EENG 428 Introduction to Robotics Laboratory

Lab Session 2

Objective

This Session aims to explore the usage of symbolic Matlab in order to simplify the robotic analysis.

1- Introduction to symbolic variables, expressions and function

Usually, mathematical formulas are derived using symbols. It would be very useful if a computer helps in the derivation and/or the simplification of symbolic formulas rather than only calculating numbers.

Defining the Symbolic: Using Matlab, the Matlab expression ***syms*** (***sym*** for single variable) allows us to define symbolic variables, and the function ***sym ()*** allows us to define symbolic expressions.

Example 1.1

Write the following expression in matlab using symbolic variables

$$G = ax^4 + bx^3 + e^x + c$$

```
syms G x a b c
G= a*x^4 + b*x^3 + exp(x)+ c
```

- We can directly write G as a symbolic expression

```
G=sym('a*x^4 + b*x^3 + exp(x)+ c')
```

- We also can express G as a symbolic function as follow

```
syms  G(x) a b c
G(x)=a*x^4 + b*x^3 + exp(x)+ c
G(3)
```


Prepared by: eyad.almasri@emu.edu.tr

Substituting Numbers: After defining an expression, we can substitute values instead of some symbols using the command ***subs (S, OLD, NEW)***. This function replaces OLD with NEW in the symbolic expression S.

Example 1.2

Write the following expression in matlab using symbolic variables

$$G = e^x + \log(y)$$

And find the value of G by replacing x and y by 1 and 2 respectively.

```
syms x y
G = exp(x) + log(y)
f=subs(G,[x y],[1 2])
numeric=double(f)
```

Plotting Symbolic Formulas: Usually, a functions can be plotted verses its independent variable within some range by hand. Matlab symbolic toolbox offers a similar service using the function ***ezplot ()***.

Making the function to be more similar to human handwriting: Usually, humans do not express formulas as computers do. The function ***pretty ()*** helps us to see the formulas as we are used to on paper.

Example 1.3

Write the following expression in matlab using symbolic variables

$$G = \frac{\sin(x)^2 - 1}{2\,x^2 + x}$$

And plot G when x varying from 2 to 4

```
syms x y
G = ((sin(x)^2)-1)/((2*x^2)+x)
pretty(G)
ezplot(G,[2 4])
```


Prepared by: eyad.almasri@emu.edu.tr

Symbolic Matrices: In this course, we are highly interested in working with symbolic matrices.

Example 1.4

Given the following linear system

$$\boldsymbol{Ax = b}$$

Where A is a 2×2 invertible matrix, x is a column vector with 2 entries, and b is a column vector with 2 entries. Represent and solve this system using symbolic matlab.

```
% we assume A is invertible 2*2 matrix
A=sym('a',[2 2])
% syms a b c d
% A=[a b;c d]
x=sym('x',[2 1])
b=sym('b',[2 1])
S=inv(A)*b %% the solution of the system
```

Symbolic solve: The Symbolic Math Toolbox supports the solving of equations and systems of equations using **solve( )**. It supports solving multivariate equations, solving inequalities and solving with assumptions. Solutions can be found symbolically or numerically

Example 1.5

solve the following equation using matlab symbolic solve ( )

$$ax^2 - b = 0$$

```
s=solve('a*x^2 - b == 0')
pretty(s)
```

You can observe that the matlab automatically recognize that the unknown is x and solve the equation based on that assumption. We can change this presumption by explicit the desired variable inside the solve statement.

```
s=solve('a*x^2 - b == 0', 'a'); % solve for a
pretty(s)
```


Prepared by: eyad.almasri@emu.edu.tr

Solving a set of algebraic equations: Matlab can help solving a set of equations as in the following example

Example 1.6

Solve the following set of equation

$$\begin{bmatrix} 1 & 1 & 1 \\ 5 & 5 & 7 \\ 3 & 1 & 3 \end{bmatrix} . \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ 18 \\ 3 \end{bmatrix}$$

```
F=solve('x+y+z=4','5*x+5*y+7*z=18','3*x+1*y+3*z=3')
solution=[f.x
          f.y
          f.z]
```

In Calculus: The Symbolic Math Toolbox has a full set of calculus tools for applied mathematics. It can perform multivariate symbolic integration and differentiation.

Example 1.7

Consider the function

$$f(x,y) = \sin\left(\frac{\sqrt{e^{xy}+1}}{2xy}\right)$$

Use Matlab to find

$$\frac{\partial f(x,y)}{\partial y}$$

```
syms f(x,y)
f(x,y)=sin(sqrt(exp(x*y)+1)/(2*x*y));
pretty(f)
D=diff(f,y);
pretty(simplify(D))
```


Prepared by: eyad.almasri@emu.edu.tr

Example 1.8

Consider the function

$$f(x) = x\ log(1 + x)$$

Use Matlab to find

$$\int_0^1 f(x)dx$$

```
syms f(x)
f(x)=x*log(1+x)
int(f,x,0,1)
```

Differential Equations: The Symbolic Math Toolbox can analytically solve systems of ordinary differential equations using *dsolve*().

