MEAM 520

Rotation Matrices

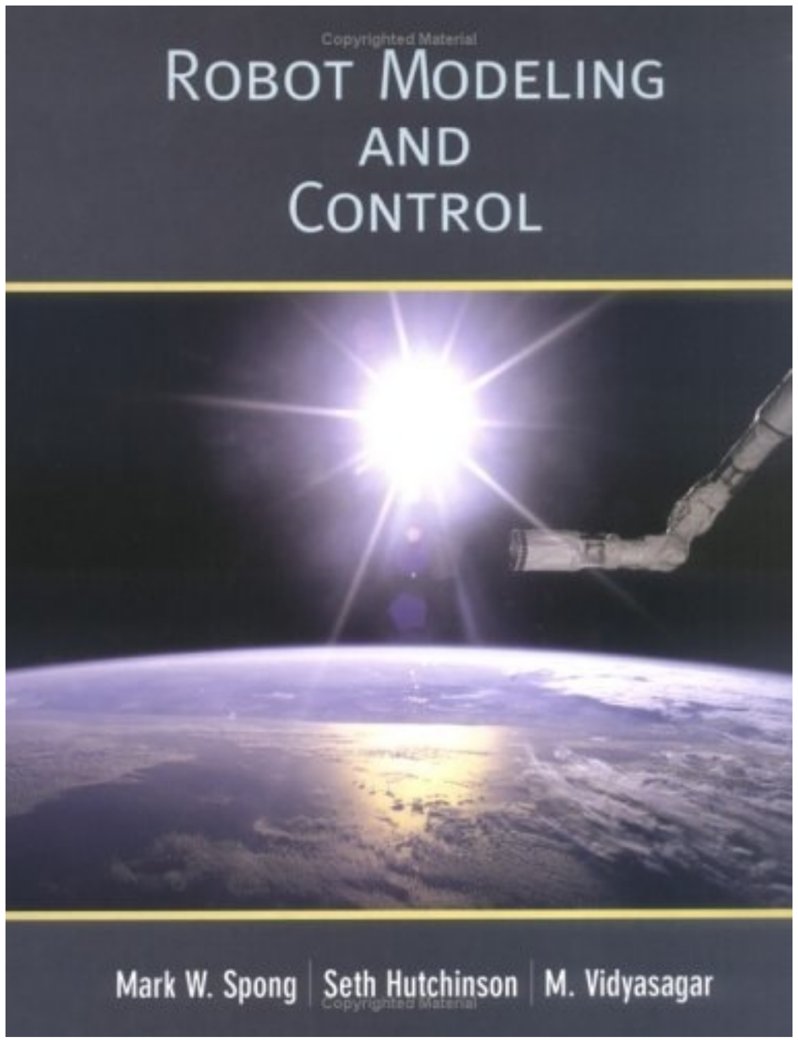
Katherine J. Kuchenbecker, Ph.D.
Mechanical Engineering and Applied Mechanics Department
SEAS, University of Pennsylvania

LEGES SINE MORIBUS VANAE

GRASP
LABORATORY

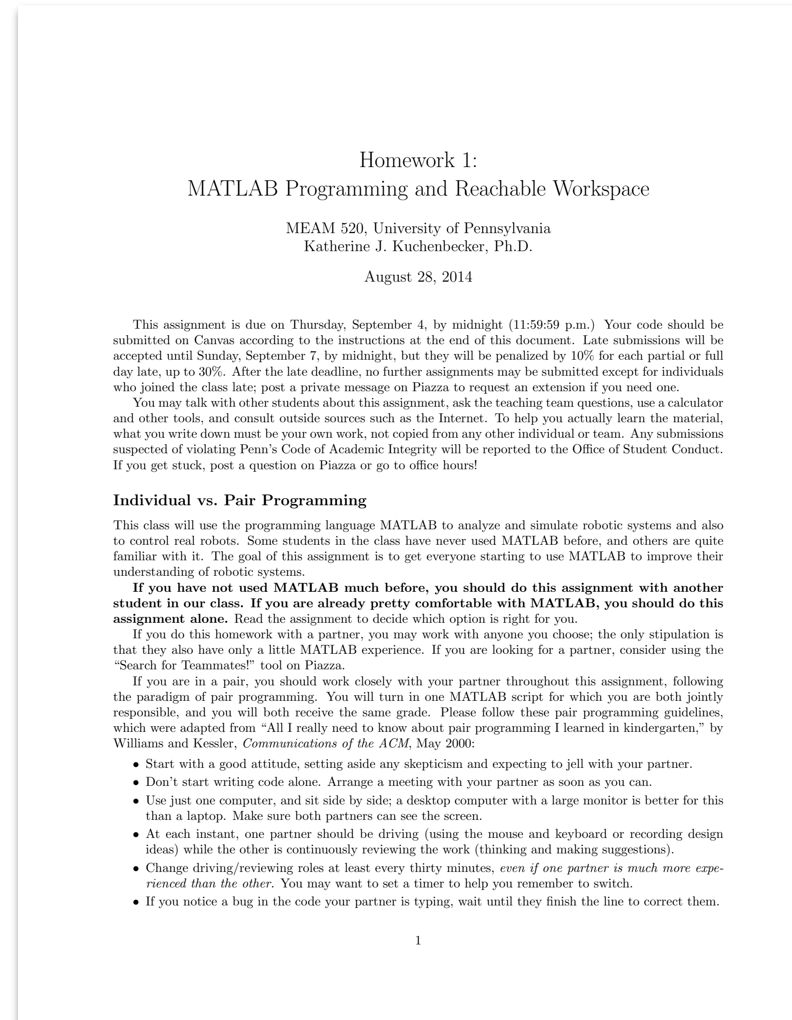
Lecture 3: September 4, 2014

Reading



For today,
review Sections B.1 & B.4
and read 2.1, 2.2, and 2.3.

Homework



Homework 1:
MATLAB Programming and Reachable Workspace

MEAM 520, University of Pennsylvania
Katherine J. Kuchenbecker, Ph.D.

August 28, 2014

This assignment is due on Thursday, September 4, by midnight (11:59:59 p.m.) Your code should be submitted on Canvas according to the instructions at the end of this document. Late submissions will be accepted until Sunday, September 7, by midnight, but they will be penalized by 10% for each partial or full day late, up to 30%. After the late deadline, no further assignments may be submitted except for individuals who joined the class late; post a private message on Piazza to request an extension if you need one.

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 write down 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. If you get stuck, post a question on Piazza or go to office hours!

Individual vs. Pair Programming

This class will use the programming language MATLAB to analyze and simulate robotic systems and also to control real robots. Some students in the class have never used MATLAB before, and others are quite familiar with it. The goal of this assignment is to get everyone starting to use MATLAB to improve their understanding of robotic systems.

If you have not used MATLAB much before, you should do this assignment with another student in our class. If you are already pretty comfortable with MATLAB, you should do this assignment alone. Read the assignment to decide which option is right for you.

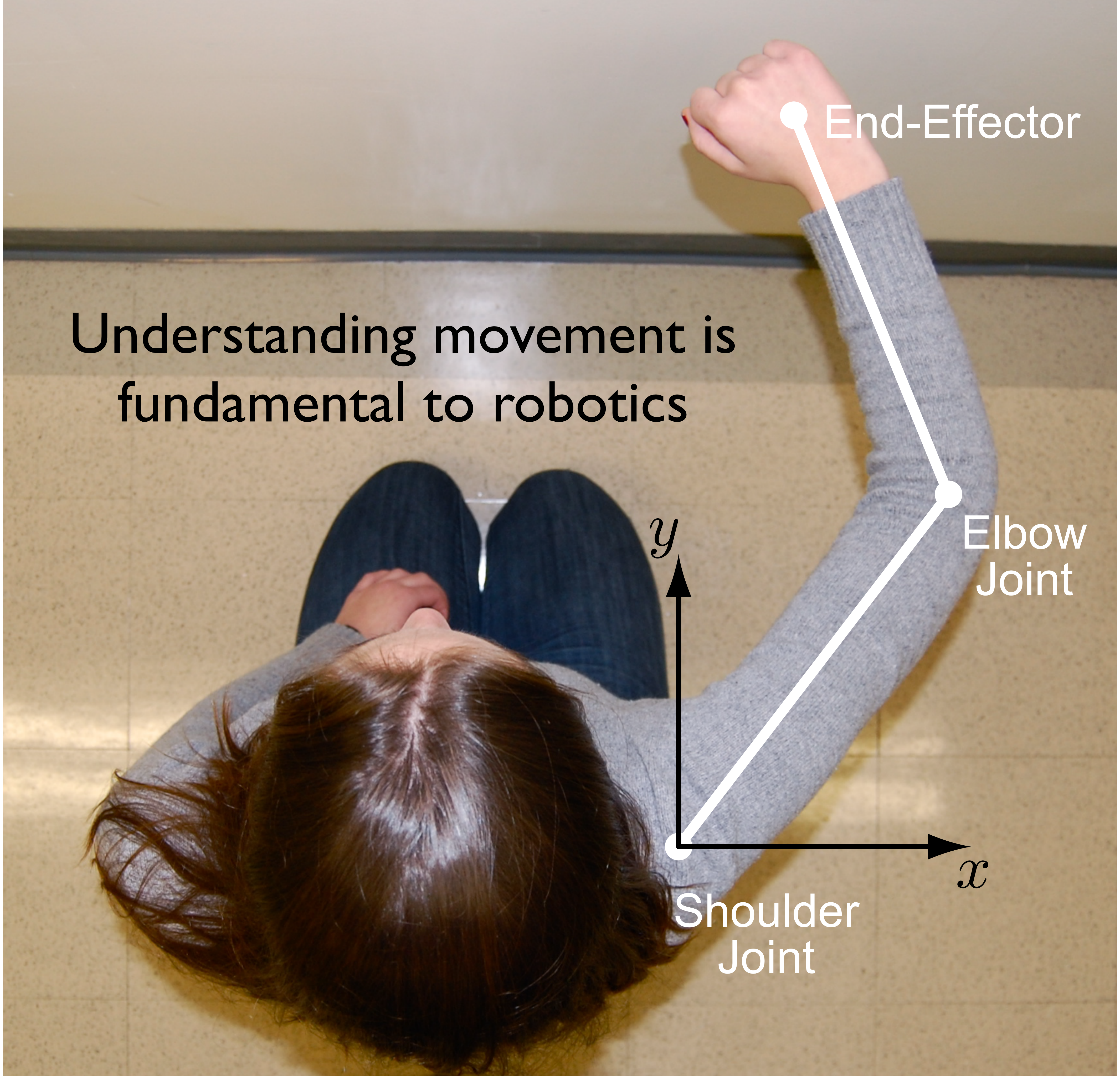
If you do this homework with a partner, you may work with anyone you choose; the only stipulation is that they also have only a little MATLAB experience. If you are looking for a partner, consider using the "Search for Teammates!" tool on Piazza.

If you are in a pair, you should work closely with your partner throughout this assignment, following the paradigm of pair programming. You will turn in one MATLAB script for which you are both jointly responsible, and you will both receive the same grade. Please follow these pair programming guidelines, which were adapted from "All I really need to know about pair programming I learned in kindergarten," by Williams and Kessler, *Communications of the ACM*, May 2000:

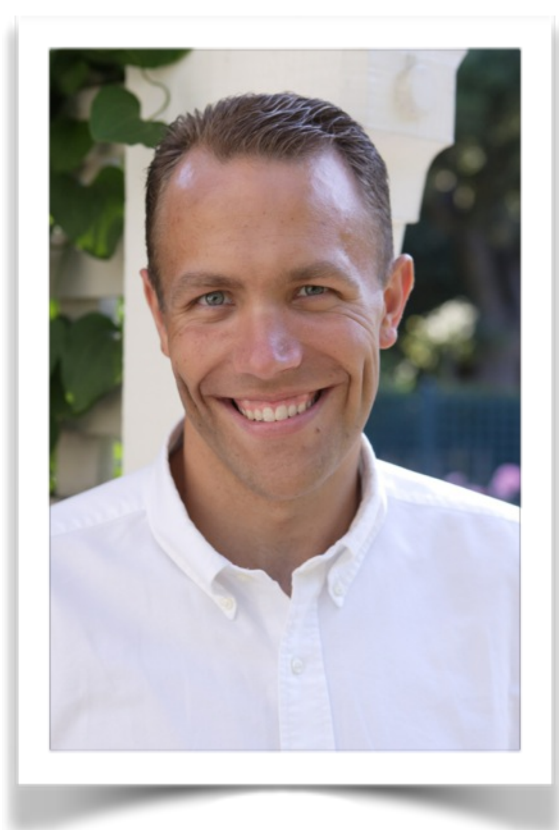
- Start with a good attitude, setting aside any skepticism and expecting to jell with your partner.
- Don't start writing code alone. Arrange a meeting with your partner as soon as you can.
- Use just one computer, and sit side by side; a desktop computer with a large monitor is better for this than a laptop. Make sure both partners can see the screen.
- At each instant, one partner should be driving (using the mouse and keyboard or recording design ideas) while the other is continuously reviewing the work (thinking and making suggestions).
- Change driving/reviewing roles at least every thirty 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 a bug in the code your partner is typing, wait until they finish the line to correct them.

1

Homework 1 is due by
11:59 p.m. today.


Understanding movement is fundamental to robotics
End-Effector
Elbow Joint
y
x
Shoulder Joint

Many of today's slides are adapted from ones created by Jonathan Fiene for MEAM 520 in Spring 2012.

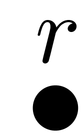


Points & Frames

SHV 2.1

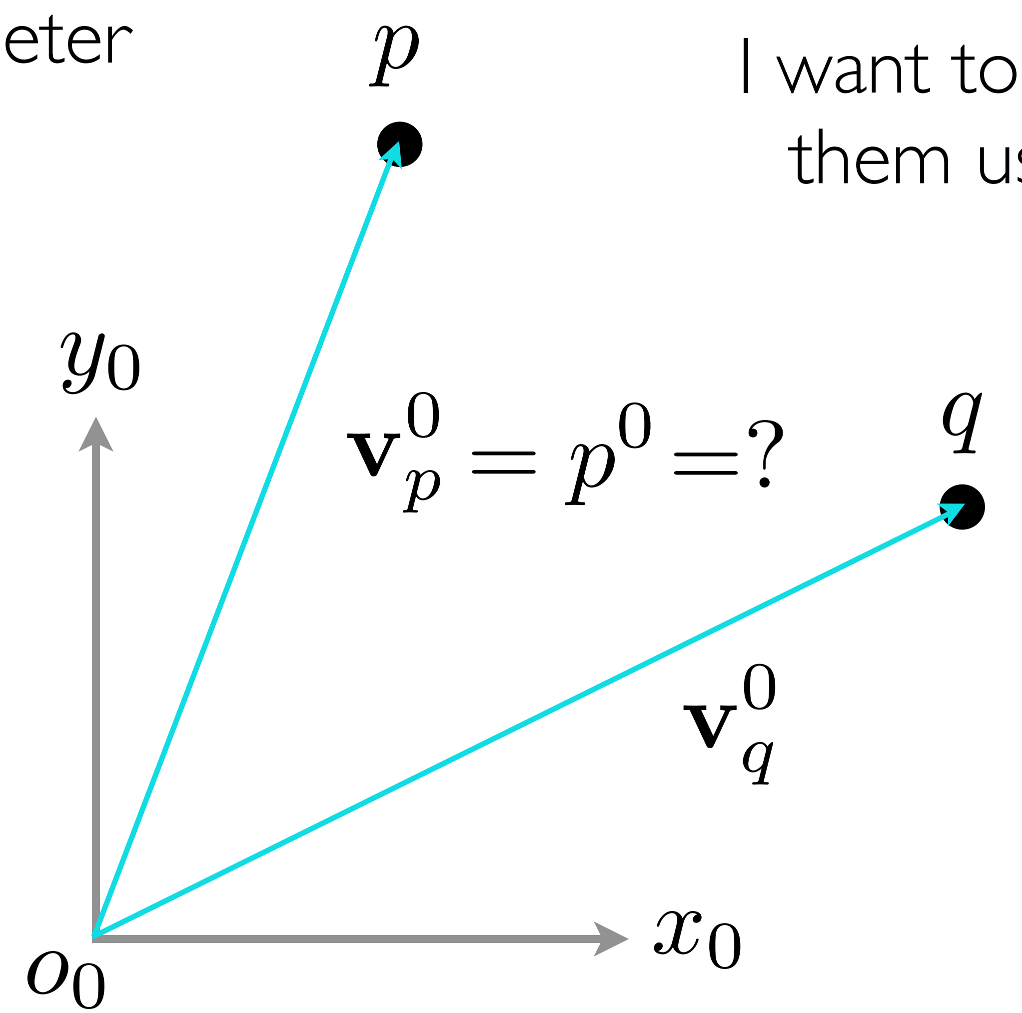
Representing positions

A **point** exists in space as a geometric entity



I can reason directly about these points, but if I want to **analyze** them, I must represent them using coordinates or equations.

