Description: http://www.siga.frba.utn.edu.ar/imag/utnba.png

**ROBOTICA**

**Proyecto robot SCARA**

Curso: R6055

Año: 2013

**Docente:**

Ing. Hernan Giannetta

**Ayudante:**

Ing. Damian Granzella

**Alumnos:**

Ignacio Tamayo

Javier Garbini

**ÍNDICE**

[Revision 3](#_Toc358051787)

[1 Introducción 4](#_Toc358051788)

[2 Memoria descriptiva 4](#_Toc358051789)

[Funcionamiento 4](#_Toc358051790)

[Modelo adoptado 4](#_Toc358051791)

[Medidas del robot 5](#_Toc358051792)

[3 Cinematica del robot 5](#_Toc358051793)

[3.1 Cinemática directa 5](#_Toc358051794)

[Algoritmo Denavit-Hartenberg 5](#_Toc358051795)

[Desplazamiento de los ejes 5](#_Toc358051796)

[Coordenadas articulares válidas 7](#_Toc358051797)

[3.2 Cinemática Inversa 8](#_Toc358051798)

[Selección de método 8](#_Toc358051799)

[Método geométrico 9](#_Toc358051800)

[Ecuaciones 10](#_Toc358051801)

[4 Espacios de trabajo 12](#_Toc358051802)

[Proyecciones sobre los planos 13](#_Toc358051803)

[Generación de trayectorias articulares 13](#_Toc358051804)

[5 Conclusiones 17](#_Toc358051805)

# Revision

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Nombre | Descripción | Versión |
| 03/06/2013 | TP1 | Cinemática Directa e Inversa | 1.0 |
|  |  |  |  |
|  |  |  |  |

# Introducción

A fin de cumplimentar los requerimientos de cursada para para cátedra de la asignatura electiva de 6° nivel Robótica se realiza la presentación del Proyecto un robot Scara con 5 grados de libertad.

Se toma como base el trabajo realizado en el año 2012 con esta misma arquitectura, agregando un grado más de libertad.

# Memoria descriptiva

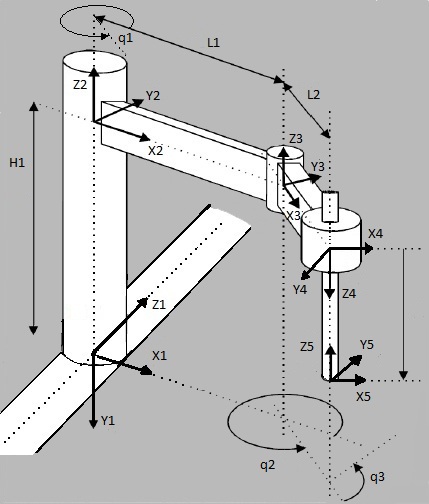
### Funcionamiento

SCARA es un tipo de configuración de robot manipulador las siglas en inglés significan Selective Compliance Assembly Robot Arm. Este robot fue creado por un grupo de industrias electrónicas japonesas, en colaboración con dos universidades, para insertar los componentes de forma vertical.

La configuración seleccionada para realizar el proyecto está formada por dos articulaciones de rotación con respecto a dos ejes paralelos entre sí y perpendiculares al plano de trabajo, dos de desplazamiento una dirección paralela a la de los ejes de rotación y la otra dirección en sentido perpendicular a los ejes de rotación.

### Modelo adoptado

Para el proyecto de la cátedra se eligió un SCARA con el agregado de un grado más de libertad, permitiendo que el robot se desplace linealmente sobre el eje Y del sistema de referencia inicial, como se muestra en la siguiente figura:



### Medidas del robot

H1 = 25 cm Altura de la barra.

L1 = 30 cm Longitud de la primera articulación.

L2 = 20 cm Longitud de la segunda articulación.

# Cinematica del robot

## Cinemática directa

### Algoritmo Denavit-Hartenberg

Para realizar el cálculo de la cinemática directa utilizaremos el algoritmo de Denavit-Hartenberg, el cual define los movimientos de las articulaciones de la siguiente manera:

* Rotación alrededor del eje zi-1 un ángulo θi
* Traslación a lo largo de zi-1 una distancia di ; vector di (0,0,di).
* Traslación a lo largo de xi una distancia ai ; vector ai (0,0,ai).
* Rotación alrededor del eje xi un ángulo αi.

