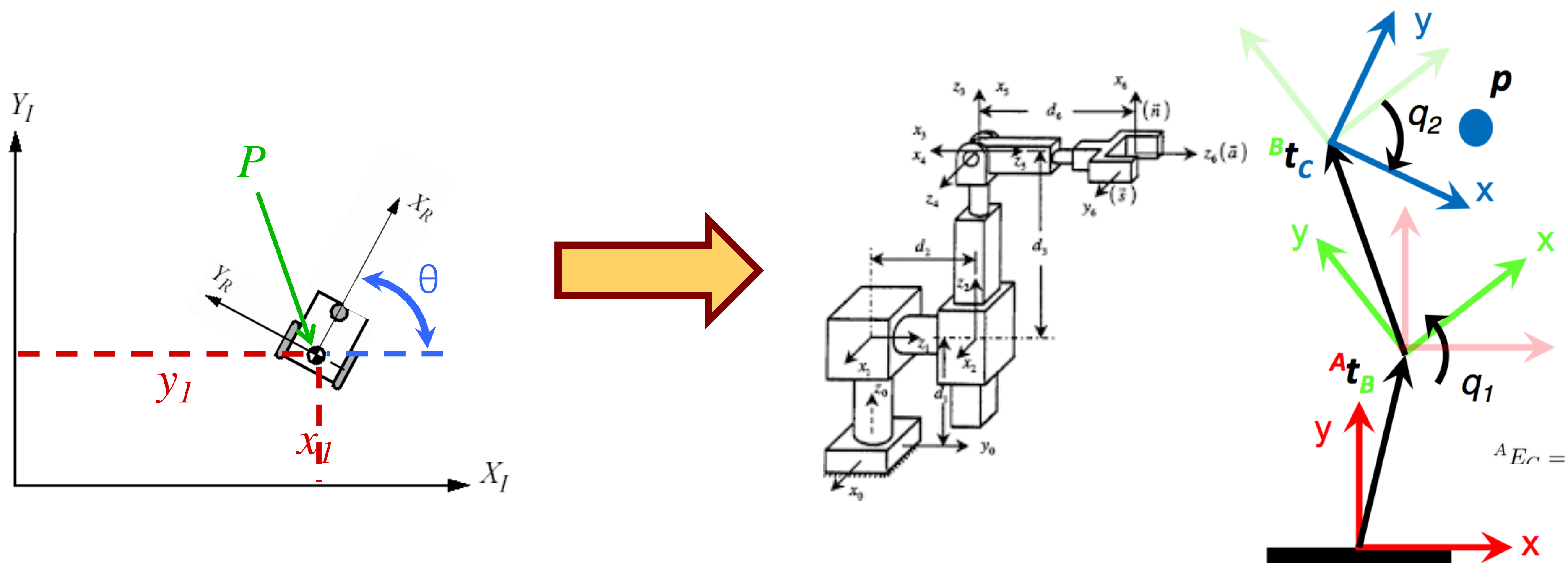
Kinematics: Solving Sequences

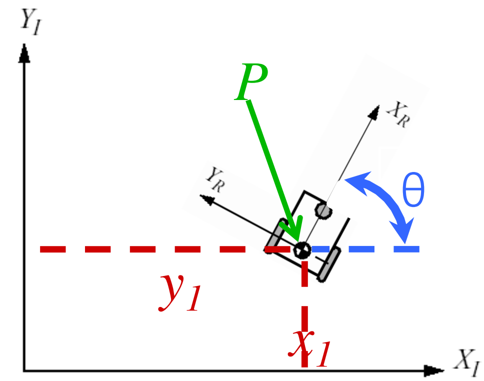
Manipulators & DH parameters



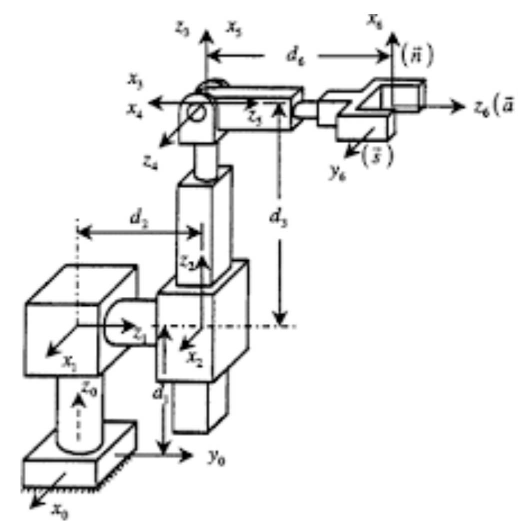
Many slides, graphics, and ideas adapted (with thanks!) from:
Siegwart, Nourbakhsh and Scaramuzza, Autonomous Mobile Robots
Renata Melamud, An Introduction to Robot Kinematics, CMU
Rick Parent, Computer Animation, Ohio State
Steve Rotenberg, Computer Animation, UCSD
Angela Sodemann, www.youtube.com/watch?v=lVjFhNv2N8o, ASU

Review: Kinematics

- **Goal:** Figure out *where robot or end effector is* in the global (world) frame of reference

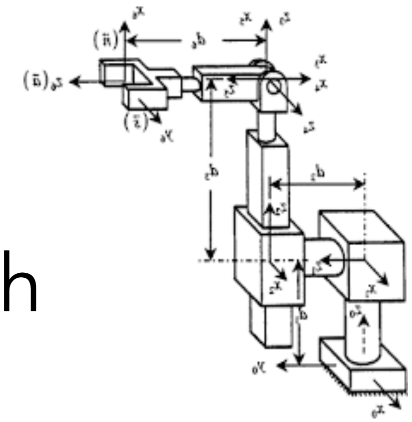


- **Method:** Treat robot and world as having different *frames of reference* that we convert between
  - Use matrices that represent complete information about world and robot
    - Where it is (location of the origin)
    - Which way axes are pointed (orientation)

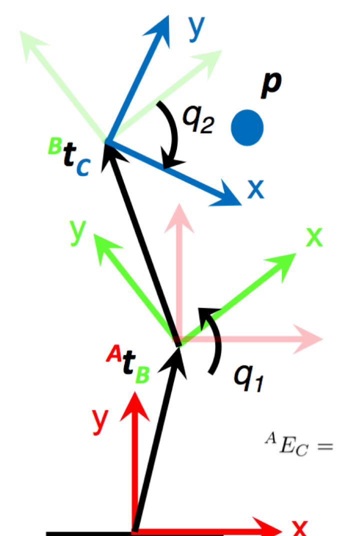


- **Why:** so we can put the robot where we want it

Review: Manip. Kinematics



- **Goal:** Figure out where end effector is
- **Method:** Treat each joint as having its own independent frame that moves with respect to the previous one
  - Joint 1 moves wrt. the world
  - Joint 2 moves wrt. joint 1, etc.



- Then, each transform from joint i to joint i+1 is a single rotation or translation
  - Each joint has one degree of freedom
  - Multi-axis joints are treated as 2+ joints connected by zero-length links

Review: Rotation

- Any frame of reference can be represented by:
  - 3 numbers for planar movement: (x, y, θ)
  - 6 numbers for 3D: $(x, y, z, \phi, \theta, \psi)$ (roll/pitch/yaw)
- Any rotation can be broken down into single rotations around one axis each
- We transform rotations between frames by multiplying by a rotation matrix
  - Derived trigonometrically

$$R_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix} \quad R_Y = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \quad R_Z = \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Review: Translation

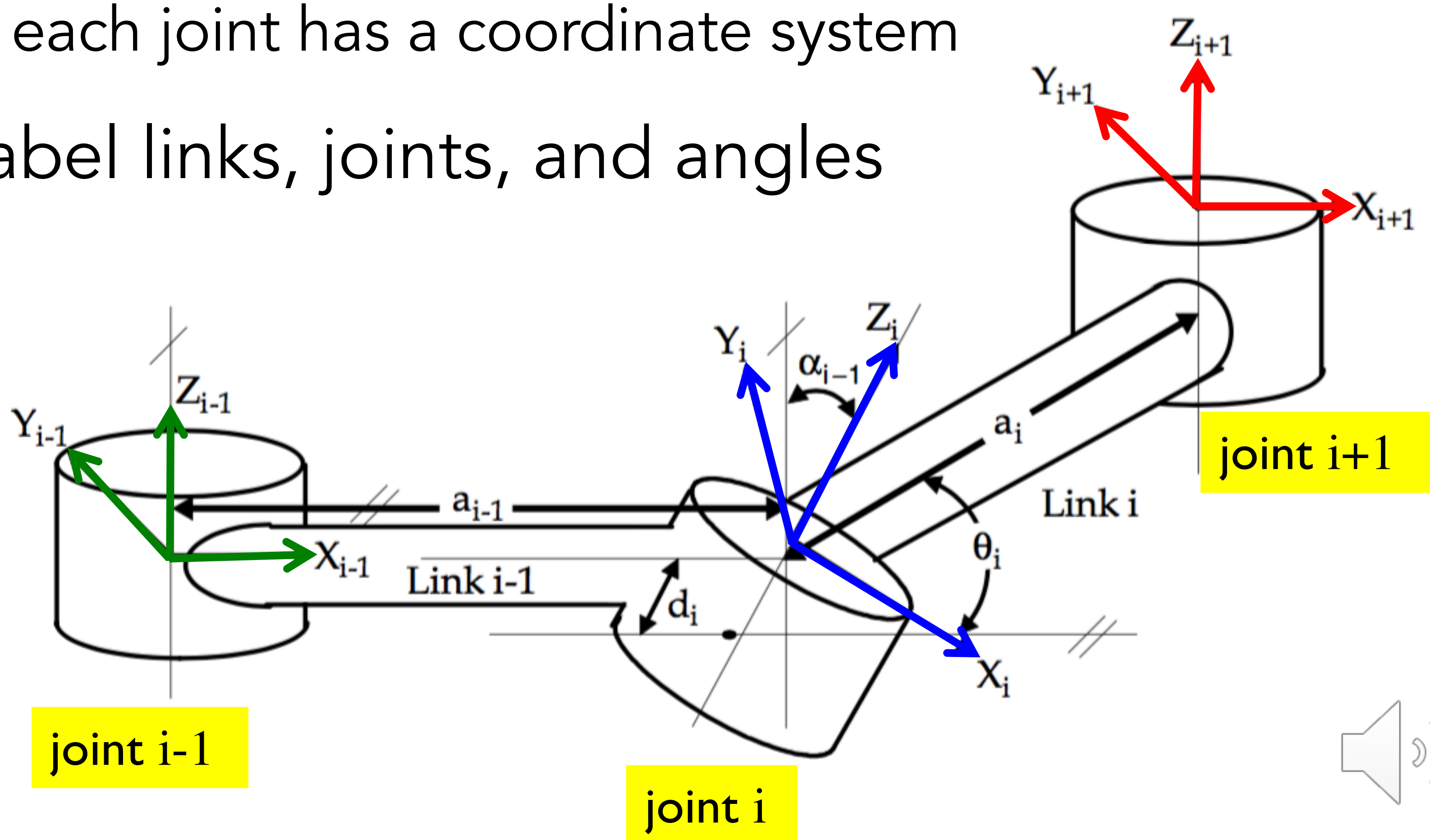
- Any frame of reference can be represented by:
  - 3 numbers for planar movement: (x, y, θ)
  - 6 numbers for 3D: $(x, y, z, \phi, \theta, \psi)$
- Any translation (movement) can be broken down into single moves in one axis each
- We transform translations by adding location information to the matrices, as follows

$$\begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(we'll do rotation and translation exercises in class, as well as DH)

Review: Describing A Manipulator

- Arm made up of links in a chain
- Joints each have <x,y,z> and roll/pitch/yaw
  - So, each joint has a coordinate system
- We label links, joints, and angles



Review: Kinematics

- You'll sometimes see vector $\mathbf{\Phi}$ to represent the array of M joint values:

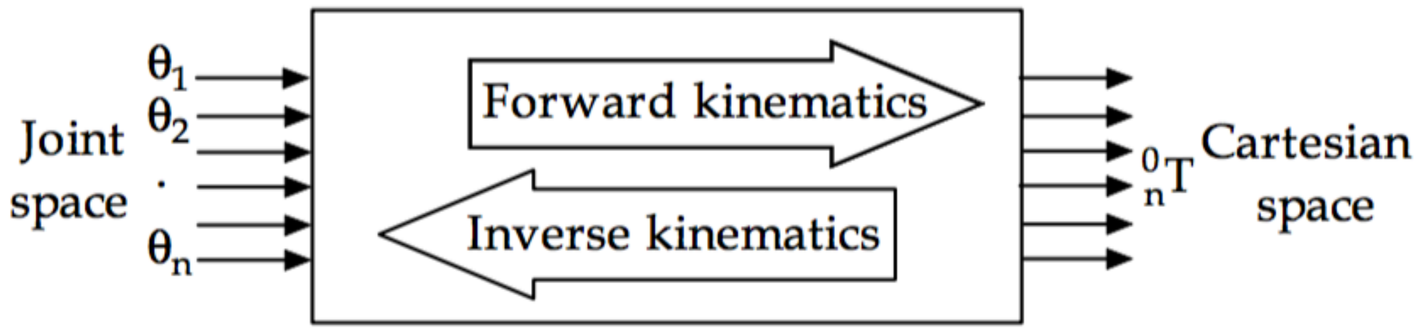
$$\mathbf{\Phi} = \begin{bmatrix} \phi_1 & \phi_2 & \dots & \phi_M \end{bmatrix}$$

