EXAMEN PARCIAL 2

Miguel Ángel Bucio Macías /A01633021

Algoritmo de Cinemática directa

Para poder resolver el problema de cinemática directa se dividió el plano cartesiano de 3 dimensiones que puede alcanzar el robot en dos planos de 2 dimensiones, un plano en donde el eje y se encuentra conformado por la coordenada “Z” y el eje x se encuentra formado por la magnitud de los vectores del plano xy, es decir “$p = \sqrt{x^2 y^2}$”.

El segundo plano se encontrará formado por la coordenada “X” en el eje “Y” y la coordenada “Y” en el eje x

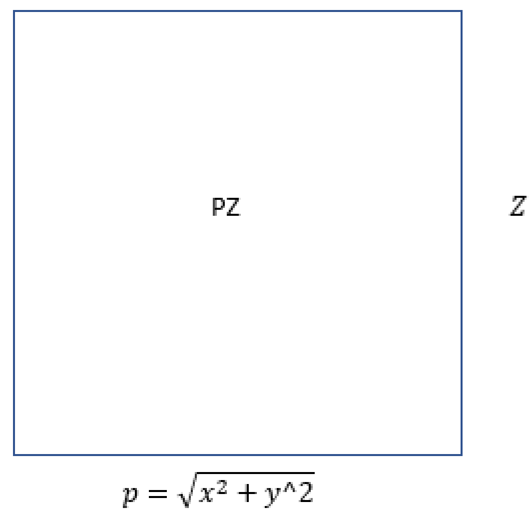


Figura 1. Plano PZ

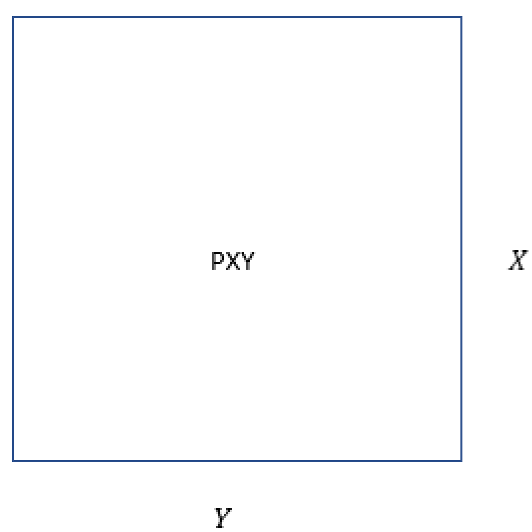


Figura 2. Plano PXY

Primero definimos las variables que utilizaremos para encontrar la cinemática directa, en el caso de la programación del algoritmo todas las variables serán de tipo float

- Ang1
- Ang2
- Ang3
- Ang4
- Ang5
- D1
- D2
- D3
- D4
- D5
- D6
- L1
- L2
- X
- Y
- Z

Primero inicializamos las variables conocidas las cuales son

- Ang1 = coordenada angular unión 1
- Ang2 = coordenada angular unión 2
- Ang3 = coordenada angular unión 3
- L1 = Longitud brazo del robot (longitud entre unión 2 y 3)
- L2 = Longitud antebrazo del robot (longitud entre unión 3 y efector final)
- D2 = Altura del robot, (Altura de unión 2 con respecto a referencia W)

Primero Analizamos el plano PZ, por lo que podemos formar los siguientes triángulos para empezar a analizarlos

Para analizarlo más fácilmente se analizará el sistema tomando como referencia global la unión 2, por lo que las coordenadas cartesianas quedarán con respecto a este punto, al final se realizan los ajustes necesarios para referenciar el punto con respecto a W

