CSCI 6650:
Intelligent Agents and Multi-Agent Systems

Paper 5:

Basics of Motion Planning: A Student Guide

Ted Holmberg

March 2023

Abstract

This paper introduces fundamental concepts for motion planning orientation and trajectory. It guides the reader from path planning to motion planning, beginning with C-space and its applications. The use of special orthogonal groups, including $SO(3)$, is explored to compare orientations and differentiate motion planning from path planning. Homeomorphism is introduced to explain the mapping between geometric space and C-space. Finally, a practical example of motion planning is provided.

Introduction

Motion planning involves determining a sequence of configurations to move a robot from its initial to its final configuration while avoiding obstacles. While path planning is a fundamental component of motion planning, additional considerations, such as the orientation of the robot, must also be taken into account to ensure a successful motion plan. In this paper, we aim to guide and motivate readers to understand the additional complexities of motion planning beyond path planning.

Starting with the concept of C-space and its applications, we step through the use of special orthogonal groups to define orientations that separate motion planning from path planning. We delve into $SO(3)$ and quaternions, which are essential for comparing orientations. Moreover, we introduce the concept of homeomorphism, which is vital in mapping between geometric space and C-space.

In summary, this paper will provide a comprehensive overview of the fundamental concepts necessary for understanding motion planning beyond path planning. We aim to make these concepts accessible and understandable to our target audience, which includes students and researchers interested in robotics and motion planning.

1 C-Space

Motivation Motion planning with robots involves computing a collision-free path from the robot's initial position to its goal location. However, path planning with robotics is much more complex than simple pathfinding in digital environments. Unlike digital environments, robots have rigid bodies in the physical world and physical constraints and orientations that limit their movement and make planning more difficult than path planning. These additional considerations must be mapped on top of the spatial domain. In this section, we will explain how the concept of C-space helps to address these constraints.

What is a C-Space?

General Definition In robotics, C-space (short for configuration space) is a mathematical construct that represents the space of all possible configurations of a robot. A configuration of a robot is a complete specification of its position and orientation in space, as well as the positions of any other objects that are part of the robot.

The C-space can include both translational and rotational degrees of freedom, which makes it a high-dimensional space that can be difficult to visualize or work with directly. However, it is a fundamental concept in motion planning, as it allows us to reason about the connectivity of the space and the existence of paths that avoid collisions. By defining the C-space, we can plan paths and motions for robots that consider physical constraints and other environmental obstacles.

Mathematical Definition The mathematical definition of a C-space may be expressed as:

$$C = \mathbb{R}^n \times SO(n)$$

where n is the number of degrees of freedom of the rigid body B, and $SO(n)$ denotes the special orthogonal group of $n \times n$ rotation matrices, and C is the configuration space of B.

The dimension of C is given by:

$$m = n + \frac{n(n-1)}{2} = \frac{n(n+1)}{2}$$

which is equal to 3 when B is a 2-dimensional body and 6 when B is a 3-dimensional body.

A parametrization of C based on both positional and rotational coordinates, known as hybrid coordinates, can be used to represent C as a Euclidean space $\mathbb{R}^m$. However, since orientation matrices have rotationally cyclical properties, the representation of C using hybrid coordinates requires periodicity considerations.

What is the C-Space for a car?

Explanation A car's configuration space (C-space) must mathematically represent all possible configurations, including its position and orientation. The C-space for a car includes both translational and rotational degrees of freedom, making it a three-dimensional space. Specifically, the C-space for a car moving on a 2D plane can be represented by three dimensions: x, y, and theta.

$$C = (x, y, \theta) \in \mathbb{R}^3 : x \in \mathbb{R}, y \in \mathbb{R}, \theta \in [0, 2\pi)$$

The x and y dimensions represent the position of the car's center of mass in the 2D plane, while the theta dimension represents the car's orientation or heading. The value of theta can range from 0 to 2π radians, representing all possible angles of rotation around the z-axis. Figure 1 shows the configuration space of a car.

2 Special Orthogonal Groups

Conceptual The Special Orthogonal group, denoted by $SO(n)$, is the group of all $n \times n$ real matrices A such that:

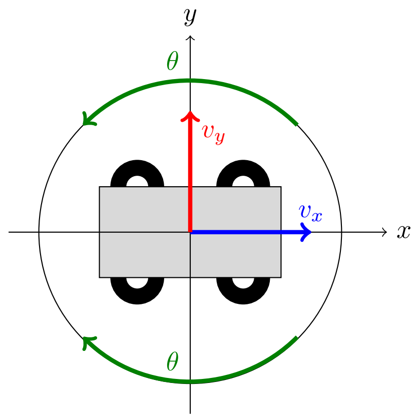


Figure 1: Illustration of a car's configuration space: $\mathbb{R}^3 : (\mathbb{R}^2 \times \mathbb{S}^1)$

$$A^T A = I \quad \text{and} \quad \det(A) = 1 \tag{1}$$

It is the set of all orthogonal matrices with determinant 1. An orthogonal matrix is a matrix whose transpose is equal to its inverse, and it represents a linear transformation that preserves length and angles. The determinant of an orthogonal matrix is either 1 or -1, and the special orthogonal group consists of the subset of matrices with determinant 1.