1 meter



Coordinate Frame

Needs an origin (a single point in space)

and two or three orthogonal coordinate axes

The coordinate frame being used is designated using **superscript** notation

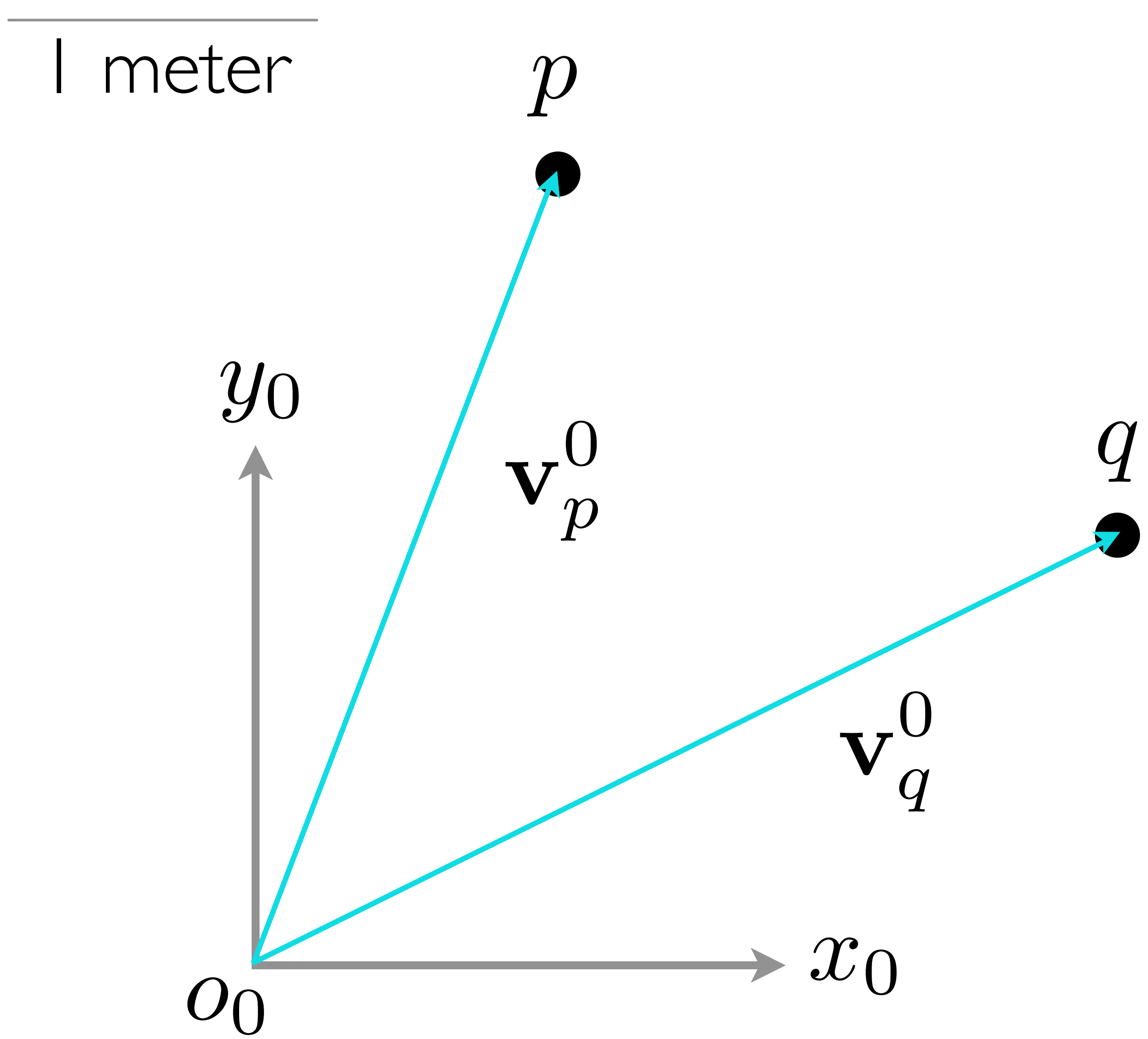
"It's *super* to specify the frame!"

What if we were doing this in **3D**.

How would the coordinate frame change?

How would the coordinates change?

How do I find the **magnitude** or **length** or **norm** of a vector?



$$\mathbf{v}_p^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad \mathbf{v}_p^0 = [x, y, z]^T$$

$$||\mathbf{v}_p^0|| = \sqrt{x^2 + y^2 + z^2}$$

$$||\mathbf{v}_p^0|| = (x^2 + y^2 + z^2)^{\frac{1}{2}}$$

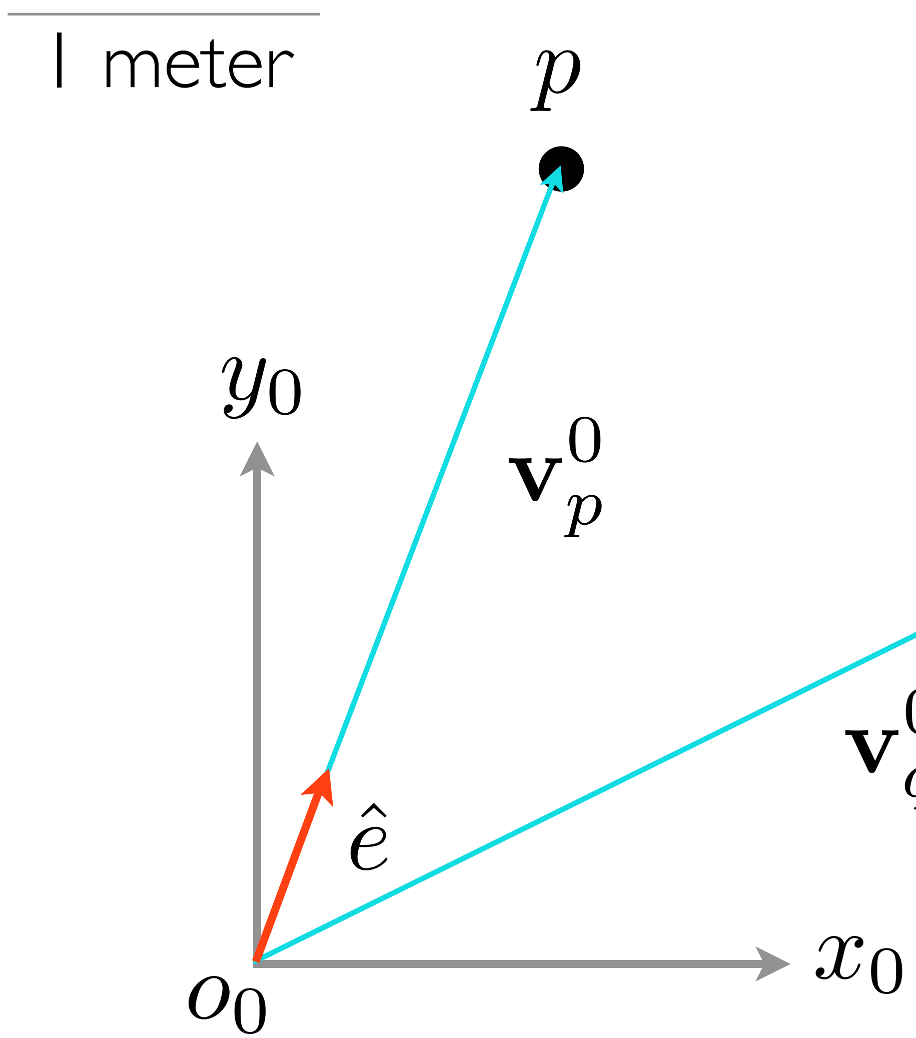
$$||\mathbf{v}_p^0|| = \left((\mathbf{v}_p^0)^T \mathbf{v}_p^0\right)^{\frac{1}{2}}$$

$$||\mathbf{v}_p^0|| = \langle \mathbf{v}_p^0, \mathbf{v}_p^0 \rangle^{\frac{1}{2}}$$

In MATLAB?

Vectors

How do I represent the **direction** of a vector?



$$\mathbf{v}_p^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad \mathbf{v}_p^0 = [x, y, z]^T$$

Create a **unit vector**.

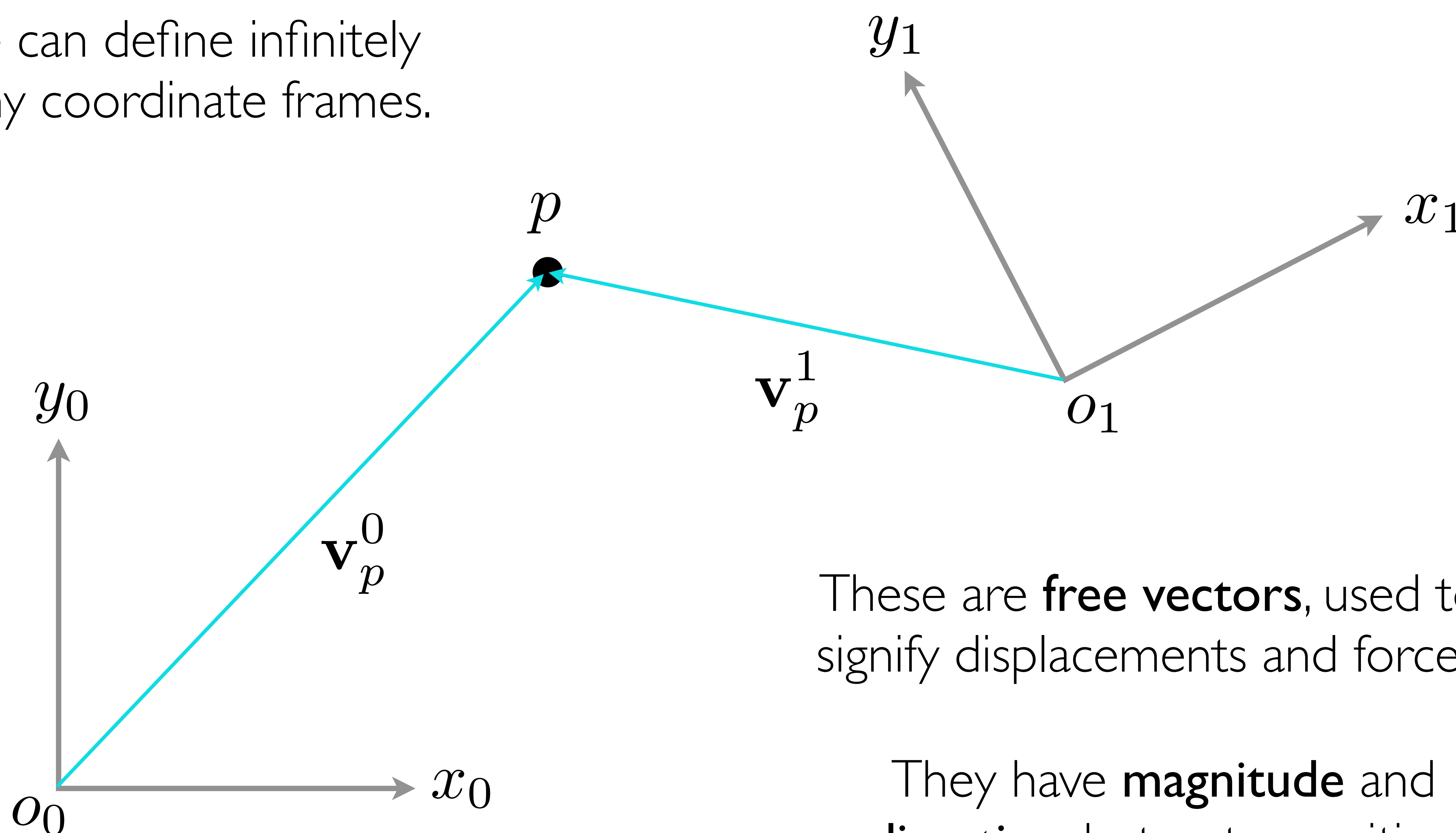
$$\hat{e} = \frac{\mathbf{v}_p^0}{||\mathbf{v}_p^0||}$$

In MATLAB?

Multiple coordinate frames

The **superscript** should make you turn your head to consider world from the orientation of the coordinate frame.

We can define infinitely many coordinate frames.

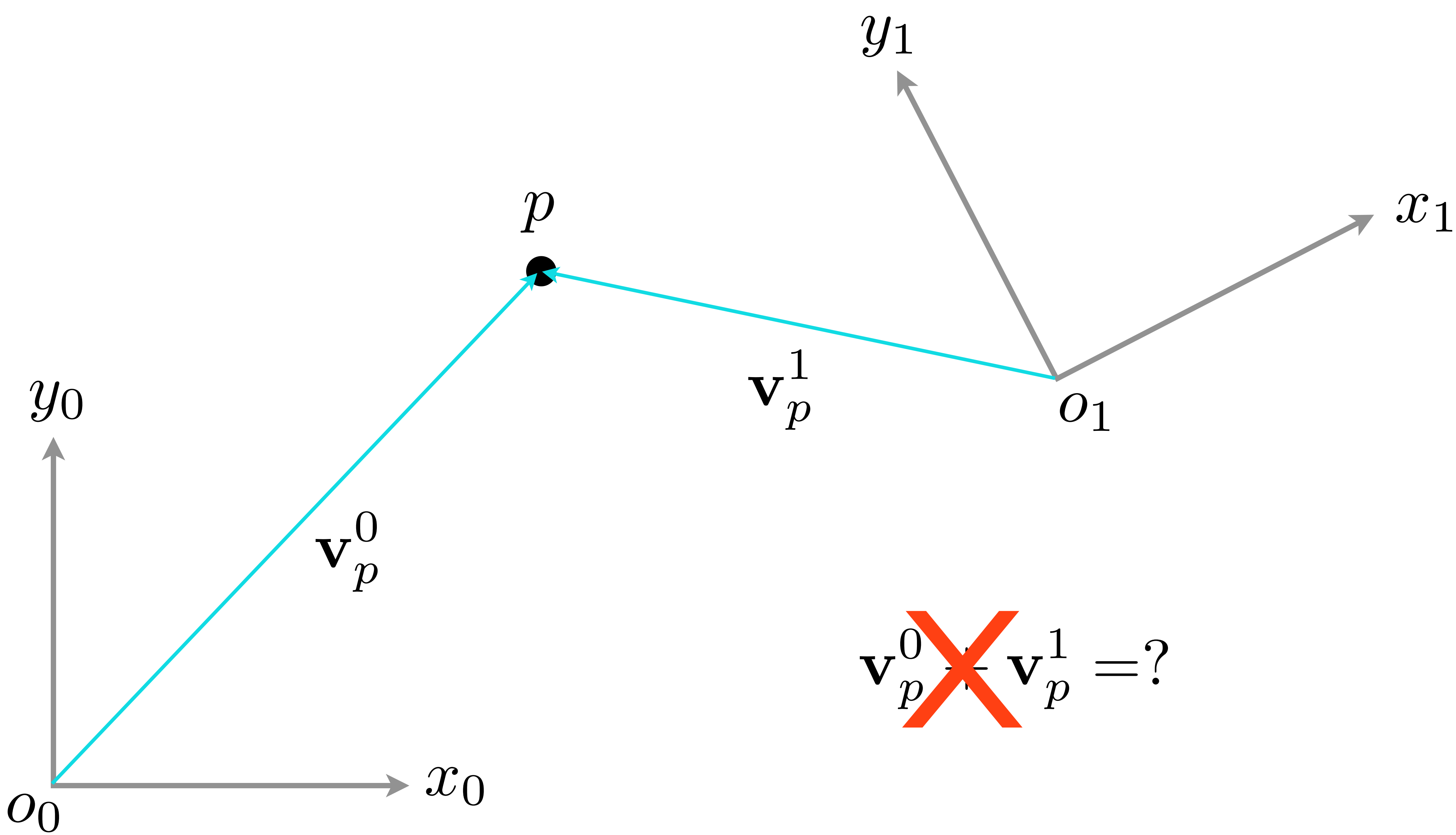


These are **free vectors**, used to signify displacements and forces.