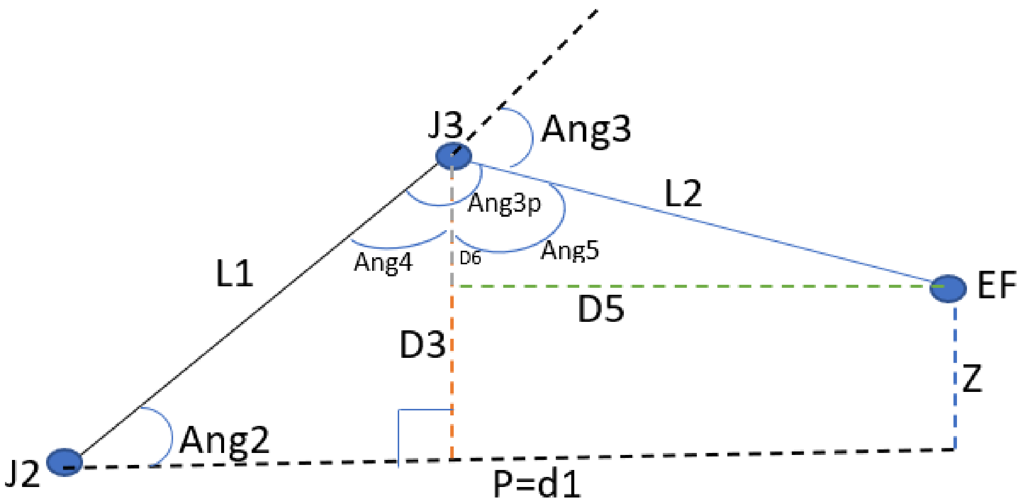


Figura 3. Triángulos a analizar en plano PZ

Nuestro robot utiliza como referencia el ángulo externo que se formaría con la extensión eje del brazo para trazar el ángulo del codo, pero es más sencillo analizar el ángulo interno que forman estos dos elementos, por lo que asignamos

$$Ang3 = 180 - Ang3$$

Primero encontramos D3 utilizando ANG2 y el triángulo rectángulo que se formó

$$D3 = \sin(Ang2) * L1$$

Posteriormente encontramos el ANG4 utilizando los otros dos ángulos de este triángulo

$$Ang4 = 180 - (Ang2 + 90)$$

Ahora encontramos ANG5 utilizando ANG3 y ANG4

$$ANG5 = ANG3 - ANG4$$

A continuación se encuentra D6, utilizando trigonometría mediante ANG5 y su triangulo rectángulo

$$D6 = \cos(ANG5) * L2$$

Encontramos la coordenada en Z utilizando las distancias D3 y D6 previamente calculadas

$$Z = D3 - D6$$

Procedemos a encontrar D4 y D5 utilizando trigonometría

$$D4 = \cos(ANG2) * l1$$

$$D5 = \sin(ANG5) * l2$$

Encontramos el último parámetro de este plano, el cual es la D1, que equivale a la magnitud de los vectores del plano XY

$$D1 = D4 + D5$$

Z ya es la coordenada final del efector, pero hay que considerar un caso en el que Z=0, lo que significaría que el efector se encuentra sobre el plano horizontal, por lo que no será necesario analizar el plano XY, se pueden utilizar los puntos calculados en este plano para encontrar “X” y “Y”, lo único que tenemos que hacer es sustituir “Z” por “X” y “P” por “Y”

$$X = Z$$

$$Y = D1$$

$$Z = 0$$

En caso de que Z sea diferente de 0 será necesario analizar los triángulos sobre el plano XY