Example 1.9

Solve the first order ODEs

$$\frac{dy}{dx} = -ay$$

with the initial condition $y(0) = b$

```
syms a b y(x)
dsolve(diff(y)== -a*y,y(0)==b)
```

2- Basic Transformation matrices using Matlab toolbox

Matlab robotic toolbox offers some helpful transformation matrices that can facilitate different applications of robotic analysis. The basic transformation matrices are:

- Rotation about x-axis. Matlab function ***trotx ()***
- Rotation about y-axis. Matlab function ***troty ()***
- Rotation about z-axis. Matlab function ***trotz ()***
- Translation along x, y and z axes. Matlab function ***trnasl ()***

Prepared by: eyad.almasri@emu.edu.tr

Example 2.1

Representation of a pure translation or a pure Rotation about an axis

Define a rotational transformation matrix about x-axis with angle t

Define a rotational transformation matrix about y-axis with angle t

Define a rotational transformation matrix about z-axis with angle t

Define a translation transformation matrix along x, y, z axes with distance $px\ py\ pz$ respectively. Replace $px\ py\ pz$ with 1,2,3 respectively.

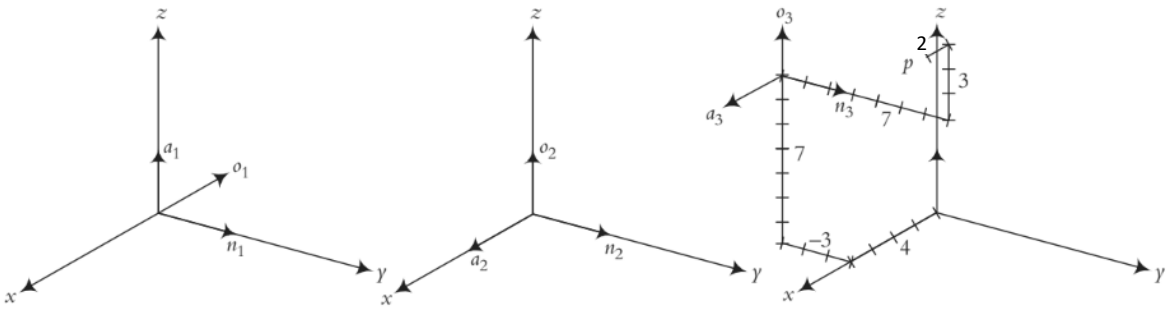
```
syms t
trotx(t)% rotate about x-axis with angle t
troty(t)% rotate about y-axis with angle t
trotz(t)% rotate about z-axis with angle t
syms px py pz
A=transl(px,py,pz) % translate along x,y and z axes
AA=subs(A,[px,py,pz],[1,2,3])
```

3- Representation of a combined Transformation

Combined transformations consist of a number of successive translations and rotations about a ***fixed reference frame*** or the ***moving current frame*** axes. Any transformation can be resolved into a set of translations and rotations in a particular order.

Example 3.1 (*fixed reference frame axes*)

A point $p = [7\ \ 3\ \ 2]^T$ is attached to a frame F_{noa} and is subjected to the following transformations. Rotation of 90° about the z-axis Followed by a rotation of 90° about the y-axis Followed by a translation of $[4\ \ -3\ \ 7]$



After the 1st transformation After the 2nd transformation After the 3rd transformation

