Forward Kinematics

Forward kinematics is determining where the tip of the robot is given it's joint angles. To do this, a reference frame is assigned to each joint. There is a standard way to do this: by having a standard, different designers will get the same result.

- First, you rotate the reference frame about the x-axis so that the two z-axis line up (twist)
- Second, move along the x axis
- Third, rotate about the z-axis, finally
- Move along the z-axis to end up at the next reference frame.

Putting these rotations and translations into one matrix gives the transformation matrix to go from joint 1 to joint 0

$$T_{01} = R_x(\alpha_0)D_x(a_0)R_z(\theta_1)D_z(d_1)$$

or, multiplying it out:

$$T_{01} = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_0 \\ s\theta_1 c\alpha_0 & c\theta_1 c\alpha_0 & -s\alpha_0 & -s\alpha_0 d_1 \\ s\theta_1 s\alpha_0 & c\theta_1 s\alpha_0 & c\alpha_0 & c\alpha_0 d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This is the subroutine Transform.m posted on Bison Academy.

```
function [T] = Transform(alpha, a, d, theta)

T1 = [cos(theta), -sin(theta), 0, a];
T2 = [sin(theta)*cos(alpha), cos(theta)*cos(alpha), -sin(alpha), -d*sin(alpha)];
T3 = [sin(theta)*sin(alpha), cos(theta)*sin(alpha), cos(alpha), d*cos(alpha)];
T4 = [0,0,0,1];

T = [T1; T2; T3; T4];

end
```

Example: Determine the transformation matrix T01 when

- alpha0 = 90 degrees (twist)
- a0 = 50cm
- d1 = 40cm
- theta1 = 30 degrees (rotation)

Solution:

```
Transform(pi/2, 50, 40, pi/4)

    0.7071   -0.7071         0   50.0000
    0.0000    0.0000   -1.0000  -40.0000
    0.7071    0.7071    0.0000    0.0000
         0         0         0    1.0000
```

Repeating this procedure for joints 1 to 2, 2 to 3, etc. allows you to find the transformation matrix to go from tip coordinates to base (earth) coordinates.

With a robot manipulator, you have N joints. Each joint will have its associate reference frame (i.e. the motor at each joint knows its own angle.) The problem is how to determine the tip position given the joint angles.

Before we can do that, however, we need to define the reference frames for each joint. For this assignment, there is a standard way of doing it.

Types of Joints:

First, determine the type of joint you're dealing with. This doesn't really affect the assignment of reference frames (it it's a rotational joint, and angle will be a variable, if it's a translational joint, a displacement will be a variable).

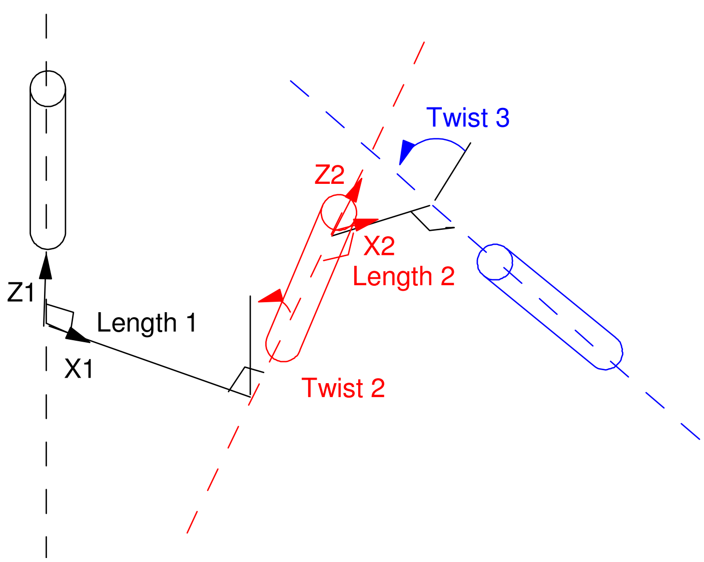
- Translational: The motor changes the length of the arm
- Rotational: The motor changes the angle of the arm
- Compound: If a joint can do several things, like a hip joint

If a joint is compound, it is treated as two separate joints with a displacement of zero.