They have **magnitude** and **direction**, but not a position from which they start.

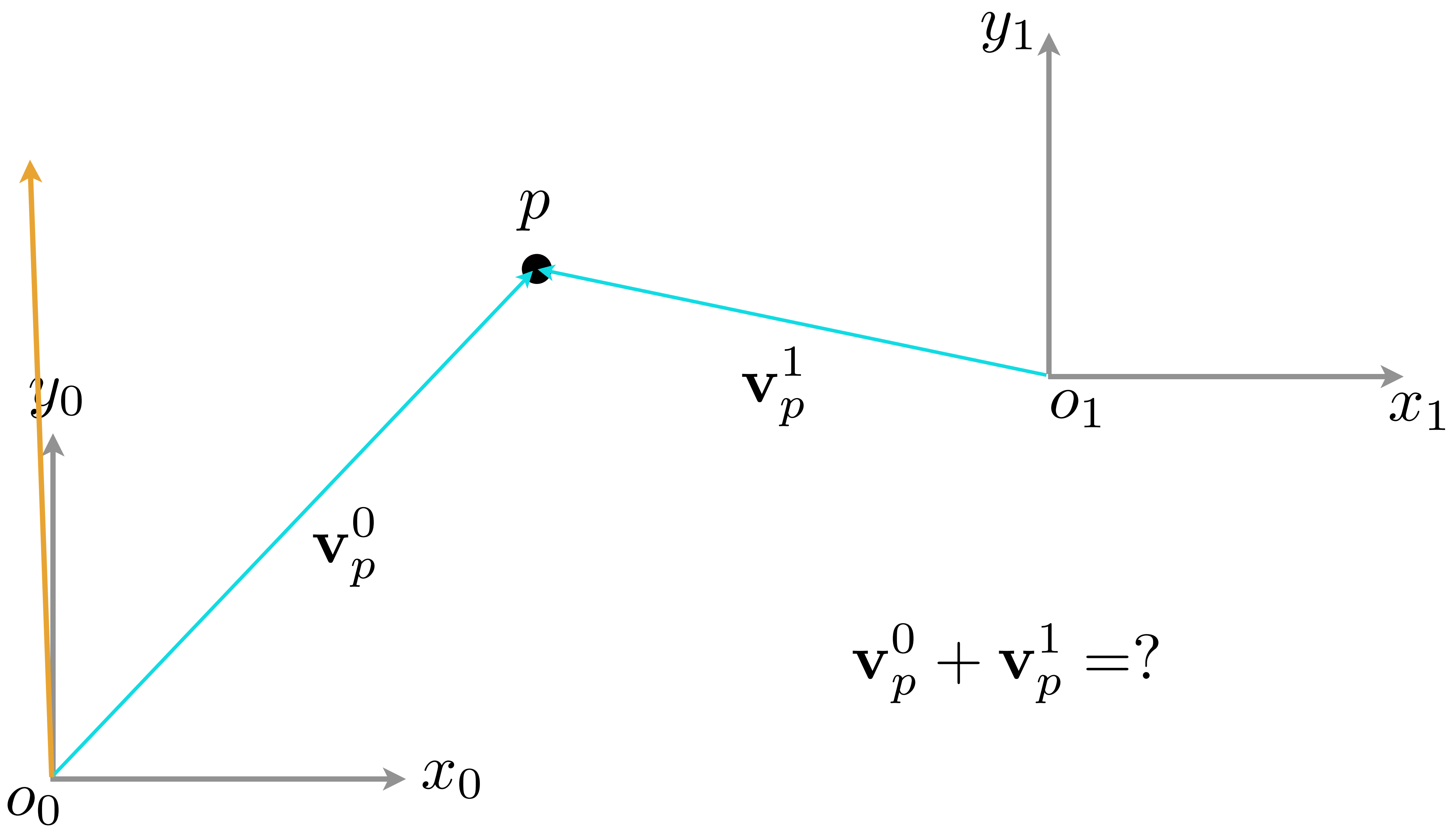
We draw them at different locations for convenience.

Multiple coordinate frames



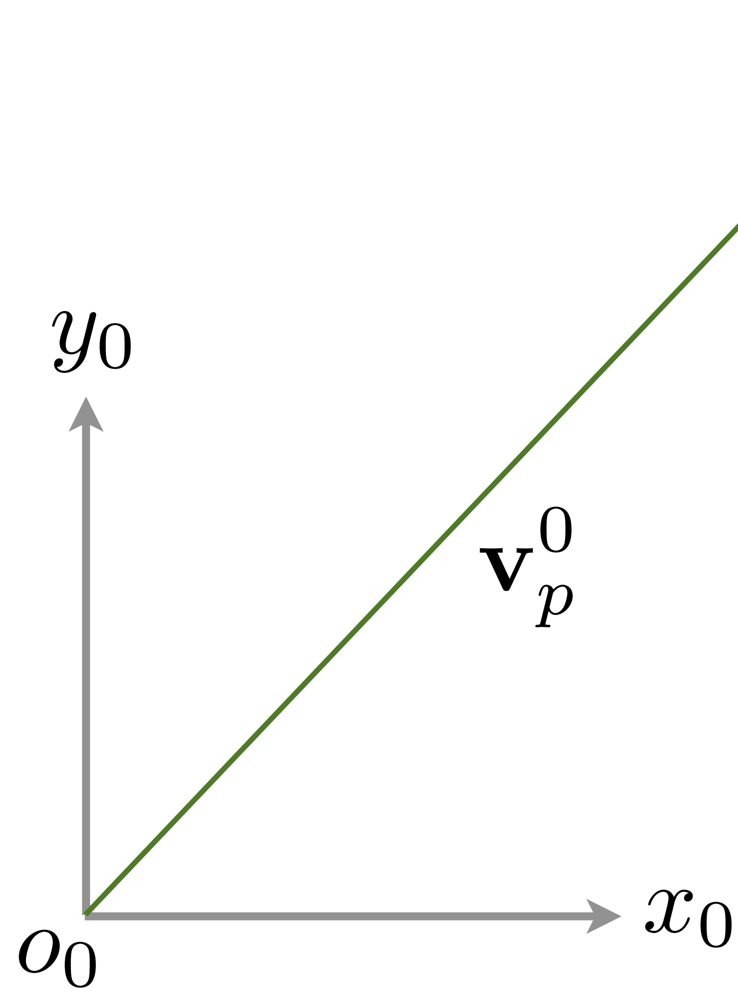
To perform algebraic manipulation such as addition or subtraction, you must express vectors in the **same frame** or in **parallel frames**

Multiple coordinate frames



Adding vectors expressed in the **same frame** or in **parallel frames** is like placing the vectors "tip-to-tail," or like applying displacements in sequence.

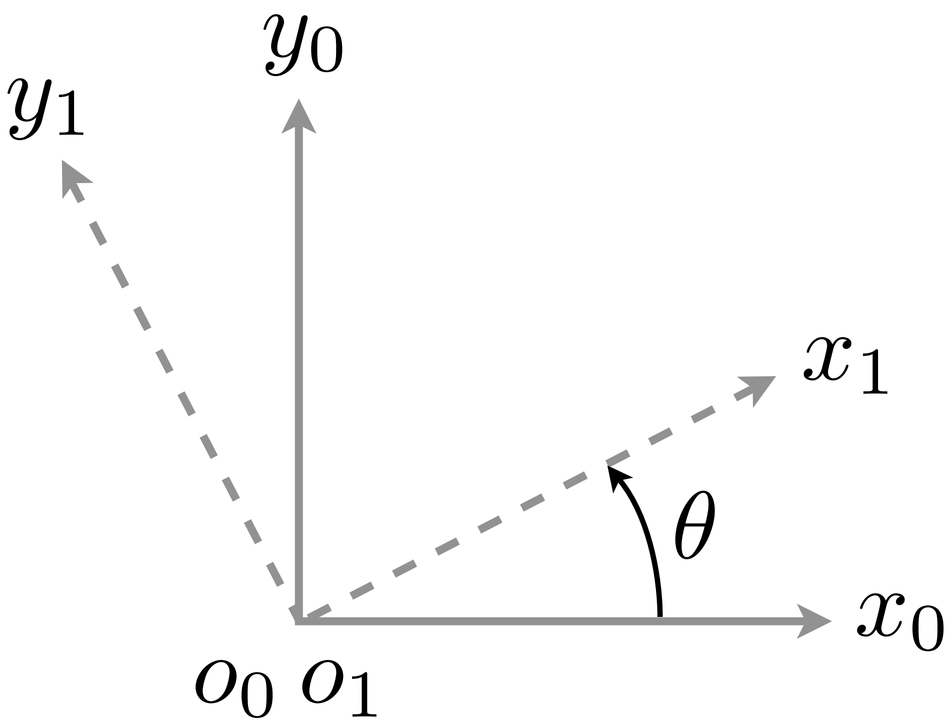
What questions do you have?