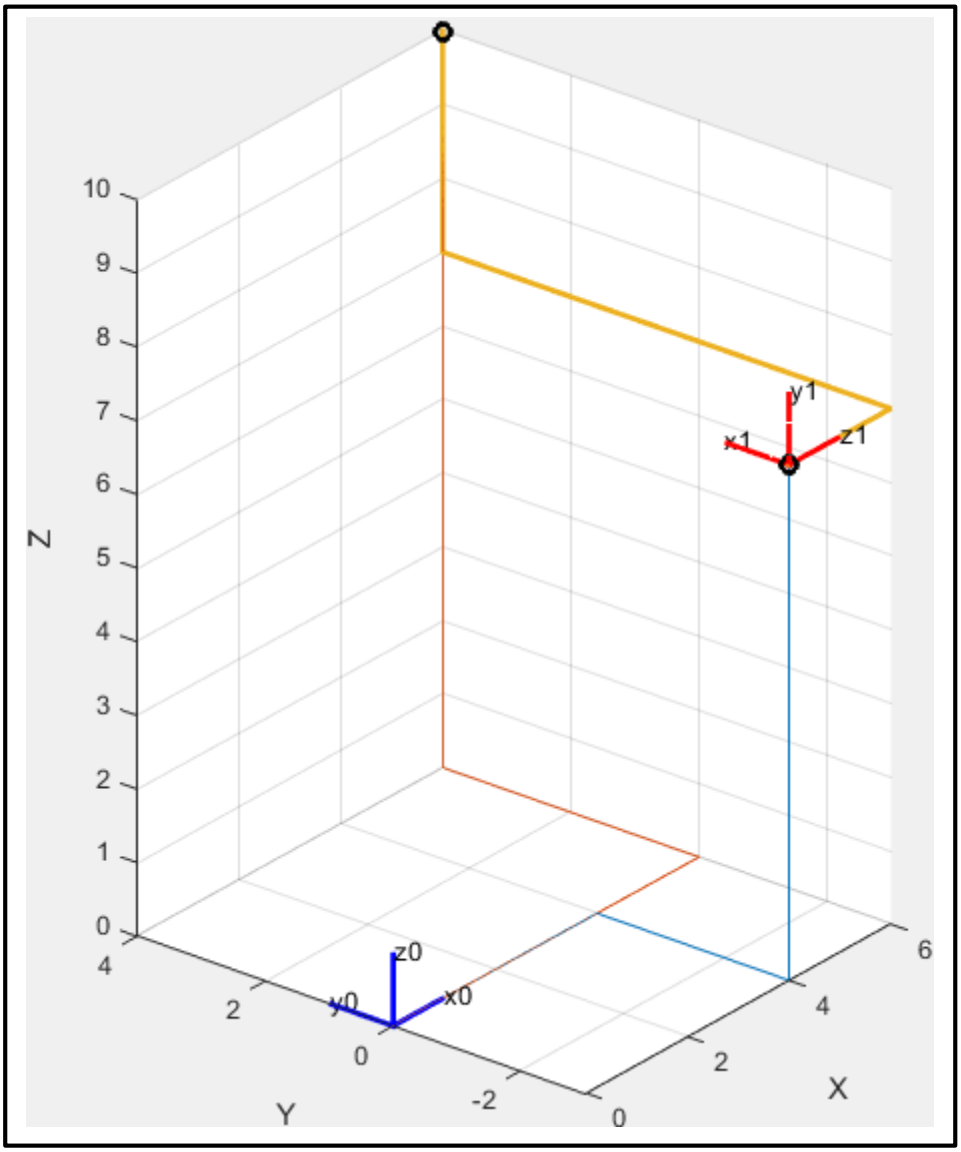
Prepared by: eyad.almasri@emu.edu.tr

Find the coordinates of the point relative to the reference frame at the conclusion of transformations.

The matrix equation representing the transformation is

$$\boldsymbol{P_{xyz} = transl\,(4,-3,7)\;Rot(y,90^{\circ})\;Rot(z,90^{\circ})\;P_{noa}}$$

```
T=transl(4,-3,7)*troty(pi/2)*trotz(pi/2)
p=T*[7;3;2;1]
```




```
T =
    0    0    1    4
    1    0    0   -3
    0    1    0    7
    0    0    0    1

p =
    6
    4
   10
    1
```

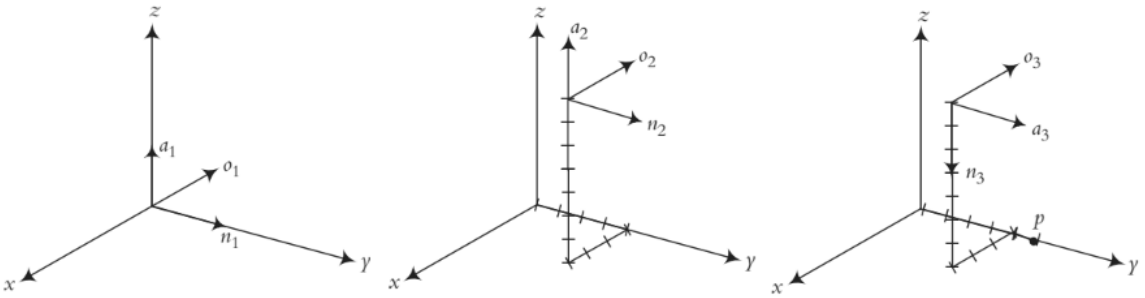

Prepared by: eyad.almasri@emu.edu.tr

Example 3.2 (*moving current frame axes*)

Assume that A point $p = [7 \;\; 3 \;\; 1]^T$ is attached to a frame F_{noa} subjected to the transformations relative to the current moving frame, as listed below.

- Rotation of 90° about the a-axis.
- translation of $[4 \;\; -3 \;\; 7]$ along $n-, o-, a-$axes respectively.
- rotation of 90 about the o-axis.

