Assignment 1: Rigid Body Motion

Course: CS128 - Robot Program and Control Theory (Spring 2025)
Instructor: Md Alimoor Reza, Assistant Professor of Computer Science, Drake University
Due: Wednesday, February 26, 11:59 PM

In this assignment, you will be doing matrix, vector calculations, and rigid body transformations.

Part 1: Matrix and Vector Operations (/4 points)

Matrix Addition

As we have seen during the lecture, in matrix-matrix addition, we need to **add** the entries *element-wise* from the two matrices. Let's assume you are given two matrices, each with n rows and m columns, as follows:

$$A_{nm} = \begin{bmatrix} a_{11} & a_{12}... & a_{1m} \\ a_{21} & a_{22}... & a_{2m} \\ & ... & \\ a_{n1} & a_{n2}... & a_{nm} \end{bmatrix}, B_{nm} = \begin{bmatrix} b_{11} & b_{12}... & b_{1m} \\ b_{21} & b_{22}... & b_{2m} \\ & ... & \\ b_{n1} & b_{n2}... & b_{nm} \end{bmatrix}$$

The size of the resulting matrix C_{nm} will be the same as A_{nm} and B_{nm} with n rows and m columns. This operation could also be expressed as follows:

$$C_{nm} = A_{nm} + B_{nm} \tag{1}$$

Your goal is to apply *matrix-matrix addition* using the following matrices $A_{33} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ and $B_{33} = \begin{bmatrix} 10 & 20 & 30 \\ 40 & 50 & 60 \\ 70 & 80 & 90 \end{bmatrix}$. More specifically, compute the missing entries in the resulting matrix C_{33} with 3 rows and 3 columns as shown below:

$$\begin{bmatrix} ? & ? & ? \\ ? & ? & ? \\ ? & ? & ? \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + \begin{bmatrix} 10 & 20 & 30 \\ 40 & 50 & 60 \\ 70 & 80 & 90 \end{bmatrix}$$

Matrix Multiplication

Let's proceed to another very useful matrix operation. Let's assume you are given two matrices, A_{np} (with n rows and p columns) and B_{pm} (with p rows and m columns) as follows:

$$A_{np} = \begin{bmatrix} a_{11} & a_{12}... & a_{1p} \\ a_{21} & a_{22}... & a_{2p} \\ & ... & \\ a_{n1} & a_{n2}... & a_{np} \end{bmatrix}, B_{pm} = \begin{bmatrix} b_{11} & b_{12}... & b_{1m} \\ b_{21} & b_{22}... & b_{2m} \\ & ... & \\ b_{p1} & b_{p2}... & b_{pm} \end{bmatrix}$$

Matrix-matrix multiplication operation could also be expressed as follows:

$$C_{nm} = A_{np} * B_{pm} \tag{2}$$

There will be n rows and m columns in the resulting matrix C_{nm}. In matrix-matrix multiplication, each entry in the resulting matrix C_{nm} can be computed from the two matrices as follows:

$$c_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj} \tag{3}$$

For example, the first entry $c_{11} = a_{11} * b_{11} + a_{12} * b_{21} + ... + a_{1p} * b_{p1}$. Your goal is to apply *matrix-matrix multiplication* using the following matrices $A_{32} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$ and $B_{24} = \begin{bmatrix} 10 & 20 & 30 & 40 \\ 50 & 60 & 70 & 80 \end{bmatrix}$. More specifically, compute the missing entries in the resulting matrix C_{34} as shown below:

$$\begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} + \begin{bmatrix} 10 & 20 & 30 & 40 \\ 50 & 60 & 70 & 80 \end{bmatrix}$$

Matrix Transpose

Let's assume you are given a matrix, A_{nm} (with n rows and m columns) as follows:

$$A_{np} = \begin{bmatrix} a_{11} & a_{12}... & a_{1p} \\ a_{21} & a_{22}... & a_{2p} \\ & ... & \\ a_{n1} & a_{n2}... & a_{nm} \end{bmatrix}$$

In Matrix transpose operation, entries in the columns of the matrix A_{nm} become entries in the rows of resulting matrix C_{mn} (or vice versa). It could also be expressed as follows:

$$C_{mn} = A_{nm}^T \tag{4}$$

In other words, there will be m rows and n columns in the resulting matrix C_{mn}. Your goal is to apply *matrix transpose* operation on the following matrices $A_{32} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$ and $B_{24} = \begin{bmatrix} 10 & 20 & 30 & 40 \\ 50 & 60 & 70 & 80 \end{bmatrix}$ and find the transposed matrices.