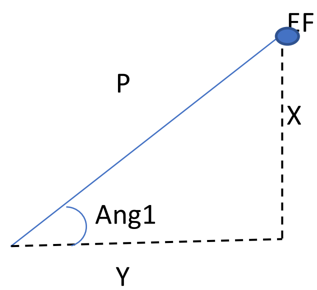


Figura 4. Triángulos a analizar en plano PZ

Encontramos “X” e “Y” mediante relaciones trigonométricas

$$x = \sin(ANG1) * D1$$

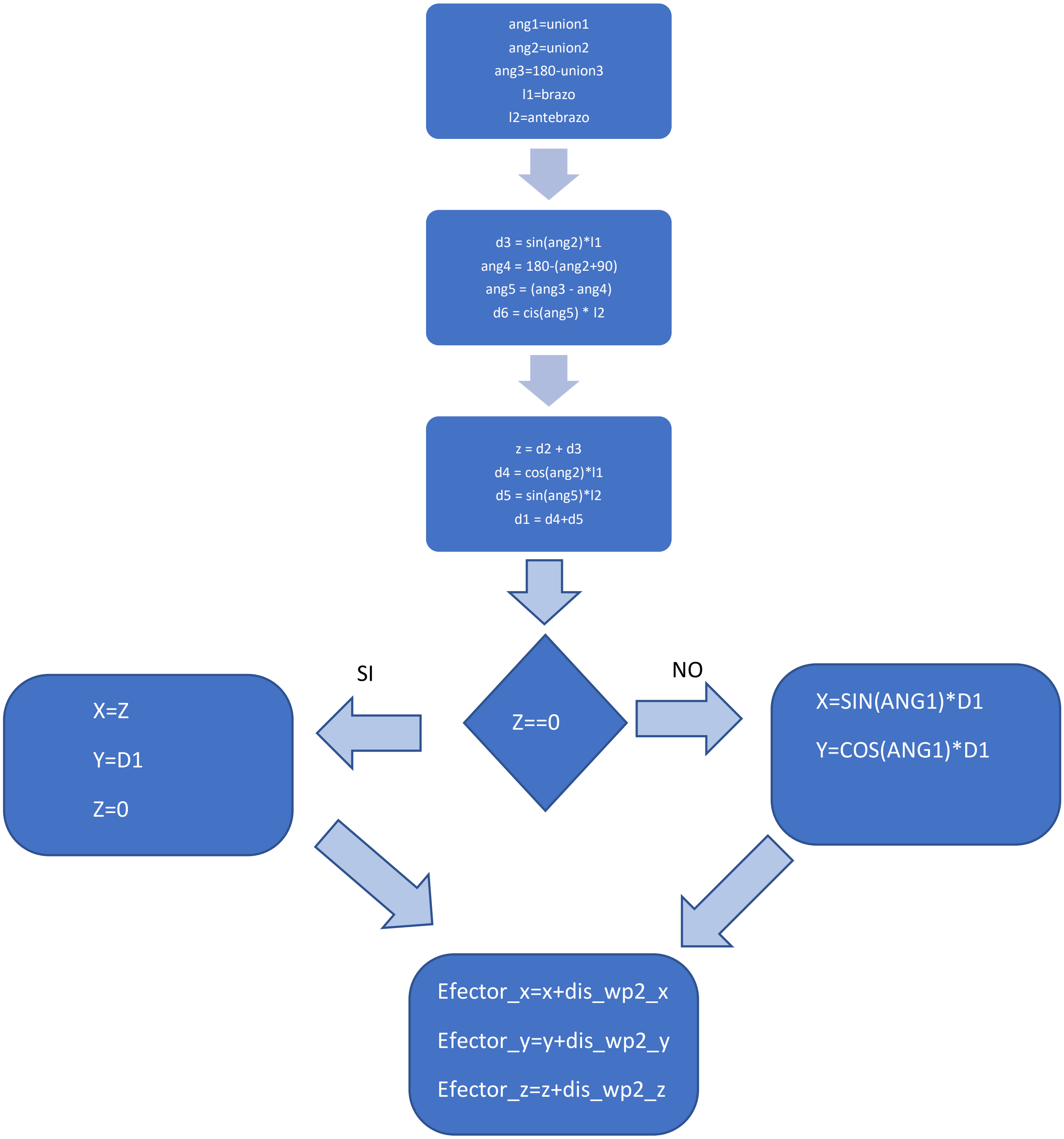
$$y = \cos(ANG1) * D1$$

Por último, agregamos las distancias entre la unión 2 y la referencia W para obtener las coordenadas del efector final con respecto a W

$$efector_x = x + dis_{w\_}p2_x$$

$$efector_y = y + dis_{w\_}p2_y$$

$$efector_z = z + dis_{w\_}p2_z$$


ang1=union1
ang2=union2
ang3=180-union3
l1=brazo
l2=antebrazo
d3 = sin(ang2)*l1
ang4 = 180-(ang2+90)
ang5 = (ang3 - ang4)
d6 = cis(ang5) * l2
z = d2 + d3
d4 = cos(ang2)*l1
d5 = sin(ang5)*l2
d1 = d4+d5
Z==0
SI
NO
X=Z
Y=D1
Z=0
X=SIN(ANG1)*D1
Y=COS(ANG1)*D1
Efector_x=x+dis_wp2_x
Efector_y=y+dis_wp2_y
Efector_z=z+dis_wp2_z

Algoritmo cinemática inversa

Para realizar la cinemática inversa se realiza la misma estrategia que para la cinemática directa de dividir el sistema en 2 planos, uno para el plano Z y otro para el plano XY, los mismos planos que en la cinemática directa.