Find the coordinates of the point relative to the reference frame after transformations are completed.



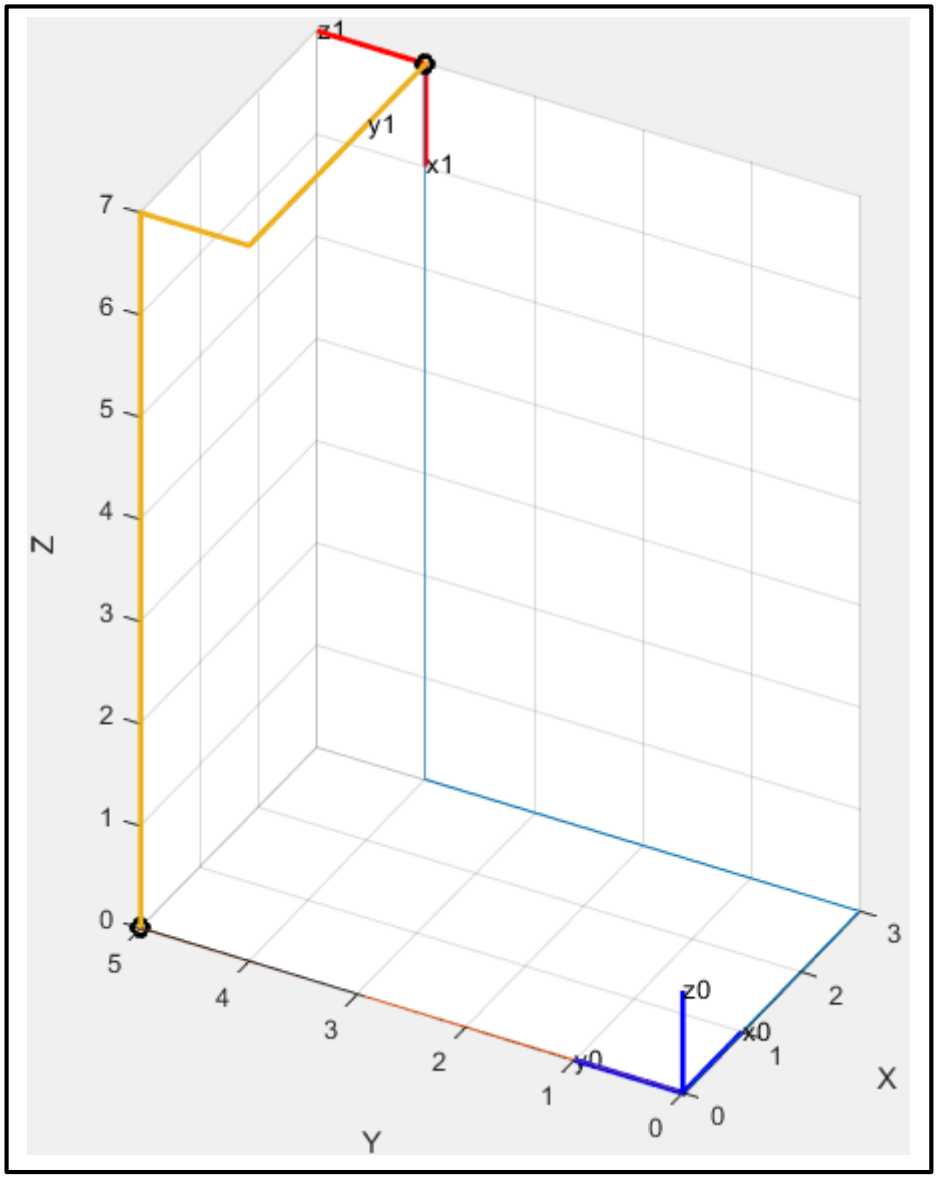
After the 1st transformation | After the 2nd transformation | After the 3rd transformation

```
T=trotz(pi/2)*transl(4,-3,7)*troty(pi/2)
P=T*[7;3;1;1]
```

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 7 \\ 3 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 5 \\ 0 \\ 1 \end{bmatrix}$$

To calculate the changes in the coordinates of a point attached to the ***current frame relative to the reference*** frame, the transformation matrix is ***Pre-multiplied*.** However, since the position of a point or an object attached to a moving frame is always measured relative to that ***moving frame***, the position matrix describing the point is ***post-multiplied*.**

Prepared by: eyad.almasri@emu.edu.tr



T =

0	-1	0	3
0	0	1	4
-1	0	0	7
0	0	0	1

P =

0

5

0

1

References:

1- Moore, H. (2017). *MATLAB for Engineers*. Pearson.

2- Niku, S. (2010). *Introduction to robotics*. John Wiley & Sons.

Prepared by: eyad.almasri@emu.edu.tr