- We sometimes use vector $\mathbf{e}$ to represent array of N values describing end effector
  - Position/orientation
  - E.g., configuration

$$\mathbf{e} = \begin{bmatrix} e_1 & e_2 & \dots & e_N \end{bmatrix}$$

- Example:
  - For end effector position and orientation, $\mathbf{e}$ would contain 6 DOFs: 3 translations and 3 rotations
  - If we only need position, $\mathbf{e}$ would contain 3 translations

Review: Forward & Inverse



Joint space (robot space – previously R)

$\theta_1, \theta_2, \ldots, \theta_n$

This is what we can directly control

Cartesian space (global space – previously I)

(x,y,z), r/p/y

This is where things in the robot's environment are

Forward: i → i-1

- We are we looking for transformation matrix (or transform) T that converts between frame i and frame i-1:

$$T_i^{i-1} \qquad (\text{or } {}^{i-1}_{i}T\,) \qquad (\text{or } {}^{i-1}T_i\,)$$

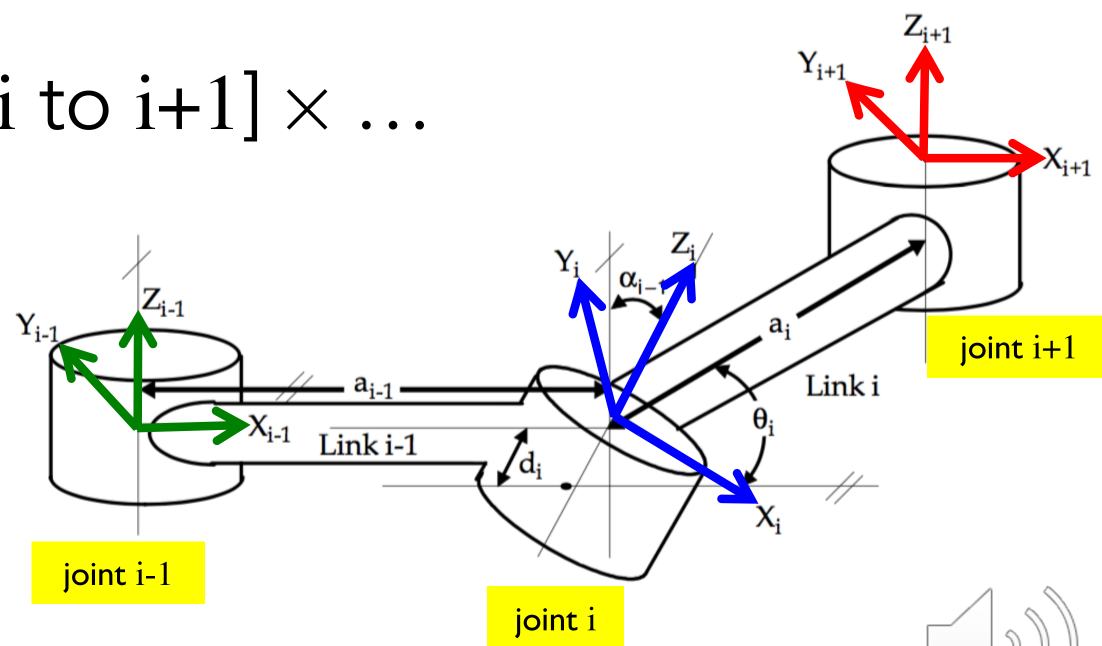
- Determine position and orientation of end-effector as function of displacements in joints

Forward Kinematics and IK

- Joint angles ⇆ end effector configuration in I
- Can string together rotations with multiplication
  - So, can get end effector rotation by ?
- Finding rotation from
  [joint i-1 to i] × [joint i to i+1] × …

$$\boxed{R_2^0} = R_1^0 R_2^1$$

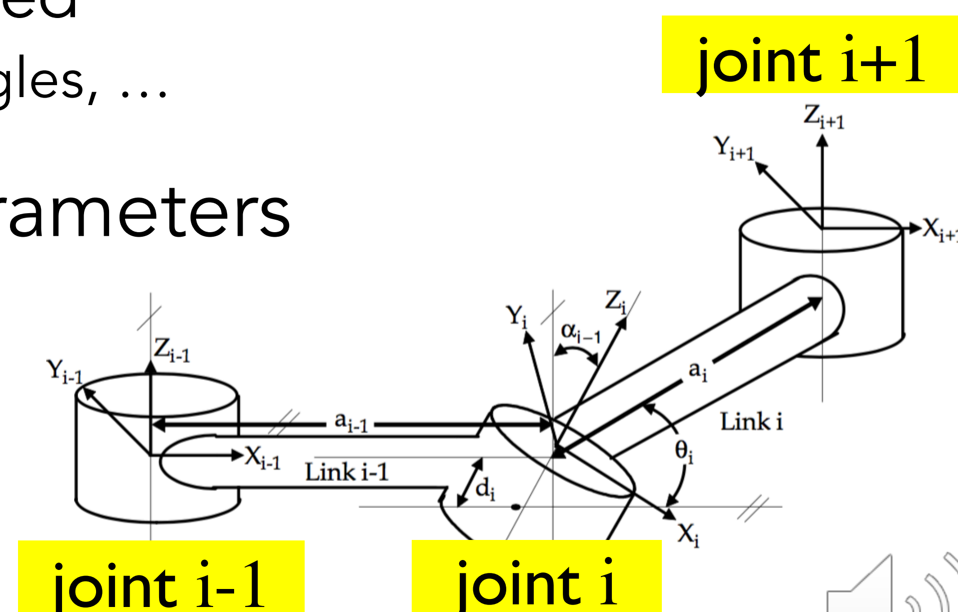
- **Rotation of end effector frame**, relative to **base frame**



www.youtube.com/watch?v=lVjFhNv2N8o

Describing A Manipulator

- But where do these frames come from?
- An arm is made up of links in a chain
  - How to describe each link?
  - Many choices exist
  - DH parameters widely used
    - Also: quaternions, Euler angles, …
- Denavit-Hartenberg parameters
  - DH parameters let you describe each ${}^{t-i}T_i$ with only 4 values
  - $a_{i-1}, \alpha_{i-1}, d_i, \theta_2$



Denavit-Hartenberg Method

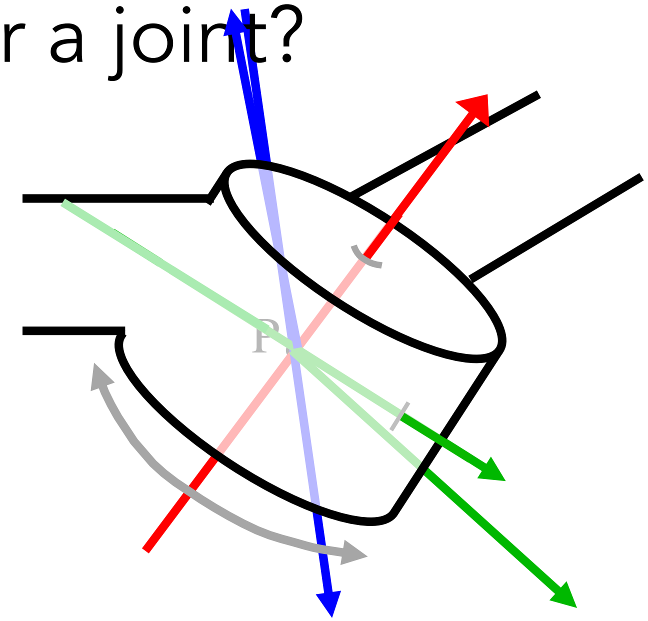
- Efficient way of finding transformation matrices

1. **Set** frames for all joints
   - This is actually the tricky part
2. **Calculate** all DH parameters from frames
   - 4 DH parameters (not 6!) define position/orientation*
3. **Popul**
4. **Popul**
   - Matr
5. **Multiply** all matrices together, in order
   - 0-1 × 1-2 × 2-3 × …

* How is this possible?
We already have some constraints on joints – i.e., that they're connected by a rigid link that can rotate or displace (but not both)

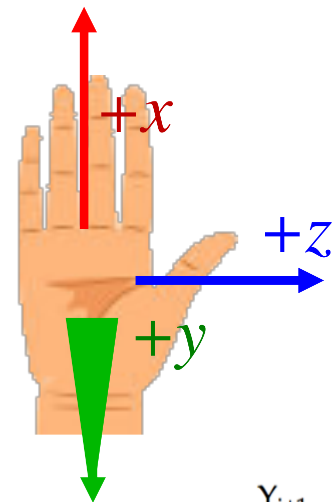
Defining Frames for Joints

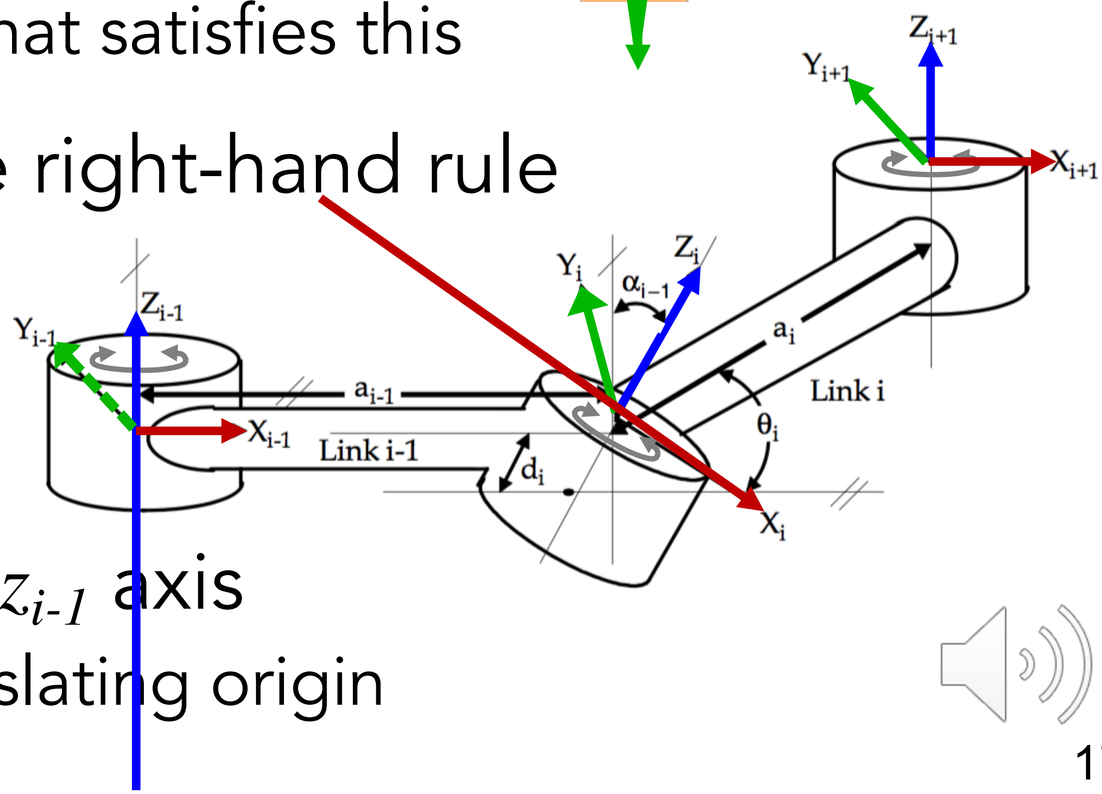
- What's the frame of reference for a joint?
  - Actually, completely flexible
- We usually choose:
  - 1 axis through the center of rotation/direction of displacement
  - 2 more perpendicular to that
    - Which can be any orientation!
- We can move the origin
  - P is no longer <0, 0, 0>
- To use DH method, choose frames carefully