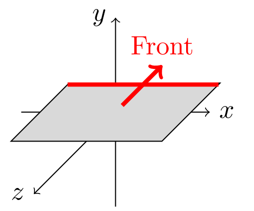
What is $SO(3)$?

Definition $SO(3)$ is a mathematical group that consists of all 3×3 rotation matrices in three-dimensional Euclidean space. These matrices preserve both the length and orientations of vectors, which means that they describe rotations without any stretching or skewing. Because of this, $SO(3)$ is often referred to as the 3D rotation group. Note that each rotational realization may have multiple orientations obtained by the combination of planar rotation and reflection operations, as seen in the figure below.

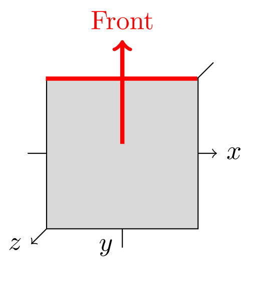
General Form An axis of rotation and an angle of rotation about this axis can specify the general form of a rotation in 3-dimensional space. Each nontrivial proper rotation fixes a unique 1-dimensional linear subspace of $\mathbb{R}^3$, which is called the axis of rotation. Each such rotation acts as a 2-dimensional rotation in the plane orthogonal to this axis. An arbitrary 3-dimensional rotation can thus be represented by an axis of rotation together with an angle of rotation about this axis. The matrix can represent counterclockwise rotation about the positive x-axis, y-axis or z-axis by an angle ϕ:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \qquad R_y(\phi) = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \qquad R_z(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

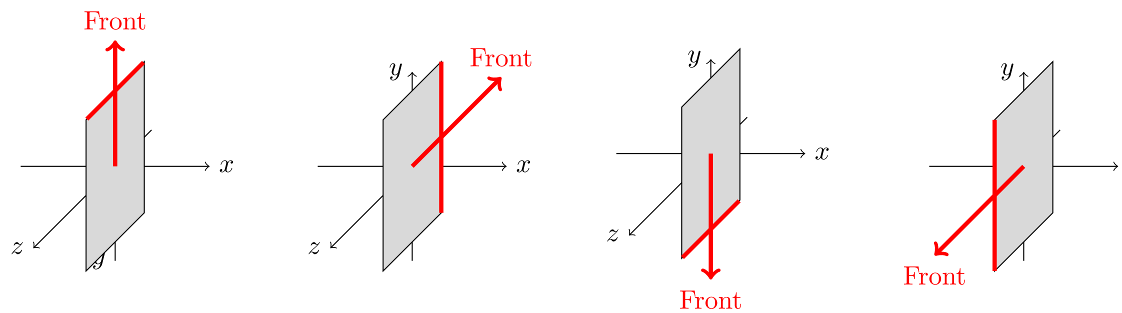
Using some properties of rotations, any rotation can be represented by a unique angle ϕ in the range $0 \le \phi \le \pi$ and a unit vector $\mathbf{n}$ such that $\mathbf{n}$ is arbitrary if $\phi = 0$, $\mathbf{n}$ is unique if $0 < \phi < \pi$, and $\mathbf{n}$ is unique up to a sign if $\phi = \pi$. Figure 3 provides a visual representation of the minimal and maximal bounds on rotational values. The horizontal line spans from 0 to π, indicating that an angle within this range can



Same planar rotation on X-axis, with 4 possible orientations



Same planar rotation on Y-axis, with 4 possible orientations



Same planar rotation on Z-axis, with 4 possible orientations

uniquely represent any rotation. The arc shows the intuition that it maps back within the range when the rotation angle goes beyond this range.

Example Here is an example of a matrix in $SO(3)$ that represents a rotation of $\pi/2$ radians around the z-axis:

$$R_z\left(\frac{\pi}{2}\right) = \begin{bmatrix} \cos\frac{\pi}{2} & -\sin\frac{\pi}{2} & 0 \\ \sin\frac{\pi}{2} & \cos\frac{\pi}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

This matrix R represents a rotation of $\pi/2$ radians around the z-axis because it sends the unit vector $\mathbf{i} = (1, 0, 0)^T$ to the unit vector $\mathbf{j} = (0, 1, 0)^T$, and leaves the unit vector $\mathbf{k} = (0, 0, 1)^T$ fixed. Applying this matrix to a point in three-dimensional space will rotate the point $\pi/2$ radians counterclockwise around the z-axis.