Link Numbering:

Next, determine how many links you need. The links are numbered starting from the immobile base, frame 0. Coordinate frame 0 is also called the earth reference frame, since it is usually where the robot is bolted to the floor.

- Each subsequent reference frame is numbered from 1 on up.
- Link i connects axis i to i+1



Length i: The length of link i is the distance from axis i to i+1. This is the shortest distance between the two lines that pass through the axis of rotation.

Twist i: Relative to the line which is perpendicular to both axis (i.e. looking down this line), the twist is the angle between the two axis.

Reference Frame Assignment:

The origin of reference frame i is the perpendicular line from reference frame i to i+1

- The Z axis aligns with the rotation of the joint (the joint rotates about the Z axis). If the joint is translational, the joint slides along the Z axis.
- The X axis points to the next axis.
- The Y axis follows the right-hand rule

$$X \times Y = Z$$

Procedure:

1. Identify the joint axis. Draw an infinite line along each axis.
2. Identify the perpendicular lines between each axis. At the point of intersection, define a reference frame.
3. Assign the Zi axis pointing along the ith joint axis.
4. Assign the Xi axis pointing to the next axis (i.e. along the perpendicular line between axis). If the joint axis intersect, Xi is perpendicular to both Z axis.
5. Assign the Yi axis following the right-hand rule

$$X \times Y = Z$$

6. Assign axis 0 to match axis 1 when the first joint angle is zero.
7. The zero position is when a all X axis are pointing in the same direction.

Link Parameters:

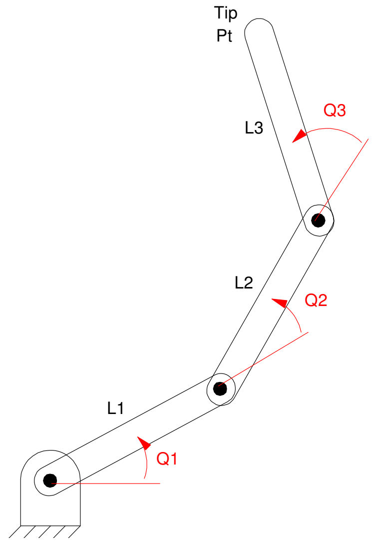
a_i The distance from Zi to Zi+1 measured along the Xi axis

α_i The angle between the Zi and Zi+1 axis (twist)

d_i The distance from Xi-1 to Xi measured along the Zi axis

θ_i The angle between Xi-1 and Xi measured about the Zi axis

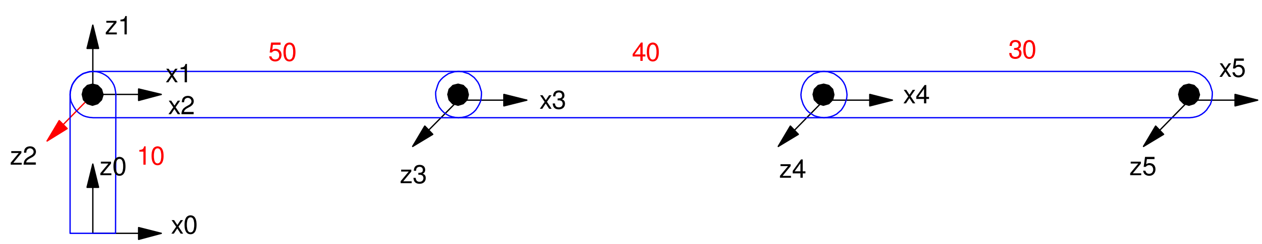
Example: Define the reference frames for the following robot (a model for a human finger):



Example 1: Modeling a human finger

Solution in zero-position: Note that if assigned correctly

- You can only move in the x and z direction. If you try to move in the y-diretion, you made a mistake in assigning the reference frames.
- In zero position, all of the x-axis point in the same direction