1. Choose Frames for DH

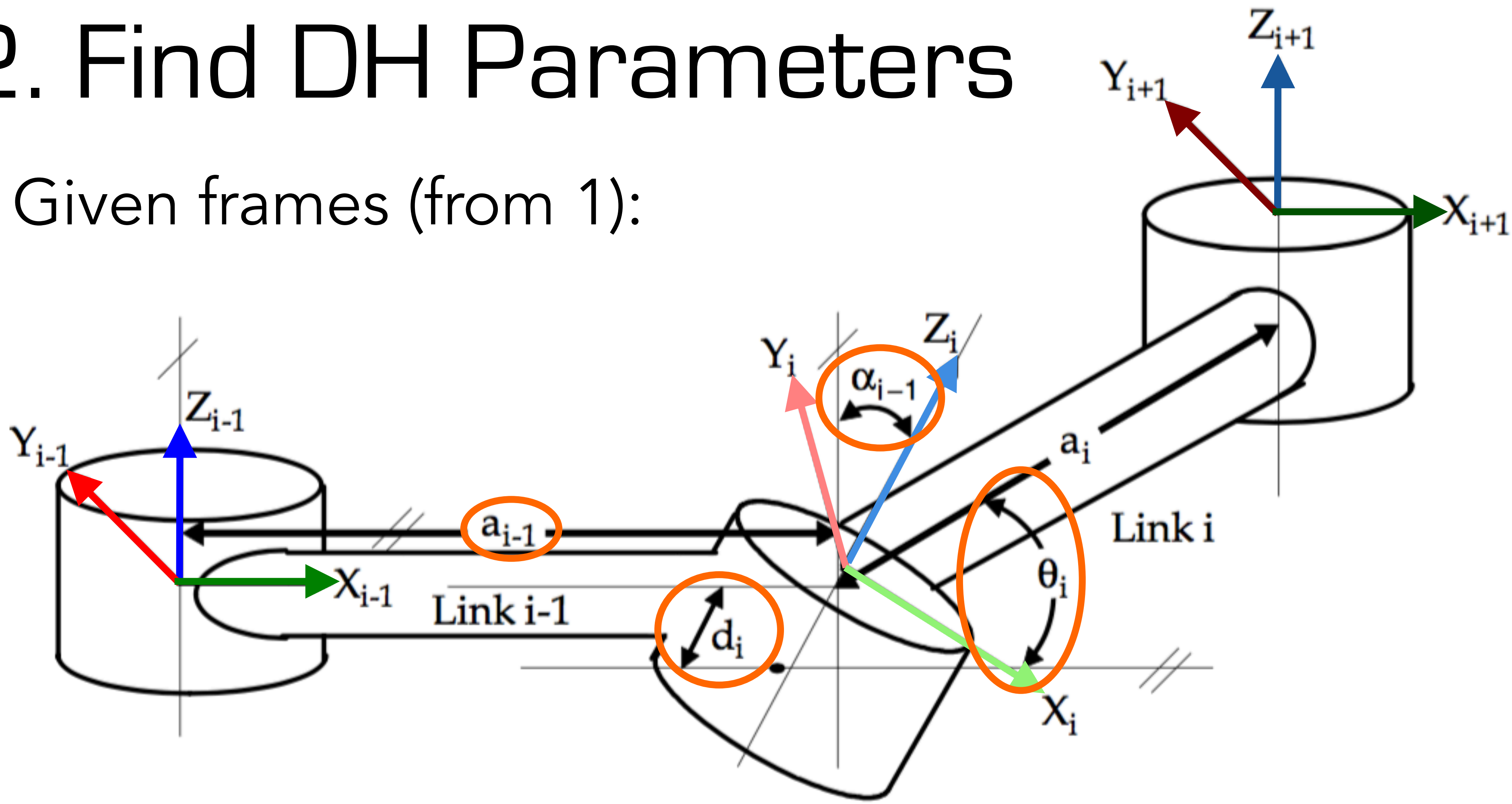
- z axis must be axis of motion
  - Rotation around z for revolute
  - Translation along z for prismatic
- x_i axis orthogonal to z_i **and** z_{i-1}
  - There's always a line that satisfies this
- y axis must follow the right-hand rule
  - Fingers point $+x$
  - Thumb points $+z$
  - Palm faces $+y$
- x_i axis must intersect z_{i-1} axis
  - Which may mean translating origin





2. Find DH Parameters

Given frames (from 1):



a_{i-1} : link length – distance $Z_{i-1} \leftrightarrows Z_i$ along X_i

α_{i-1} : link twist – angle $Z_{i-1} \leftrightarrows Z_i$ around X_i

d_i : link offset – distance X_{i-1} to X_i along Z_i

θ_i : joint angle – angle X_{i-1} and X_i around Z_i

3. DH Parameter Table

Given parameters:

- Create a parameter table
  - # of rows = (# of frames) – 1
  - Columns = 4 (always) ← DH parameters θ, α, a, d

	θ	**α**	**a**	**d**
frame 0-1	$\theta_{0\text{-}1}$	$\alpha_{0\text{-}1}$	$a_{0\text{-}1}$	$d_{0\text{-}1}$
frame 1-2	$\theta_{1\text{-}2}$	$\alpha_{1\text{-}2}$	$a_{1\text{-}2}$	$d_{1\text{-}2}$
frame 2-3	…	…	…	…

4. Make Transform Matrix

Given parameter table:

- Fill DH transformation matrix* for each transition:

$$R_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_{i,i+1} & \sin\theta_i \sin\alpha_{i,i+1} & a_{i,i+1}\cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_{i,i+1} & -\cos\theta_i \sin\alpha_{i,i+1} & a_{i,i+1}\sin\theta_i \\ 0 & \sin\alpha_{i,i+1} & \cos\alpha_{i,i+1} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- And multiply (e.g., $R_2^0 = R_1^0 R_2^1$)
- R_2^0 is the same matrix as would be found by other methods. DH is fast and efficient.

** Conceptual derivation is at the end*

Transformation i to i-1

a_{i-1} : distance Z_{i-1} and Z_i along X_i
α_{i-1} : angle Z_{i-1} and Z_i around X_i

together: *screw displacement*

$$[X_i] = \text{Trans}_{x_i}(a_{i,i+1})\text{Rot}_{x_i}(a_{i,i+1})$$

d_i : distance X_{i-1} to X_i along Z_i
θ_2 : angle X_{i-1} and X_i around Z_i

together: *screw displacement*

$$[Z_i] = \text{Trans}_{z_i}(a_{i,i+1})\text{Rot}_{z_i}(a_{i,i+1})$$

- Coordinate transformation:

$${}^{i-1}_{i}T = [Z_i][X_i] = \text{Trans}_{z_i}(d_i)\text{Rot}_{z_i}(\theta_i)\text{Trans}_{x_i}(a_{i,i+1})\text{Rot}_{x_i}(a_{i,i+1})$$

Transformation i to i-1

$$\mathrm{Trans}_{z_i}(d_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{Rot}_{x_i}(\alpha_{i,i+1}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_{i,i+1} & -\sin\alpha_{i,i+1} & 0 \\ 0 & \sin\alpha_{i,i+1} & \cos\alpha_{i,i+1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{Trans}_{x_i}(a_{i,i+1}) = \begin{bmatrix} 1 & 0 & 0 & a_{i,i+1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{Rot}_{z_i}(\theta_i) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Transformation in DH:

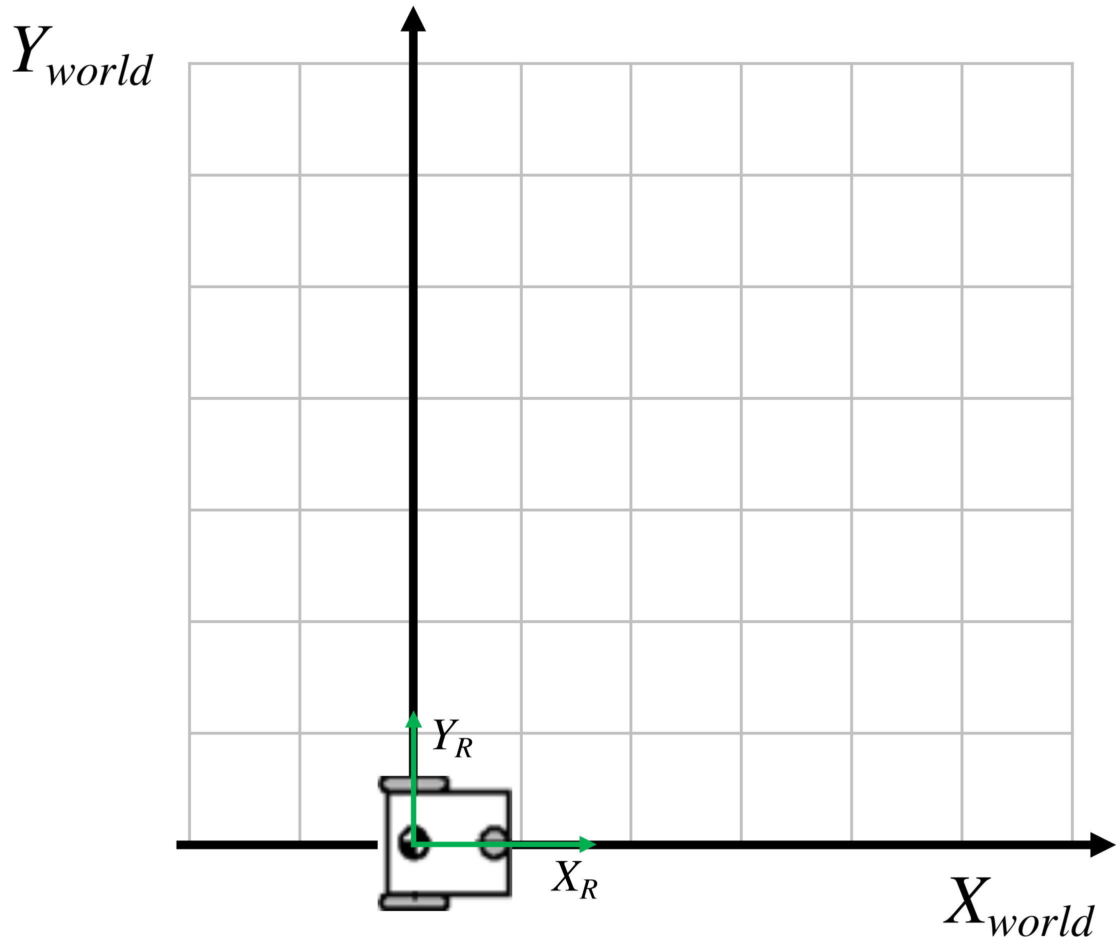
$$R_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_{i,i+1} & \sin\theta_i\sin\alpha_{i,i+1} & a_{i,i+1}\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_{i,i+1} & -\cos\theta_i\sin\alpha_{i,i+1} & a_{i,i+1}\sin\theta_i \\ 0 & \sin\alpha_{i,i+1} & \cos\alpha_{i,i+1} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Exercises

Exercise 1

- **Homogeneous transformations** translate and rotate simultaneously.

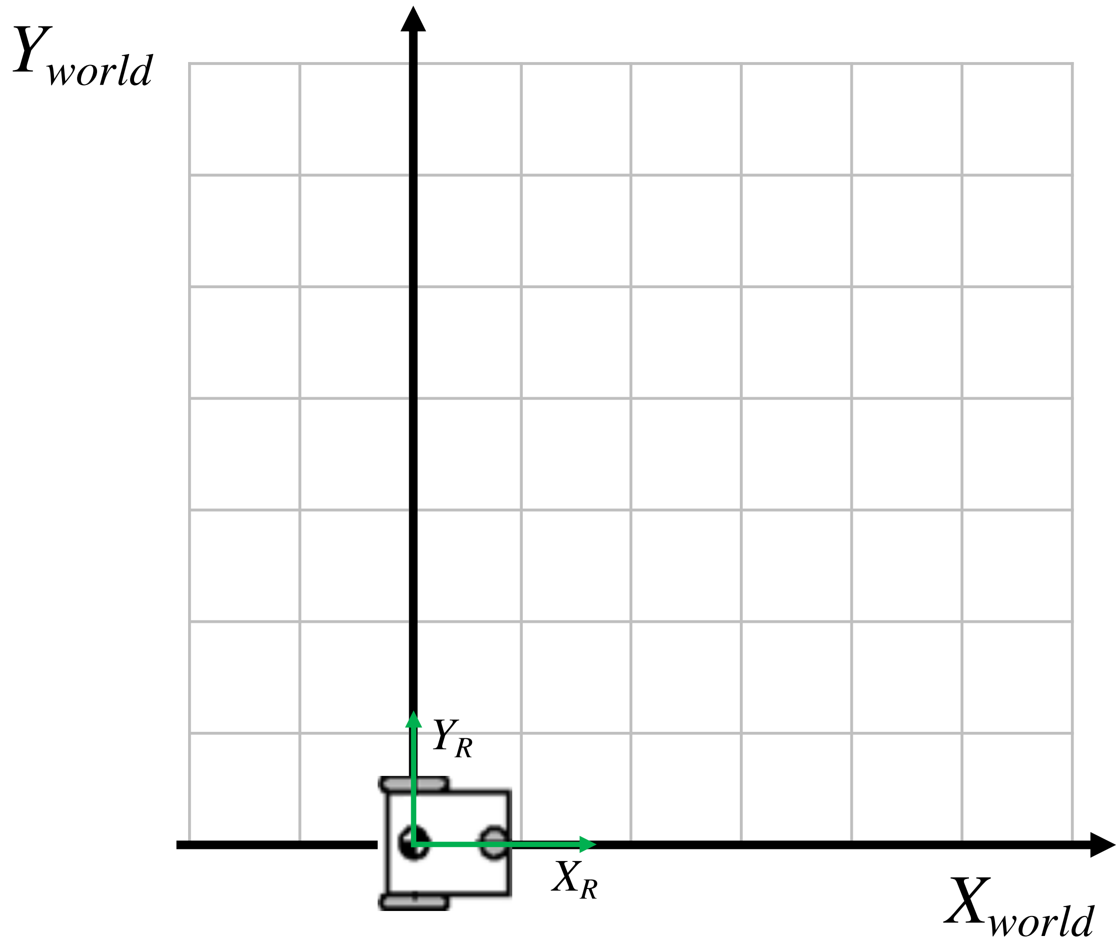
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$^{0}_{1}T =$

