ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA MECÁNICA

MODELO SIMULACIÓN Y CONTROL DE UN BRAZO ROBOTICO MEDIANTE MATLAB Y SIMULINK PARA SOLDADURA DE ARCO

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO MECÁNICO

DAVID ALEJANDRO JARAMILLO ROJAS
dajar_2000@hotmail.com

DIRECTOR: ING. IVAN ZAMBRANO
oscarz@epn.edu.ec

Quito, Febrero 2010

DECLARACIÓN

Yo, David Alejandro Jaramillo Rojas declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

David Jaramillo Rojas

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por David Alejandro Jaramillo Rojas bajo nuestra supervisión.

Ing. Iván Zambrano	Ing. Jorge Escobar	Dr. Álvaro Aguinaga
DIRECTOR DE PROYECTO	**COLABORADOR**	**COLABORADOR**

AGRADECIMIENTOS

A ti mi Dios, porque nunca me abandonaste ni en el momento más oscuro de mi vida, me devolviste el aliento y me llenaste fuerza para vencer los obstáculos y enemigos.

A mi Madre, mi apoyo, mi guía y amiga incondicional en todo momento. Gracias por tu paciencia, comprensión y cariño. Gracias a ti estoy hoy aquí.

A mi Padre, por su oportuno apoyo y leal amistad en los momentos de debilidad y necesidad durante mi vida y mi carrera. Gracias por tus enseñanzas.

A mis hermanos Francisco y René, mis aliados, amigos y confidentes cuya invaluable amistad, lealtad y auxilio en los momentos difíciles me han permitido continuar desarrollándome a lo largo de mi vida y carrera.

A Vanessa, tu humildad y sencillez me han enseñado a sobrellevar muchas cosas.

Un especial agradecimiento al Ingeniero Iván Zambrano quien tuvo mucha certeza al dirigir este proyecto de titulación y a la Escuela Politécnica Nacional, institución muy prestigiosa que forma profesionales de primera.

A mis amigos verdaderos con los que he contado a lo largo de mi carrera y me han brindado su sincera y desinteresada amistad.

DEDICATORIA

Este logro se lo dedico a Dios mi único e infalible amigo con el que cuento siempre, a mi Madre quien con sus consejos llenos de amor siempre me guiaron por el buen camino y se ha convertido en un pilar fundamental de mi existencia y a mi padre quien con sus consejos siempre supo alentarme en momentos difíciles.

RESUMEN

El presente proyecto tiene por objetivo el estudio de la ingeniería mecánica, especialización robótica como punto de partida al desarrollo de la automatización de procesos de ingeniería. Esto con el fin de mejorar los procesos de producción y manufactura que se llevan a cabo dentro del país, con tecnología que se encuentre al alcance de la mayoría de industrias.

Como primer paso se estudia los tipos de control automático adecuados para el funcionamiento de un robot de base inmóvil. Luego se investiga los fundamentos de cinemática y dinámica de robots y nociones generales sobre la generación de trayectorias. Después se procede a la revisión del toolbox SimMechanics de Simulink que será empleado durante el modelado y simulación del robot. A continuación se plantea una tarea concreta que deberá ejecutar el robot, para lo cual se empieza determinando la configuración mecánica del robot y dimensionándolo, a partir de esto se elabora el modelo cinemático y dinámico del brazo robótico, luego con ayuda del toolbox SimMechanics se crea un algoritmo que resuelve la cinemática inversa del robot para lograr la tarea propuesta. Finalmente se construyó el diagrama de bloques del sistema físico del robot en la plataforma de Simulink. En este punto se recurrió a la solución de la cinemática inversa que fue incluida en el diagrama de bloques para que el robot pueda funcionar. Al finalizar el modelado se ejecuta la simulación del robot bajo ciertos parámetros de funcionamiento donde se observó el comportamiento de este durante la simulación de la tarea inicialmente planteada. Como complemento se planteo una tarea adicional para observar el desenvolvimiento del robot en esta.

PRESENTACIÓN

La robótica se ha convertido en una poderosa y común herramienta en la producción mecánica en general, siendo la soldadura una de las áreas donde ha protagonizado con gran desempeño.

La adquisición de un equipo de soldadura robotizada es muy difícil para la mediana y pequeña industria en el Ecuador por el costo y la tecnología intrínseca a estos equipos. Sin embargo la necesidad de producir y obtener mayores ganancias económicas ha llevado a la búsqueda de nuevas alternativas que reemplacen a la mano de obra en ciertas tareas de manufactura.

En la mayoría de casos de aplicaciones de soldadura en el país un robot se adaptaría de gran manera, ya que la versatilidad de estos robots radica en gran parte en la programación que estos tienen. Además el avance de la robótica favorecería el incremento de la productividad de alimentos, ropa, calzado, aparatos electrónicos, en la elaboración de perfumes, etc., industrias que están floreciendo o que ya están establecidas en el país.

Este proyecto es el primer paso para un posterior diseño concreto de un robot soldador que pueda ser producido por la industria ecuatoriana, aprovechando la gran apertura que se le está dando a la pequeña y mediana empresa a acceder a créditos para mejoramiento de la misma.