This rotation matrix may be applied on a given three-dimensional point, e.g. $p = (1, 0, 0)$ to find its

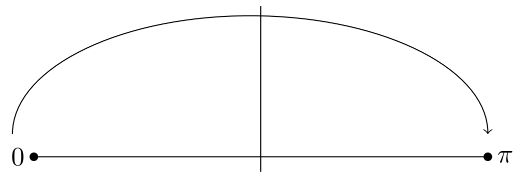


Figure 3: Representation of the minimal and maximal bounds on unique rotational values

transformed position p^*.

$$R_z\left(\frac{\pi}{2}\right) \times p = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$$

In the above example, the transformed point is $p^* = (0, 1, 0)$

Quaternions Quaternions represent 3D rotations using 4 real values to define a hypercomplex number. Unlike complex numbers that have only one imaginary component (e.g. $a + bi$), a hypercomplex number has more than one imaginary component (e.g. $a + bi + cj + dk$). A quaternion can be written in the form $q = a + bi + cj + dk$, where a, b, c, d are real numbers and i, j, k are imaginary units that satisfy the rules $i^2 = j^2 = k^2 = ijk = -1$. The quaternion 4×4 multiplication table:

	1	i	j	k
1	1	i	j	k
i	i	-1	k	$-j$
j	j	$-k$	-1	i
k	k	j	$-i$	-1

Table 1: Quaternion Multiplication Table

Based on Table 1, the Quaternion formula can be expressed with 4×4 matrices by isolating each of the terms: $1, i, j, k$.

$$\begin{bmatrix} a & b & c & d \\ b & -a & d & -c \\ c & -d & -a & b \\ d & c & -b & -a \end{bmatrix} = a \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} + b \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix} + c \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} + d \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Quaternion Intuition Informally, for $SO(3)$ in $\mathbb{R}^3$, i, j, and k are assumed to be $i = (1, 0, 0)$, $j = (0, 1, 0)$, and $k = (0, 0, 1)$, the identity vectors that isolate the x-axis, y-axis, and z-axis in geometrical space.

In three-dimensional Cartesian coordinates, a shape's orientation can be described as a set of three linear equations. These equations can be combined to form a 3×3 matrix representing the shape's orientation.

The 3×3 Identity matrix, denoted as I, represents the unit origin orientation in three-dimensional space. It is defined as:

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The identity matrix is used as the unit reference matrix for performing operations such as rotation and scaling. The diagonal entries are all equal to 1, and the off-diagonal entries are all equal to 0.

The orthogonal dimensions represented by the Identity matrix are individually represented by the vectors:

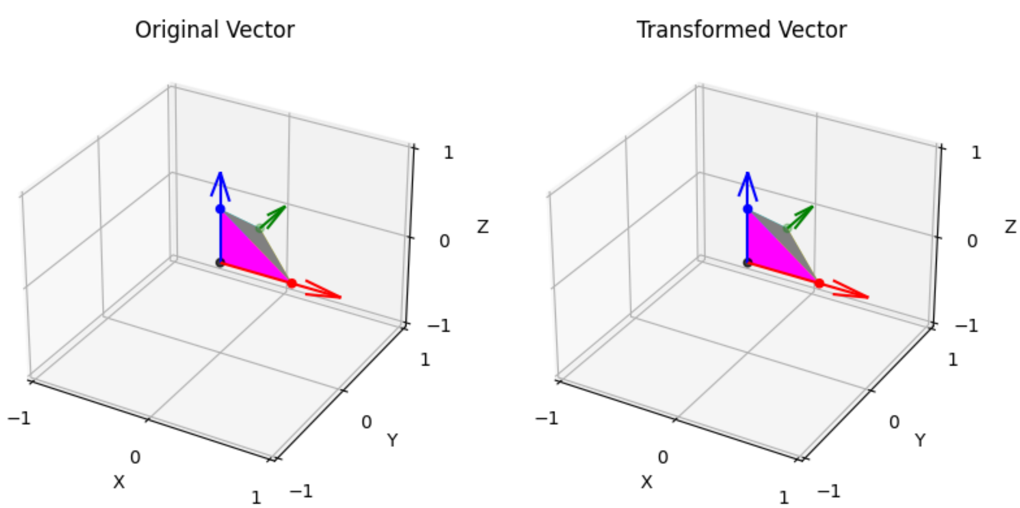
$$\vec{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \vec{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \vec{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

These vectors form the basis for the Cartesian coordinate system and can be used to describe the orientation of objects in three-dimensional space, each isolating the x-axis, y-axis, and z-axis. A hypercomplex number with three imaginary terms (i.e. i, j, k) enables the encoding of the three orthogonal rotational vectors into a single quaternion. Thus a unit quaternion, denoted by the vector $h = (a, b, c, d)$, represents a 3D rotation where the axis of rotation is parallel to the vector (b, c, d).