Las variables a utilizar en este procedimiento son:

- R
- X
- Y
- Z
- Base
- Elbow
- Shoulder
- L1
- L2
- Beta
- Alfa
- Gamma

Lo primero a realizar es la asignación de las variables conocidas, las cuales son

- L1 = brazo (longitud union2 a union3)
- L2 = antebrazo(Longitud union3 a efector final)
- X = efector_x
- Y = efector_y
- Z = efector_z

Posteriormente restamos la distancia de W a la unión 2 para tener como referencia la unión 2 y realizar los cálculos más fácilmente

- X = X-dist_wp2_x
- Y = Y-dist_wp2_y
- Z = Z-dist_wp2_z

En este caso primero analizaremos el triángulo del plano XY

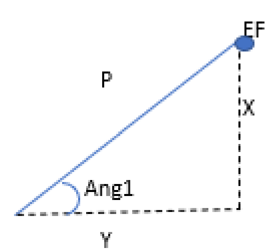


Figura 5. Triángulo a analizar plano XY

Obtenemos el ángulo de la unión 1 mediante trigonometría

$$Base = \operatorname{atan}\left(\frac{x}{y}\right)$$

Procedemos a analizar los triángulos del plano PZ

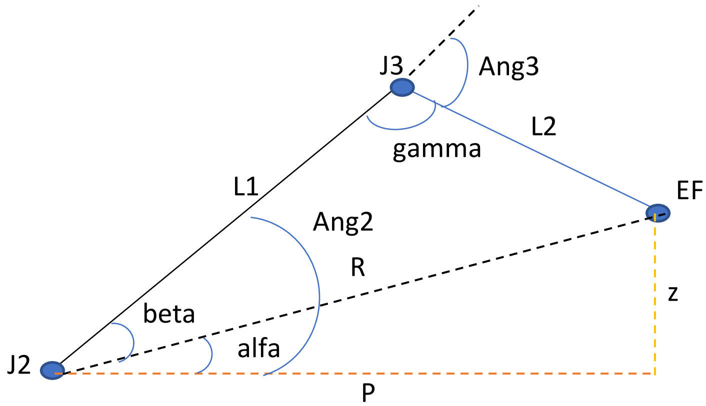


Figura 6. Triángulo a analizar en plano PZ

Primero obtenemos el radio del efector

$$R = \sqrt{x^2 + y^2 + z^2}$$

También consideramos un case en el que z es igual a 0, lo que significa que el punto se encuentra sobre el plano horizontal, por lo que asignamos el ángulo de la base a 90 para que realice recorridos horizontales e igualamos z a x

$$z = x$$

$$base = 90$$

Posteriormente calculamos alfa utilizando trigonometría

$$alfa = \operatorname{asin}\left(\frac{z}{r}\right)$$

A continuación, calculamos Beta y gamma utilizando el teorema del coseno

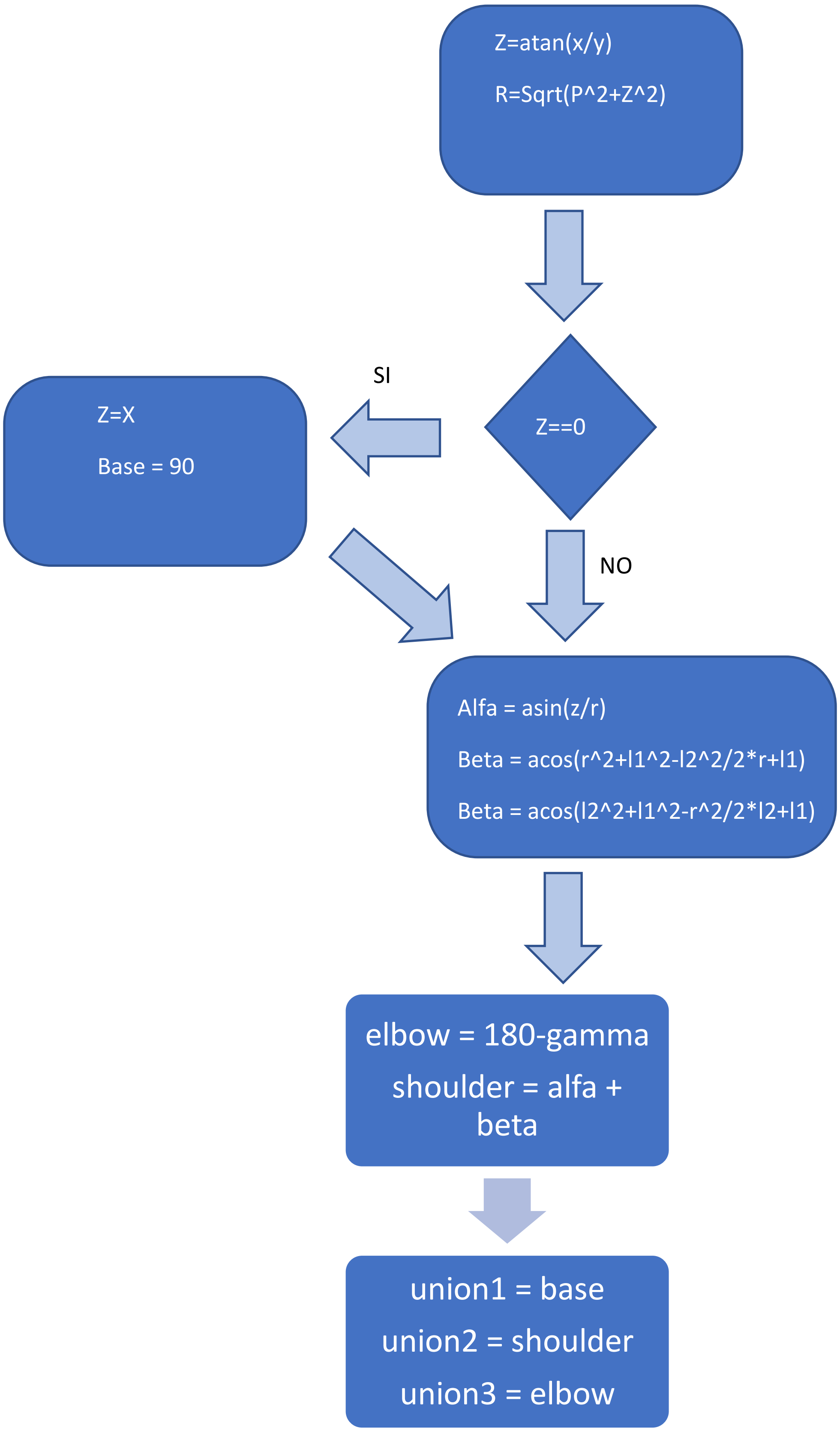
$$beta = \operatorname{acos}\left(\frac{r^2 + l1^2 - l2^2}{2 * r * l1}\right)$$