Definition of reference frames and zero position for modeling a human finger

The link parameters are then

Link i	α_{i-1} The angle between the Zi-1 and Zi axis (twist)	a_{i-1} The distance from Zi-1 to Zi measured along the Xi-1 axis	d_i The distance from Xi-1 to Xi measured along the Zi axis	θ_i The angle between Xi-1 and Xi measured about the Zi axis
1	0	0	10	0
2	+90 degrees	0	0	θ_1
3	0	50	0	θ_2
4	0	40	0	θ_3
5	0	30	0	0

Once you define the robot, you can

- draw the robot and
- determine the tip position

given the joint angles. This is the program RRR.m

```
function [Tip] = RRR(W, TIP)

alpha = [0, pi/2,  0,   0,   0];
a     = [0,    0,  50, 40, 30];
d     = [10,   0,   0,   0,   0];
Q     = [0, W(1), W(2), W(3), 0];

T01a = Transform(alpha(1), a(1), 0,    Q(1));
T01  = Transform(alpha(1), a(1), d(1), Q(1));
T12a = Transform(alpha(2), a(2), 0,    Q(2));
T12  = Transform(alpha(2), a(2), d(2), Q(2));
etc.
```

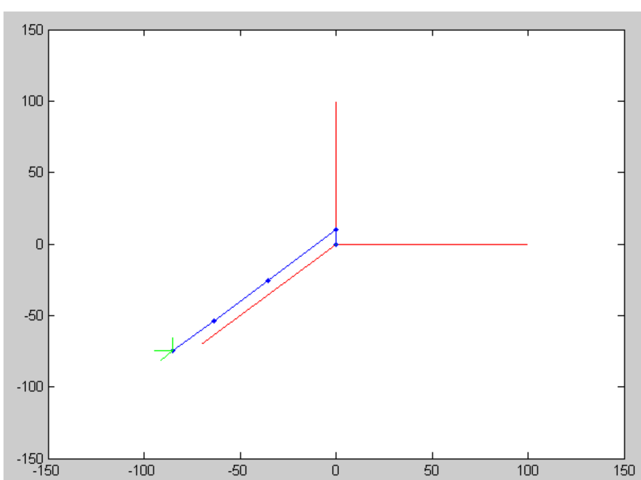
Note that the previous table defines the robot: simply change this table and you have a whole new robot.

The way this program works is, once you define the refrence frames (alpha, a, d, Q), it

- Defines the transformation matrix to reference frame #1,
- Then #2, etc.
- It then draws a line from each reference frame to the next, and
- It returns the tip position.

For example, the tip position in zero position is:

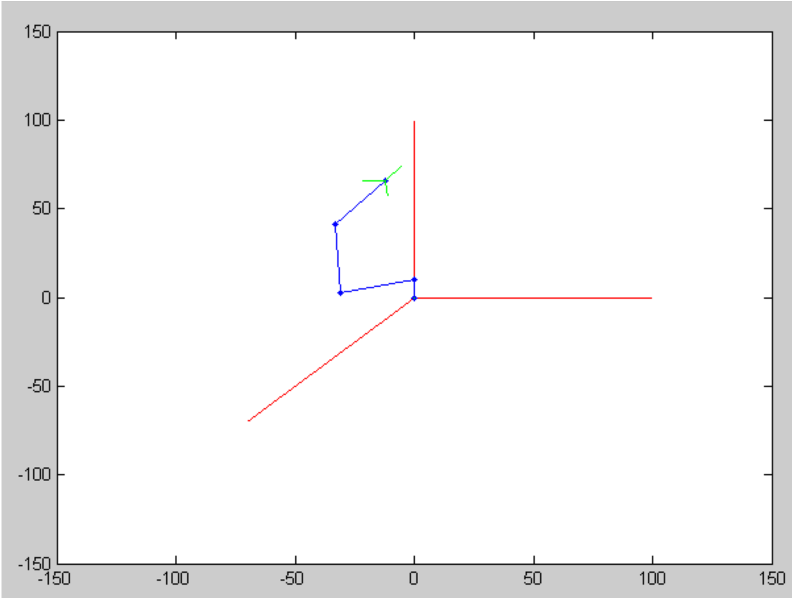
```
TIP = zeros(3,1);   % we will use this later
RRR([0,0,0], TIP)
x   120
y     0
z    10
      1
```