Quaternions & SO(3) A one-to-one correspondence exists between quaternions and rotations in three-dimensional space. This correspondence arises from the fact that every unit quaternion q can be written as $q = \cos(\theta/2) + u \sin(\theta/2)$, where θ is the angle of rotation and u is a unit vector representing the axis of rotation.

$$\text{Axis} = (x, y, z) \qquad \text{Angle} = \theta \qquad \text{Quaternion} = \begin{bmatrix} x \sin(\theta/2) \\ y \sin(\theta/2) \\ z \sin(\theta/2) \\ \cos(\theta/2) \end{bmatrix} \tag{4}$$

This one-to-one correspondence establishes a similarity between quaternions and the special orthogonal group in three dimensions, denoted by $SO(3)$.



Continued on next page

Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (1, 0, 0, 0)$, the quaternion equation $1 + 0i + 0j + 0k$ which represents an identity rotation (i.e., no rotation). The norm is computed as $1 = \sqrt{1^2 + 0^2 + 0^2 + 0^2}$, which means that the transformed vector is parallel to the original vector across all planes (i.e., planar alignment). Additionally, the orientation across all planes is the same, as all imaginary terms are zeroed out. This 3×3 rotation matrix and the 4×4 quaternion matrix is in the form of:

$$rotation_{3\times3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

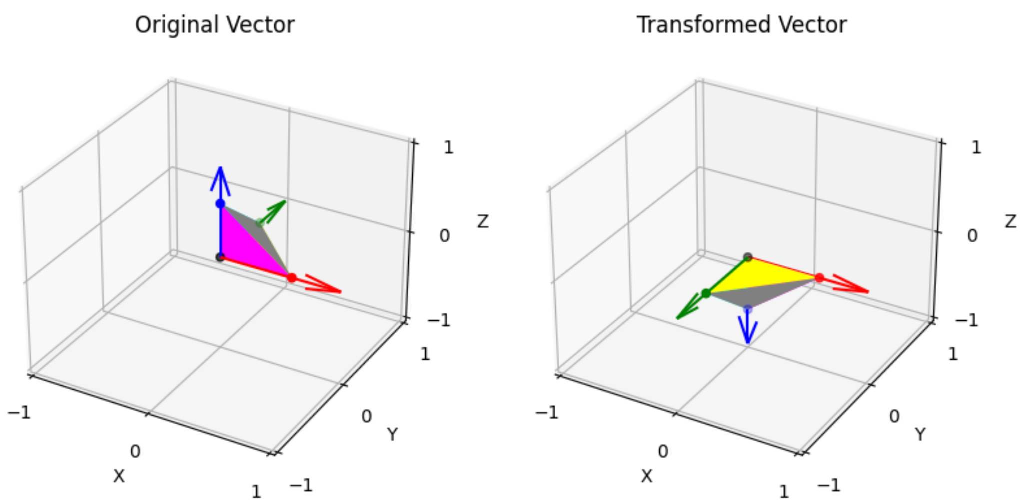


Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (0, 1, 0, 0)$, the quaternion equation $0 + 1i + 0j + 0k$ which results in a 180° rotation across x-axis. The norm is computed as $1 = \sqrt{0^2 + 1^2 + 0^2 + 0^2}$. A norm of 1 means that the transformed vector is parallel to the original vector across all planes (i.e., planar alignment). However, the orientation across the y-z plane is inverted (i.e. planar inversion). This 3×3 rotation matrix and the 4×4 quaternion matrix is in the form of:

$$rotation_{3\times3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Continued on next page