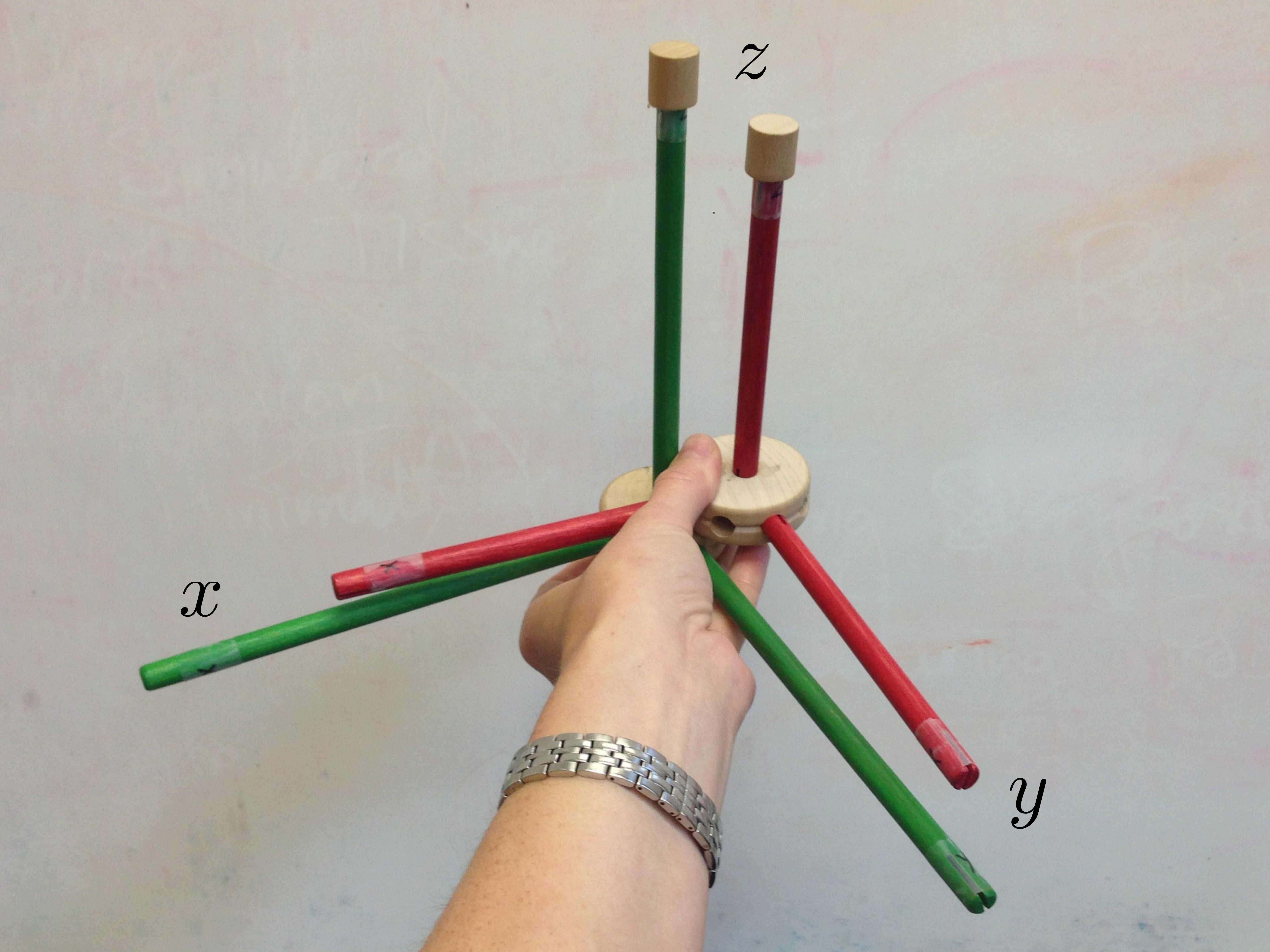


Rotation Matrices

SHV 2.2 & 2.3

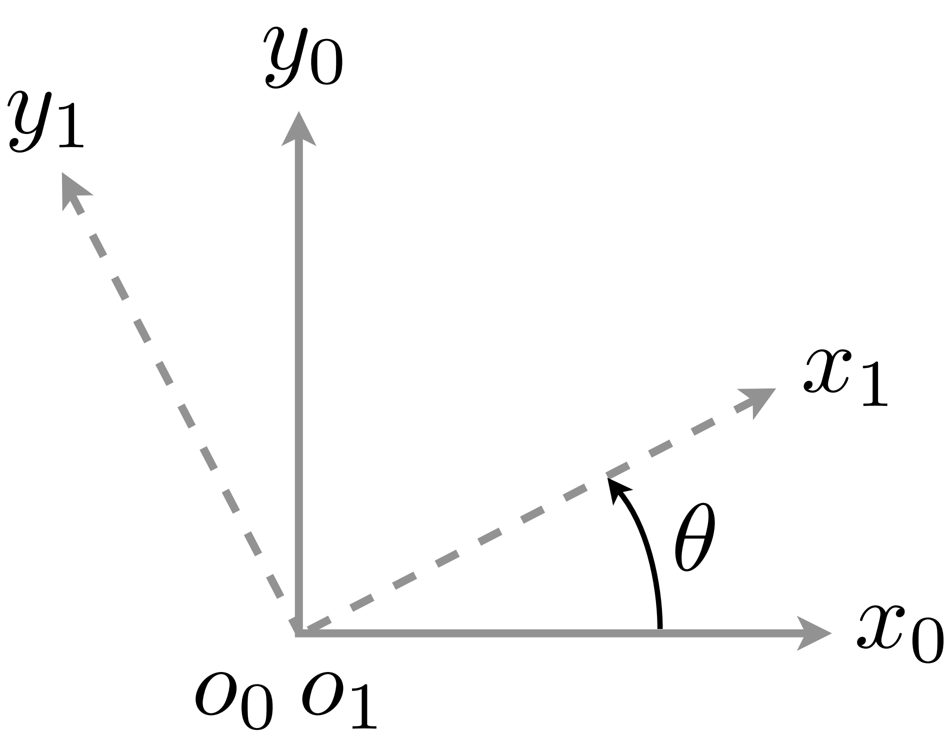
Planar Coordinate Rotations





z
x
y

Planar Coordinate Rotations



project frame 1 into frame 0

$$\mathbf{x}_1^0 =$$

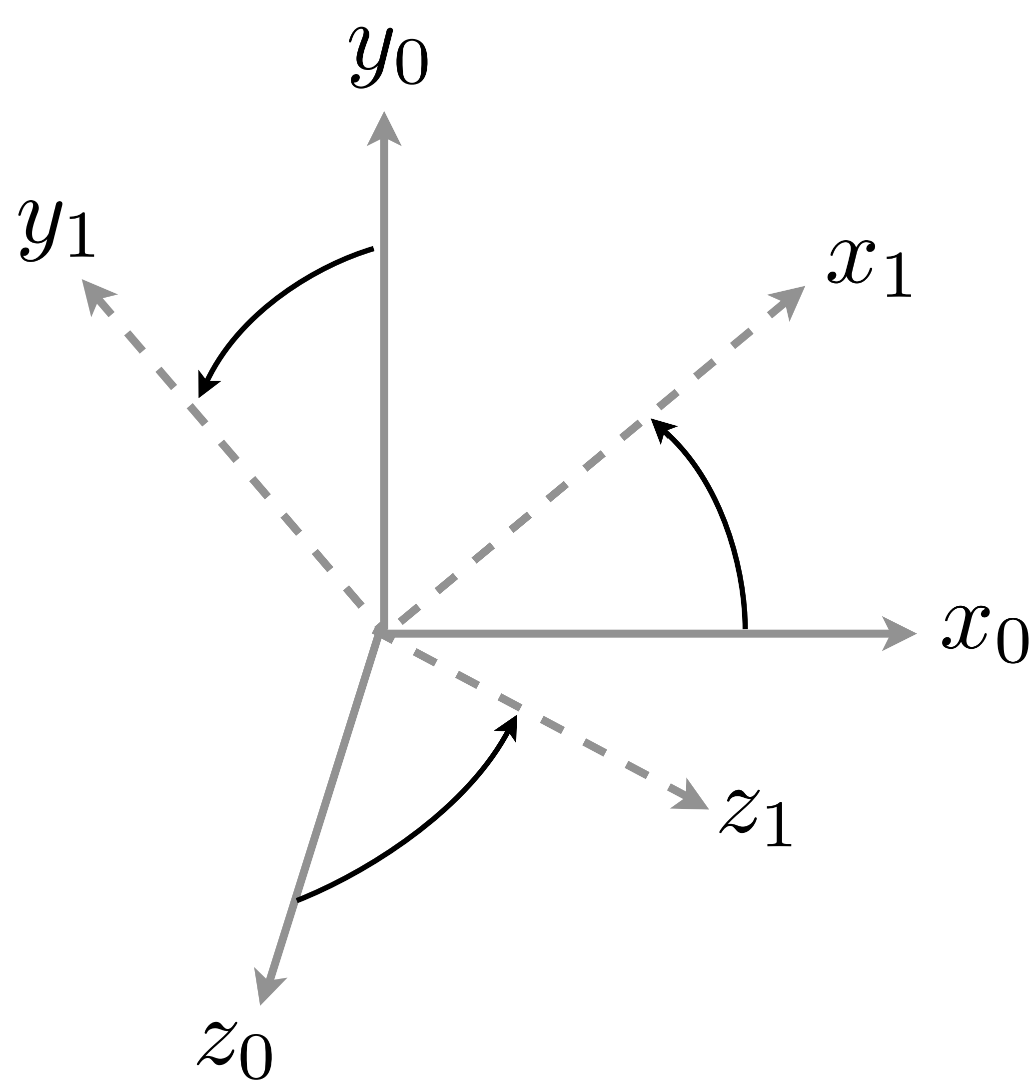
$$\mathbf{y}_1^0 =$$

which can be expressed as a **rotation matrix**

$$\mathbf{R}_1^0 = \left[\begin{array}{cc} \mathbf{x}_1^0 & \mathbf{y}_1^0 \end{array} \right] = \left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array} \right]$$

the inverse of which is the matrix transpose

$$\mathbf{R}_0^1 = (\mathbf{R}_1^0)^\top = \left[\begin{array}{cc} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{array} \right]$$



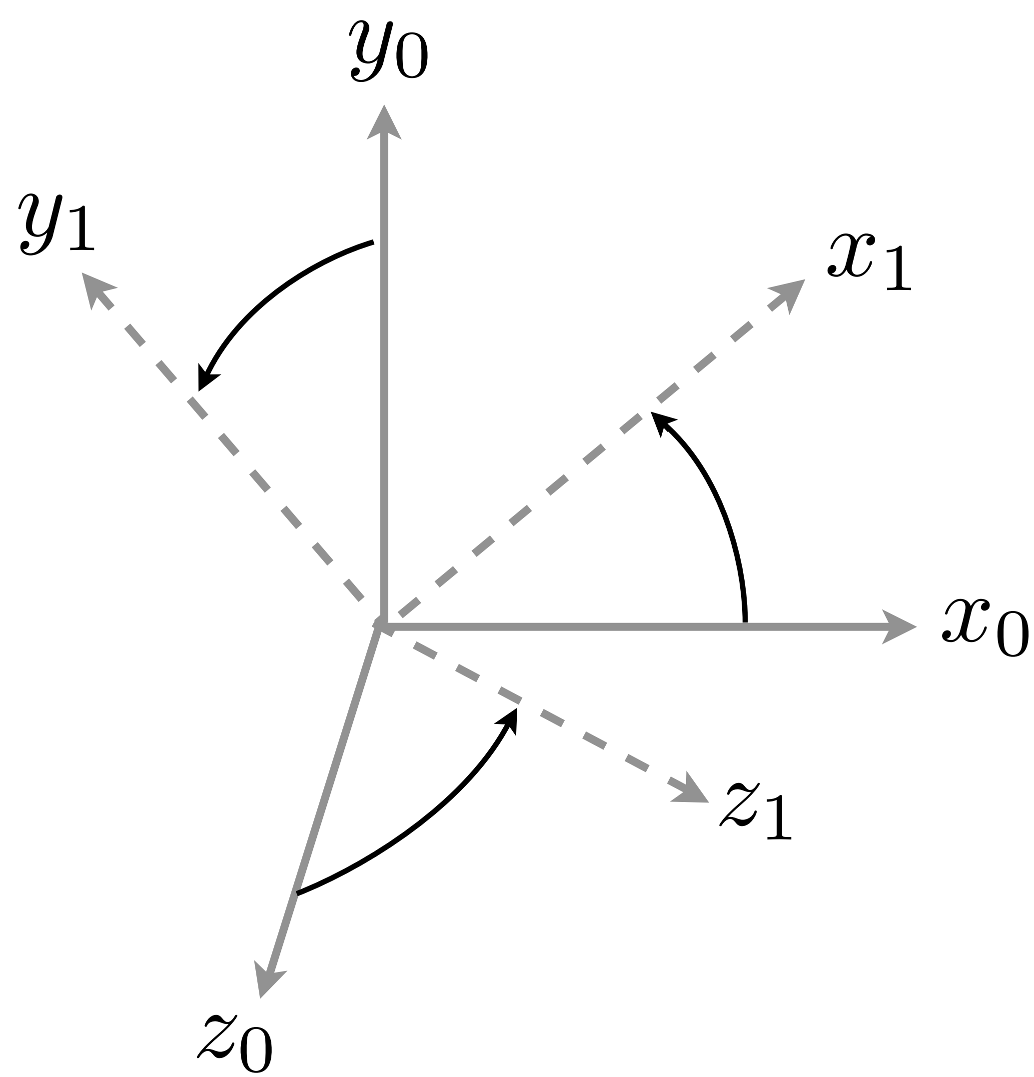
Represent orientation of one frame with respect to another frame

$$\mathbf{R}_1^0 = \begin{bmatrix} x_1 \cdot x_0 & y_1 \cdot x_0 & z_1 \cdot x_0 \\ x_1 \cdot y_0 & y_1 \cdot y_0 & z_1 \cdot y_0 \\ x_1 \cdot z_0 & y_1 \cdot z_0 & z_1 \cdot z_0 \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

SO(3)

Special Orthogonal group of order 3



Represent orientation of one frame with respect to another frame

$$\mathbf{R}_1^0 = \begin{bmatrix} x_1 \cdot x_0 & y_1 \cdot x_0 & z_1 \cdot x_0 \\ x_1 \cdot y_0 & y_1 \cdot y_0 & z_1 \cdot y_0 \\ x_1 \cdot z_0 & y_1 \cdot z_0 & z_1 \cdot z_0 \end{bmatrix}$$

$$\mathbf{R}_{x,\theta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

$$\mathbf{R}_{y,\theta} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$\mathbf{R}_{z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

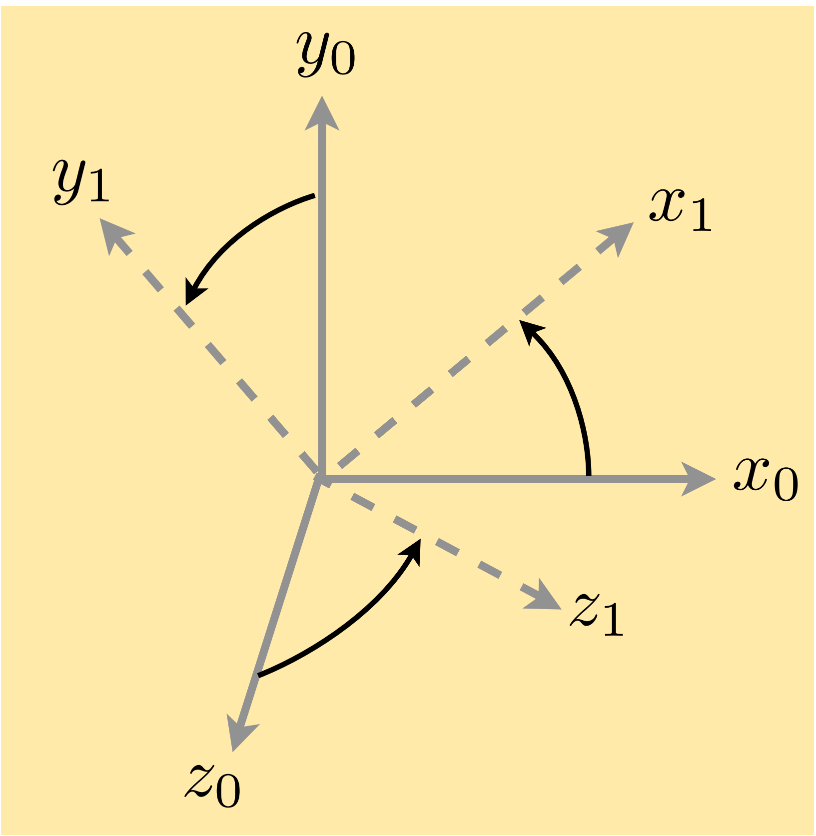
The **basic rotation matrices** define rotations about the three coordinate axes

The **0's and 1** are always in the row and column of the rotational axis.

How can you remember where the **sines and cosines** are?

+/– sin?

What questions do you have?



Represents the orientation of one coordinate frame with respect to another frame

$$\mathbf{R}_1^0 = \begin{bmatrix} x_1 \cdot x_0 & y_1 \cdot x_0 & z_1 \cdot x_0 \\ x_1 \cdot y_0 & y_1 \cdot y_0 & z_1 \cdot y_0 \\ x_1 \cdot z_0 & y_1 \cdot z_0 & z_1 \cdot z_0 \end{bmatrix}$$

Orientation of frame 1 w.r.t. frame 0

$$\mathbf{R}_0^1 = \left(\mathbf{R}_1^0\right)^T$$

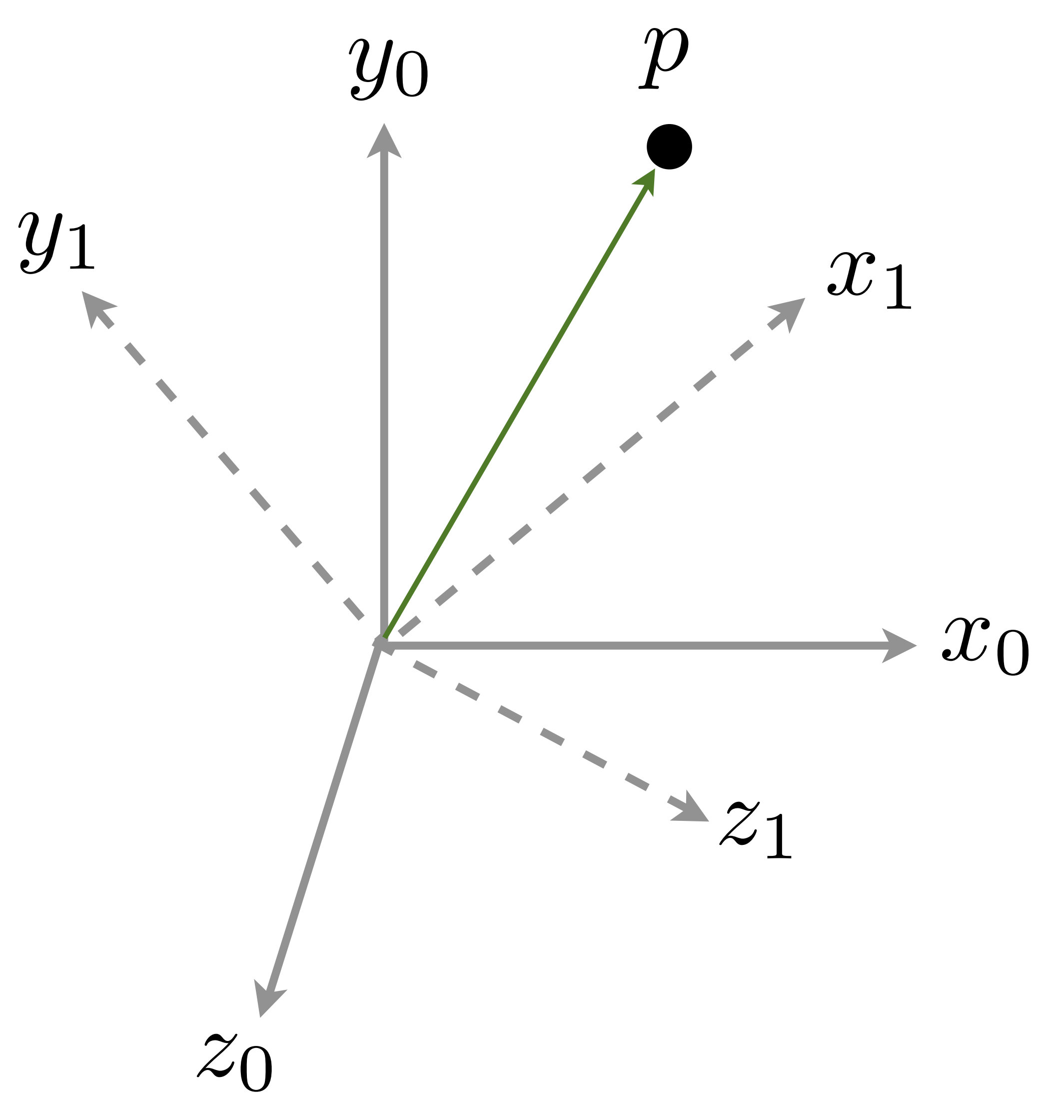
$$\left(\mathbf{R}_1^0\right)^T = \left(\mathbf{R}_1^0\right)^{-1}$$

columns are of unit length

columns are mutually orthogonal

$$\det \mathbf{R}_1^0 = +1$$

Rotational Transformations



For pure coordinate rotation, a point in frame 1 can be expressed in frame 0 using the rotation matrix

$$\mathbf{R}_1^0 \qquad \mathbf{v}_p^1$$

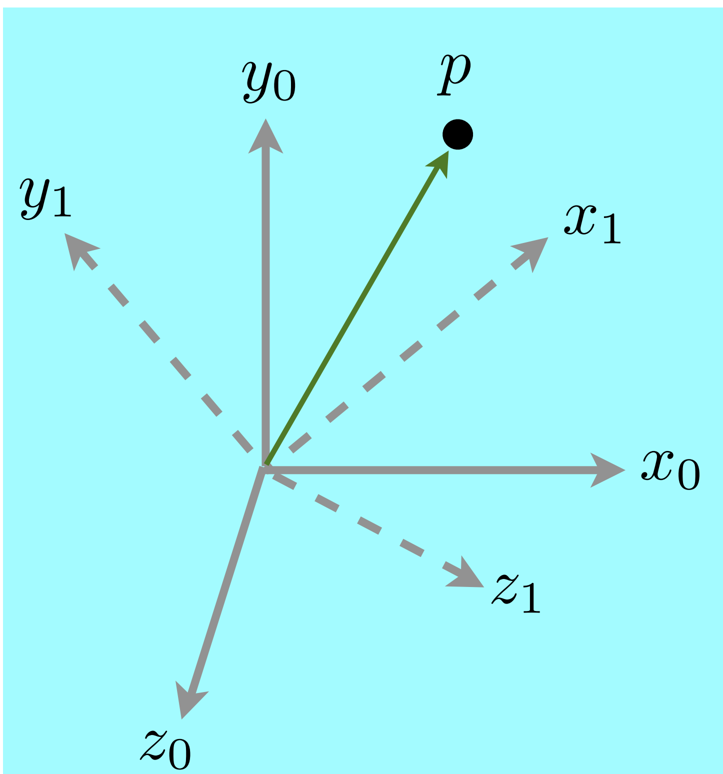
$$\mathbf{v}_p^0 = ?$$

$$\mathbf{v}_p^0 = \mathbf{R}_1^0 \, \mathbf{v}_p^1$$

Pre-multiply the vector by the rotation matrix that gives the **frame you have** in terms of the **frame you want**.