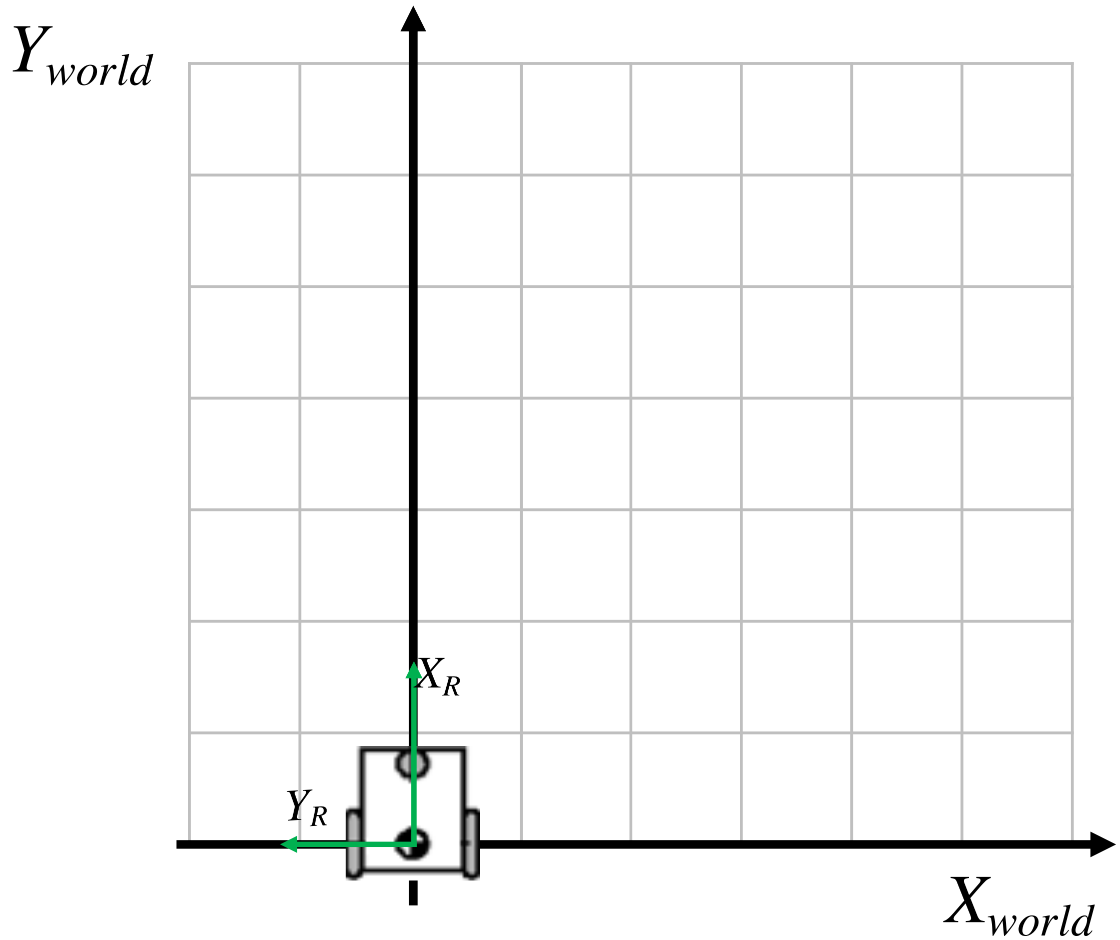
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = $$

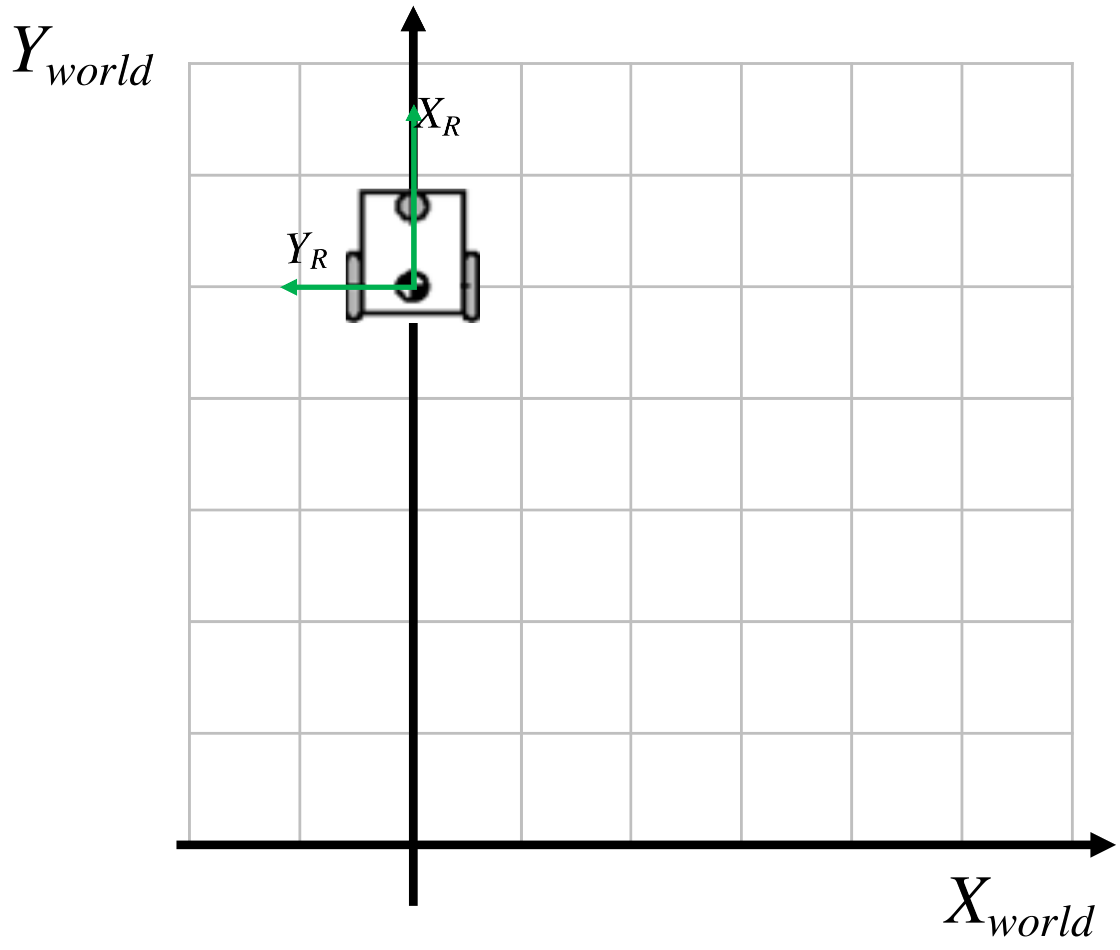
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} \quad \end{bmatrix} $$

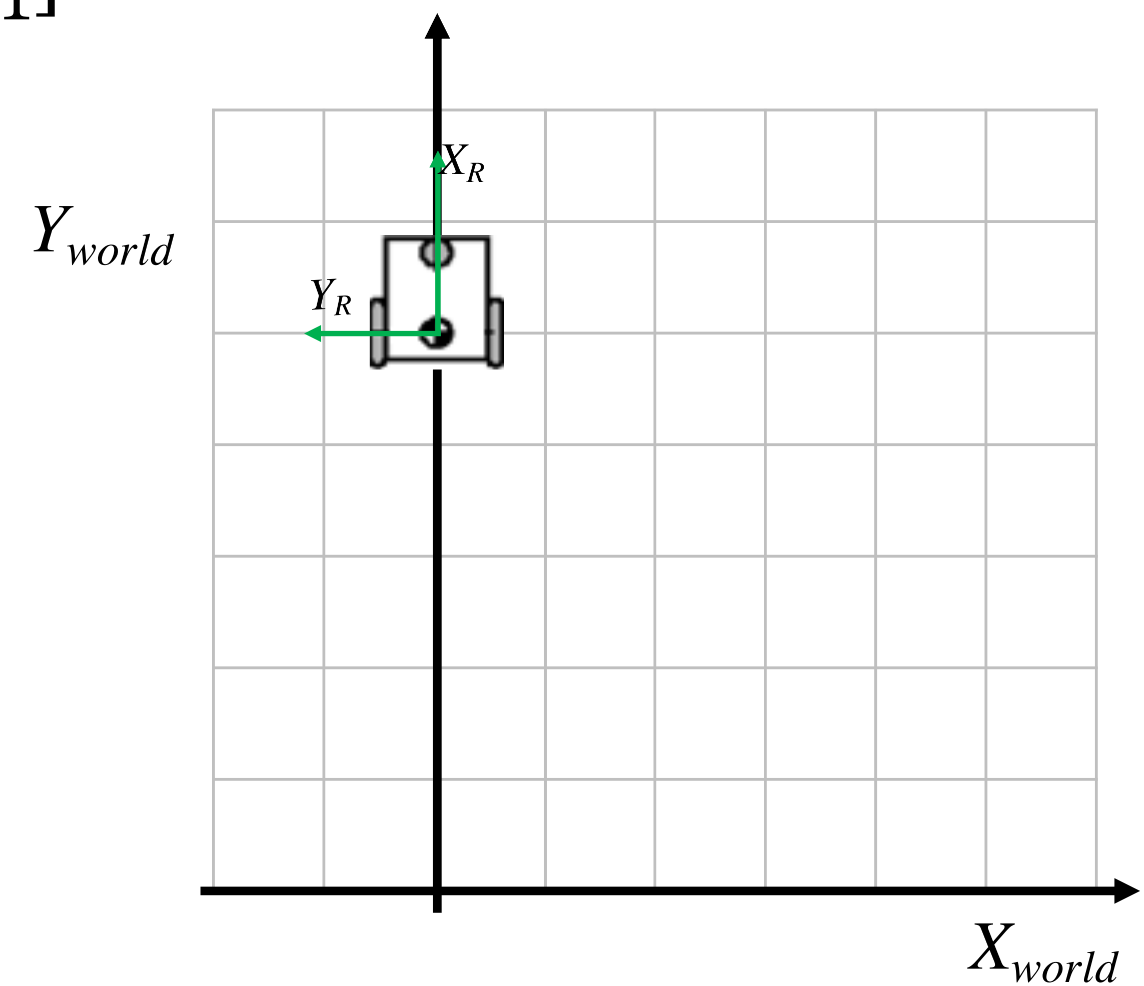
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^0_1T = \begin{bmatrix} \cos\theta & -\sin\theta & x \\ \sin\theta & \cos\theta & y \\ 0 & 0 & 1 \end{bmatrix} $$

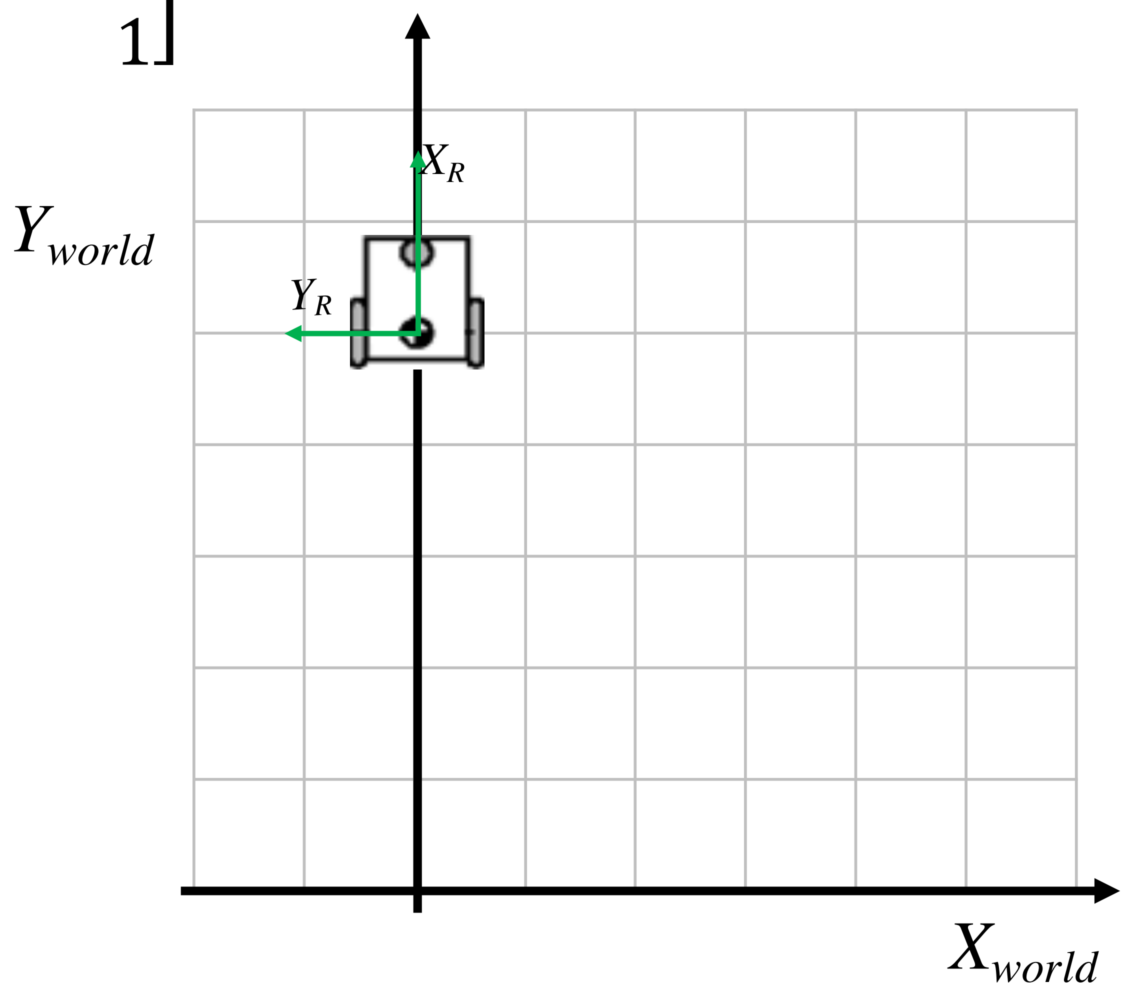
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} \cos {}^{\pi}/_{2} & -\sin {}^{\pi}/_{2} & 0 \\ \sin {}^{\pi}/_{2} & \cos {}^{\pi}/_{2} & 5 \\ 0 & 0 & 1 \end{bmatrix} $$