Draw the robot and calculate the tip position for angles of (0.5, 1, 1.5 radians)

Solution:

```
>> RRR([0.5,1,1.5],TIP)

x   17.0088
y    0.0000
z   78.1047
     1.0000
```



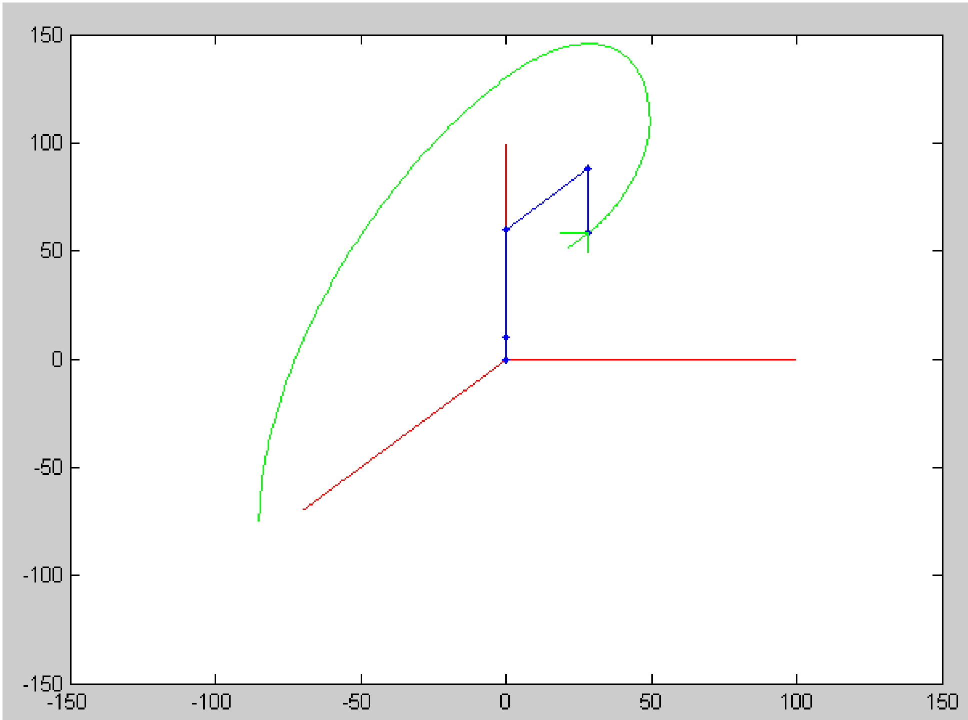
You can also show the path of the tip during a move:

- Start from zero position
- Rotate Q1 to 90 degrees, then
- Rotate Q2 to 90 degrees, then
- Rotate Q3 to 90 degrees

Code:

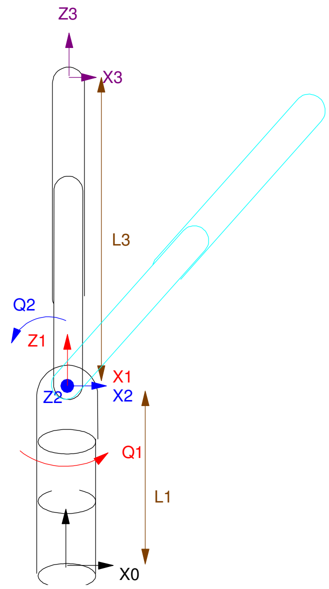
```
TIP = RRR([0,0,0], zeros(4,1));
for i=1:90
    T = RRR([i*pi/180, 0, 0], TIP);
    TIP = [TIP,T];
end

for i=1:90
    T = RRR([pi/2, i*pi/180, 0], TIP);
    TIP = [TIP,T];
end

for i=1:90
    T = RRR([pi/2, pi/2, i*pi/180], TIP);
    TIP = [TIP,T];
end
```