Rotation Matrices - Interpretation 2 of 3

What questions do you have?



Coordinate transformation relating the coordinates of a point p in two different frames

$$\boxed{\mathbf{v}_p^0 = \mathbf{R}_1^0 \mathbf{v}_p^1}$$

Subscript and superscript cancel

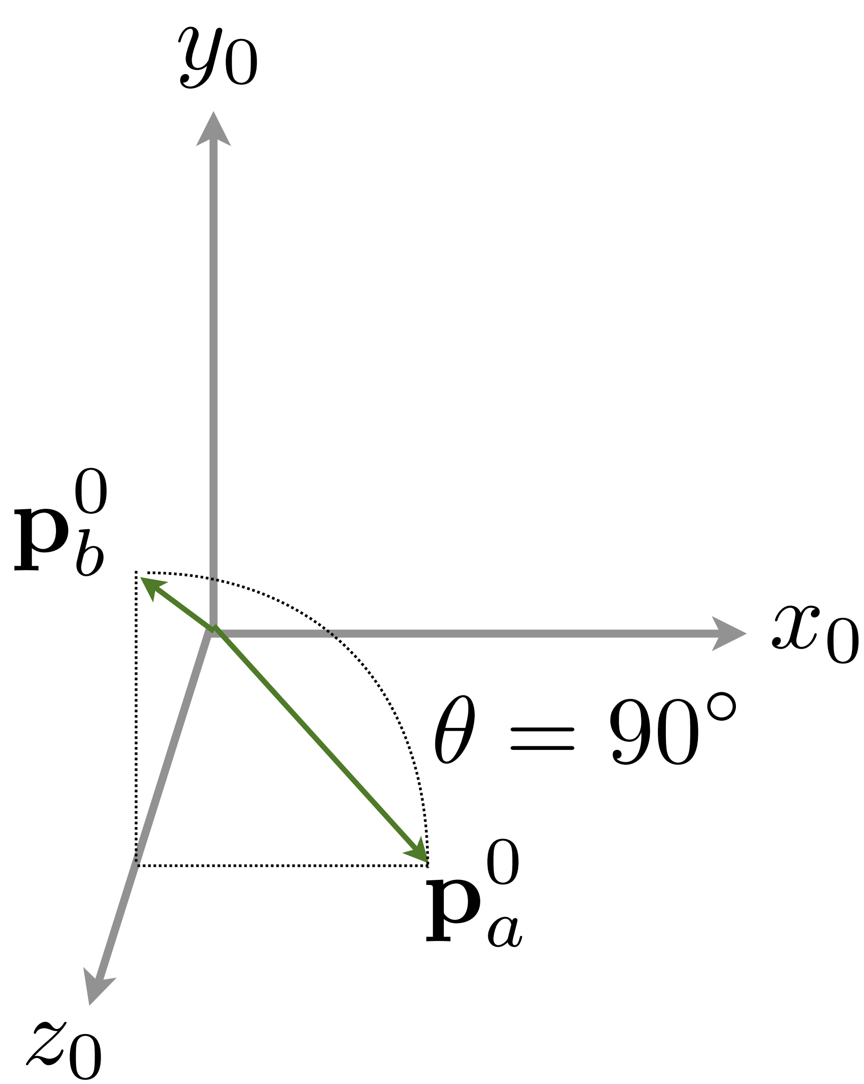
$$\mathbf{v}_p^1 = \left(\mathbf{R}_1^0\right)^T \mathbf{v}_p^0$$

$$\boxed{\mathbf{v}_p^1 = \mathbf{R}_0^1 \mathbf{v}_p^0}$$

$$\mathbf{v}_p^1 = ?$$

$$\left(\mathbf{R}_1^0\right)^{-1} \mathbf{v}_p^0 = \cancel{\left(\mathbf{R}_1^0\right)^{-1} \mathbf{R}_1^0} \mathbf{v}_p^1$$

$$\left(\mathbf{R}_1^0\right)^T = \left(\mathbf{R}_1^0\right)^{-1}$$



The rotation matrix can also be used to perform rotations on vectors

$$\mathbf{p}_b^0 = \mathbf{R}\,\mathbf{p}_a^0$$

$$\mathbf{p}_a^0 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

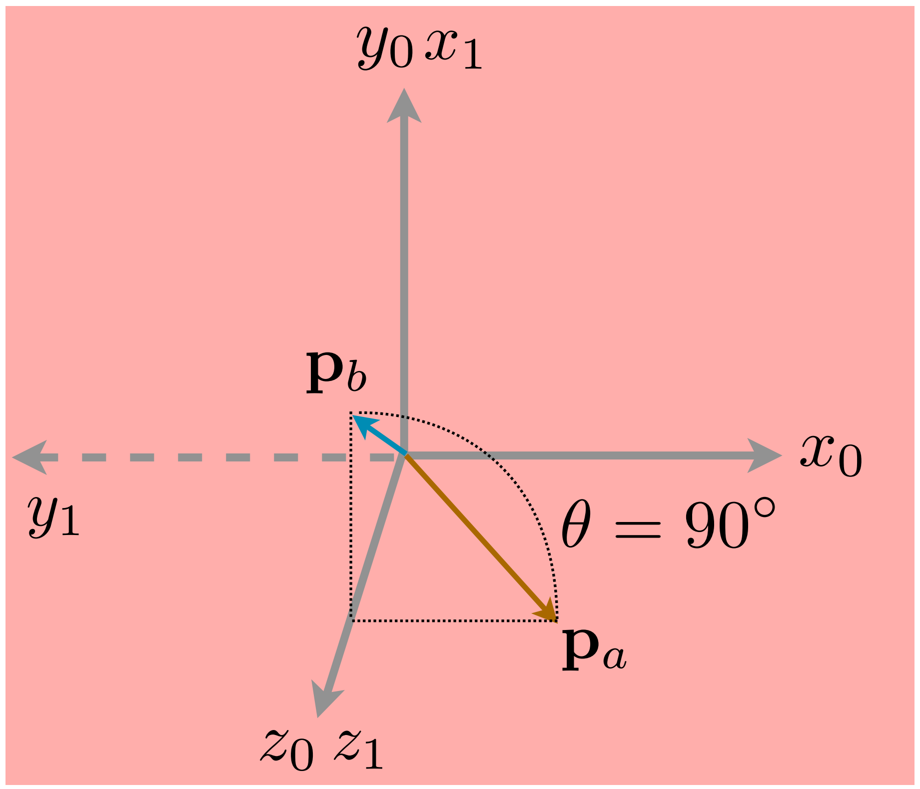
$$\mathbf{R}_{z,90^\circ} = ?$$

$$\mathbf{R}_{z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{p}_b^0 = \mathbf{R}_{z,\theta}\ \mathbf{p}_a^0 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

Rotation Matrices - Interpretation 3 of 3

What questions do you have?



Operator taking a vector and rotating it to yield a new vector in the same coordinate frame

$$\mathbf{p}_a^0 \qquad \mathbf{R}_1^0$$

$$\mathbf{p}_b^0 = ?$$

$$\mathbf{p}_b^0 = \cancel{\mathbf{R}_1^0\, \mathbf{p}_b^1}$$

$$\mathbf{p}_b^1 = \mathbf{p}_a^0$$

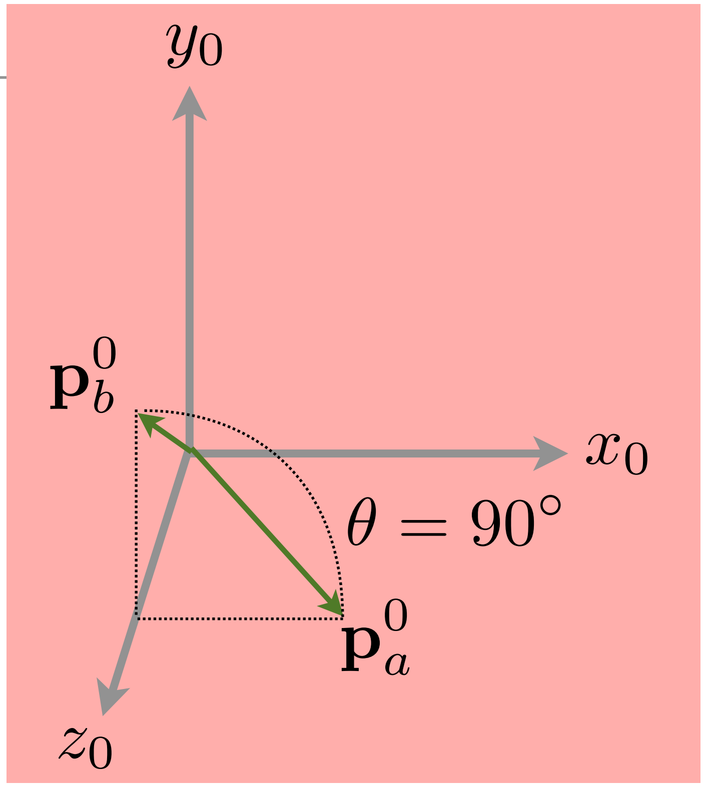
$$\mathbf{p}_b^0 = \mathbf{R}_1^0\, \mathbf{p}_a^0 \qquad \boxed{\mathbf{p}_b^0 = \mathbf{R}\, \mathbf{p}_a^0}$$

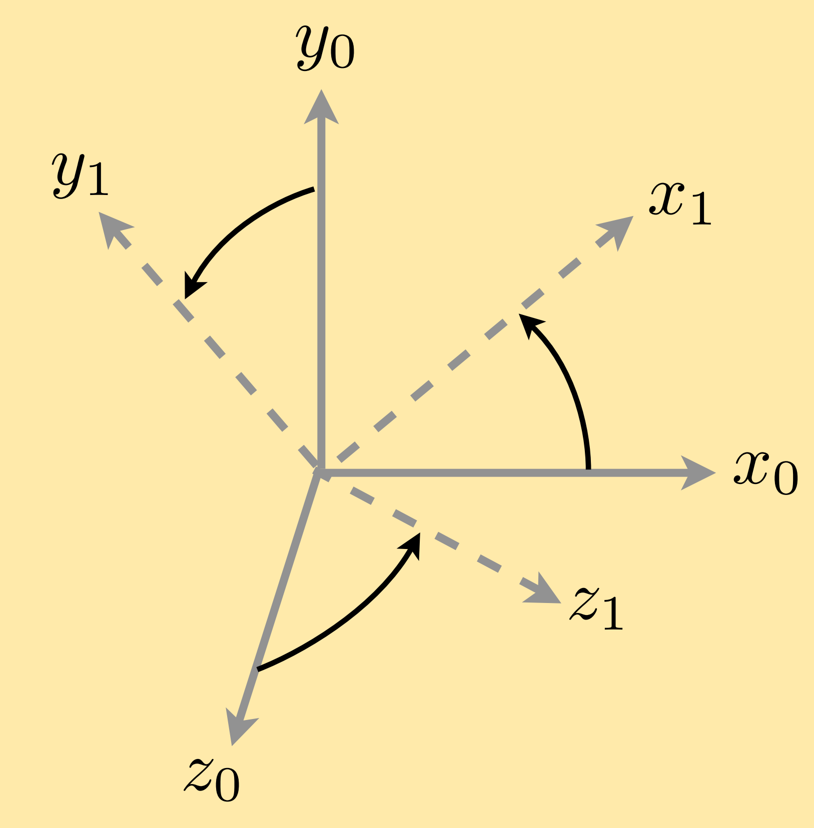
Rotation matrices serve three purposes (p. 47 in SHV):

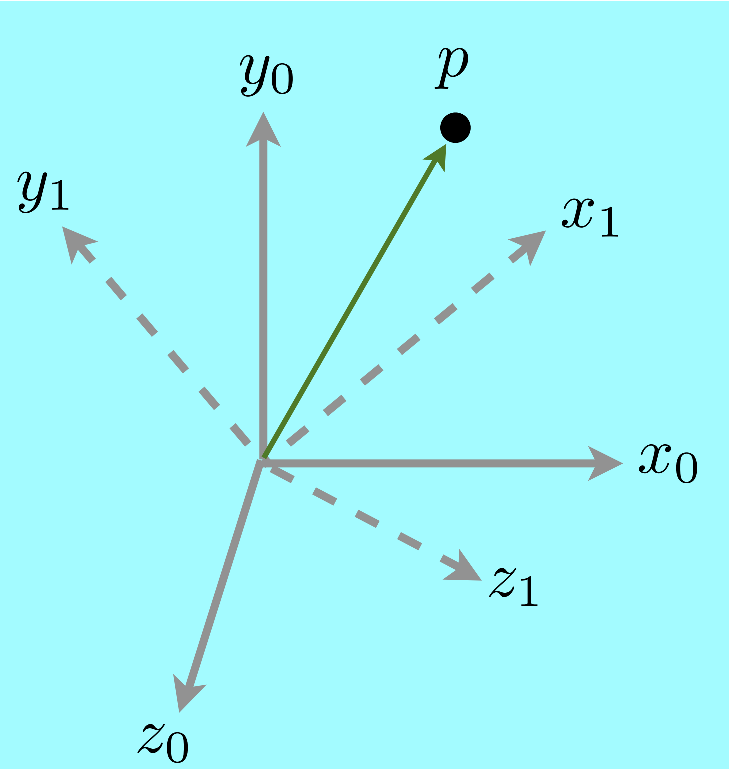
1. Coordinate transformation relating the coordinates of a point p in two different frames

2. Orientation of a transformed coordinate frame with respect to a fixed frame

3. Operator taking a vector and rotating it to yield a new vector in the same coordinate frame





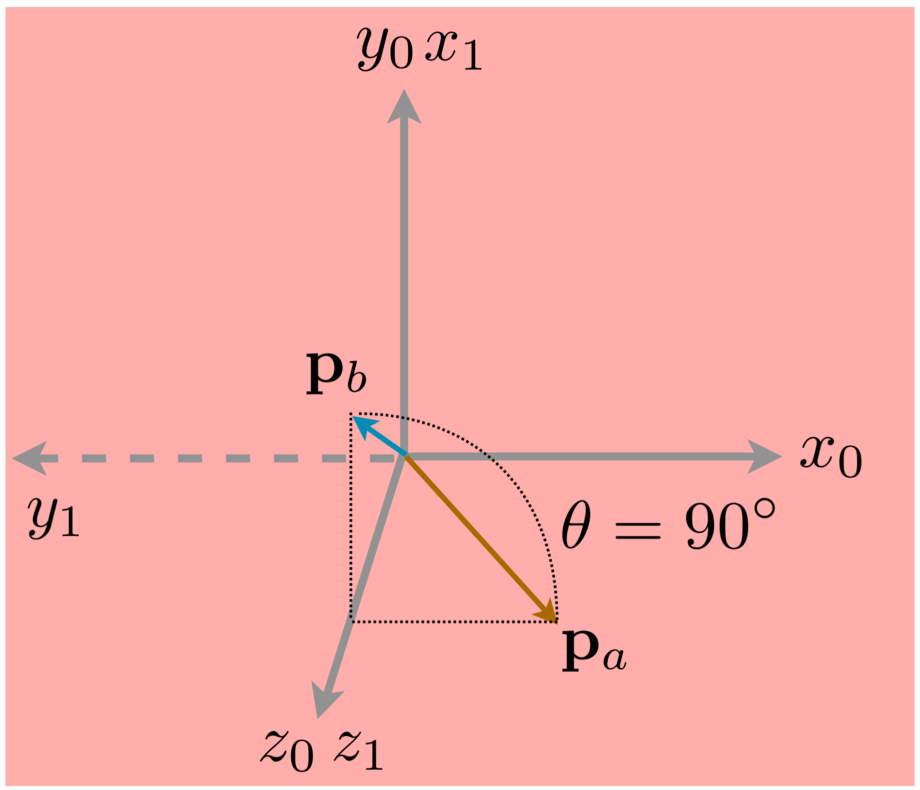


Composite Rotations

SHV 2.4

Rotation Matrices - Interpretation 3 of 3

What if I want to apply another rotation to this vector? Method depends on which axis I want to rotate around.