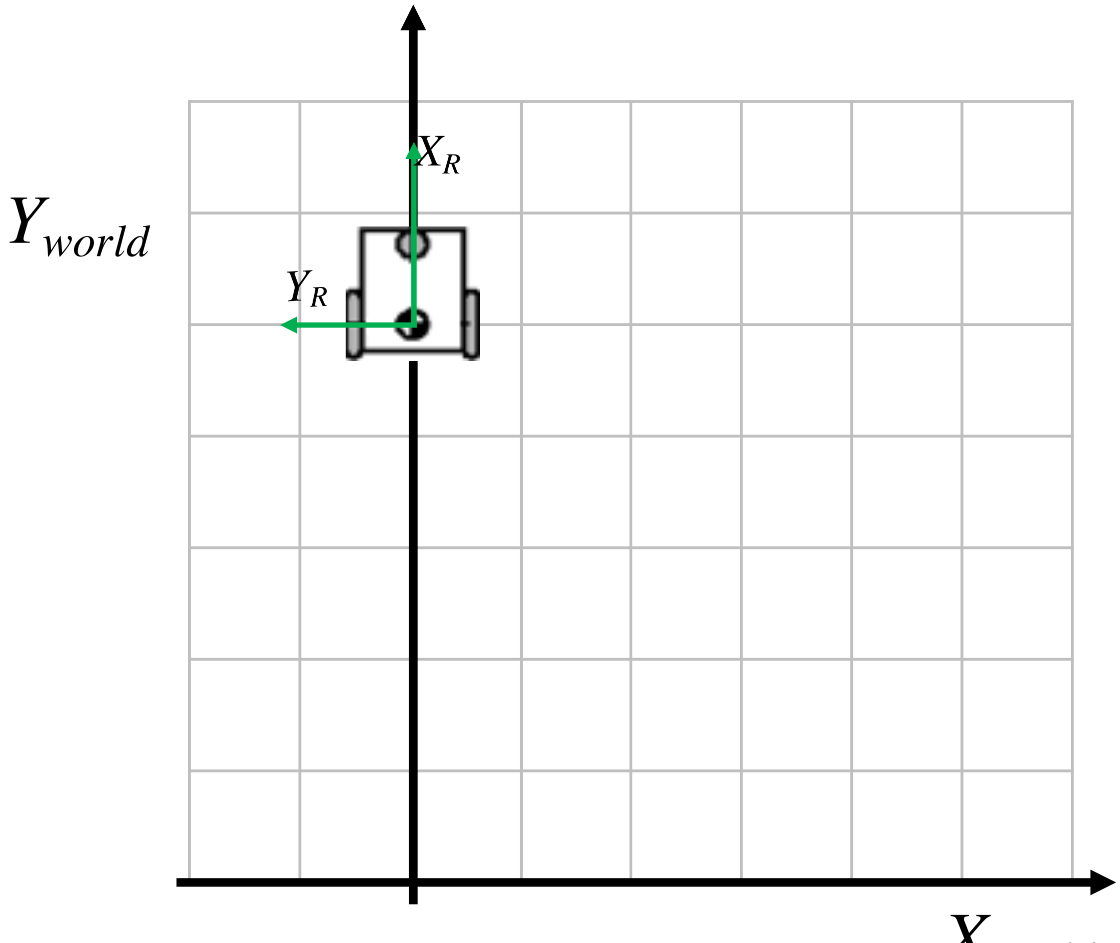
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} $$

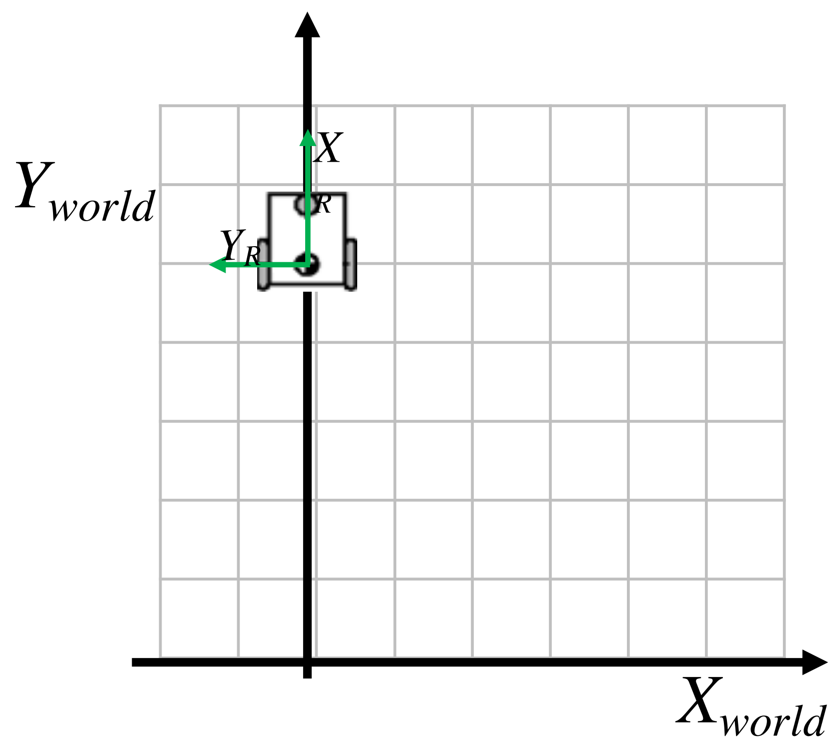
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} $$

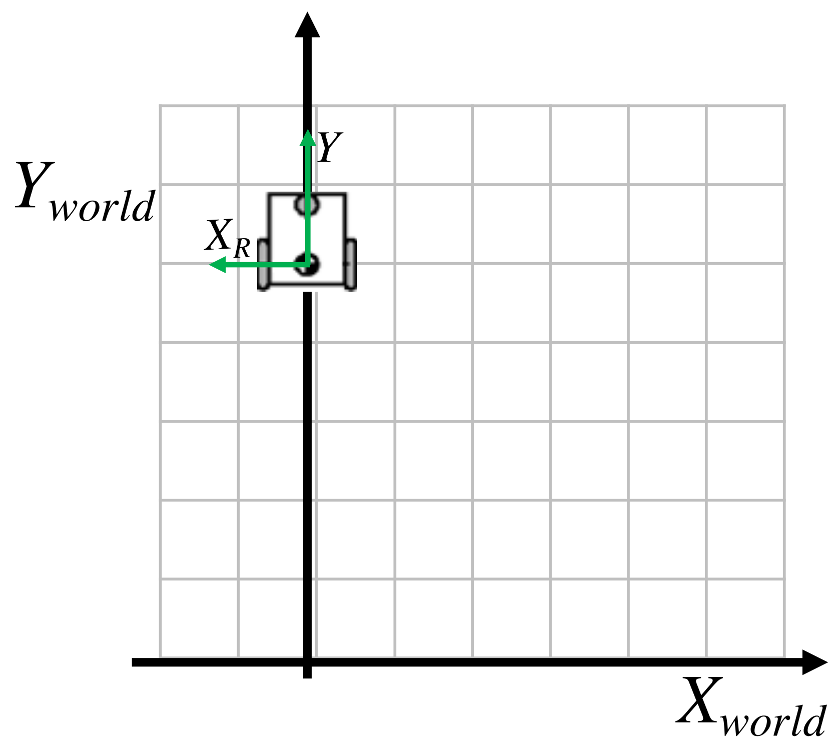
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} $$

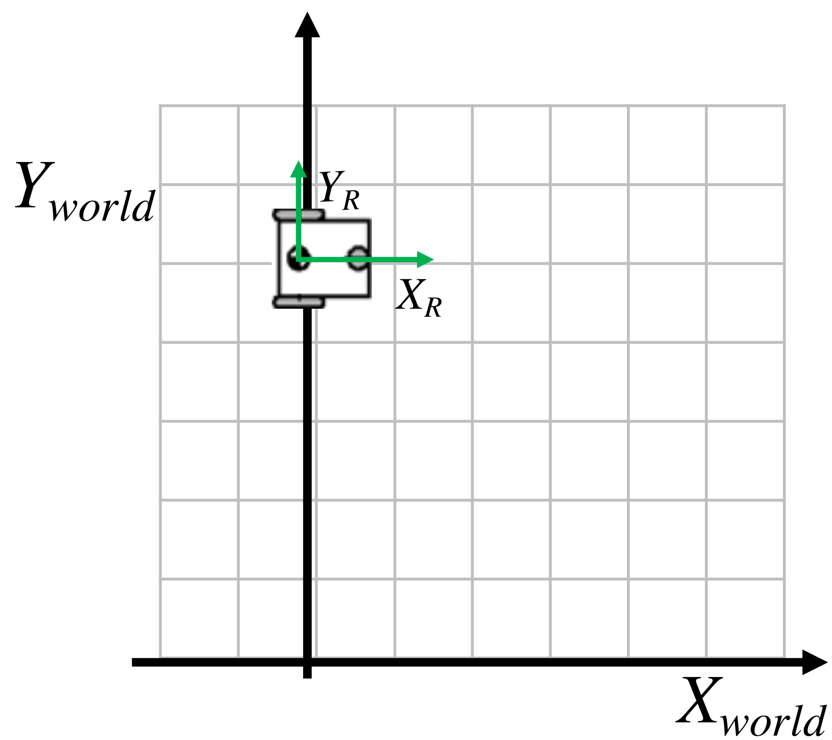
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^0_1T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} \quad {}^1_2T = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} $$

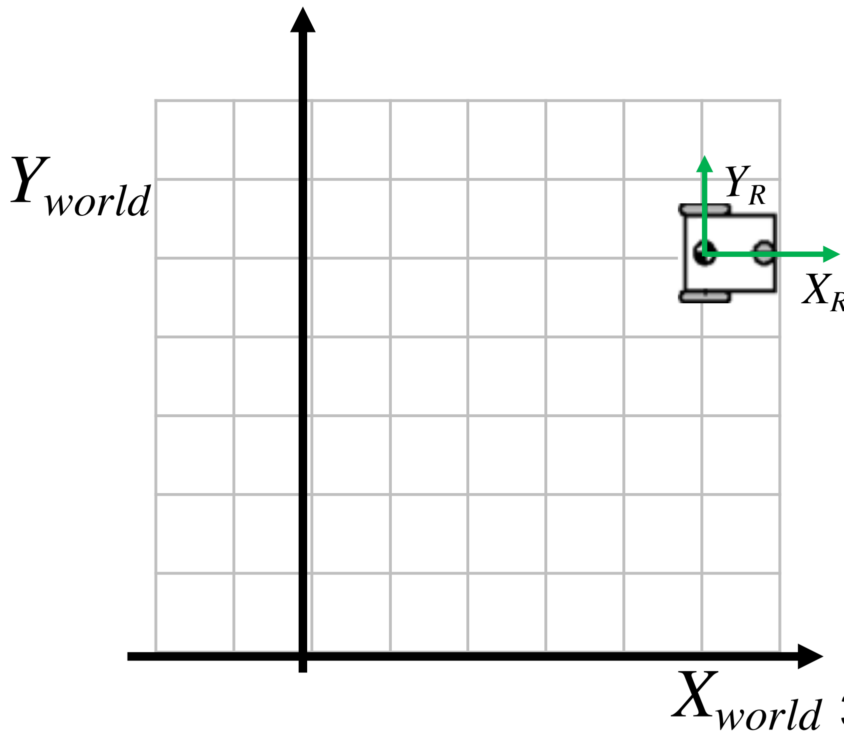
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} $$