Matrix Vector Multiplication

Let's assume you are given a matrix, A_{nm} (with n rows and m columns) and a vector X (one column vector with p rows) as follows:

$$A_{nm} = \begin{bmatrix} a_{11} & a_{12}... & a_{1m} \\ a_{21} & a_{22}... & a_{2m} \\ & ... & \\ a_{n1} & a_{n2}... & a_{nm} \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \\ ... \\ x_m \end{bmatrix}$$

Matrix-vector multiplication operation could also be expressed as follows:

$$Y = A_{nm} * X \tag{5}$$

The result will be a column vector Y with n rows. Each entry in the resulting column vector Y can be computed from the matrix's rows and vector X as follows:

$$y_i = \sum_{k=1}^{p} a_{ik} x_k \tag{6}$$

For example, the first entry $y_1 = a_{11} * x_1 + a_{12} * x_2 + ... + a_{1m} * x_m$. Your goal is to apply *matrix-vector multiplication* using the following matrix $A_{33} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ and $X = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix}$. More specifically, compute the missing entries in the resulting column vector Y as shown below:

$$\begin{bmatrix} ? \\ ? \\ ? \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix}.$$

Part 2: 2D and 3D Rigid Body Transformations (/8 points)

In this part, you are required to finish the transformations in Cartesian coordinates using 2D and 3D rotations and translations.

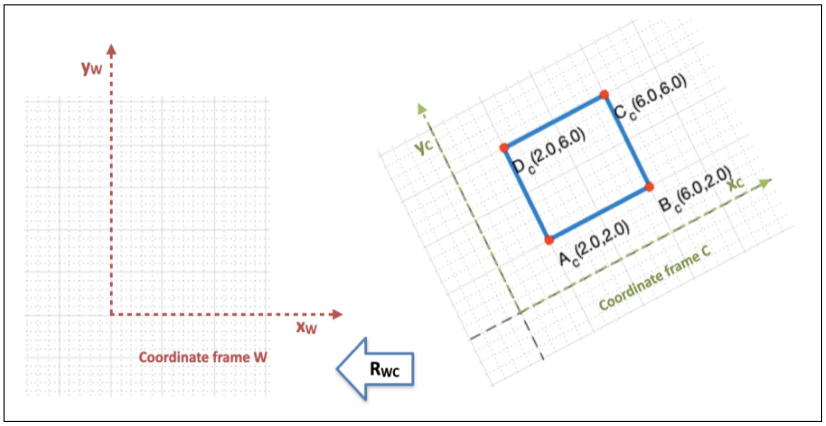

Figure 1: 2D rotational transformation between coordinate frame C and Coordinate frame W.

Transformation using only 2D Rotation

Let's assume coordinate frame C (attached to the rigid body) is rotated by an angle of 30 with respect to coordinate frame W as shown in Figure 1. As we have seen during the lecture, you can transform 2D points from in Coordinate frame C to 2D points in Coordinate frame W. For example, 4 corners of the rectangle in Coordinate frame C A_c, B_c, C_c, and D_c (as shown in Figure 1) can be transformed to coordinate A_w, B_w, C_w, and D_w in Coordinate frame W. More specifically, you can transform the homogeneous coordinates of these 4 corners using 3x3 2D rotation matrix using the following formula:

$$X_W = \begin{bmatrix} R_{WC} & & 0 \\ & & 0 \\ 0 & 0 & 1 \end{bmatrix} X_C \tag{7}$$

where $R_{WC} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$. Your goal is to apply *2D rotational transformation* on these 4 corner points in Coordinate frame C (in homogeneous coordinate) ie, $A_c = \begin{bmatrix} 2.0 \\ 2.0 \\ 1 \end{bmatrix}$, $B_c = \begin{bmatrix} 6.0 \\ 2.0 \\ 1 \end{bmatrix}$, $C_c = \begin{bmatrix} 6.0 \\ 6.0 \\ 1 \end{bmatrix}$, and $D_c = \begin{bmatrix} 2.0 \\ 6.0 \\ 1 \end{bmatrix}$. to find the transformed coordinates (in coordinate frame W) A_w, B_w, C_w, and D_w. Plot those transformed points on a graph. You can draw it on paper and upload the digital copy of it on Blackboard.