Result:

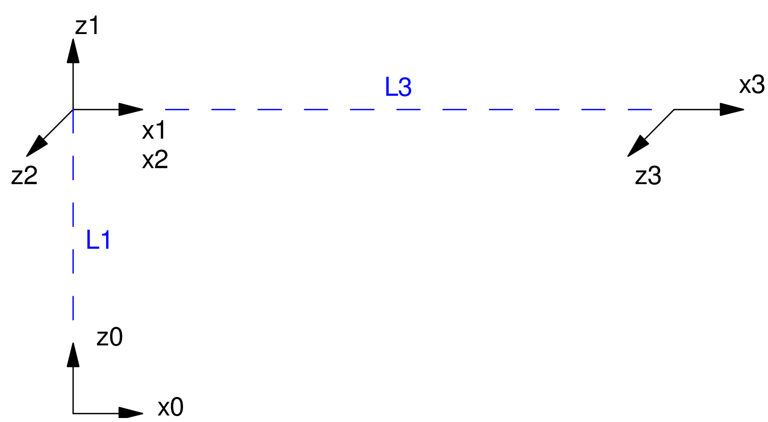


Trajectory of the tip (green) as you rotate Q1, then Q2, then Q3

Example 2: RRP Robot



First, redraw the robot in zero position, assigning reference frames



Reference frame definitions at zero position

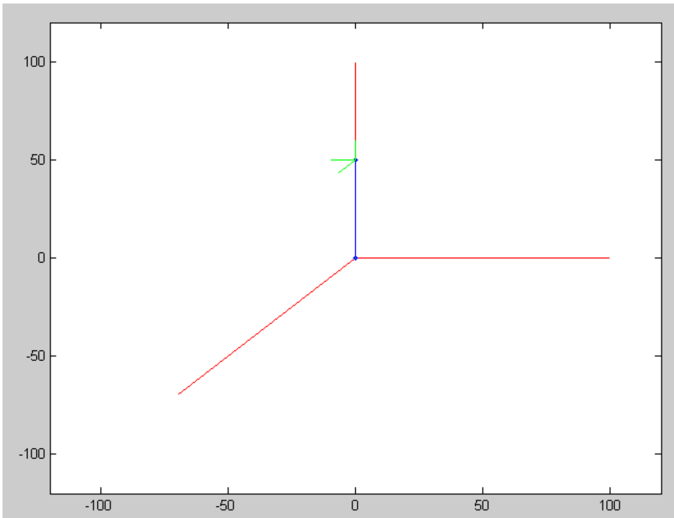
Link i	α_{i-1} The angle between the Zi-1 and Zi axis (twist)	a_{i-1} The distance from Zi-1 to Zi measured along the Xi-1 axis	d_i The distance from Xi-1 to Xi measured along the Zi axis	θ_i The angle between Xi-1 and Xi measured about the Zi axis
1	0	0	L1	θ_1
2	+90 degrees	0	0	θ_2
3	0	L3	0	0

To simulate this robot, change the first four lines of code in RRR (creating RRP)

```
function [Tip] = RRP(W, TIP)

alpha = [0, pi/2,    0,   0,   0];
a     = [0,    0,  W(3),  0,   0];
d     = [50,   0,     0,  0,   0];
Q     = [W(1), W(2), 0,   0,   0];

etc.
```

Determine the tip position at zero position:

```
>> TIP = RRP([0,0,0],zeros(4,1))

x      0
y      0
z     50
       1
```