$$gamma = \operatorname{acos}\left(\frac{l2^2 + l1^2 - r^2}{2 * l1 * l2}\right)$$

Por último, calculamos los ángulos del codo y hombro con los complementos de los ángulos beta, alfa y gamma

$$shoulder = ANG2 = alfa + beta$$

$$elbow = ANG3 = 180 - gamma$$


Z=atan(x/y)
R=Sqrt(P^2+Z^2)
Z==0
SI
Z=X
Base = 90
NO
Alfa = asin(z/r)
Beta = acos(r^2+l1^2-l2^2/2*r+l1)
Beta = acos(l2^2+l1^2-r^2/2*l2+l1)
elbow = 180-gamma
shoulder = alfa + beta
union1 = base
union2 = shoulder
union3 = elbow

Para probar estos algoritmos primero elimine las distancias de la unión w a la unión 2 para tener una referencia sencilla, además coloque la distancia del brazo y antebrazo en 5cm cada una para observar más fácilmente los movimientos del brazo.

Estos fueron los resultados

Testeo sobre eje vertical únicamente

```
float test_x[num_tests] = {10,   5,      0,  -7.07,      -10};
float test_y[num_tests] = {0,   6.5,   10, 7.07,   0};
float test_z[num_tests] = {0,  0,   0,  0, 0};
```

```
Point 0
1: 90 2: 90 3: 0
x: 10 y: 2.69235e-08 z: 0
1: 90 2: 90 3: 0
Point 1
1: 90 2: 72.4777 3: 69.8182
x: 5 y: 6.5 z: 0
1: 90 2: 72.4777 3: 69.8182
Point 2
1: 90 2: 0 3: 0
x: -8.97448e-09 y: 10 z: 0
1: 90 2: -5.142e-08 3: 0
Point 3
1: 90 2: -44.0046 3: 1.99115
x: -7.07002 y: 7.06998 z: 0
1: 90 2: -44.0047 3: 1.99115
Point 4
1: 90 2: -90 3: 0
x: -10 y: 8.97448e-09 z: 0
1: 90 2: -90 3: 0
Error promedio
x: 3.05355e-06
y: 3.04458e-06
z: 0
```