Table 2 – *Continued from previous page*

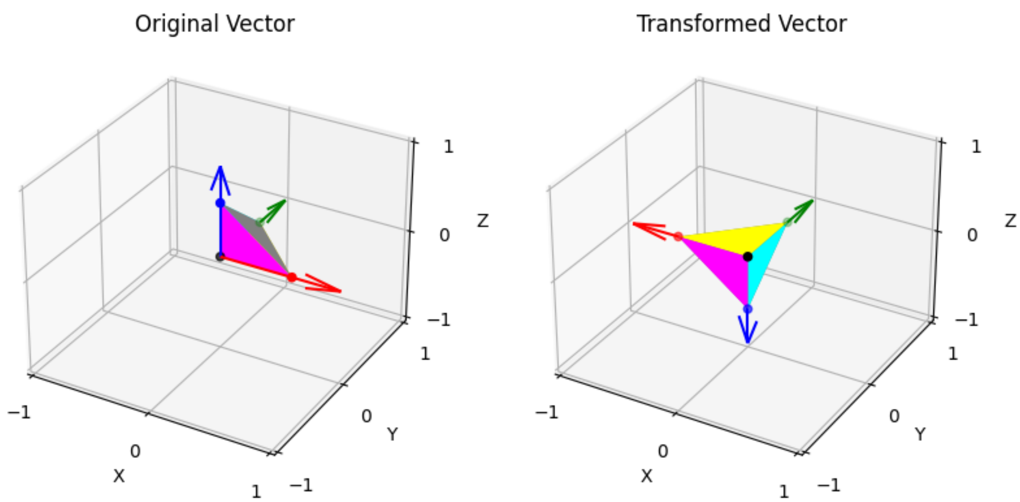
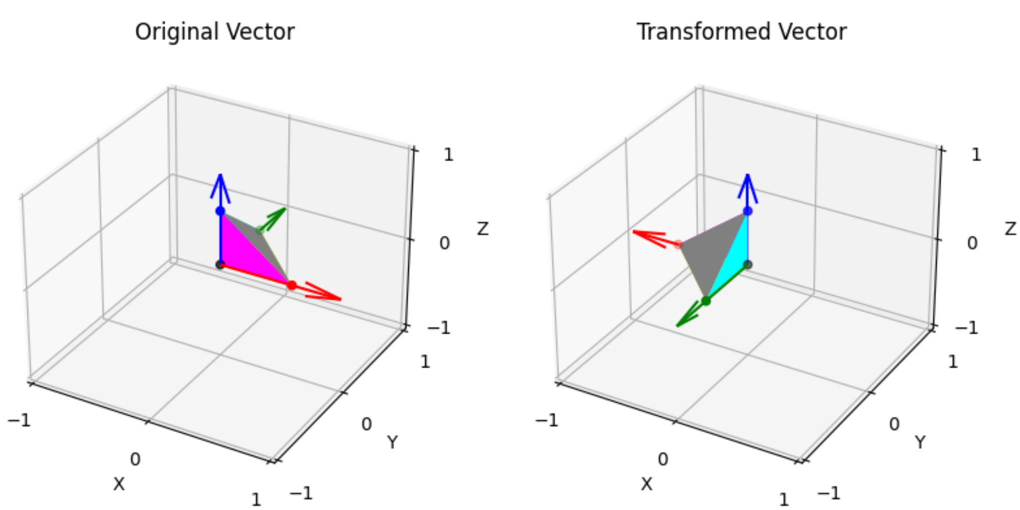


Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (0, 0, 1, 0)$, the quaternion equation $0 + 0i + 1j + 0k$ which results in a 180° rotation across x-axis. The norm is computed as $1 = \sqrt{0^2 + 0^2 + 1^2 + 0^2}$. A norm of 1 means that the transformed vector is parallel to the original vector across all planes (i.e., planar alignment). However, the orientation across the x-z plane is inverted (i.e. planar inversion). This 3×3 rotation matrix and the 4×4 quaternion matrix is in the form of:

$$rotation_{3\times3} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$



Continued on next page

Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (0, 0, 0, 1)$, the quaternion equation $0 + 0i + 0j + 1k$ which results in a 180° rotation across x-axis. The norm is computed as $1 = \sqrt{0^2 + 0^2 + 0^2 + 1^2}$. A norm of 1 means that the transformed vector is parallel to the original vector across all planes (i.e., planar alignment). However, the orientation across the x-y plane is inverted (i.e. planar inversion). This 3×3 rotation matrix and the 4×4 quaternion matrix is in the form of:

