Practice Midterm 1

EECS/BioE C106A/206A
Introduction to Robotics

Due: October 1, 2020

Problem	**Max. Score**
Short Answers	20
Order of Operations	10
ROS Turtlesim Wrapper	15
Reference Frames	10
Forward Kinematics	8
Inverse Kinematics	12
When all else fails	15
Total	90

Problem 1. Short Answers (20 points)

(a) (5 points) Show that if $A \in \mathfrak{so}(n)$ is a skew-symmetric matrix then $R = e^A \in SO(n)$.

(b) (5 points) Let $g = (R, p) \in SE(3)$ be such that $R = R_Z(\pi)$ and $p = (0, 2, 0)$. Find a set of exponential coordinates for g.

(c) (5 points) Show that $R_X(\theta_1)R_X(\theta_2) = R_X(\theta_1 + \theta_2)$.

Hint: You may use the fact that for any square matrices A, B, if $AB = BA$ then $e^A e^B = e^{A+B}$.

(d) (5 points) Your friend from MIT asserts that for $a \in \mathbb{R}^3$, the matrix $B = (I - \hat{a})^{-1}(I + \hat{a})$ is in $SO(3)$; ie. B is a rotational matrix. True or False.

Hint 1: Don't try to brute-force this.
Hint 2: Remember the properties of skew-symmetric matrices
Hint 3: Does $(I - \hat{a})(I + \hat{a}) = (I + \hat{a})(I - \hat{a})$?

Solution:

(a) We need to show that $R^T R = I$ and that $\det R = 1$. We know that $A^T = -A$. So $R^T = e^{A^T} = e^{-A}$. Therefore, $R^T R = e^{-A} e^A = I$. Morevoer, we know that if $R^T R = I$ then $\det R = \pm 1$. Since $\det e^0 = \det I = 1$ and 0 is a skew symmetric matrix, by the continuity of the determinant, $\det A$ must be 1 for all skew symmetric matrices A.

(b) Since R is just $R_Z(\pi)$, we immediately have $\omega = (0, 0, 1)^T$ and $\theta = \pi$. Now, note that there is no translation in the Z direction at all. Therefore, the associated screw must be a pure rotation, since otherwise there would be some translation in the ω direction. So all we need to do is find a q that works. We see that q must be such that if we rotate by π radians about the point q we move the origin to the location $(0, 2, 0)$. This is achieved if we pick $q = (0, 1, 0)^T$ i.e. we need to pick q so that it is halfway between the intial and final location of the origin. The exponential coordinates are then $(\xi = (v, \omega), \theta)$ where $\omega = (0, 0, 1), v = -\omega \times q$, and $\theta = \pi$.

(c) $R_X(\theta)$ is just a rotation by θ about the vector $x = (1, 0, 0)^T$. So $R_X(\theta) = e^{\hat{x}\theta}$. Then we have

$$\begin{aligned} R_X(\theta_1)R_X(\theta_2) &= e^{\hat{x}\theta_1} e^{\hat{x}\theta_2} \\ &= e^{\hat{x}\theta_1 + \hat{x}\theta_2} \\ &= e^{\hat{x}(\theta_1 + \theta_2)} \\ &= R_X(\theta_1 + \theta_2) \end{aligned}$$

as needed.

(d) **True**. Firstly, Let's prove $(I+\hat{a})(I-\hat{a})$ commutes. This is clear by direct computation.

$$(I+\hat{a})(I-\hat{a}) = I - \hat{a}^2$$
$$(I-\hat{a})(I+\hat{a}) = I - \hat{a}^2$$

Then

$$\begin{aligned}
[(I-\hat{a})^{-1}(I+\hat{a})][(I-\hat{a})^{-1}(I+\hat{a})]^T &= (I-\hat{a})^{-1}(I+\hat{a})(I+\hat{a})^T((I-\hat{a})^{-1})^T \\
&= (I-\hat{a})^{-1}(I+\hat{a})(I-\hat{a})(I+\hat{a})^{-1} \\
&= (I-\hat{a})^{-1}(I-\hat{a})(I+\hat{a})(I+\hat{a})^{-1} \\
&= I
\end{aligned}$$

In order to prove it's a rotational matrix, the rest we should do is prove the determinant is 1, where

$$\det(B) = \det((I-\hat{a})^{-1})\det(I+\hat{a}) = \frac{\det(I+\hat{a})}{\det(I-\hat{a})} = \frac{\det(I+\hat{a})}{\det(I+\hat{a})} = 1$$

where we use the property that $\det(K) = \det(K^T)$ for a given square matrix K

Remark: This problem is a mainly about (link) Cayley's representation. The Cayley's transformation for rotations is often invertible, which means that it can be useful for doing control on rotations.