Zero position for the RRP robot

Display the position at { 45 degrees, 60 degrees, 50cm }

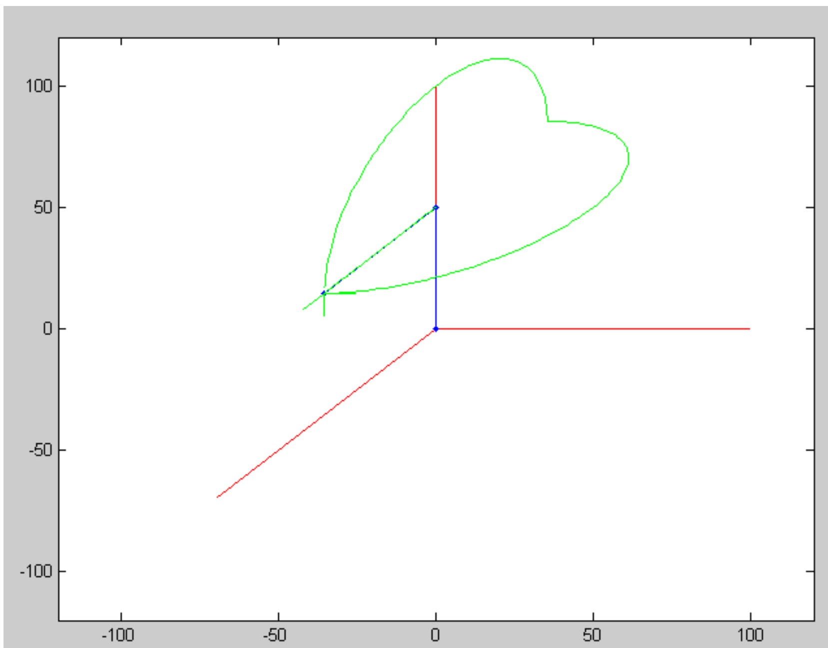
```
TIP = RRP([pi/4, pi/3, 50],zeros(4,1))

x   17.6777
y   17.6777
z   93.3013
     1.0000
```

Trace out the tip position as

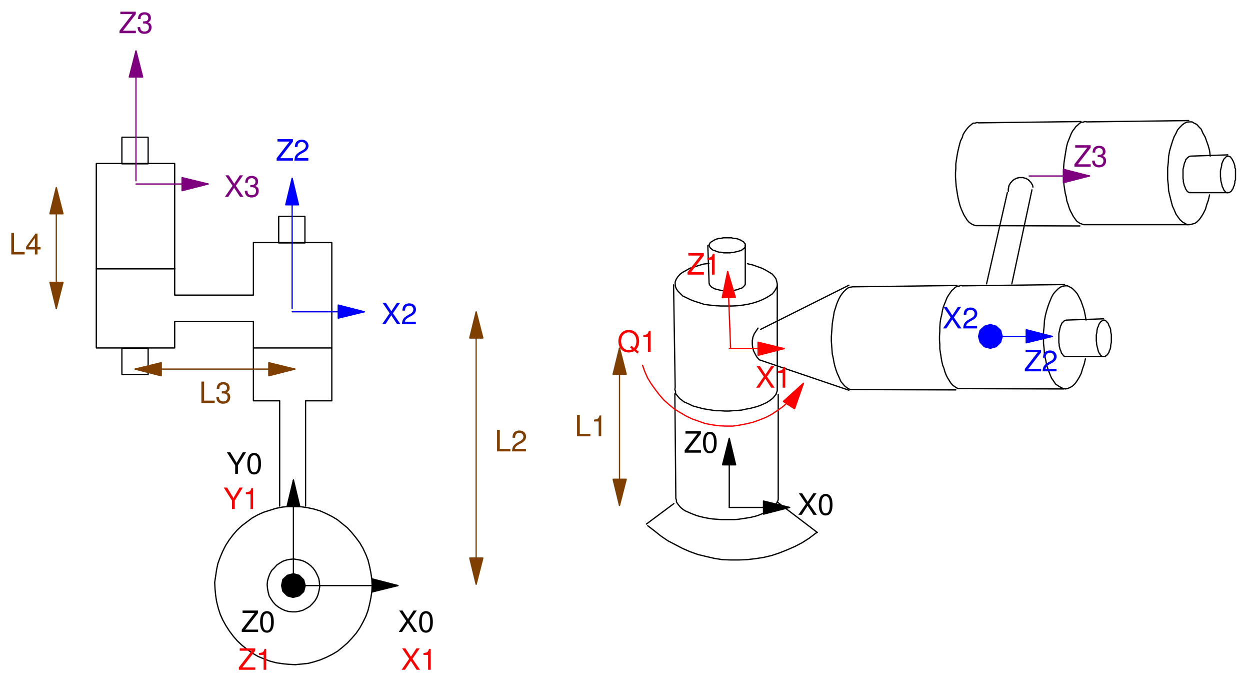
- L3 goes from 0 to 50cm, then
- Q1 goes from 0 to 180 degrees, then
- Q2 goes from 0 to 180 degrees