### Desplazamiento de los ejes

Se muestra el gráfico se la secuencia de sistemas para el algoritmo

Art1 d1

Art2 q2

Art3 q3

Art4 d2

Art5 q3

S0

X0

Z0

Y0

S1

Y1

X1

Z1

S2

X2

Y2

Z2

S3

X3

Y3

Z3

S4

X4

Z4

Y4

S5

X5

Y5

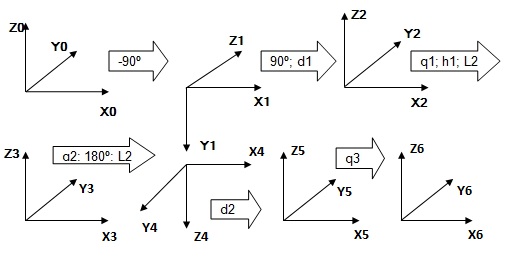
Z5

S6

X6

Z6

Y6



Para el caso de nuestro robot los parámetros de la matriz son los siguientes:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **θ** | **d** | **a** | **α** |
| **T0** | -90 ° | 0 | 0 | 0 |
| **T1** | 90 ° | 0 | 0 | **d1** |
| **T2** | 0 | L1 | **q1** | H1 |
| **T3** | 180 ° | L2 | **q2** | 0 |
| **T4** | -180 ° | 0 | 0 | **d2** |
| **T5** | 0 | 0 | **q3** | 0 |

Se tiene una aparente articulación extra (6 filas en lugar de 5) pero esta articulación sirve solo para orientar el eje de coordenadas en el sentido del desplazamiento de la primera articulación.

Por medio de Matlab se obtiene la siguiente matriz de transformación homogénea:

**Matriz DH del robot**

[cos(q1+q2), -sin(q1+q2), 0, L2\*cos(q1+q2)+L1\*cos(q1) ]

[sin(q1+q2), cos(q1+q2), 0, L2\*sin(q1+q2)+L1\*sin(q1)+d1 ]

[0, 0, 1, H1-d2 ]

[0, 0, 0, 1 ]

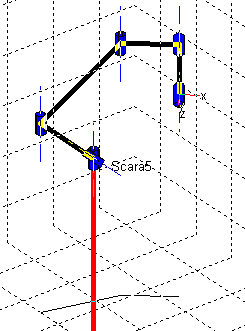
También usando MatLab se obtienes su matriz Jacobiana correspondiente

[ 0, -L2\*sin(q1+q2)-L1\*sin(q1), -L2\*sin(q1+q2), 0 ]

[ 1, L2\*cos(q1+q2)+L1\*cos(q1), L2\*cos(q1+q2), 0 ]

[ 0, 0, 0, -1 ]

Se valida el diseño y las matrices obtenidas usando el Toolbox Corke, del cual se valida el modelo obtenido



L0 = link([-pi/2 0 0 0]) ;

L1 = link([pi/2 0 0 0 1]);

L2 = link ([0 l1 0 h1]);

L3 = link([pi l2 0 0]);

L4 = link([-pi 0 0 0 1]);

L5 = link([0 0 0 0]);

Scara5 = robot({L0,L1,L2,L3,L4,L5},'Scara5');

drivebot(Scara5);

### Coordenadas articulares válidas

Usando la matriz Jacobiana obtenida, se usa un proceso recursivo para encontrar las coordenadas articulares [d1 q1 q2 d2 q3] que causan que el determinante del Jacobiano sea nulo, y son estas las coordenadas articulares que se deben evitar

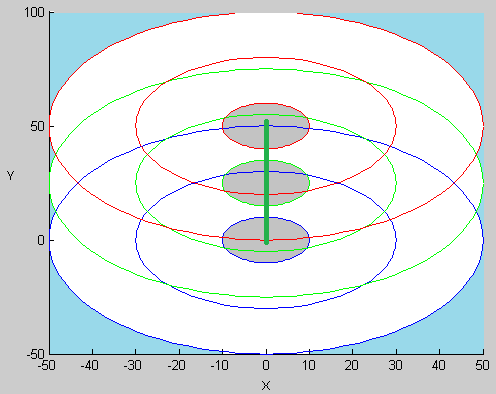
Al no ser el Jacobiano una matriz cuadrada, se usa el siguiente determinante

Det( inv(Jacob\*Jacob') )

Los límites del espacio Externos son los puntos que exceden el radio de los brazos del robot estirados por completo

Los límites del espacio Internos de encuentran son para todo q1, con q2 entre **175-190** grados. Es decir, cuando el segundo brazo se rebate sobre el primero. Estos valores se obtienen del proceso iterativo de anular el determinante.