Operator taking a vector and rotating it to yield a new vector in the same coordinate frame

$$\mathbf{p}_a^0 \qquad \mathbf{R}_1^0$$

$$\mathbf{p}_b^0 = ?$$

$$\mathbf{p}_b^0 = \mathbf{R}_1^0 \, \mathbf{p}_b^1$$

$$\mathbf{p}_b^1 = \mathbf{p}_a^0$$

$$\mathbf{p}_b^0 = \mathbf{R}_1^0 \, \mathbf{p}_a^0$$

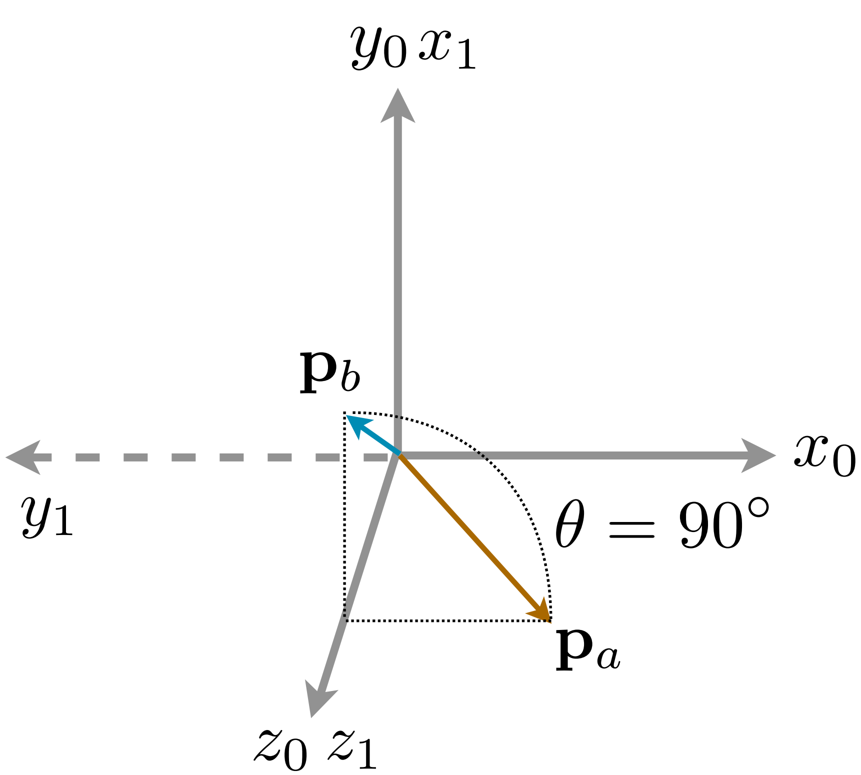
$$\boxed{\mathbf{p}_b^0 = \mathbf{R} \, \mathbf{p}_a^0}$$

For example:
rotate 45° around y_0
vs.
rotate 45° around y_1

The **order** in which a sequence of rotations
is performed is crucial.

Thus, the **order** in which the rotation matrices are
multiplied together is crucial.

What if I want to apply another rotation to this vector?
Method depends on which axis I want to rotate around.



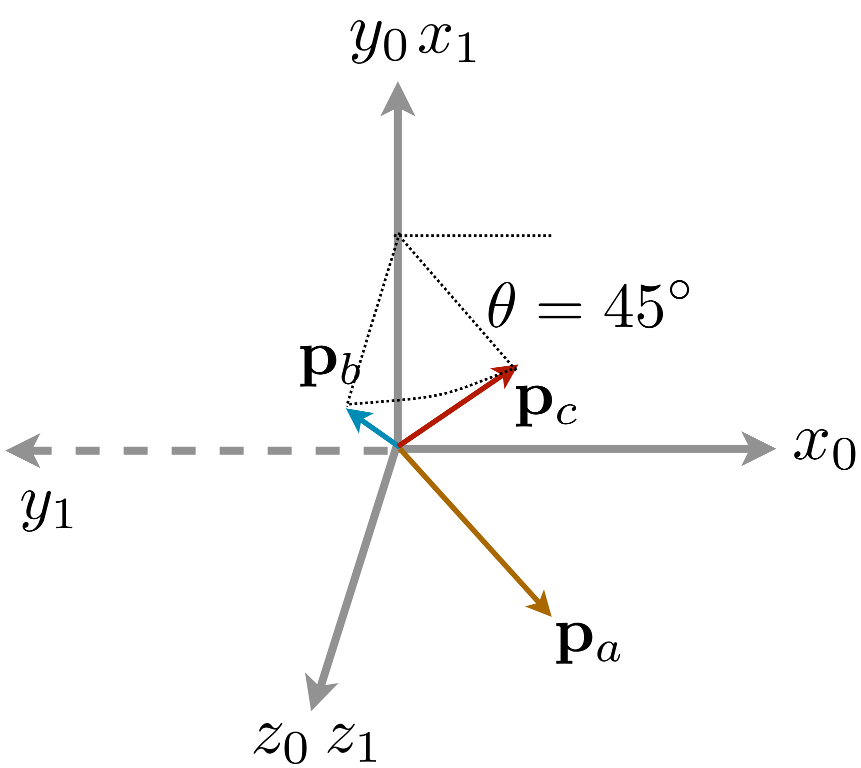
For example:

rotate 45º around y_0

vs.

rotate 45º around y_1

What if I want to apply another rotation to this vector?
Method depends on which axis I want to rotate around.



For example:
rotate 45° around y_0
vs.
rotate 45° around y_1

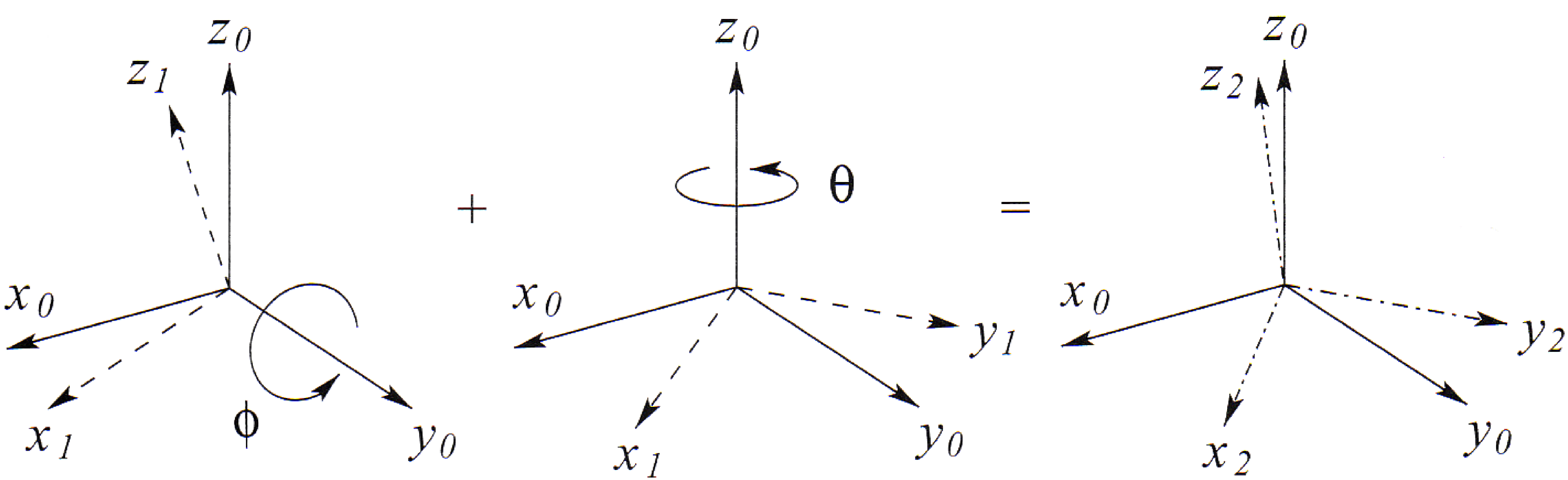
$$\mathbf{p}_b^0 = \mathbf{R}\,\mathbf{p}_a^0 \qquad \mathbf{R}' = \mathbf{R}_{y,45^\circ}$$

$$\mathbf{p}_c^0 = ?$$

$$\boxed{\mathbf{p}_c^0 = \mathbf{R}'\mathbf{p}_b^0}$$

$$\boxed{\mathbf{p}_c^0 = \mathbf{R}'\,\mathbf{R}\,\mathbf{p}_a^0}$$

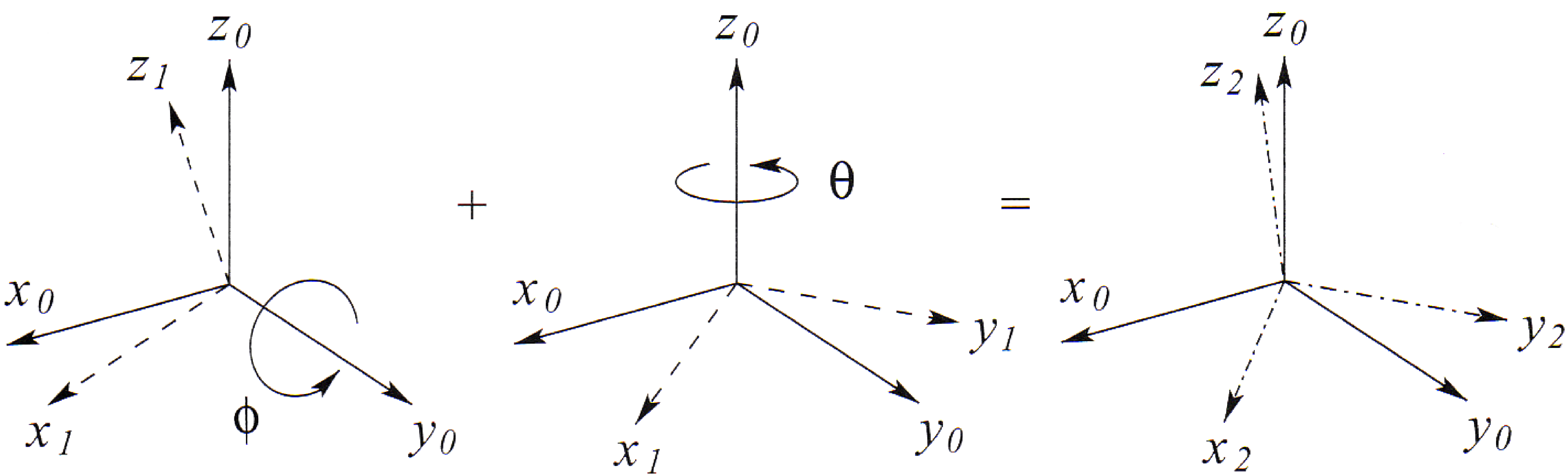
Composition of Rotations with Respect to a Fixed Frame



Composition of Rotations with Respect to a Fixed Frame

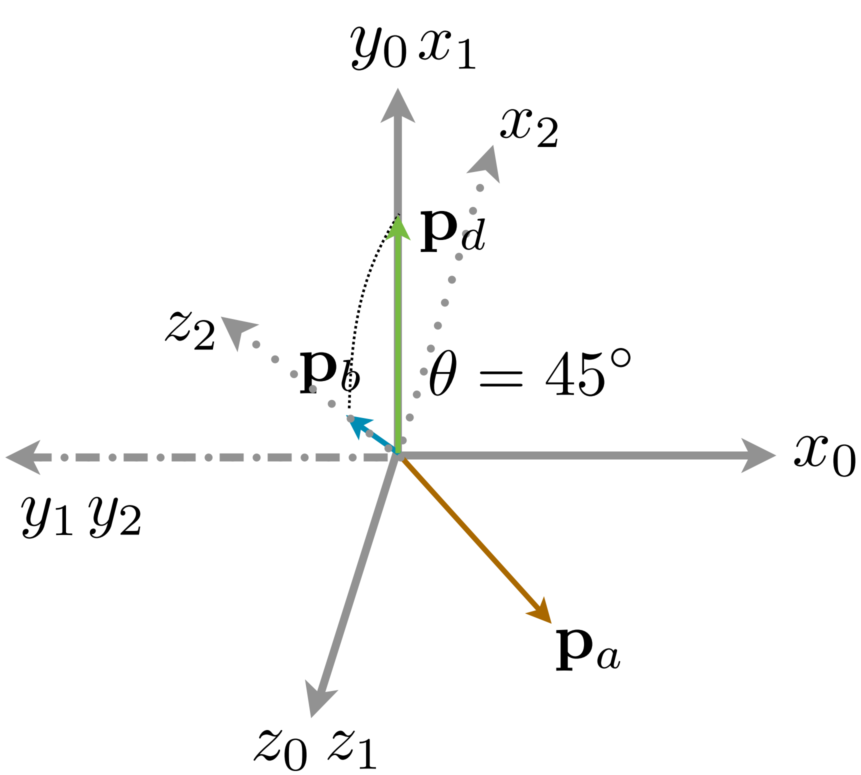
the result of a successive rotation about a fixed frame can be found by **pre-multiplying** by the corresponding rotation matrix

$$\mathbf{R}_2^0 = \mathbf{R}\,\mathbf{R}_1^0$$



Note that $\mathbf{R}$ is a rotation about the original frame

What if I want to apply another rotation to this vector?
Method depends on which axis I want to rotate around.



$$\mathbf{p}_d^0 = ?$$

For example:

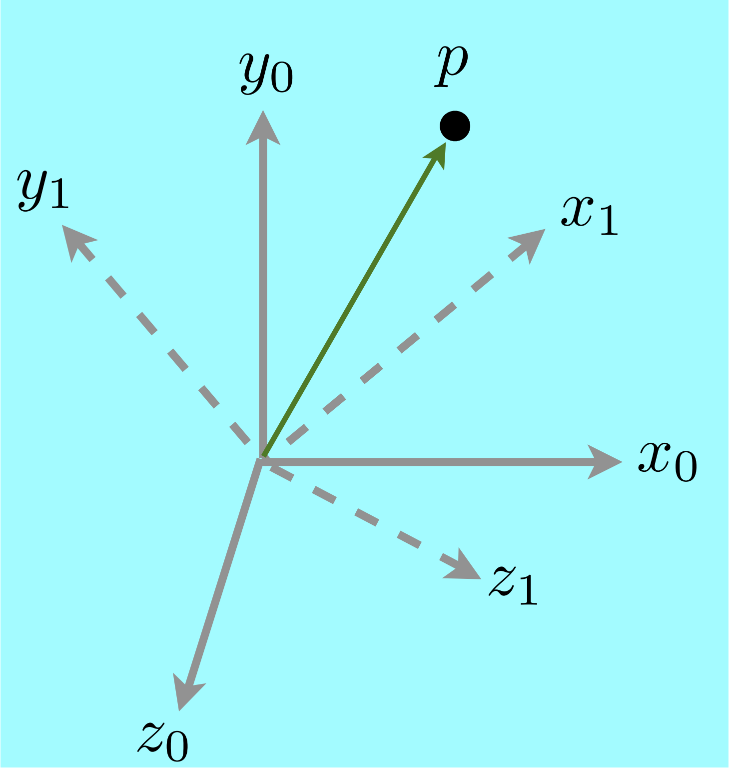
rotate 45° around y_0

vs.

rotate 45° around y_1

Rotation Matrices - Interpretation 2 of 3

Remember...