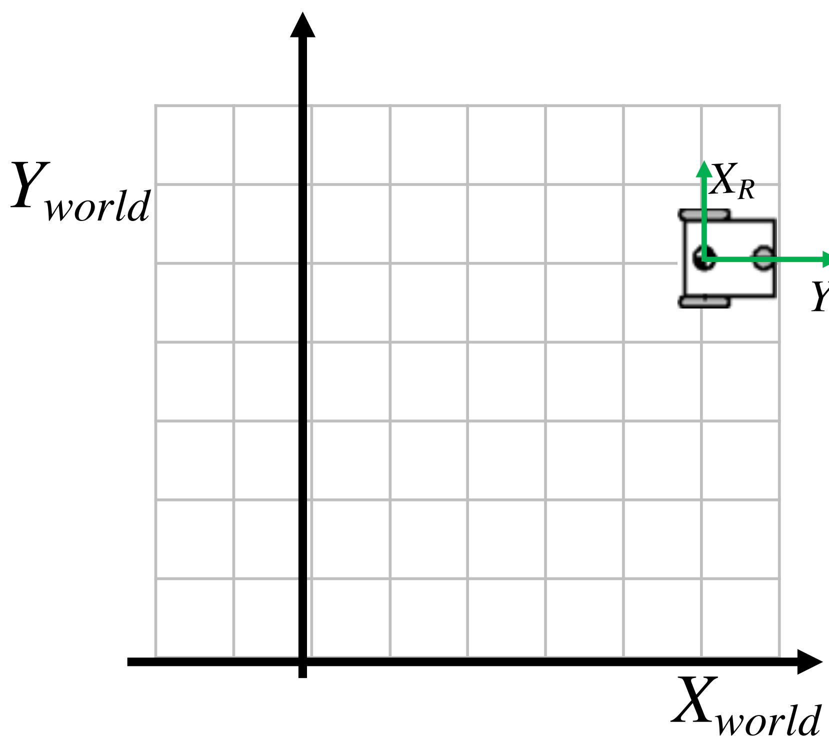
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{1}T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} \quad {}^{1}_{2}T = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -5 \\ 0 & 0 & 1 \end{bmatrix} $$

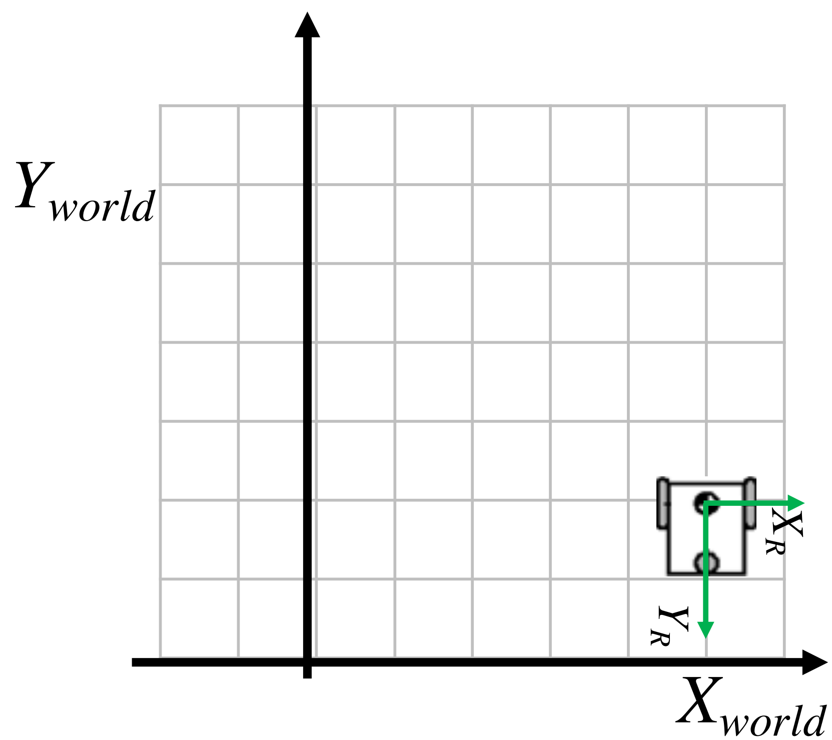
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^0_1T = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 5 \\ 0 & 0 & 1 \end{bmatrix} \quad {}^1_2T = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -5 \\ 0 & 0 & 1 \end{bmatrix} \quad {}^2_3T = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -3 \\ 0 & 0 & 1 \end{bmatrix} $$

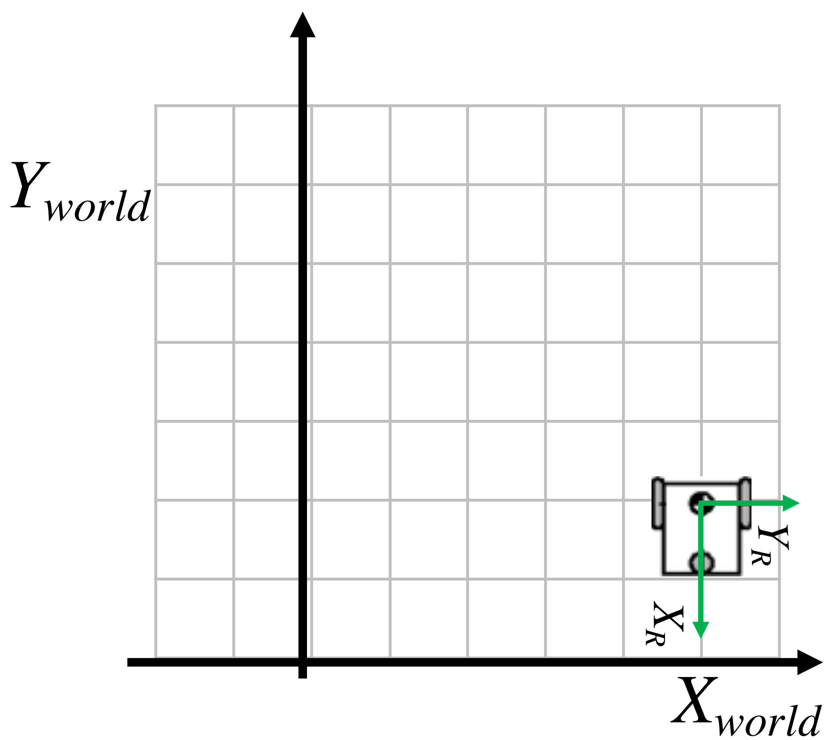
1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3



Exercise 1

$$ {}^{0}_{2}T = {}^{0}_{1}T \times {}^{1}_{2}T \times {}^{2}_{3}T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$

1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3

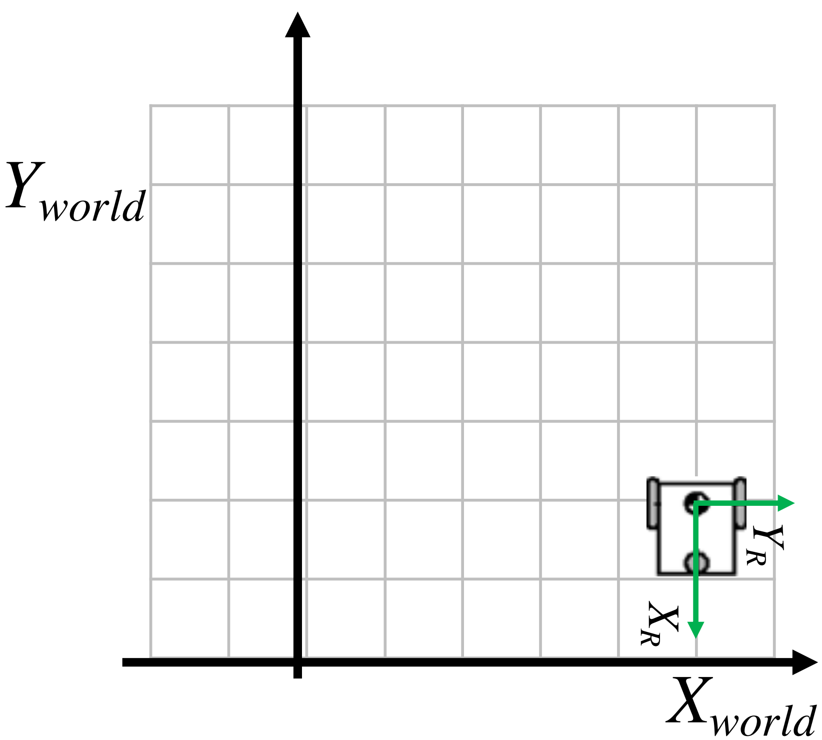


Exercise 1

$$ {}^0_2T = {}^0_1T \times {}^1_2T \times {}^2_3T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$

…which is consistent with what we see.

1. Rotates 90°
2. Moves forward 5
3. Rotates -90
4. Moves forward 5
5. Rotates -90
6. Moves forward 3

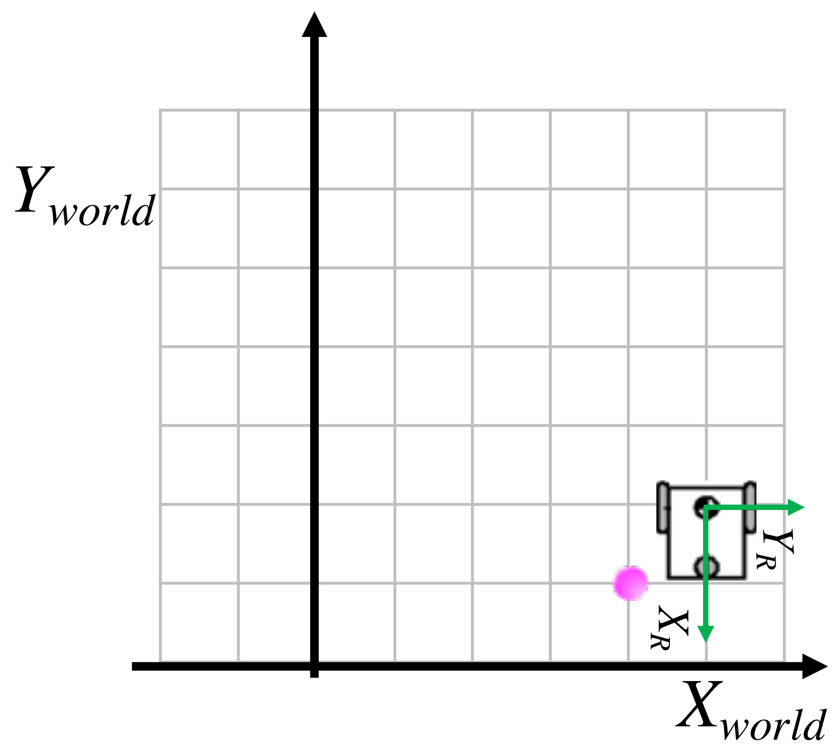


Exercise 2

$$ {}^{0}_{2}T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$

Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
2. What are its coordinates in ***W***?

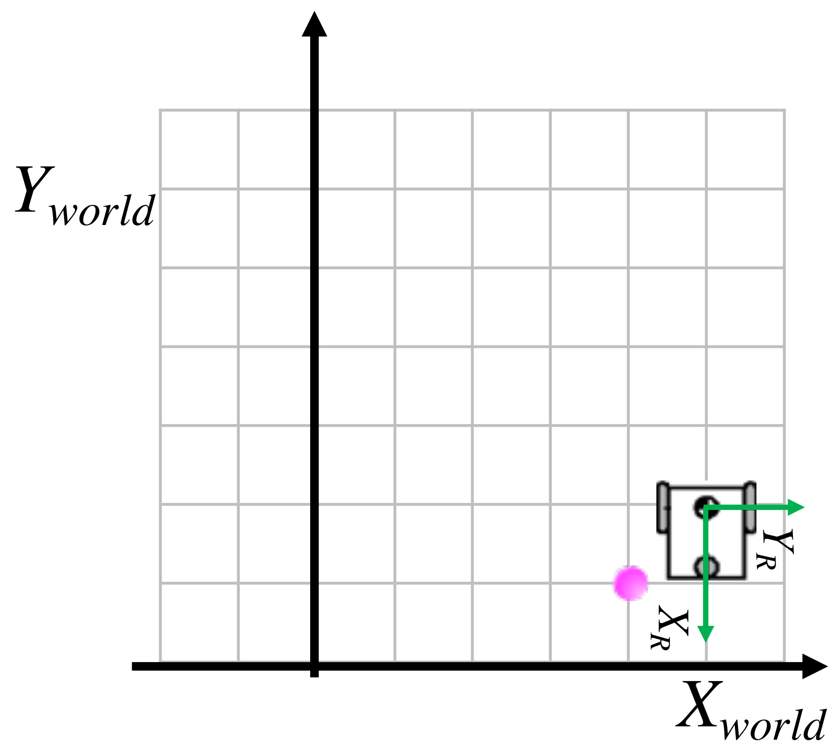


Exercise 2

$$ {}^0_2T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$

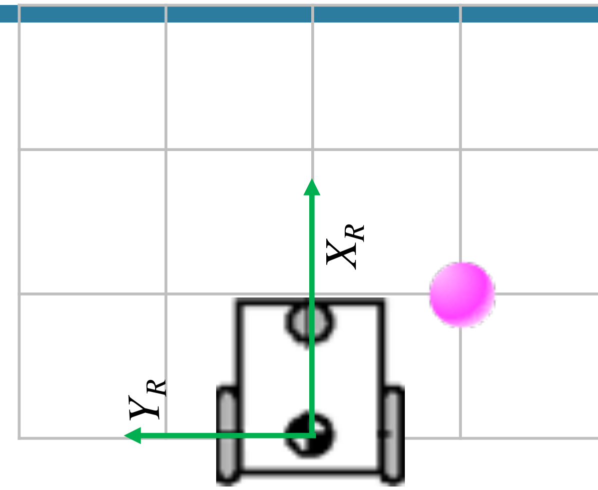
Now let's say we sense an obstacle.