Problem 2. Order of Operations (10 points)

(a) (3 points) Select all operations that are always commutative:

- ☐ Multiple rotation matrices, about orthogonal axes
- ☐ Multiple rotation matrices, about parallel axes
- ☐ Multiple homogenous transforms, where all $R = I$
- ☐ Multiple homogenous transforms, where all $R = R_X(\frac{\pi}{4})$
- ☐ Multiple exponential mappings, with parallel revolute joints.
- ☐ Multiple exponential mappings, with parallel prismatic joints.

(b) (3 points) Select all options that are always associative:

- ☐ Multiple rotation matrices, about orthogonal axes
- ☐ Multiple rotation matrices, about parallel axes
- ☐ Multiple homogenous transforms, where all $R = I$
- ☐ Multiple homogenous transforms, where all $R = R_X(\frac{\pi}{4})$
- ☐ Multiple exponential mappings, with parallel revolute joints.
- ☐ Multiple exponential mappings, with parallel prismatic joints.

(c) (4 points) Answer the following questions.

(i) You are given the rotation matrices: R_{AB}, R_{CB}. Write an expression for R_{CA}.

(ii) You are given the rotation matrices: R_{AB}, R_{CA}. Write an expression for R_{BC}.

(iii) You are given the rotation matrices: R_{AB}, R_{BC}. Write an expression for R_{AA}.

(iv) You are given the rotation matrices: R_{AB}^{-1}, R_{BC}^{T}. Write an expression for R_{AC}.

Solution:

(a) The options that are commutative are: 2, 3, 6.

(b) All options are associative.

(c) (i) $R_{CA} = R_{CB}R_{AB}^T$

(ii) $R_{BC} = R_{AB}^T R_{CA}^T$

(iii) $R_{AA} = R_{AB}R_{AB}^T = I$

(iv) $R_{AC} = \left(R_{AB}^{-1}\right)^T \left(R_{BC}^T\right)^T$

Problem 3. Turtle Wrapper Node (15 points)

In Lab 1, we asked you to write a publisher node that would send a `geometry_msgs/Twist` over the `/turtle1/cmd_vel` topic in order to control your simulated turtle. You may recall only having to set two values of the twist: the linear x velocity, and the angular z velocity. This made sense at the time because our robot was entirely simulated in a 2D environment, reflecting the fact that it was a *unicycle modeled* robot; at any time, we may model the turtle's velocity relative to its own reference frame as

$$\vec{V} = \begin{bmatrix} v \\ \omega \end{bmatrix}.$$

where v is the linear x velocity and ω is the angular velocity. For a unicycle modeled robot, we always assume that the linear y velocity is 0. By controlling the turtle through directly manipulating a 6D Twist, you were breaking the abstraction between the perceived model of the robot and the commands the simulator needed to receive in order to control the turtle! To remedy this, you are now tasked with writing a "wrapper" node: you will construct a node that will listen for linear velocity and angular velocity commands published over a topic of your choice, and will publish that information to `/<turtle_name>/cmd_vel`, where `<turtle_name>` denotes the name of the turtle you want to control. Assume that your wrapper node has access to the desired turtle name through a command line argument. Assume this node will be written as a `.py` file placed in the appropriate location of a package named `midterm_1`.

(a) (3 points) What is the name of your node, what topic(s) do you want it to subscribe to, and what topic(s) do you want it to publish to? Remember that you want to be able to run multiple instances of your node if someone wants to use your wrapper node for multiple turtles.