$$rotation_{3\times3} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

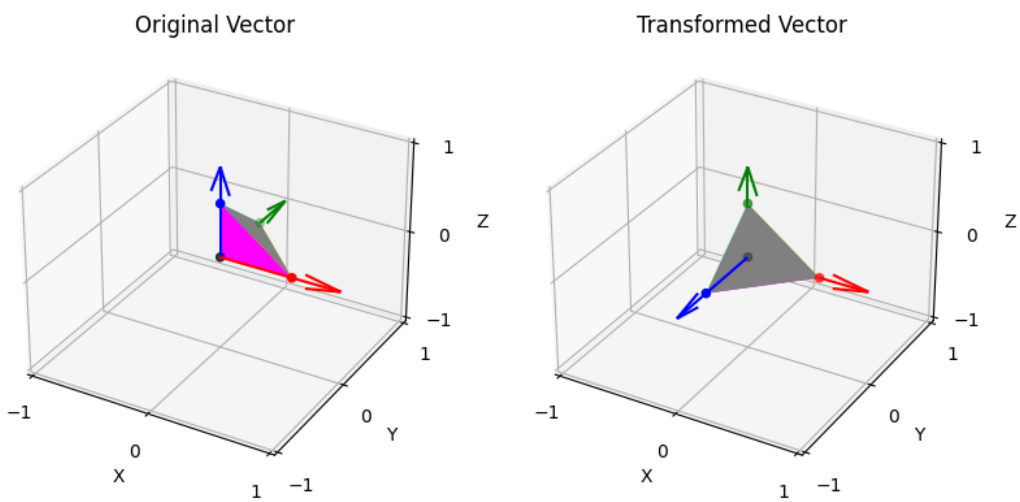


Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (1, 1, 0, 0)$, the quaternion equation $1 + 1i + 0j + 0k$ which results in a 90° rotation across x-axis. The norm is computed as $\sqrt{2} = \sqrt{1^2 + 1^2 + 0^2 + 0^2}$. A norm of $\sqrt{2}$ means that the transformed vector is the perpendicular angle to the original vector on the x-axis (i.e., an orthogonal planar rotation). Thus the x-z plane is swapped (i.e. planar reversal). This 3×3 rotation matrix and the 4×4 quaternion matrix are in the form of:

$$rotation_{3\times3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Continued on next page

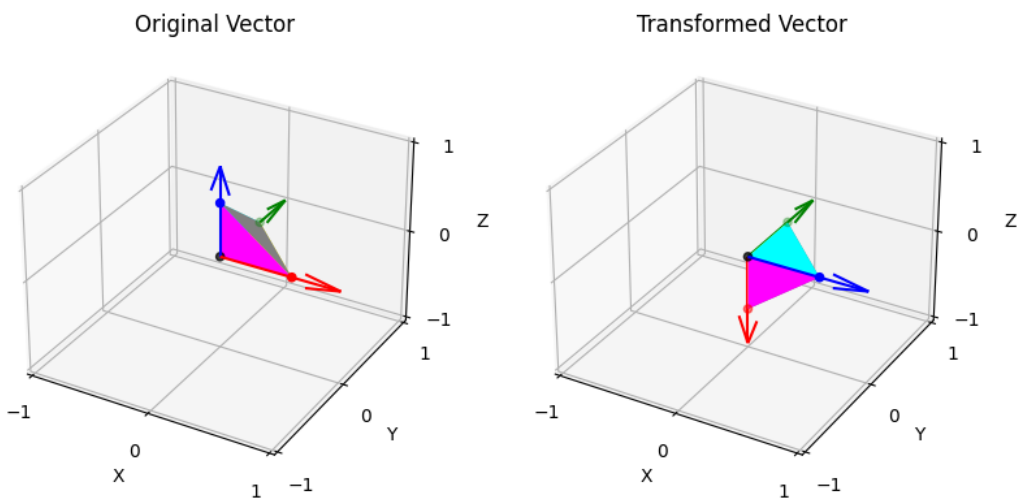
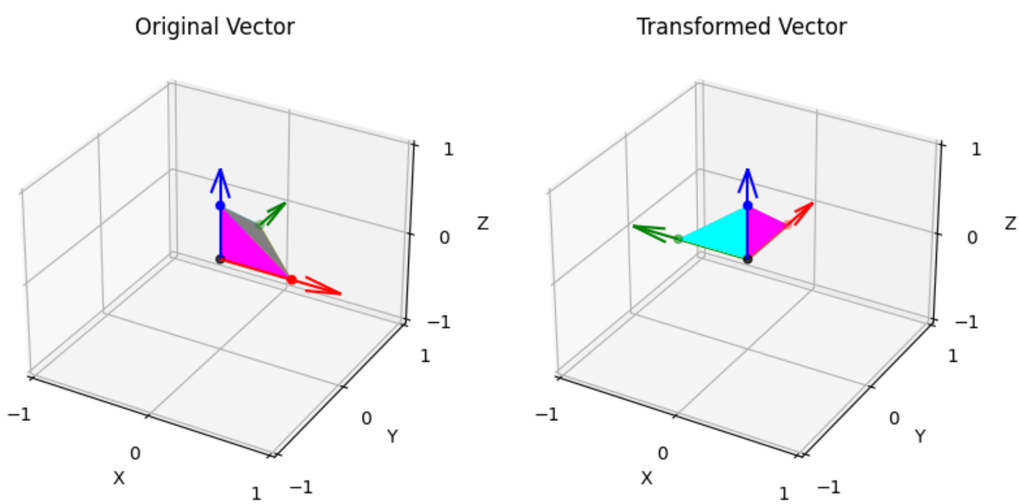


Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (1, 0, 1, 0)$, the quaternion equation $1 + 0i + 1j + 0k$ which results in a 90° rotation across y-axis. The norm is computed as $\sqrt{2} = \sqrt{1^2 + 0^2 + 1^2 + 0^2}$. A norm of $\sqrt{2}$ means that the transformed vector is the perpendicular angle to the original vector on the y-axis (i.e., an orthogonal planar rotation). Thus the y-z plane is swapped (i.e. planar reversal). This 3×3 rotation matrix and the 4×4 quaternion matrix are in the form of:

$$rotation_{3\times3} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \qquad quaternion_{4\times4} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$



Continued on next page

Table 2 – *Continued from previous page*

Table 2: *Left:* The identity matrix in 3D Cartesian coordinates. *Right:* $(a, b, c, d) = (1, 0, 0, 1)$, the quaternion equation $1 + 0i + 0j + 1k$ which results in a 90° rotation across z-axis. The norm is computed as $\sqrt{2} = \sqrt{1^2 + 0^2 + 0^2 + 1^2}$. A norm of $\sqrt{2}$ means that the transformed vector is the perpendicular angle to the original vector on the z-axis (i.e., an orthogonal planar rotation). Thus the x-y plane is swapped (i.e. planar reversal). This 3×3 rotation matrix and the 4×4 quaternion matrix are in the form of:

$$rotation_{3\times 3} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad quaternion_{4\times 4} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

How to compute distances in $SO(3)$?

Spherical Distance Measure To compare two rotations h_1 and h_2 in $SO(3)$, we can use the formula:

$$\rho_s(h_1, h_2) = \cos^{-1}(a_1 a_2 + b_1 b_2 + c_1 c_2 + d_1 d_2) \tag{5}$$

This formula measures the angle between the two rotations, which is a measure of their difference. However, this formula does not take into account the fact that h and $-h$ represent the same rotation. To account for this, we can use the formula:

$$\rho(h_1, h_2) = \min \rho_s(h_1, h_2), \rho_s(h_1, -h_2) \tag{6}$$

This formula measures the minimum distance between the two rotations, considering that h and $-h$ represent the same rotation.

Example: Suppose we have two rotations R_1 and R_2 in $SO(3)$ represented by the following 3×3 rotation matrices:

$$R_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \qquad R_2 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

To compare the two rotations, we can use the spherical distance measure: $\rho_s(h_1, h_2)$ as defined in Equation 5, where the values of a_1, b_1, c_1, d_1 come from the matrix R_1, and the values of a_2, b_2, c_2, d_2 come from the matrix R_2. So for the given matrices, we have:

$$\begin{aligned} a_1 = 1, \quad b_1 = 0, \quad c_1 = 0, \quad d_1 = 0 \\ a_2 = 0, \quad b_2 = 1, \quad c_2 = 0, \quad d_2 = -1 \end{aligned} \tag{8}$$

Substituting these values into the equation, we get:

$$\rho_s(h_1, h_2) = \cos^{-1}(1 \cdot 0 + 0 \cdot 1 + 0 \cdot 0 + 0 \cdot (-1)) = \cos^{-1}(0) = \frac{\pi}{2} \text{ radians} \tag{9}$$

However, this formula does not account for the fact that R and $-R$ represent the same rotation. To get the true minimum distance between R_1 and R_2, we need to use the formula for the minimum distance measure:

$$\rho(R_1, R_2) = \min\left(\rho_s(R_1, R_2), \rho_s(R_1, -R_2)\right) \tag{10}$$

Where The matrix for $-R_2$ can be obtained by negating each element of R_2:

$$R_2 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad -R_2 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{11}$$

Thus:

$$\begin{aligned} a_1 = 1, \quad b_1 = 0, \quad c_1 = 0, \quad d_1 = 0 \\ a_2 = 0, \quad b_2 = -1, \quad c_2 = 0, \quad d_2 = 1 \end{aligned} \tag{12}$$

$$\begin{aligned} \rho_s(h_1, -h_2) &= \cos^{-1}(1 \cdot 0 + 0 \cdot (-1) + 0 \cdot 0 + 0 \cdot 1) \\ &= \cos^{-1}(0) \\ &= \frac{\pi}{2} \text{ radians} \end{aligned} \tag{13}$$

Substituting in the values for R_1 and R_2, we get:

$$\rho(R_1, R_2) \quad = \quad \min\left(\frac{\pi}{2}, \frac{\pi}{2}\right) \quad = \quad \frac{\pi}{2} \text{ radians} \tag{14}$$

Therefore, the minimum distance between R_1 and R_2 and R_1 and $-R_2$ is $\frac{\pi}{2}$ radians, indicating that R_1 and R_2 are 180 degrees apart from each other.