Estos espacios terminan generando un círculo interior y exterior que no alcanza el robot y son los límites de la zona de trabajo. La siguiente figura muestra los círculos de las zonas de trabajo para distintos valores de desplazamiento d1 sobre el eje Y (vista superior, plano XY)



Sin embargo, los círculos interiores se pueden alcanzar de maneras más cómodas desplazando la articulación d1 de manera que el punto espacial sea alcanzable con q2 fuera de los ángulos críticos

En esta premisa se basa el algoritmo cinemático inverso usado para nuestro robot Scara

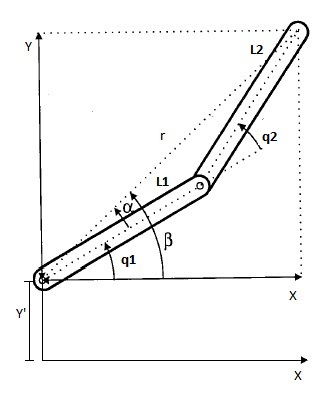
## Cinemática Inversa

### Selección de método

Para el cálculo de la cinemática inversa se utilizó el método geométrico, ya que resulta ser más sencillo de implementar. Como el robot posee un grado de libertad sobre el plano XY es muy tedioso utilizar la matriz de transformación inversa incluyendo esta articulación y a los fines didácticos el método geométrico es mucho más académico al momento de analizarlo.

### Método geométrico

Vista superior del plano XY:



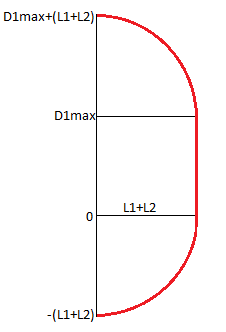
Como se puede observar en la figura el robot se desplaza Y’ para llegar al punto deseado, realizando el anélisis geométrico obtenemos las siguientes ecuaciones:

Analíticamente el desplazamiento Y’ es muy difícil despejar por lo que se eligió para el proyecto trabaja con diferentes zonas de trabajo, dependiendo de la ubicación del punto (px, py, pz).

Los límites de la zona se definieron de la siguiente manera:

De 0 a D1max es el desplazamiento máximo que realiza el robot sobre el eje Y, llegando con las 2 articulaciones extendidas a las posiciones [0-(L1+L2)] y [D1max +(L1+L2)] sobre dicho eje

Siendo las dimensiones del robot L1 + L2 = D1max = 50 cm se tiene una zona de trabajo principal de geometría cuadrada. Se tiene acceso a los puntos de la zona cuadrada de 50cm x 50cm, más un cuarto de circunferencia de radio 50cm sobre esta área y otro de iguales características en la parte inferior. En la siguiente figura se observa la zona de trabajo sobre el plano XY.



### Ecuaciones

Para el cálculo de las ecuaciones se parte de las coordenadas donde se quiere posicionar el extremo del robot (px; py; pz).

Como primer aproximación se determina la posición sobre el eje “y” dependiendo de los valores del punto.

* Para tener el primer brazo flexionado hacia arriba (q1 > 0)

Si py >= D1max+L1: Se desplaza al robot sobre el eje “y” hasta el punto D1max.

Sino Si py >= L1: Se desplaza al robot sobre el eje “y” hasta el punto py – L1.

Sino: Se desplaza al robot sobre el eje “y” hasta el punto 0

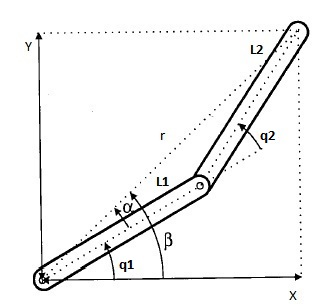
* Para tener el primer brazo flexionado hacia abajo (q1 < 0)

Si py <= -L1: Se desplaza al robot sobre el eje “y” hasta el punto 0.

Sino Si py <= D1max -L1: Se desplaza al robot sobre el eje “y” hasta el punto py + L1.

Sino: Se desplaza al robot sobre el eje “y” hasta el punto D1max

Con estos posicionamientos se logran las ecuaciones geométricas sin tener en cuenta el desplazamiento Y’ siendo de este modo mucho más simple el análisis de la cinemática inversa.