(b) (2 points) You will be designing a new message type for the topic you choose to subscribe to. Define your `.msg` file. Make sure to indicate the name of the file somewhere.

(c) (5 points) Assume someone wants to control the turtle named "jturtle". A node named `user_control` is running that will send data of the appropriate message type to the topic your wrapper node subscribes to. Assuming that your wrapper node, turtlesim, and a `rostopic echo` node listening to the output of `user_control` for debugging purposes are all running. Draw the an approximate RQT graph that fits this scenario.

(d) (5 points) Time to code up your node! Fill in the appropriate blanks:

```
#!/bin/env python

import rospy
import sys
from geometry_msgs.msg import Twist
from __________ import __________

class TurtleWrapper:
```

```python
    def __init__(self, turtle_name):
        rospy.Subscriber(__________, __________, receive_command)
        self.pub = rospy.Publisher(__________, __________, queue_size=10)

    def receive_command(self, cmd_vel_2D):
        cmd_vel = Twist()
        cmd_vel.linear.x = __________
        cmd_vel.angular.z = __________
        self.pub.publish(cmd_vel)

if __name__ == '__main__':
    rospy.init_node(_________, anonymous=True)
    turtle_name = sys.argv[1]
    wrapper = TurtleWrapper(turtle_name)
    rospy.spin()
```

Solution:

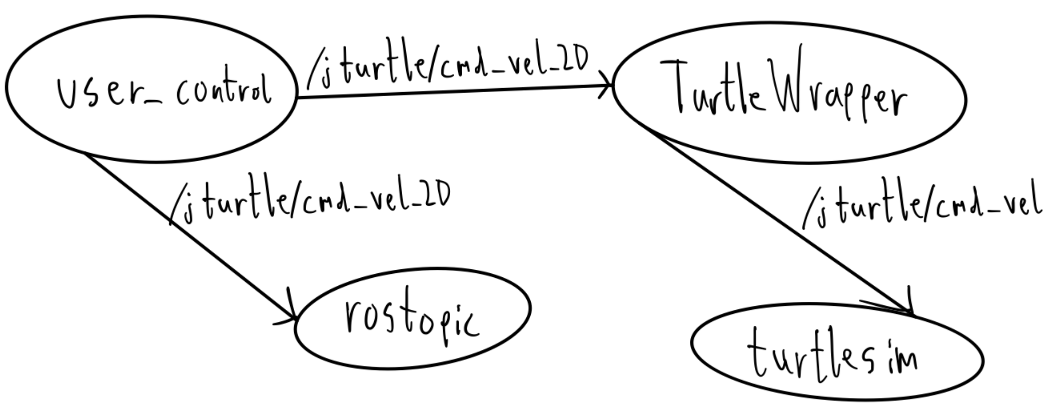
Answers will vary depending on how things are named. This is one correct solution.

(a)
- I will name my node `TurtleWrapper`.
- It will subscribe to /<turtle_name>/cmd_vel_2D.
- It will publish to /<turtle_name>/cmd_vel.

(b) My message file is `TurtleVelocity.msg`. Here is the message

```
float32 v
float32 w
```

(c) Here is how the RQT graph could look



(d) Here is how my code looks

```python
#!/bin/env python

import rospy
import sys
```

```python
from geometry_msgs.msg import Twist
from midterm_1.msg import TurtleVelocity

class TurtleWrapper:

    def __init__(self, turtle_name):
        rospy.Subscriber('/{}/cmd_vel_2D'.format(turtle_name),
        TurtleVelocity, receive_command)
        self.pub = rospy.Publisher('/{}/cmd_vel'.format(turtle_name),
        Twist, queue_size=10)

    def receive_command(self, cmd_vel_2D):
        cmd_vel = Twist()
        cmd_vel.linear.x = cmd_vel_2D.v
        cmd_vel.angular.z = cmd_vel_2D.w
        self.pub.publish(cmd_vel)

if __name__ == '__main__':
    rospy.init_node('TurtleWrapper', anonymous=True)
    turtle_name = sys.argv[1]
    wrapper = TurtleWrapper(turtle_name)
    rospy.spin()
```

Problem 4. Reference Frames (10 points)

Figure 1 shows four reference frames in the workspace of a robot: the fixed frame $\{a\}$, the end-effector frame $\{b\}$, the camera frame $\{c\}$, and the workpiece frame $\{d\}$.