1. What are its coordinates in $\boldsymbol{R}$?
   - We can transform between F_R and F_S
   - Right now let's eyeball it



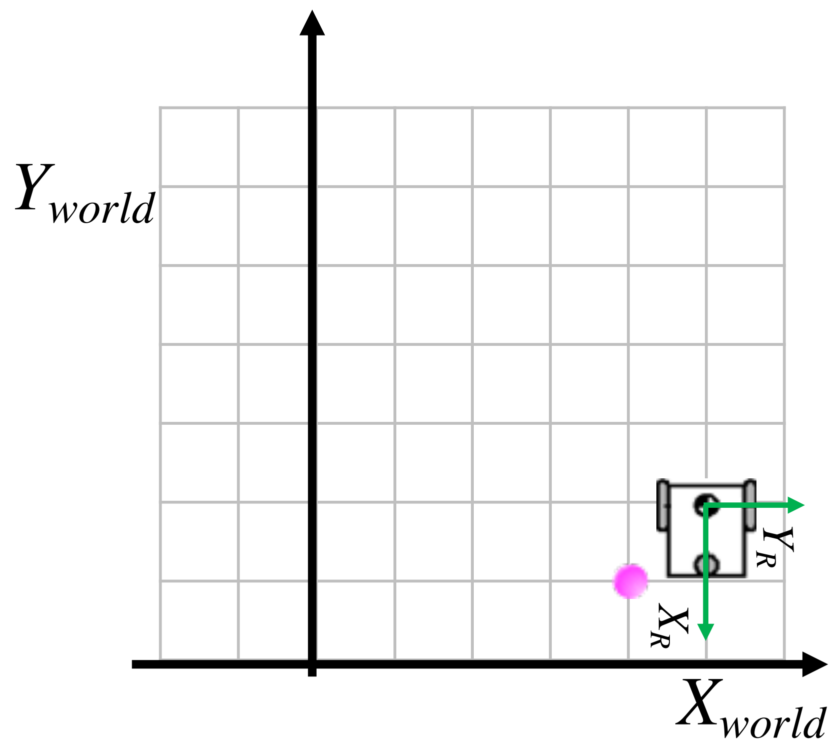
Exercise 2

$$ {}^{0}_{2}T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$



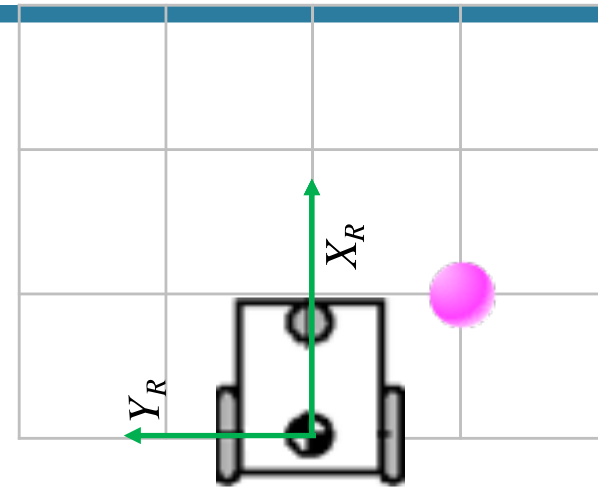
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - We can transform between F_R and F_S
   - Right now let's eyeball it
   - Looks like it's 1m ahead (+x) and 1m in -y



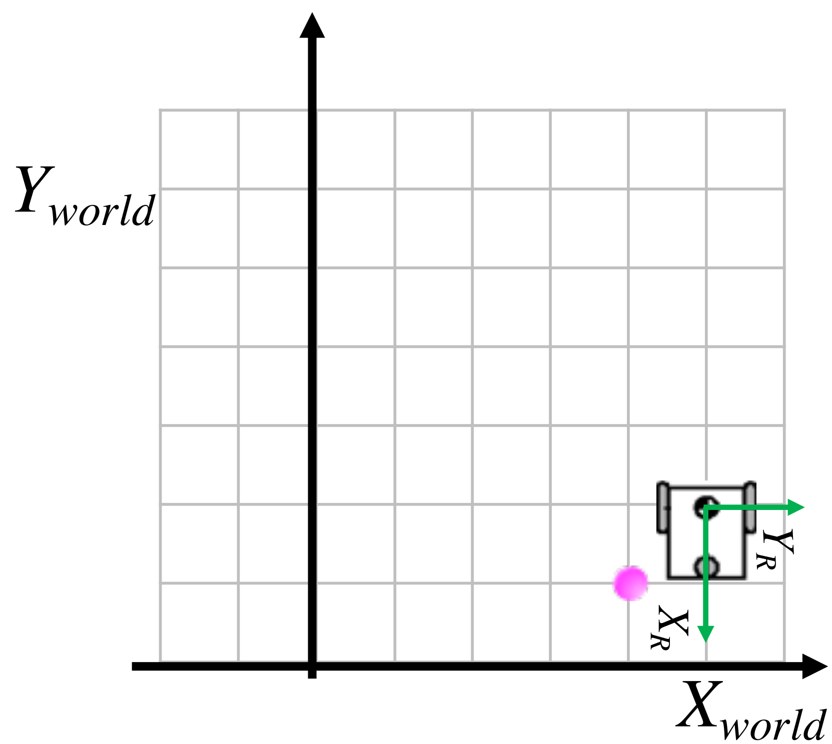
Exercise 2

$${}^0_2T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix}$$



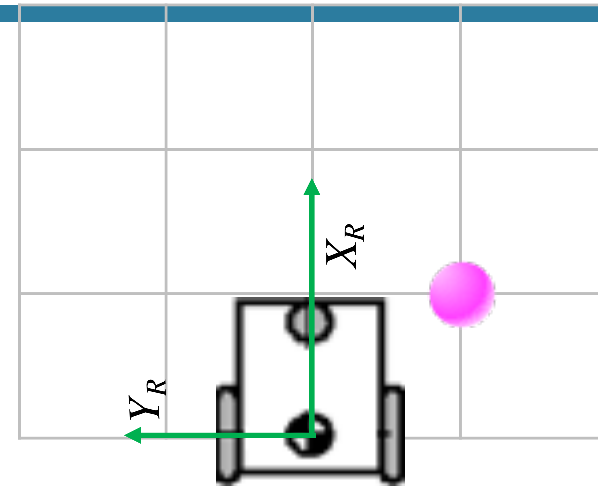
Now let's say we sense an obstacle.

1. What are its coordinates in F_R?
   - We can transform between F_R and F_S
   - Right now let's eyeball it
   - Looks like it's 1m ahead ($+x$) and 1m in $-y$
   - We can ignore rotation



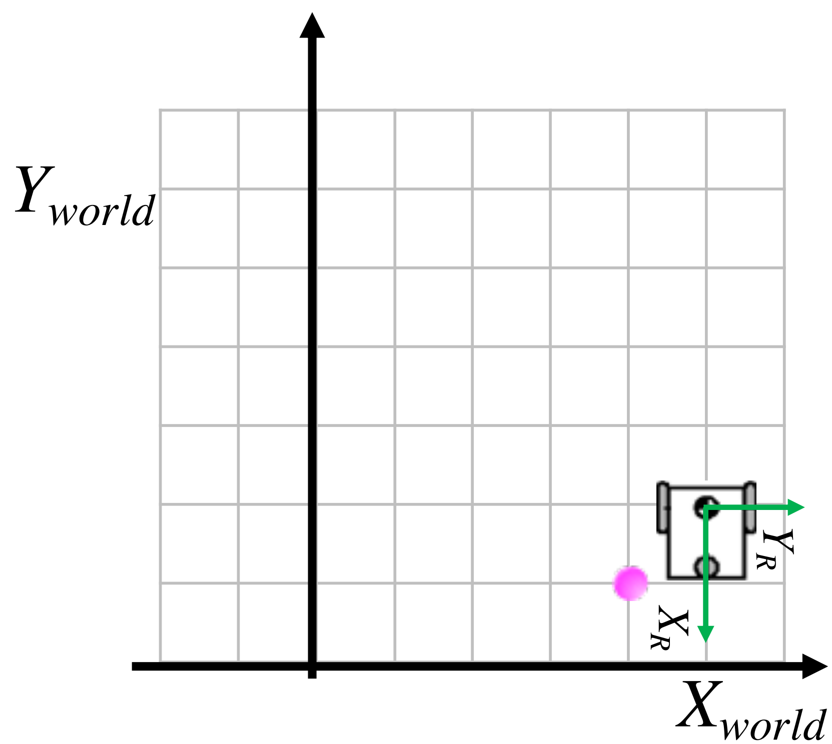
Exercise 2

$$ {}^{0}_{2}T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$



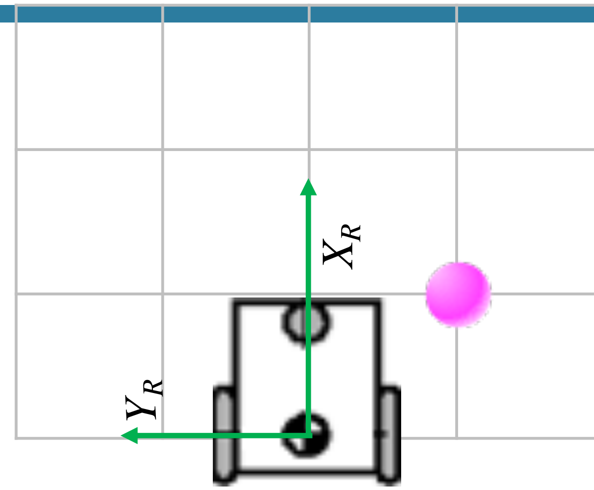
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$



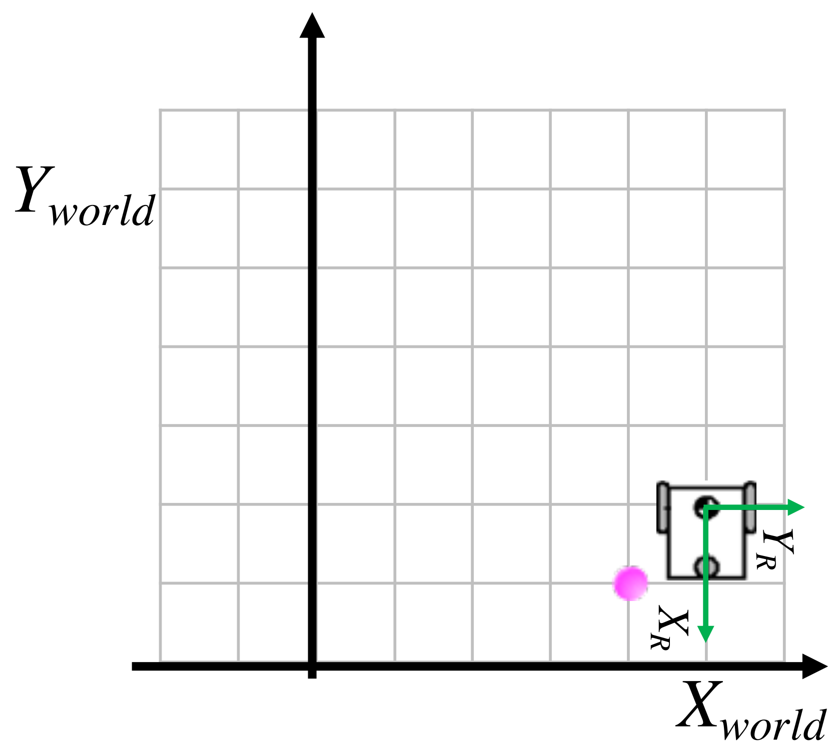
Exercise 2

$$ {}^{W}_{R}T = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} $$



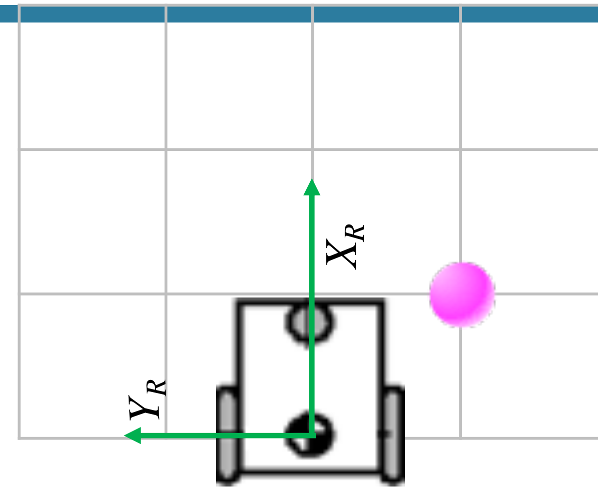
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?