Code

```
TIP = RRP([0,0,0], zeros(4,1));
for i=1:50
    T = RRP([0, 0, i], TIP);
    TIP = [TIP,T];
end
for i=1:90
    T = RRP([i*pi/90, 0, 50], TIP);
    TIP = [TIP,T];
end
for i=1:90
    T = RRP([pi, i*pi/90, 50], TIP);
    TIP = [TIP,T];
end
```

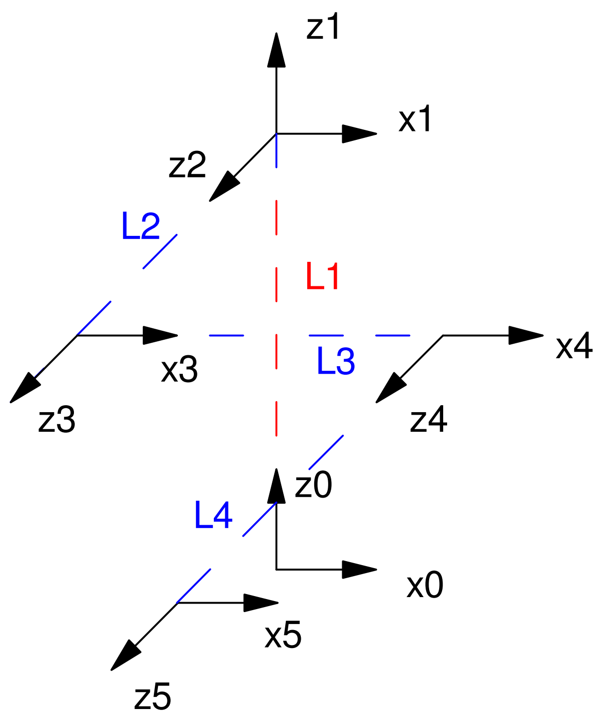


Example 3: Another RRR Robot (RRR3)



Top View - Zero Position

It helps to redraw in zero position with the reference frames. It's not necessary, but I found it easier to add five frames