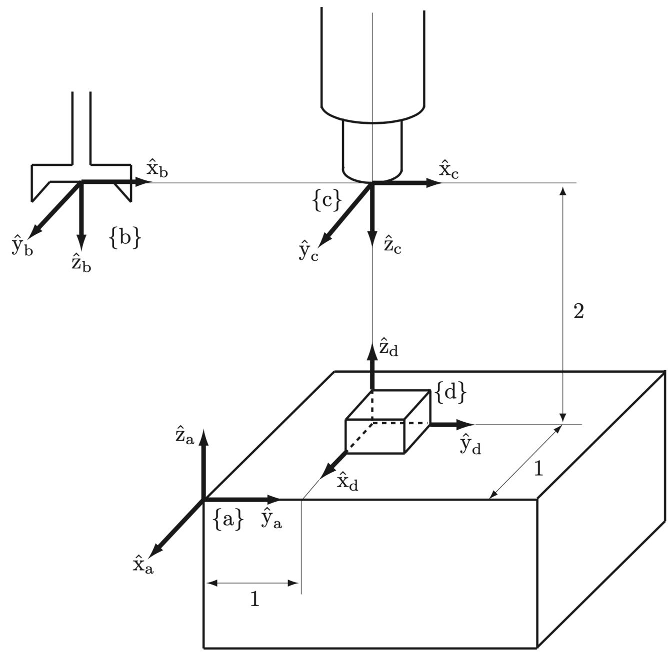


Figure 1: Four reference frames defined in a robot's workspace.

(a) (6 points) Find the $SE(3)$ poses g_{ad} and g_{cd} in terms of the dimensions given in the figure.

(b) (4 points) Find g_{ab} given that

$$g_{bc} = \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Solution:

(a)

$$g_{ad} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, g_{cd} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(b) We can compute $g_{ab} = g_{ad}g_{dc}g_{cb} = g_{ad}g_{cd}^{-1}g_{bc}^{-1}$.

Problem 5. Forward Kinematics (8 points)

Solve for the forward kinematics map of the 4DOF manipulator shown in the appendix in its initial configuration. The robot has three revolute and one prismatic joint. Do this by finding:

(a) (2 points) The initial configuration $g_{WT}(0) \in SE(3)$ of the robot.

(b) (4 points) The twists $\xi_1, \xi_2, \xi_3, \xi_4$ corresponding to each joint of the robot.

(c) (2 points) An expression for the forward kinematics map $g_{WT}(\theta)$ in terms of the vector of joint angles $\theta = (\theta_1, \theta_2, \theta_3, \theta_4)$. You may leave your answer in terms of the exponentials and products of known matrices.

Solution:

(a) By direct inspection, we an write

$$g_{WT}(0) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_1 + l_2 \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(b) For each joint, we have

$$\begin{aligned}
&q_1 = [0,0,0]^T, \omega_1 = [0,0,1]^T, v_1 = [0,0,0]^T, \xi_1 = [0,0,0,0,0,1]^T \\
&q_2 = [0,0,l_0]^T, \omega_2 = [-1,0,0]^T, v_2 = [0,-l_0,0]^T, \xi_2 = [0,-l_0,0,-1,0,0]^T \\
&q_3 = [0,0,l_0]^T, \omega_3 = [0,1,0]^T, v_3 = [-l_0,0,0]^T, \xi_3 = [-l_0,0,0,0,1,0]^T \\
&v_4 = [0,1,0]^T, \xi_4 = [0,1,0,0,0,0]^T
\end{aligned}$$

(c)

$$g_{WT}(\theta) = e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} e^{\hat{\xi}_4\theta_4} g_{WT}(0)$$

Problem 6. Inverse Kinematics (12 points)

Consider the 4DOF manipulator shown in the appendix. Assume that $0 \leq \theta_4 \leq d_{max}$.