Coordinate transformation relating the coordinates of a point p in two different frames

$$\mathbf{R}_1^0 \qquad \mathbf{v}_p^1$$

$$\mathbf{v}_p^0 = ?$$

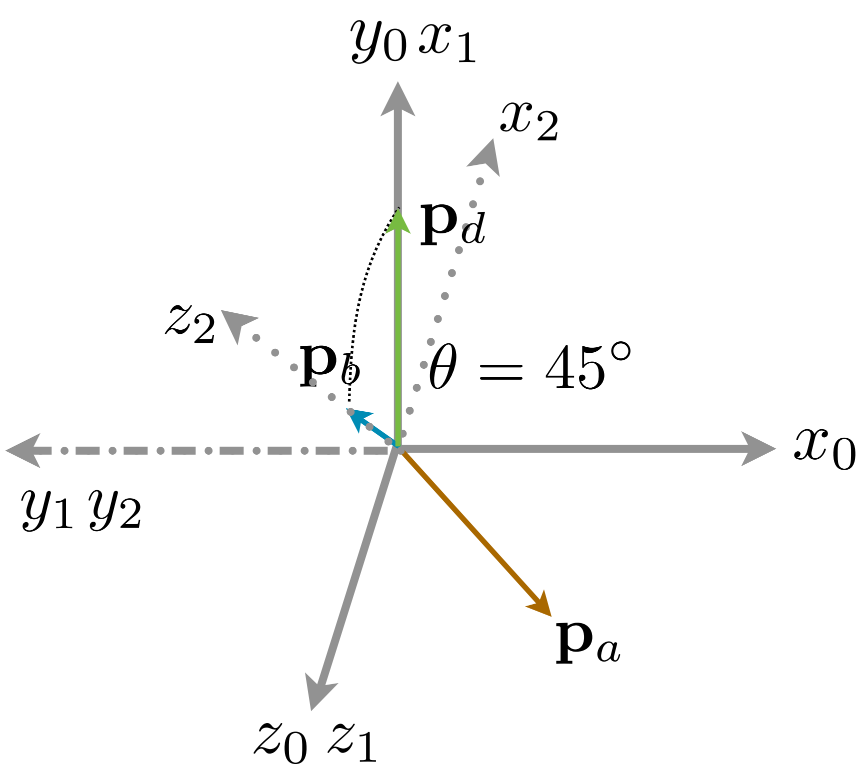
$$\boxed{\mathbf{v}_p^0 = \mathbf{R}_1^0 \mathbf{v}_p^1}$$

Subscript and superscript cancel

$$\mathbf{v}_p^1 = ? \qquad \left(\mathbf{R}_1^0\right)^{-1} \mathbf{v}_p^0 = \cancel{\left(\mathbf{R}_1^0\right)^{-1} \mathbf{R}_1^0} \mathbf{v}_p^1 \qquad \mathbf{v}_p^1 = \left(\mathbf{R}_1^0\right)^T \mathbf{v}_p^0$$

$$\boxed{\mathbf{v}_p^1 = \mathbf{R}_0^1 \mathbf{v}_p^0}$$

What if I want to apply another rotation to this vector?
Method depends on which axis I want to rotate around.



For example:

rotate 45° around y_0

vs.

rotate 45° around y_1

$$\mathbf{p}_d^2 = \mathbf{p}_a^0$$

$$\boxed{\mathbf{p}_d^0 = \mathbf{R}_1^0\, \mathbf{R}_2^1\, \mathbf{p}_a^0}$$

interp. 3 (operator)

$$\mathbf{p}_d^0 = ?$$

$$\mathbf{p}_d^0 = \mathbf{R}_2^0\, \mathbf{p}_d^2$$

$$\mathbf{p}_d^0 = \mathbf{R}_1^0\, \mathbf{p}_d^1$$

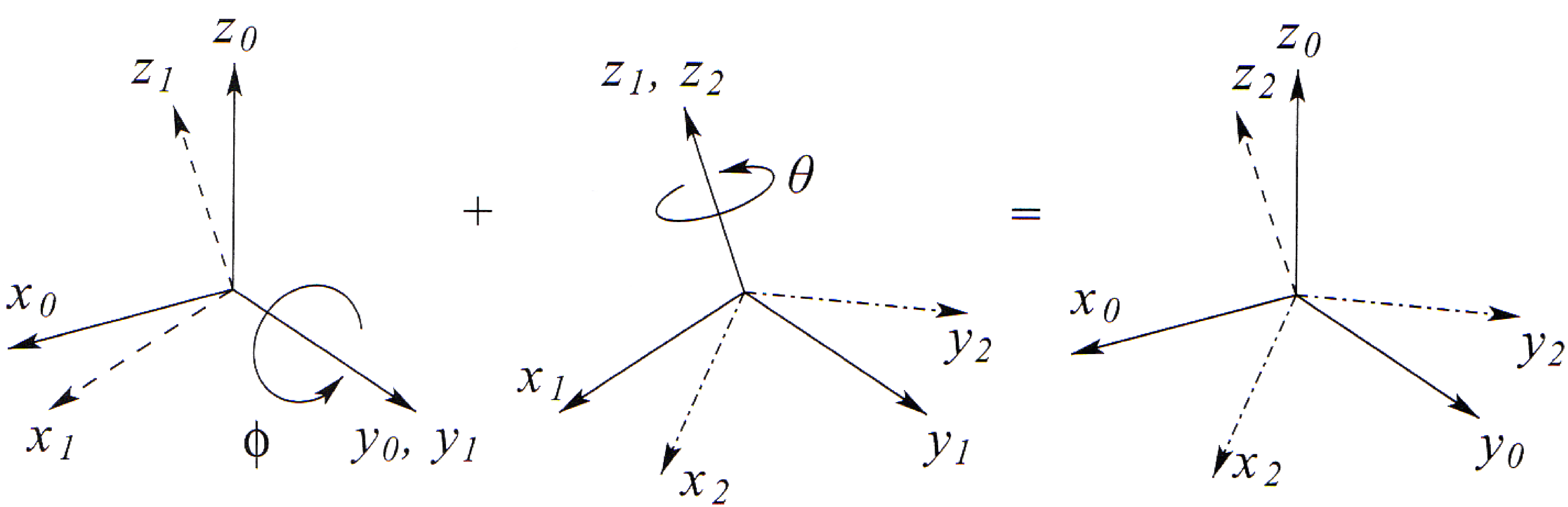
$$\mathbf{p}_d^1 = \mathbf{R}_2^1\, \mathbf{p}_d^2$$

$$\boxed{\mathbf{R}_2^0 = \mathbf{R}_1^0\, \mathbf{R}_2^1}$$

$$\boxed{\mathbf{p}_d^0 = \mathbf{R}_1^0\, \mathbf{R}_2^1\, \mathbf{p}_d^2}$$

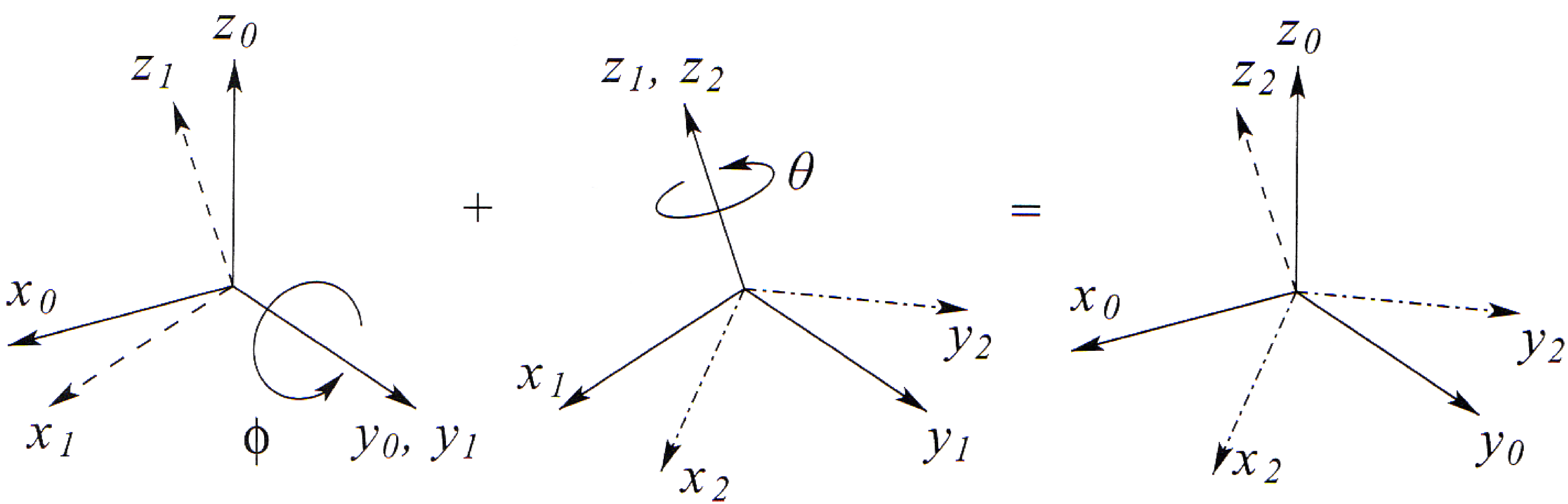
interp. 2 (frames)

Composition of Rotations with Respect to the Current Frame



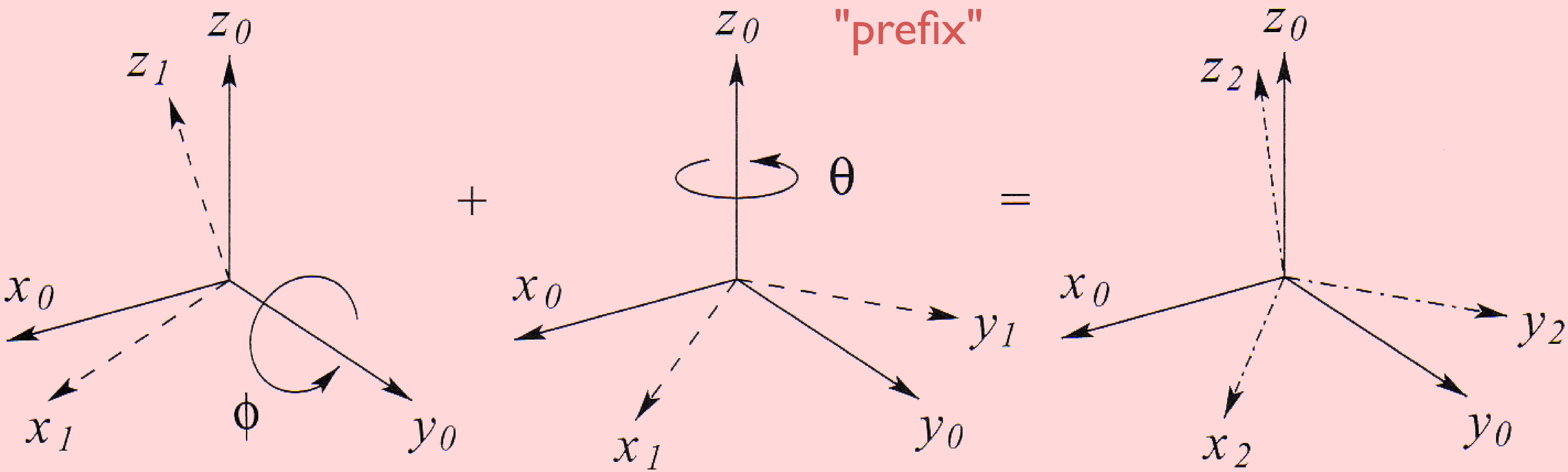
the result of a successive rotation about the current (intermediate) frame can be found by **post-multiplying** by the corresponding rotation matrix

$$\mathbf{R}_2^0 = \mathbf{R}_1^0 \, \mathbf{R}_2^1$$



successive rotation about fixed frame? **pre-multiply** $\mathbf{R}_2^0 = \mathbf{R}\,\mathbf{R}_1^0$

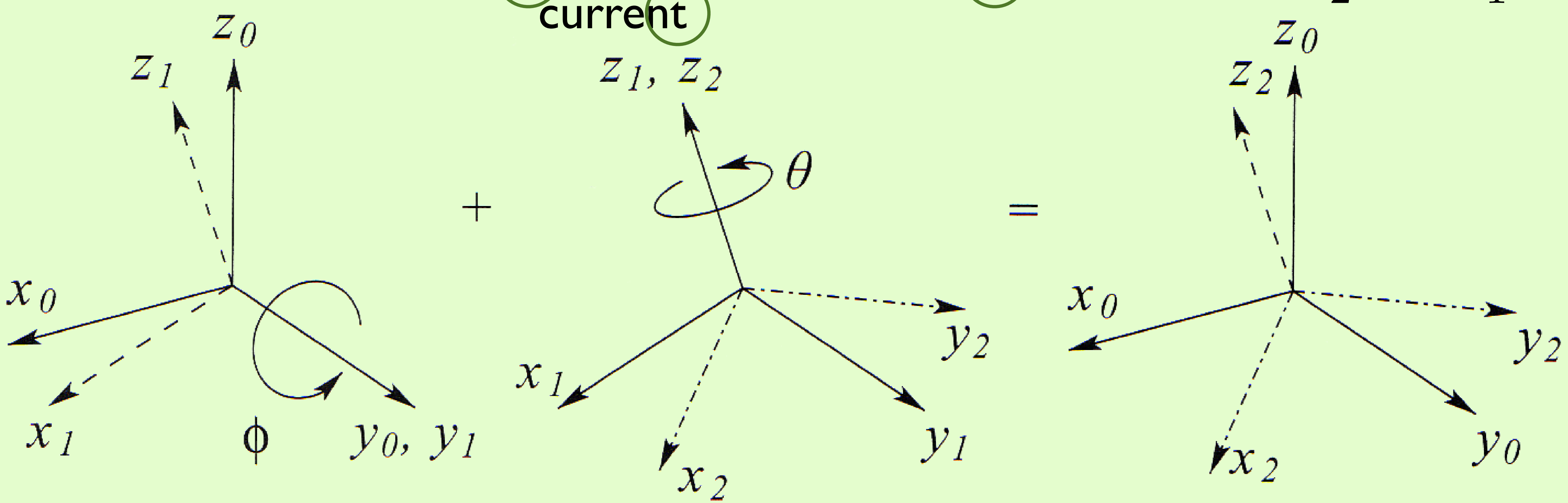
"prefix"



Which of these is more commonly used in robotics?