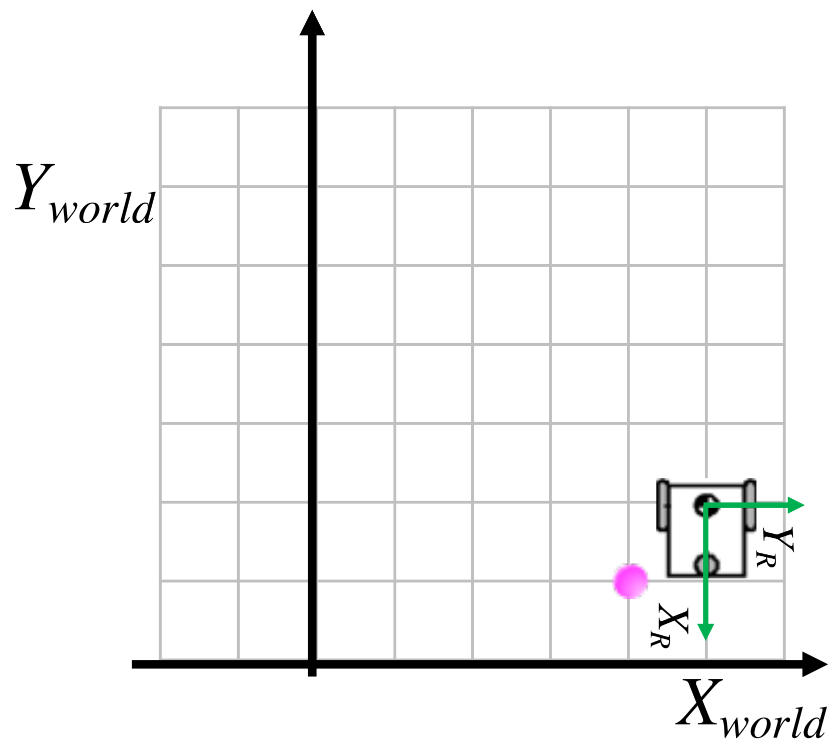
Exercise 2

$\xi_{Ob} =$



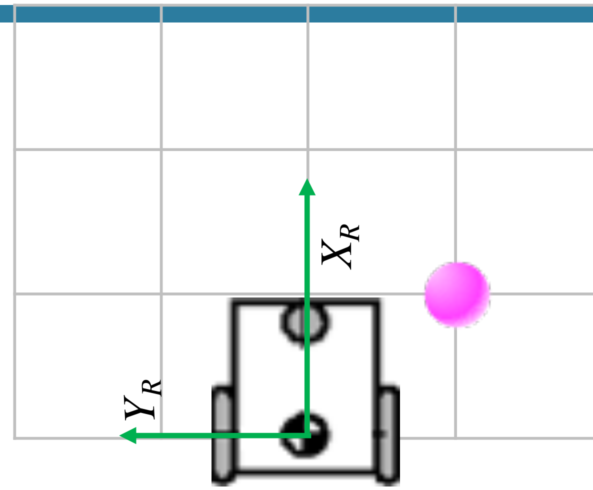
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?



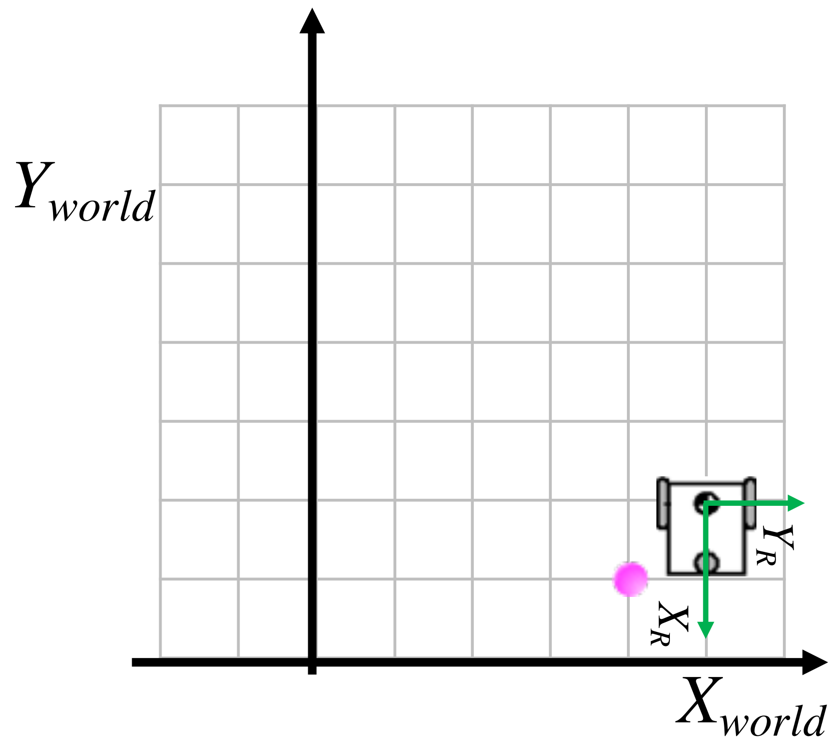
Exercise 2

$$\xi_W = {}^W_R T \times \xi_R$$



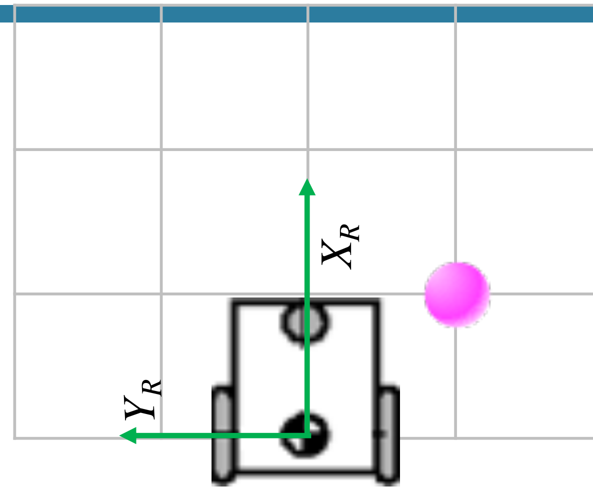
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?



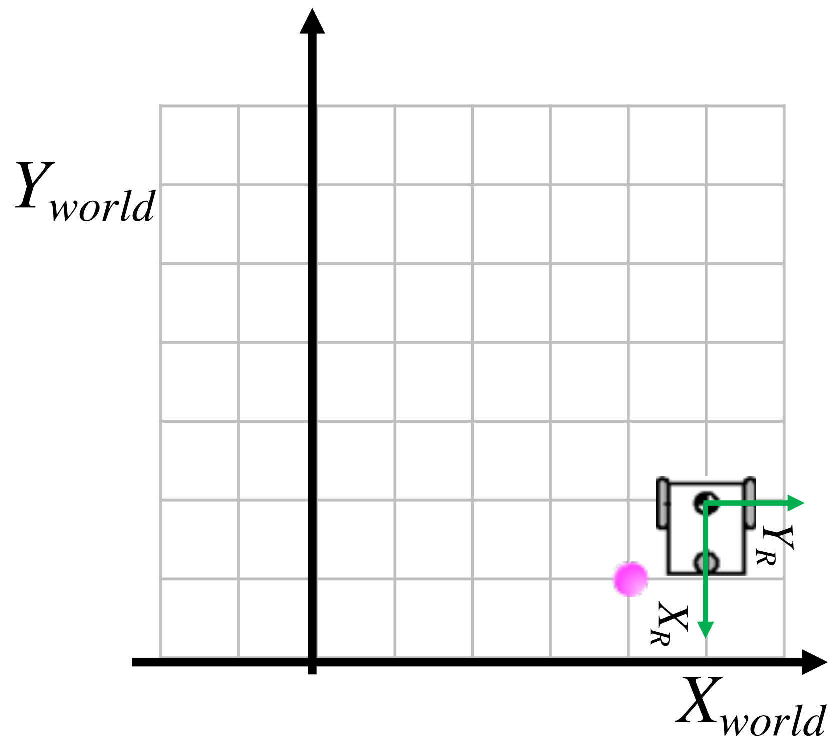
Exercise 2

$$\xi_O = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} =$$



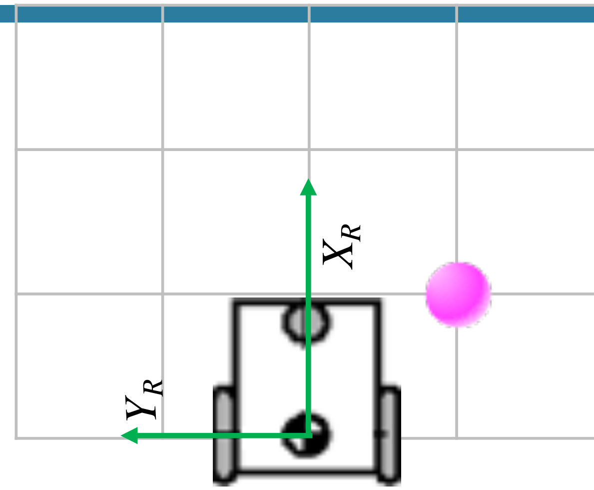
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?



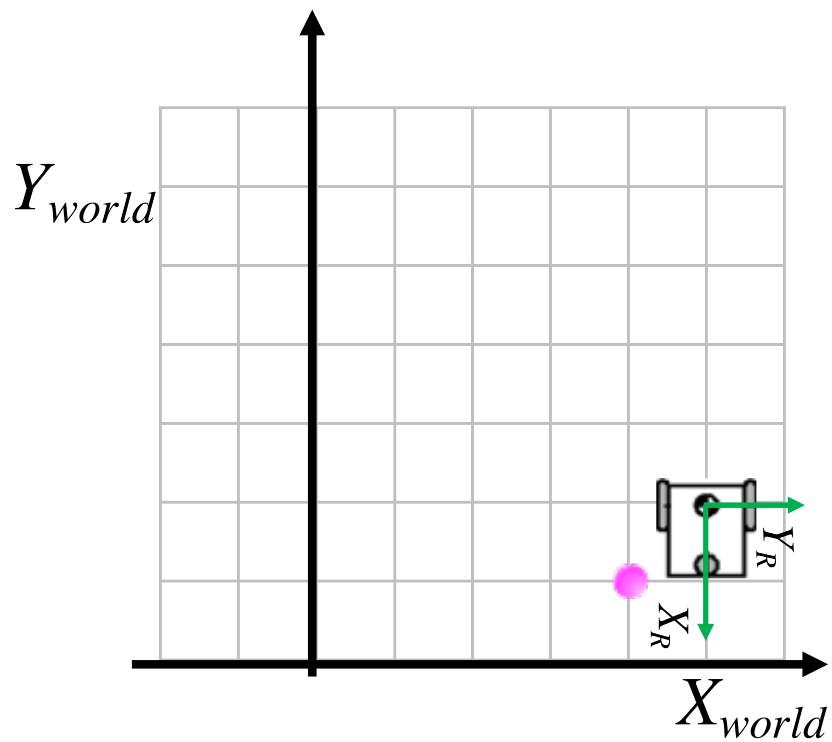
Exercise 2

$$\xi_O = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \\ 1 \end{bmatrix}$$



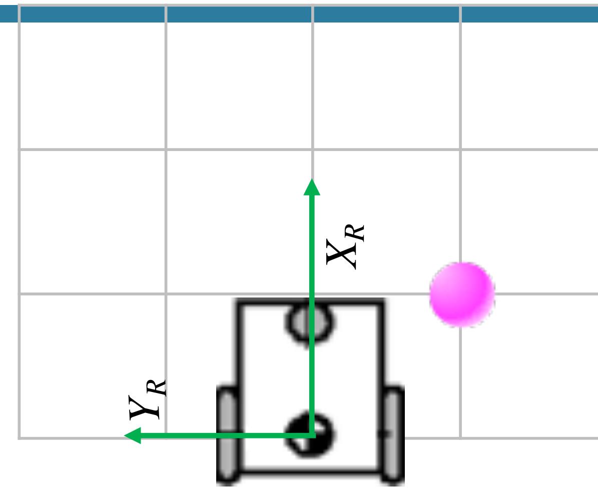
Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?
   - Which, again, looks right!



Exercise 3

$$\xi_O = \begin{bmatrix} 0 & 1 & 5 \\ -1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \\ 1 \end{bmatrix}$$



Now let's say we sense an obstacle.

1. What are its coordinates in ***R***?
   - $\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$
2. What are its coordinates in ***W***?
   - Which, again, looks right!