Euclidean Distance Measure Another way to compute distances in $SO(3)$ is to use Euclidean distance in other spaces that can represent rotations. For example, we can use Euclidean distance in the space of yaw-pitch-roll angles, which represent the three angles of rotation needed to describe the orientation of an object in 3D space. This distance is computed as:

$$\rho_E(h_1, h_2) = \sqrt{(\theta_1 - \theta_2)^2 + (\phi_1 - \phi_2)^2 + (\psi_1 - \psi_2)^2} \tag{15}$$

where θ, ϕ, and ψ are the yaw, pitch, and roll angles respectively.

Alternatively, we can use Euclidean distance in $\mathbb{R}^9$, the space of 3 by 3 matrices, which can represent rotations in $SO(3)$. This distance is computed as:

$$\rho_E(R_1, R_2) = \sqrt{\sum_{i,j=1}^{3} (r_{1_{i,j}} - r_{2_{i,j}})^2} \tag{16}$$

where R_1 and R_2 are the 3 by 3 matrices representing the two rotations, and $r_{1_{i,j}}$ and $r_{2_{i,j}}$ are their corresponding entries.

Haar Measure The Haar measure for $SO(3)$ is a way to measure the volume or area of the space of all possible rotations in $SO(3)$. It is obtained as the standard area (or 3D volume) on the surface of the 3-sphere S^3. In other words, it is a way of assigning a probability measure to each rotation in $SO(3)$ based on its "size" or "volume" in the 3-sphere.

One important property of the Haar measure is that it is invariant under left or right multiplication by any rotation. This means that if we apply a rotation to a set of uniformly distributed random points on

S^3, the resulting set of rotations will also be uniformly distributed according to the Haar measure. This property makes the Haar measure an important tool for generating uniformly distributed random rotations on $SO(3)$, which can be useful in many applications such as robotics and computer graphics.

3 Homeomorphism

What is homeomorphism?

Informal Definition Homeomorphism is a type of continuous mapping between two topological spaces that preserves the shape of objects and the continuity of the space. This means that a space can be continuously deformed without tearing or cutting. In simpler terms, a homeomorphism matches up the points of two spaces in a way that preserves their connectivity and continuity. For example, a sphere and a cube are homeomorphic because they can be continuously deformed into each other without tearing or cutting. However, a sphere and a torus are not homeomorphic because they have different topological properties, such as the number of holes or handles they possess.

Formal Definition A function $f : X \to Y$ between two topological spaces X and Y is a homeomorphism if and only if it satisfies the following conditions:

1. f is bijective
2. f is continuous
3. f^{-1} is continuous

A homeomorphism is a bijective function that preserves the topological structure of the spaces. That is, for any open set U in X, $f(U)$ is open in Y, and for any open set V in Y, $f^{-1}(V)$ is open in X. If there exists a homeomorphism between two topological spaces X and Y, then X and Y are said to be homeomorphic and are denoted by $X \cong Y$.

Geometric Space and C-Space Homeomorphism is a useful concept that allows us to map one spatial domain onto another. Specifically, it enables us to project our coordinate points from one realization into abstracted forms, according to a set of formal rules that determine if a project from one space into another is possible. When motion planning, it is necessary to consider the space in different ways at different times. The world coordinates that the robot exists in may need to be converted into gridded coordinates for algorithmic planning, or may need to be adjusted as a projection into the C-space domain. Although the path may look different in each mapping, the source and destination points, obstacles, hazards, and other objects of interest remain the same.

Give examples of homeomorphic and non-homeomorphic subspaces?

Here are some examples of homeomorphic and non-homeomorphic subspaces:

Homeomorphic subspaces:

- A square and a circle are homeomorphic because you can continuously deform one into the other without tearing or cutting.

- A solid disk and a solid rectangle with rounded corners are homeomorphic because you can stretch and compress the rectangle to form the disk.
- Any two open intervals in the real line are homeomorphic because they both have the same topology.

Non-homeomorphic subspaces:

- A circle and a line segment are not homeomorphic because removing a point from a line segment disconnects it, while removing a point from a circle does not.
- A solid sphere and a torus are not homeomorphic because a torus has a hole while a sphere does not.
- The closed interval $[0, 1]$ and the open interval $(0, 1)$ are not homeomorphic because removing a point from the closed interval disconnects it, while removing a point from the open interval does not.

4 Motion Planning in Application

Follow a geometric path from the D* algorithm (HW-4) using a quadcopter UAV.

Hint: Useful API calls

- sim.createPath
- sim.getPathLengths
- sim.getSimulationTimeStep
- sim.setInt32Signal (optional)
- sim.getInt32Signal (optional)
- sim.readCustomDataBlock (optional)
- sim.unpackDoubleTable (optional)
