operatorname{atan2}(a_1 + a_2c_2, a_2s_2) \tag{3.86}$$

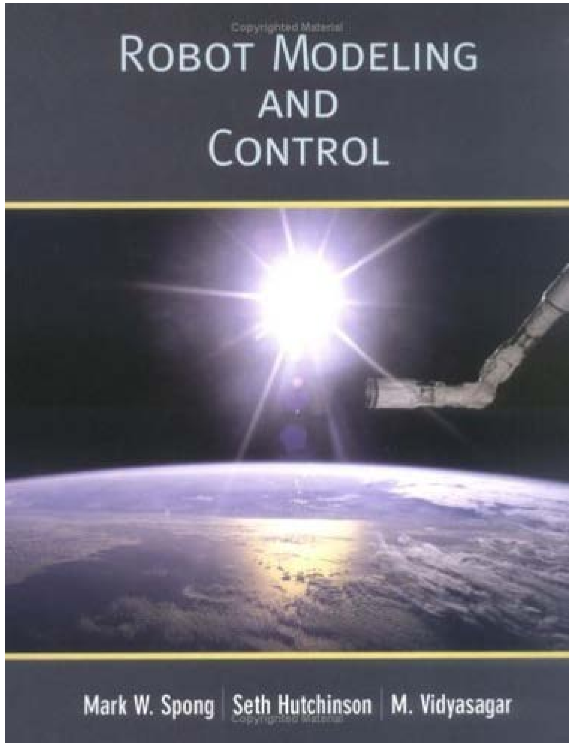
We may then determine θ_4 from (3.83) as

$$\begin{aligned} \theta_4 &= \theta_1 + \theta_2 - \alpha \\ &= \theta_1 + \theta_2 - \operatorname{atan2}(r_{11}, r_{12}) \end{aligned} \tag{3.87}$$

Finally d_3 is given as

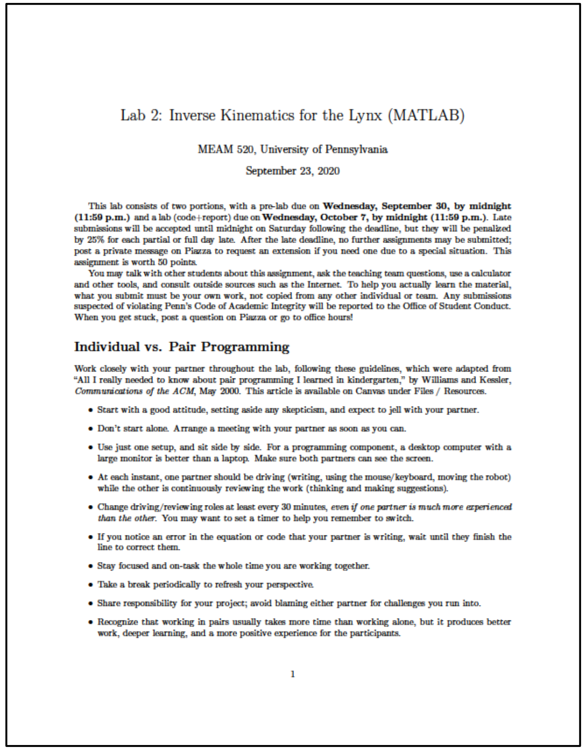
$$d_3 = o_z + d_4 \tag{3.88}$$

Next time: Inverse Orientation Kinematics



Chapter 3: Forward and Inverse Kinematics

- Read 3.3 – 3.4



Lab 2: Inverse Kinematics for the Lynx (MATLAB)

MEAM 520, University of Pennsylvania

September 23, 2020

This lab consists of two portions, with a pre-lab due on **Wednesday, September 30, by midnight (11:59 p.m.)** and a lab (code+report) due on **Wednesday, October 7, by midnight (11:59 p.m.)**. Late submissions will be accepted until midnight on Saturday following the deadline, but they will be penalized by 25% for each partial or full day late. After the late deadline, no further assignments may be submitted; post a private message on Piazza to request an extension if you need one due to a special situation. This assignment is worth 50 points.

You may talk with other students about this assignment, ask the teaching team questions, use a calculator and other tools, and consult outside sources such as the Internet. To help you actually learn the material, what you submit must be your own work, not copied from any other individual or team. Any submissions suspected of violating Penn's Code of Academic Integrity will be reported to the Office of Student Conduct. When you get stuck, post a question on Piazza or go to office hours!

Individual vs. Pair Programming

Work closely with your partner throughout the lab, following these guidelines, which were adapted from "All I really needed to know about pair programming I learned in kindergarten," by Williams and Kessler, *Communications of the ACM*, May 2000. This article is available on Canvas under Files / Resources.

- Start with a good attitude, setting aside any skepticism, and expect to jell with your partner.
- Don't start alone. Arrange a meeting with your partner as soon as you can.
- Use just one setup, and sit side by side. For a programming component, a desktop computer with a large monitor is better than a laptop. Make sure both partners can see the screen.
- At each instant, one partner should be driving (writing, using the mouse/keyboard, moving the robot) while the other is continuously reviewing the work (thinking and making suggestions).
- Change driving/reviewing roles at least every 30 minutes, *even if one partner is much more experienced than the other*. You may want to set a timer to help you remember to switch.
- If you notice an error in the equation or code that your partner is writing, wait until they finish the line to correct them.
- Stay focused and on-task the whole time you are working together.
- Take a break periodically to refresh your perspective.
- Share responsibility for your project; avoid blaming either partner for challenges you run into.
- Recognize that working in pairs usually takes more time than working alone, but it produces better work, deeper learning, and a more positive experience for the participants.

1

Lab 2: Inverse Kinematics for Lynx

- You can now do the pre-lab