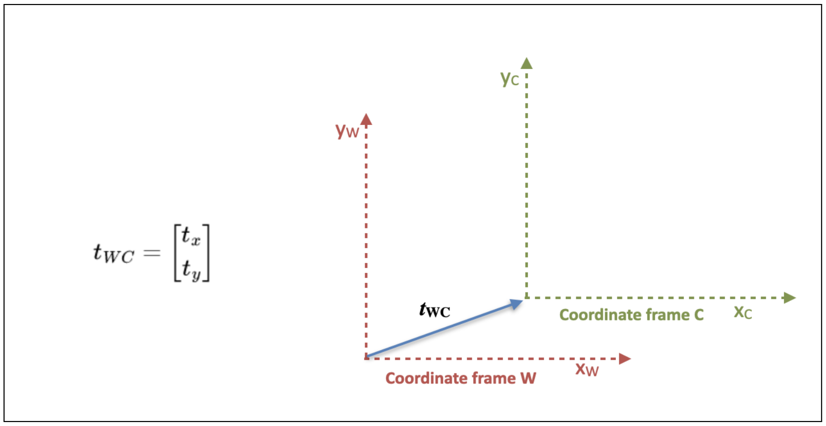

Figure 2: 2D translational transformation between coordinate frame C and Coordinate frame W.

Transformation using only 2D Translation

Let's assume coordinate frame C (attached to the rigid body) is translated by a vector t_{WC} with respect to coordinate frame W as shown in Figure 2. As we have seen during the lecture, you can transform 2D points from in Coordinate frame C to 2D points in Coordinate frame W. For example, 4 corners of a rectangle in Coordinate frame C A_c, B_c, C_c, and D_c can be transformed to coordinate A_w, B_w, C_w, and D_w in Coordinate frame W. More specifically, you can transform the homogeneous coordinates of these 4 corners using 3x3 2D translation matrix using the following formula:

$$X_W = \begin{bmatrix} 0 & 0 & t_x \\ 0 & 0 & t_y \\ 0 & 0 & 1 \end{bmatrix} X_C \tag{8}$$

Assuming $\begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$, your goal is to apply *2D translational transformation* on these 4 corner points in Coordinate frame C (in homogeneous coordinate) ie, $A_c = \begin{bmatrix} 2.0 \\ 2.0 \\ 1 \end{bmatrix}$, $B_c = \begin{bmatrix} 6.0 \\ 2.0 \\ 1 \end{bmatrix}$, $C_c = \begin{bmatrix} 6.0 \\ 6.0 \\ 1 \end{bmatrix}$, and $D_c = \begin{bmatrix} 2.0 \\ 6.0 \\ 1 \end{bmatrix}$ to find the transformed coordinates (in coordinate frame W) A_w, B_w, C_w, and D_w. Plot those transformed points on a graph. You can draw it on paper and upload a digital copy of it on Blackboard.

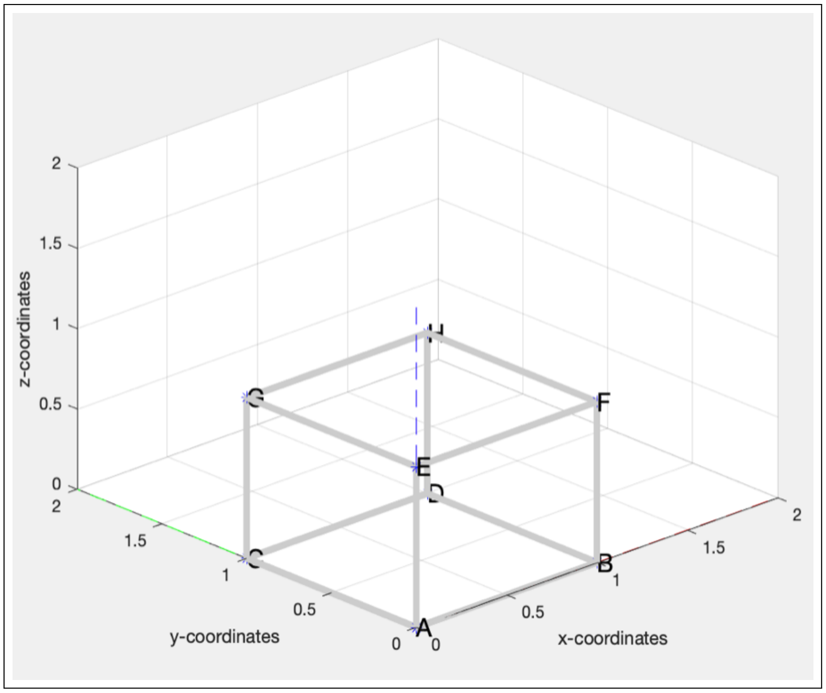

Figure 3: 8 corners (A, B, C, D, E, F, G, and H) of a 3D cube in a coordinate frame C.