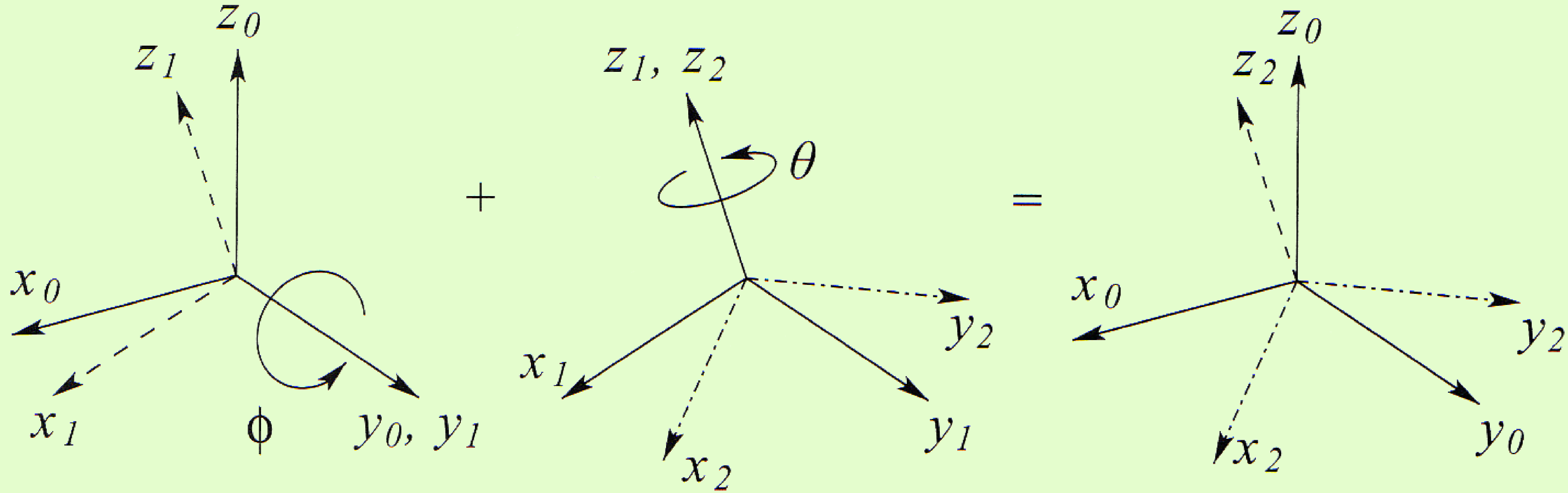
successive rotation about intermediate frame? **post-multiply** $\mathbf{R}_2^0 = \mathbf{R}_1^0\,\mathbf{R}_2^1$

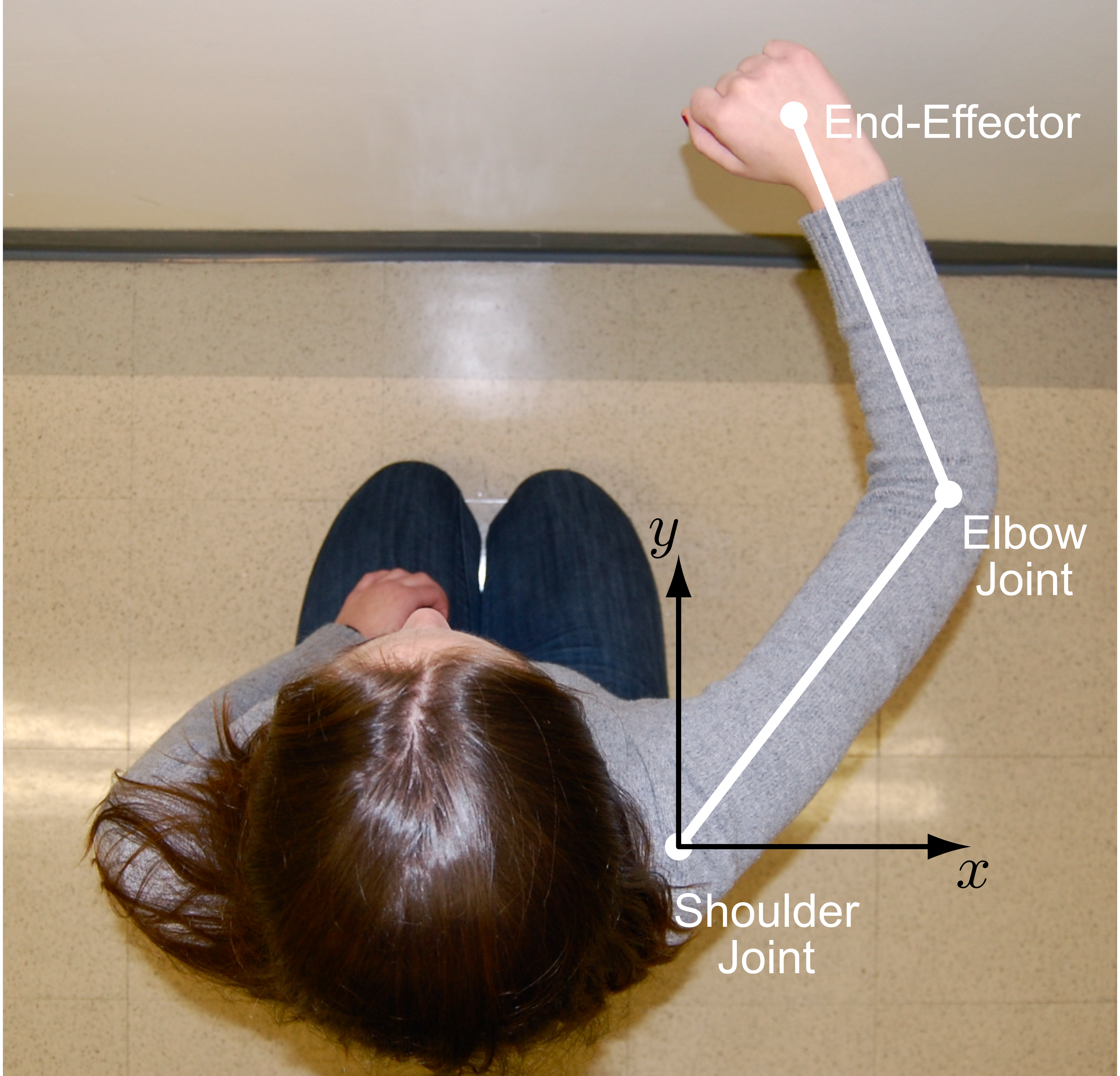
current



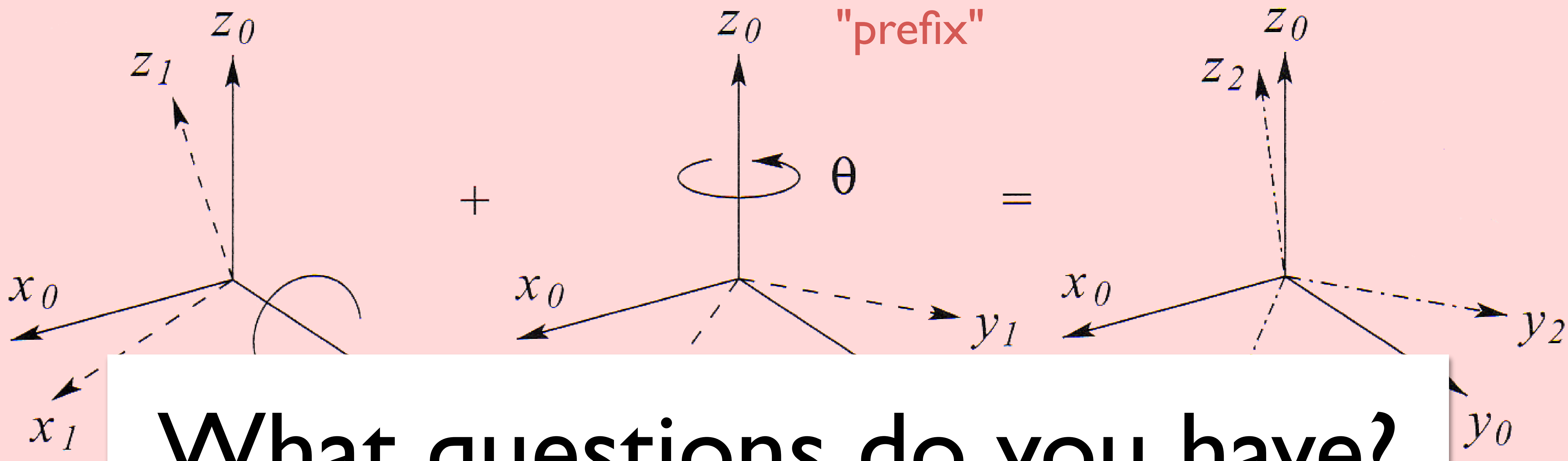
Which of these is more commonly used in robotics?

successive rotation about intermediate frame? **post-multiply** $\mathbf{R}_2^0 = \mathbf{R}_1^0\,\mathbf{R}_2^1$




End-Effector
Elbow Joint
y
x
Shoulder Joint

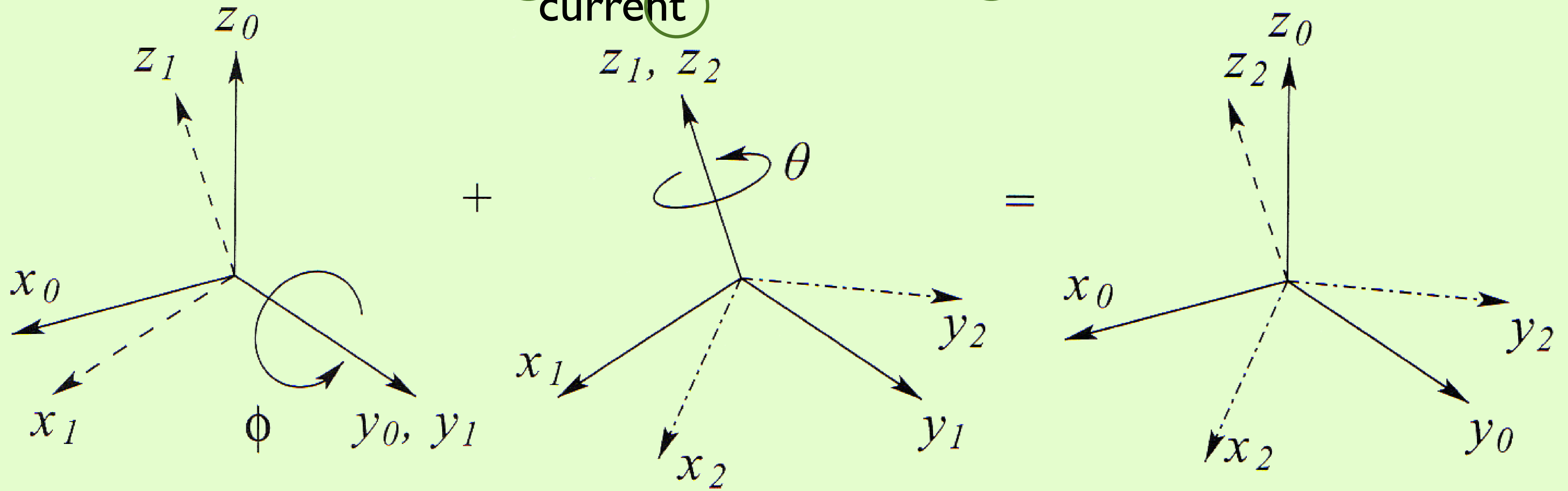
successive rotation about fixed frame? **pre-multiply** $\mathbf{R}_2^0 = \mathbf{R}\,\mathbf{R}_1^0$



What questions do you have?

successive rotation about intermediate frame? **post-multiply** $\mathbf{R}_2^0 = \mathbf{R}_1^0\,\mathbf{R}_2^1$

current

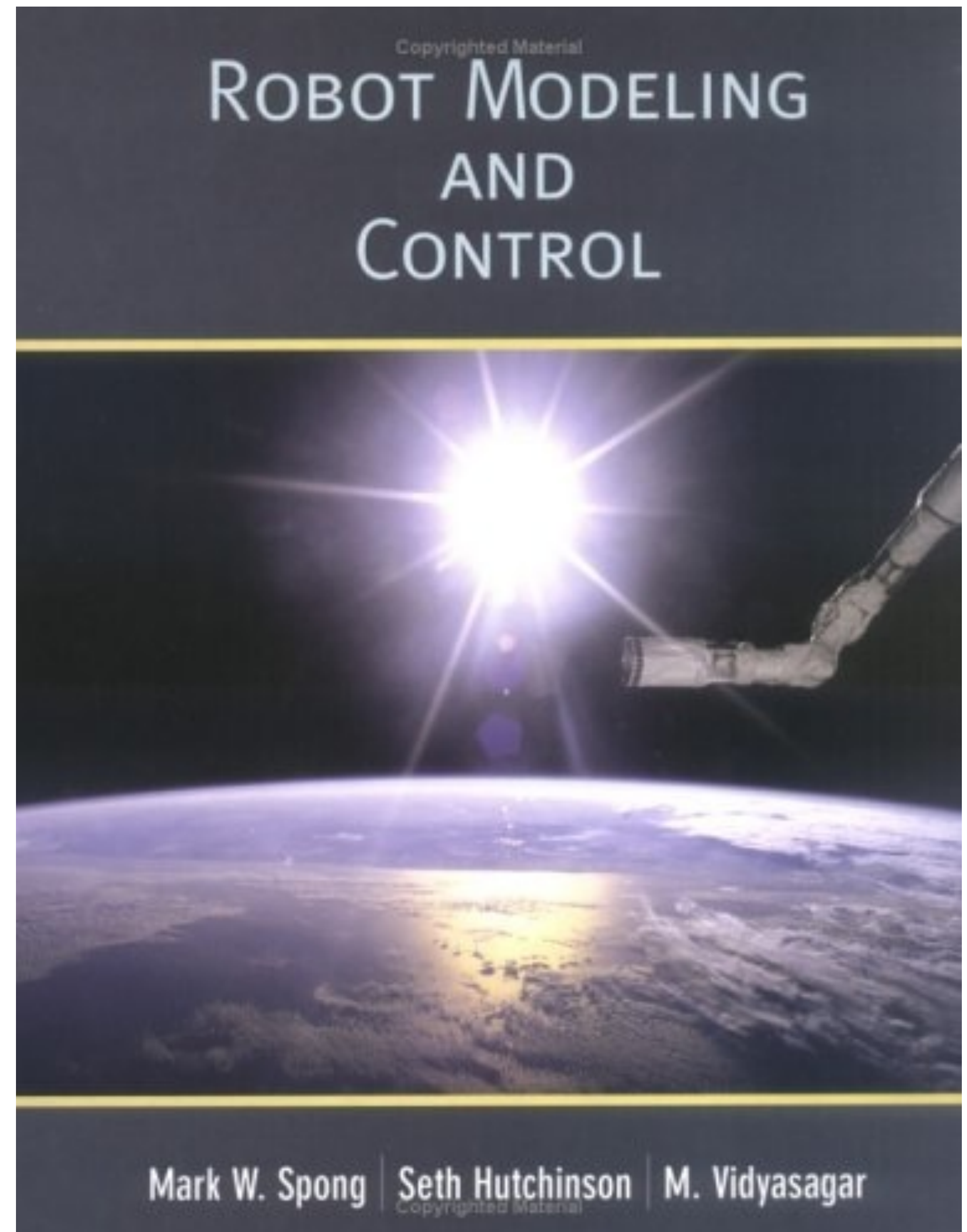


Reading Assignment

Within Chapter 2: Rigid Motions and Homogeneous Transformations

- Read Secs. 2.4 and 2.5

Deadline: Tuesday lecture



Homework 2

- Due by midnight (11:59 p.m.) on Thursday, September 11.
- Done individually, but you can help each other; you must understand everything you write down, not copy from others.
- Write in pencil, show your work, box your answers, and staple all pages.
- See Piazza for instructions on how to apply to submit electronically.
- I have not yet taught the material for problems 6, 7, and 8; those topics will be covered in lecture on Tuesday.

Homework 2:
Robotic Manipulators and Rotation Matrices

MEAM 520, University of Pennsylvania
Katherine J. Kuchenbecker, Ph.D.

September 4, 2013

This paper-based assignment is due on Thursday, September 11, by midnight (11:59:59 p.m.) You should aim to turn it in during class that day. If you don't finish until later in the day, you can turn it in to Professor Kuchenbecker's office, Towne 224, in the assignment submission box or under the door. Late submissions will be accepted until Sunday, September 14, by midnight (11:59:59 p.m.), but they will be penalized by 10% for each partial or full day late, up to 30%. After the late deadline, no further assignments may be submitted.

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 write down must be your own work, not copied from any other individual or a solution manual. Any submissions suspected of violating Penn's Code of Academic Integrity will be reported to the Office of Student Conduct. If you get stuck, post a question on Piazza or go to office hours!

These problems are a mix of custom problems and augmented problems from the textbook, *Robot Modeling and Control* by Spong, Hutchinson, and Vidyasagar (SHV). Please follow the extra clarifications and instructions when provided. Write in pencil, show your work clearly, $\boxed{\text{box your answers}}$, and staple together all pages of your assignment. This assignment is worth a total of 25 points.

1. Custom problem – **Kinematics of Baxter** *(2 points)*
   Rethink Robotics sells a two-armed manufacturing robot named Baxter. Watch YouTube videos of Baxter (e.g., "Baxter Robot Folds a Shirt", "Intera 3 Baxter Software Update Signature Moves") to learn about its kinematics. Draw a schematic of the serial kinematic chain of Baxter's left arm (the one the woman is touching in the picture below.) Use the book's conventions for how to draw revolute and prismatic joints in 3D.

Assignment continued on reverse.