(a) (3 points) Describe the reachable workspace of the manipulator, that is the subset of $\mathbb{R}^3$ that the origin of the tool frame can reach. Ignore any self-collisions.

(b) (7 points) Use the Paden Kahan sub-problems to solve the inverse kinematics of this manipulator. You do not need to do the details of the inverse kinematics, but indicate how you would break down the inverse kinematics to get the angles.

(c) (2 points) Indicate the number of possible inverse kinematics solutions.

Solution:

(a) The reachable workspace is a spherical annulus with inner radius $l_1 + l_2$ and outer radius $l_1 + l_2 + d_{max}$, centered at q_s.

(b) Let the desired configuration be $g_d \in SE(3)$. The IK problem can be formulated as

$$e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} e^{\hat{\xi}_4\theta_4} = g_d g_{WT}^{-1}(0) := g$$

where we define g to be the known matrix $g_d g_{WT}^{-1}(0)$ for convenience.

Step 1: Solve for θ_4. Use the point q_s. This point is on the axes of ξ_1, ξ_2 and ξ_3, so we can use it to factor those out. We will additionally consider the point q_t at the tip of the end effector. We get

$$\begin{aligned} e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} e^{\hat{\xi}_4\theta_4} q_t - q_s &= g q_t - q_s \\ ||e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} \left(e^{\hat{\xi}_4\theta_4} q_t - q_s \right) || &= ||g q_t - q_s|| \\ ||e^{\hat{\xi}_4\theta_4} q_t - q_s|| &= ||g q_t - q_s|| \end{aligned}$$

Since θ_4 controls this distance directly, we can set $\theta_4 = ||g q_t - q_s|| - (l_1 + l_2)$.

Step 2: Solve for θ_1 and θ_2. Now that θ_4 is known, so is the matrix $g_4 = e^{\hat{\xi}_4\theta_4}$. So we can take it to the right hand side to get

$$e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} = g g_4^{-1} := g'$$

where we have defined g' to be the known matrix $g g_4^{-1}$. Now pick some point that is on the axis of ξ_3 but not on the axes of either ξ_1 or ξ_2. For instance, q_t works. Multiply both sides by this point to get

$$\begin{aligned} e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} e^{\hat{\xi}_3\theta_3} q_t &= g' q_t \\ e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} q_t &= g' q_t \end{aligned}$$

This is exactly the setup for PK subproblem 2 with the intersecting axes ξ_1 and ξ_2 and point q_t being taken to point $g' q_t$. So we can use PK subproblem 2 to find θ_1 and θ_2. There two possible solutions in this step.

Step 3: Solve for θ_3. Now the matrices $g_1 = e^{\hat{\xi}_1 \theta_1}$ and $g_2 = e^{\hat{\xi}_2 \theta_2}$ are also known. So we can rearrange to get

$$e^{\hat{\xi}_3 \theta_3} = g_2^{-1} g_1^{-1} g' := g''$$

now we can take any point p not on the axis of ξ_3. Then we get

$$e^{\hat{\xi}_3 \theta_3} p = g'' p$$

which is the setup for PK subproblem 1 with axis ξ_3 and point p being taken to point $g'' p$. So we can use PK1 to solve for θ_3. There is one possible solution in this step.

(c) $1 \times 2 \times 1 = 2$.

Problem 7. When all else fails (15 points)

Let $u \in \mathbb{R}^3$ be a unit vector, and let $R = I + 2\hat{u}^2$.

(a) (3 points) Show that $R^T R = I$.

Hint: Recall equation (2.13) from the textbook: $\hat{u}^3 = -\hat{u}$.

(b) (3 points) Show that $\det R = 1$, and hence conclude that R is a rotation matrix.

Hint: The function $u \mapsto \det(I + 2\hat{u}^2)$ *is continuous, thanks to the continuity of the determinant. However, from part (a) it follows that* $\det(I + 2\hat{u}^2) \in \{+1, -1\}$. *Is it possible for a continuous function to take on exactly two discrete values? Can you conclude from this that* $\det(I + 2\hat{u}^2)$ *is actually a constant, for any* u*?*

(c) (5 points) Find the exponential coordinates for R i.e. find a unit vector ω and a scalar $\theta \in [0, 2\pi)$ such that $R = e^{\hat{\omega}\theta}$.