Transformation using only 3D Rotation

Let's assume coordinate frame C (attached to the rigid body) is rotated by an angle of 30 with respect to coordinate frame W as shown in Figure 3. As we have seen during the lecture, you can transform 3D points from in Coordinate frame C to 3D points in Coordinate frame W. For example, 8 corners of the 3D cube in Coordinate frame C A_c, B_c, C_c, D_c, E_c, F_c, G_c, and H_c (as shown in Figure 3) can be transformed to coordinate A_w, B_w, C_w, D_w, E_w, F_w, G_w, and H_w in Coordinate frame W. More specifically, you can transform the homogeneous coordinates of these 4 corners using 4x4 3D rotation matrix using the following formula:

$$X_W = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 30 & -\sin 30 & 0 \\ 0 & \sin 30 & \cos 30 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} X_C \tag{9}$$

Your goal is to apply *3D rotational transformation* on these 8 corner points in Coordinate frame C (in homogeneous coordinate) ie, $A_c = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 1 \end{bmatrix}, B_c = \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ 1 \end{bmatrix}, C_c = \begin{bmatrix} 0.0 \\ 1.0 \\ 0.0 \\ 1 \end{bmatrix}, D_c = \begin{bmatrix} 1.0 \\ 1.0 \\ 0.0 \\ 1 \end{bmatrix}, E_c = \begin{bmatrix} 0.0 \\ 0.0 \\ 1.0 \\ 1 \end{bmatrix}, F_c = \begin{bmatrix} 1.0 \\ 0.0 \\ 1.0 \\ 1 \end{bmatrix}, G_c = \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 1 \end{bmatrix}$, and $H_c = \begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1 \end{bmatrix}$. to find the transformed coordinates (in coordinate frame W) A_w, B_w, C_w, D_w, E_w, F_w, G_w, and H_w. Plot those transformed points on a graph. You can draw it on paper and upload a digital copy of it on Blackboard.

Transformation using only 3D Translation

Let's assume coordinate frame C (attached to a rigid body) is translated by a vector $t_{WC} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$ with respect to coordinate frame W. You can transform 3D points from in Coordinate frame C to 3D points in Coordinate frame W. For example, 8 corners of the 3D cube in Coordinate frame C A_c, B_c, C_c, D_c, E_c, F_c, G_c, and H_c can be transformed to coordinate A_w, B_w, C_w, D_w, E_w, F_w, G_w, and H_w in Coordinate frame W. More specifically, you can transform the homogeneous coordinates of these 8 corners using 4x4 3D translation matrix using the following formula:

$$X_W = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} X_C \tag{10}$$

Assuming $\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$, your goal is to apply *3D translational transformation* on these 8 corner points in Coordinate frame C (in homogeneous coordinate) ie, $A_c = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 1 \end{bmatrix}, B_c = \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \\ 1 \end{bmatrix}, C_c = \begin{bmatrix} 0.0 \\ 1.0 \\ 0.0 \\ 1 \end{bmatrix}, D_c = \begin{bmatrix} 1.0 \\ 1.0 \\ 0.0 \\ 1 \end{bmatrix}, E_c = \begin{bmatrix} 0.0 \\ 0.0 \\ 1.0 \\ 1 \end{bmatrix}, F_c = \begin{bmatrix} 1.0 \\ 0.0 \\ 1.0 \\ 1 \end{bmatrix}, G_c = \begin{bmatrix} 0.0 \\ 1.0 \\ 1.0 \\ 1 \end{bmatrix}$, and $H_c = \begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1 \end{bmatrix}$ to find the transformed coordinates (in coordinate frame W) A_w, B_w, C_w, D_w, E_w, F_w, G_w, and H_w. Plot those transformed points on a graph. You can draw it on paper and upload a digital copy of it on Blackboard.

What to turn in:

Please submit a zip file containing your document. You could write your solution in either *.docx, *.doc, or *.pdf file. if you prefer to write your answers on paper, you could do so; in that case, please take a photo of your paper or upload a scanned copy of your paper. Again, if you have any questions, please ask or email me.