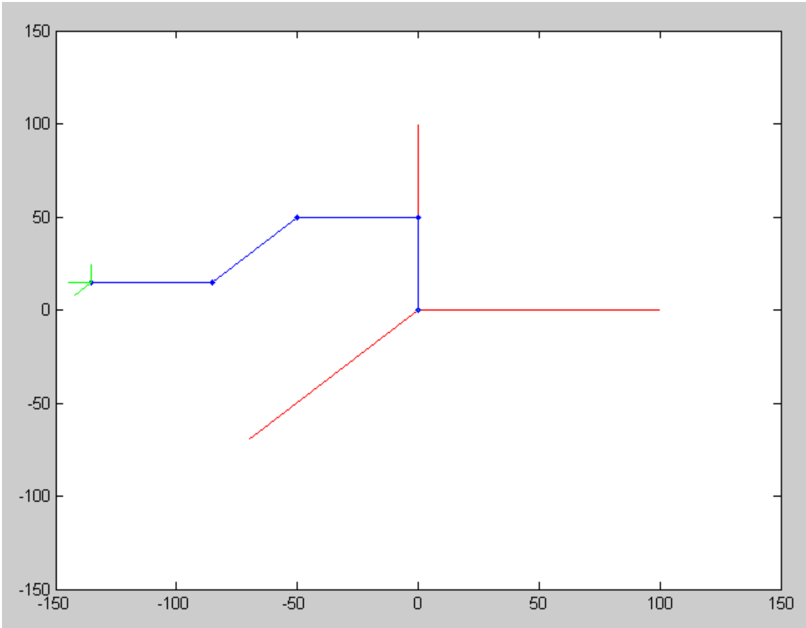


Reference frames in zero position for an RRR robot

Link i	α_{i-1} The angle between the Zi-1 and Zi axis (twist)	a_{i-1} The distance from Zi-1 to Zi measured along the Xi-1 axis	d_i The distance from Xi-1 to Xi measured along the Zi axis	θ_i The angle between Xi-1 and Xi measured about the Zi axis
1	0	0	L1	Q1
2	pi/2	0	0	0
3	0	0	L2	Q2
4	0	L3	0	0
5	0	0	L4	Q3

In zero position, the robot is at

```
RRR3([0,0,0],zeros(4,1))

    50
  -100
    50
     1
```



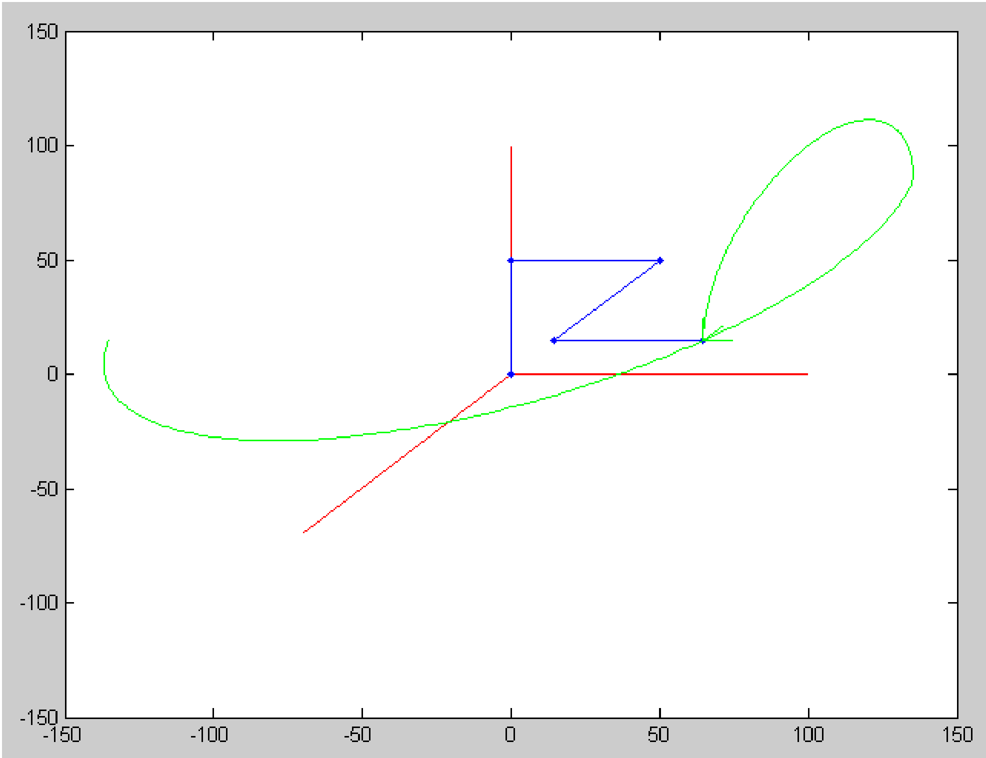
Zero position for the RRRv3 robot

Moving each joint:

- Q1 goes from 0 to 180 degrees, then
- Q2 goes from 0 to 180 degrees, then
- Q3 goes from 0 to 180 degrees

Code:

```
TIP = RRR3([0,0,0], zeros(4,1));
for i=1:180
    T = RRR3([i*pi/180, 0, 0], TIP);
    TIP = [TIP,T];
end
for i=1:180
    T = RRR3([pi, i*pi/180, 0], TIP);
    TIP = [TIP,T];
end
for i=1:180
    T = RRR3([pi, pi, i*pi/180], TIP);
    TIP = [TIP,T];
end
```



Tip trajoectory for the RRPv3 robot