Hint: What does R *look like when we switch to a new reference frame where* u *is the* x *axis?*

Hint: Recall that the determinant of a transformation is invariant under a change of basis.

(d) (2 points) Verify that when Rodrigues' formula is applied to your answer in the previous part, you get the original matrix R.

(e) (2 point) What would go wrong if you tried to use Proposition 2.9 from the textbook to compute the exponential coordinates of a rotation matrix of this form?

Solution:

(a)

$$
\begin{aligned}
R^T R &= (I + 2\hat{u}^2)^T (I + 2\hat{u}^2) && (1)\\
&= (I + 2\hat{u}^2)^2 && (2)\\
&= I + 4\hat{u}^2 + 4\hat{u}^4 && (3)\\
&= I + 4\hat{u}^2 - 4\hat{u}^2 && (4)\\
&= I && (5)
\end{aligned}
$$

(b) To show that $\det R = 1$, we can use a very similar technique as we did when showing that the exponential of a skew symmetric matrix is a rotation matrix. In particular, we know that $\det(R)$ is ± 1, but we also know that $\det(I + 2\hat{u}^2)$ is continuous in u. So, it cannot jump from 1 to -1 or vice versa as we let u vary. This means that it is either always 1 or always -1. In particular, at $u = 0$, $\det(R) = \det(I) = 1$ and so $\det R = 1$ for all u, and hence R is a rotation matrix.

(c) Lets pick any right handed orthonormal coordinate frame with x-axis u. In this frame, $u = (1, 0, 0)^T$. Then, in this new frame $R = I + 2\hat{u}^2$ has the simple form

$$R' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

By looking at the form of R' in this new reference frame we can identify that this is exactly the rotation matrix corresponding to a rotation of π radians about the x-axis. Since the x-axis in this frame is just u in the original frame, R is a rotation about u by π radians, and so the exponential coordinates are (u, π).

(d) Rodrigues formula states that for unit ω,

$$R(\omega, \theta) = I + \hat{\omega}\sin(\theta) + (1 - \cos(\theta))\hat{\omega}^2 \tag{6}$$

substituting $\omega = u$ and $\theta = \pi$ gives us

$$R(u, \pi) = I + 2\hat{u}^2 \tag{7}$$

(e) Proposition 2.9 requires a division by $\sin(\theta)$ which is zero for $\theta = \pi$. So that construction breaks down when R is in this form.

Appendix

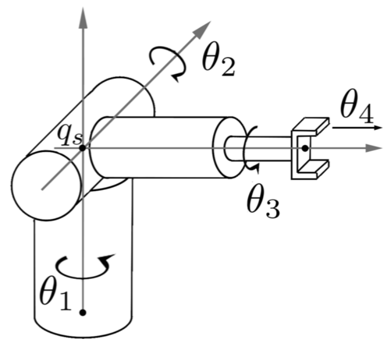


Figure 2: 4DOF Arm. Joints 1,2,3 are revolute. Joint 4 is prismatic

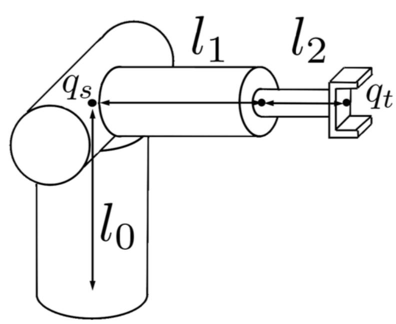


Figure 3: Manipulator lengths in zero configuration

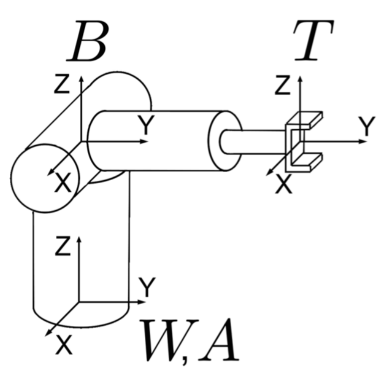


Figure 4: Coordinate Axes at zero configuration