Testeo sobre eje horizontal únicamente

```
float test_x[num_tests] = {0,   0,      0,  0,      0};
float test_y[num_tests] = {0,   7.07,   10, 7.07,   0};
float test_z[num_tests] = {10,  7.07,   0,  -0.707, -10};
```

```
Point 0
1: 0 2: 90 3: 0
x: 0 y: 2.69235e-08 z: 10
1: 0 2: 90 3: 0
Point 1
1: 0 2: 45.9954 3: 1.99115
x: 0 y: 7.07002 z: 7.06998
1: 0 2: 45.9953 3: 1.99115
Point 2
1: 90 2: 0 3: 0
x: -8.97448e-09 y: 10 z: 0
1: 90 2: -5.142e-08 3: 0
Point 3
1: 0 2: 39.0117 3: 89.4445
x: 0 y: 7.07 z: -0.707001
1: 0 2: 39.0117 3: 89.4445
Point 4
1: 0 2: -90 3: 0
x: 0 y: 8.97448e-09 z: -10
1: 0 2: -90 3: 0
Error promedio
x: 1.7949e-09
y: -3.53577e-06
z: 3.48091e-06
```

Testeo diagonal

```
float test_x[num_tests] = {0,   3,      7.07,  -3,      0};
float test_y[num_tests] = {0,   3,   7.07, 3,   0};
float test_z[num_tests] = {10,  7.07,   0,  -0.707, -10};
```

```
Point 0
1: 0 2: 90 3: 0
x: 0 y: 2.69235e-08 z: 10
1: 0 2: 90 3: 0
Point 1
1: 45 2: 93.4916 3: 68.9184
x: 3 y: 3 z: 7.07
1: 45 2: 93.4916 3: 68.9184
Point 2
1: 90 2: 45.9954 3: 1.99115
x: 7.06998 y: 7.07002 z: 0
1: 90 2: 45.9953 3: 1.99115
Point 3
1: -45 2: 55.0643 3: 129.05
x: -3 y: 3 z: -0.707
1: -45 2: 55.0642 3: 129.05
Point 4
1: 0 2: -90 3: 0
x: 0 y: 8.97448e-09 z: -10
1: 0 2: -90 3: 0
Error promedio
x: 3.14713e-06
y: -3.44041e-06
z: 1.43051e-07
```

Posteriormente se configuraron los parámetros reales del robot, se probaron puntos sobre los ejes y en el espacio 3D

```
float test_x[num_tests] = {0+incx,        0+incx,        0+incx,              0+incx,          total+incx,
float test_y[num_tests] = {0+incy,        total+incy,    total*0.707+incy,    0+incy,          0+incy,
float test_z[num_tests] = {total+altura,  0+altura,      -total*0.707+altura, -total+altura,   0+altura,
```

```
total+incx,      0+incx,              -total+incx,   7,   7,    -7  };
0+incy,          total+incy,          0+incy,        15,  15,   15};
0+altura,        0*0.707+altura,      0+altura,       7,  -7,   7};
```

```
Point 0
1: 0 2: 90.028 3: 0.039566
x: -1.161 y: 8.84838 z: 25.3737
1: -0 2: 90.028 3: 0.039566
Point 1
1: 90 2: 0.0279765 3: 0.039566
x: -1.15938 y: 32.8667 z: 1.357
1: 90 2: 0.0318331 3: 0.039566
Point 2
1: 0 2: 47.4438 3: 71.3401
x: -1.161 y: 28.5769 z: 2.33248
1: 0 2: 47.4439 3: 71.3401
Point 3
1: 0 2: 269.966 3: 180
x: -1.161 y: 8.85315 z: 6.6223
1: 0 2: 269.966 3: 180
Point 4
1: 90 2: 90.028 3: 0.039566
x: 22.8557 y: 8.84838 z: 1.357
1: 90 2: 90.028 3: 0.039566
```

```
Point 5
1: 90 2: 0.0279765 3: 0.039566
x: -1.15938 y: 32.8667 z: 1.357
1: 90 2: 0.0318331 3: 0.039566
Point 6
1: 90 2: 270 3: 180
x: 4.1043 y: 8.85 z: 1.357
1: 90 2: 270 3: 180
Point 7
1: 52.9989 2: 116.339 3: 127.202
x: 7 y: 15 z: 7
1: 52.9989 2: 116.339 3: 127.202
Point 8
1: 52.9989 2: 39.5875 3: 117.79
x: 7 y: 15 z: -7
1: 52.9989 2: 39.5875 3: 117.79
Point 9
1: -43.5141 2: 130.465 3: 136.306
x: -7 y: 15 z: 7
1: -43.5141 2: 130.465 3: 136.306
Error promedio
x: -0.000323367
y: 8.96454e-06
z: 5.72205e-07
```

Screenshots de comprobación en solid Works en carpeta data