Para el cálculo de q2

Por el teorema del coseno sabemos que r es:

Sabiendo

Siendo

Para el cálculo de q1

)-

De esta manera se tiene el algoritmo cinemático inverso, que a partir de una posición (px; py; pz) entrega las coordenadas articulares [d1 q1 q2 d2]

Adicionando un cálculo más, el algoritmo entrega la coordenada articular q3 para obtener una orientación α en la última articulación

# Espacios de trabajo

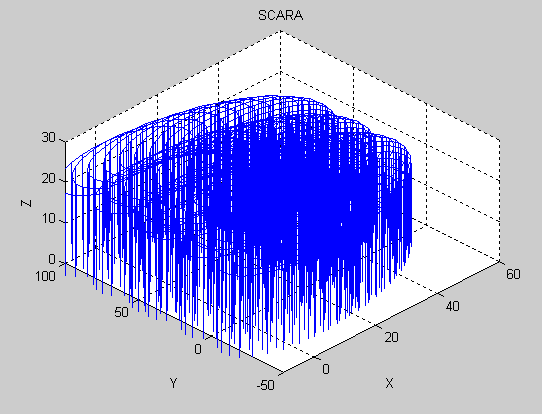
Considerando que los giros de las articulaciones están físicamente limitados por los motores que los van a realizar, además de evitar las coordenadas articulares inválidas antes halladas, se le definen excursiones máximas a las articulaciones y se itera para tener el espacio de trabajo del robot SCARA

Articulación d1: [0 a D1max] cm, Valor de D1max = 50 cm

Articulación d2: [0 a H1] cm, Valor de H1 = 25 cm

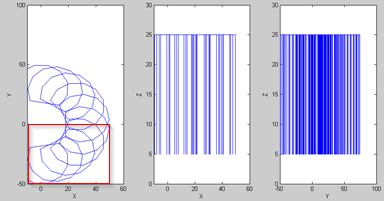
Articulaciones q1,q2,q3: [-145° a 145°]

A continuación se muestra el espacio de trabajo para 3 valores de la articulación d1, sobre el eje Y

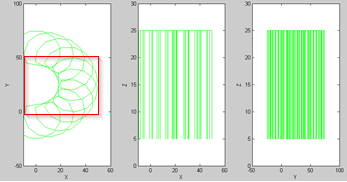


### Proyecciones sobre los planos

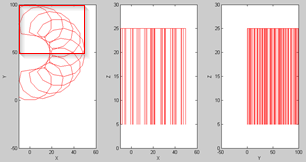
Zona inferior de 0 a -(L1+L2)



Zona media de 0 a D1max



Zona superior de D1max a D1max+(L1+L2)



### Generación de trayectorias articulares

Para poder controlar efectivamente el Robot, se necesita además del algoritmo cinemático una determinada forma de atravesar en el tiempo las coordenadas indicadas. Para nuestro robot SCARA se usa un generador de trayectorias polinómicas de 3er orden, de manera de evitar tener aceleraciones articulares bruscas.

Si usásemos una trayectoria lineal, efectivamente las aceleraciones tendrían valores infinitos en cada arranque de movimiento y esto exigiría demasiado a los motores.

Se implementa el siguiente flujo para la generación de trayectorias y las curvas temporales de velocidad y aceleraciones articulares que los motores deben proveer para conseguir el movimiento deseado

Trayectoria espacial

Espacio de trabajo

Trayectoria espacial alcanzable

Algoritmo de cinemática inversa

Valores articulares reales

Trayectoria articular

Matriz Transferencia

Trayectoria espacial real

Velocidades articulares

Matriz Jacobiana

Velocidades espaciales

Generamos las siguientes coordenadas a las que debe acceder el robot

[X Y Z] coordenada espacial para el muñón del robot

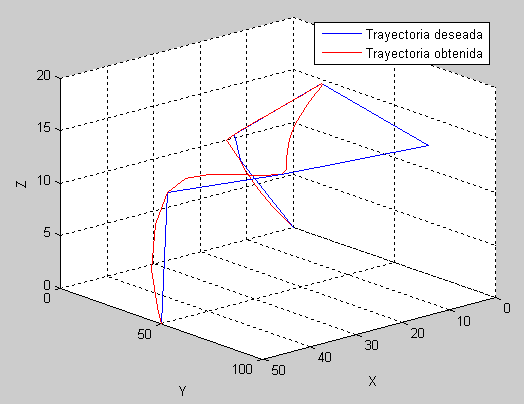
A ángulo de orientación que se desea dar al muñón

T tiempo donde ocurre la coordenada

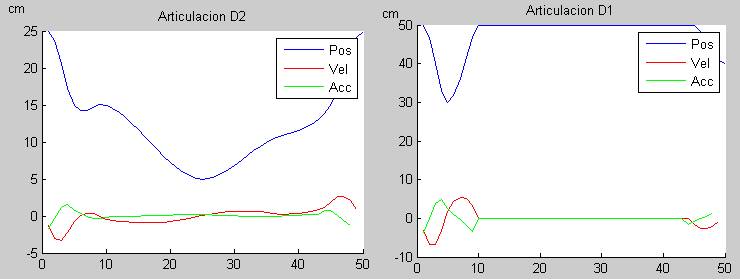
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X | 50 | 40 | 20 | 10 | 20 | 30 | 20 | 0 |
| Y | 50 | 30 | 40 | 90 | 60 | 40 | 20 | 0 |
| Z | 0 | 10 | 10 | 15 | 20 | 15 | 10 | 0 |
| A | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| T | 1 | 5 | 10 | 15 | 25 | 35 | 45 | 50 |

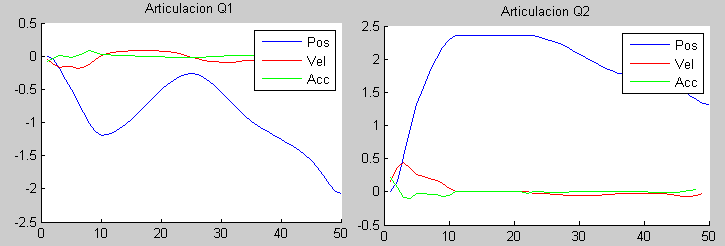
El único punto fuera del área de trabajo es (10, 90, 20). Este punto es descartado de la trayectoria

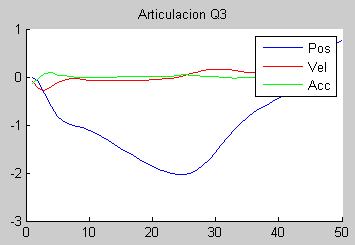
Efectivamente, el robot sigue la trayectoria deseada, salgo el punto descartado



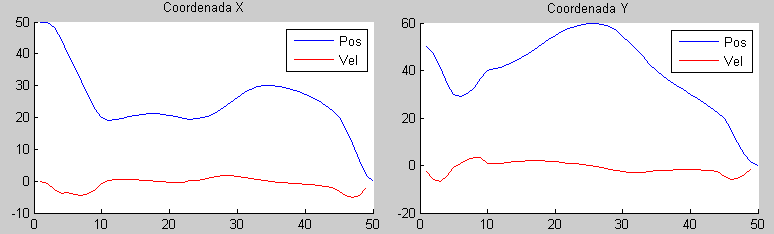
Se analizan las velocidades y aceleraciones articulares necesarias para este movimiento, donde se verifica que no hay saltos bruscos en la velocidad

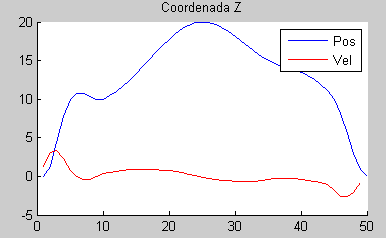






Finalmente se verifican las velocidades espaciales que adquiere el muñón del robot SCARA. Debido a la trayectoria polinomial, se obtienen velocidades sin transiciones buscas





# Conclusiones

* Se ha desarrollado adecuadamente el modelo del Robot SCARA de 5 grados de libertad, usando el algoritmo de Denavit-Hartenberg
* Se verificó el modelo con el Toolbox Corke
* Se desarrolló un algoritmo cinemático inverso
* Se usa un generador de trayectorias articulares polinomial de 3 er order, para suavizar las curvas de aceleraciones
* Efectivamente, dada una trayectoria a seguir, el algoritmo cinemático inverso es capaz de seguirlo, siempre en los puntos dentro del área de trabajo válida
* Las velocidades y aceleraciones articulares no tiene discontinuidades por ser trayectorias polinomiales
* Las velocidades espaciales no tienen discontinuidades.