INDICE DE CONTENIDO

CAPITULO I 1
1. CONTROL AUTOMATICO 1
1.1 INTRODUCCION AL CONTROL AUTOMATICO. 1
1.2 DEFINICIONES. 1
1.2.1 PLANTAS. 1
1.2.2 PROCESOS. 1
1.2.3 SISTEMAS 2
1.2.4 PERTURBACIONES 2
1.2.5 CONTROL DE REALIMENTACIÓN 2
1.2.6 SISTEMAS DE CONTROL REALIMENTADO 2
1.3 CONTROL DE LAZO CERRADO Y LAZO ABIERTO 3
1.3.1 SISTEMAS DE CONTROL DE LAZO CERRADO 3
1.3.2 SISTEMAS DE CONTROL DE LAZO ABIERTO 3
1.3.3 COMPARACIÓN ENTRE LOS SISTEMAS DE LAZO CERRADO Y LAZO ABIERTO 4
1.4 MODELOS MATEMATICOS DE SISTEMAS FISICOS. 5
1.4.1 MODELOS MATEMÁTICOS. 5
1.4.2 SISTEMAS LINEALES. 5
1.4.3 SISTEMAS NO LINEALES. 6
1.5 ALGEBRA DE BLOQUES 7
1.5.1 DIAGRAMA DE BLOQUES 7
1.6 CONTROL PID (Proporcional-Integral-Derivativo) 8
1.6.1 ESTRUCTURA DEL PID 9
1.6.2 P: ACCIÓN DE CONTROL PROPORCIONAL 9
1.6.3 I: ACCIÓN DE CONTROL INTEGRAL 10
1.6.4 PI: ACCIÓN DE CONTROL PROPORCIONAL-INTEGRAL 10
1.6.5 PD: ACCIÓN DE CONTROL PROPORCIONAL-DERIVATIVA 11
1.6.6 PID: ACCIÓN DE CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVA 11
1.7 CONTROL AUTOMATICO EN MANIUPULADORES ROBOTICOS. 12
1.7.1 TÉCNICA DE PAR COMPUTADO. 12
1.7.1.1 Sistemas lineales y no lineales. 12

1.7.1.2 Sistemas de control multientradas y multisalidas. 13
1.7.2 CONTROL ADAPTATIVO. 14
1.7.2.1 Sistemas de control adaptivo. 14
1.7.3 CONTROL POR APRENDIZAJE. 17
CAPITULO II 18
2. TEORIA DE MAQUINAS ORIENTADA A LA ROBOTICA……..27
2.1 INTRODUCCIÓN A LA ROBÓTICA. 18
2.2 DEFINICIÓN DE ROBOT INDUSTRIAL. 18
2.2.1 CLASIFICACIÓN DE ROBOTS. 19
2.2.2 ARQUITECTURA DE ROBOTS 20
2.3 DESCRIPCIONES ESPACIALES Y TRANSFORMACIONES 28
2.3.1 FORMULACIÓN MATRICIAL DE LAS TRANSFORMACIONES 28
2.3.1.1 Matriz de traslación 29
2.3.1.2 Matriz de rotación. 30
2.4 CINEMÁTICA DE ROBOTS 32
2.4.1 SÍNTESIS Y ANÁLISIS 32
2.4.2 VELOCIDAD LINEAL Y ROTACIONAL DE CUERPOS RÍGIDOS 32
2.4.2.1 Velocidad lineal. 33
2.4.2.2 Velocidad de rotación. 33
2.4.2.3 Velocidad lineal y de rotación simultáneas. 36
2.4.3 MOVIMIENTO DE LOS VÍNCULOS DE UN ROBOT. 36
2.4.3.1 Propagación de velocidad vínculo a vínculo. 37
2.4.4 JACOBIANOS. 40
2.4.4.1 Singularidades. 43
2.4.4.2 Clasificación de las singularidades. 43
2.4.4.2.1 Singularidades en la frontera del espacio de trabajo. 43
2.4.4.2.2 Singularidades dentro del espacio de trabajo. 44
2.5 CINEMÁTICA DIRECTA DE ROBOTS. 45
2.5.1 PARÁMETROS DE DENAVIT-HARTENBERG. 45
2.6 CINEMÁTICA INVERSA DE ROBOTS 47
2.7 DINAMICA DE ROBOTS 48
2.7.1 ACELERACIÓN DE UN CUERPO RÍGIDO 49

2.7.1.1 Aceleración lineal. 49
2.7.1.2 Aceleración angular. 51
2.7.2 DISTRIBUCIÓN DE LA MASA 52
2.7.3 FORMULACIÓN LAGRANGIANA DE LA DINÁMICA DE MANIPULADORES 54
CAPITULO III 58
3. PROGRAMACION Y USO DEL ENTORNO DE MATLAB Y SIMULINK……..58
3.1 INTRODUCCION AL SIMULINK 58
3.1.1 GENERALIDADES DE SIMULINK 58
3.1.2 SIMMECHANICS 59
3.1.2.1 Cuerpos (*Bodies*). 60
3.1.2.2 Uniones, juntas o pares cinemáticos (*Joints*). 61
3.1.2.3 Sensores y actuadores (*Sensors and actuators*). 62
3.1.3 ELABORACIÓN DE UN MODELO DINÁMICO SENCILLO: EL PÉNDULO SIMPLE. 62
3.1.3.1 Configuración del bloque de bancada *Ground*. 69
3.1.3.2 Configuración bloque *Revolute*. 69
3.1.3.3 Configuración elemento *Body*. 70
3.1.3.4 Configuración elemento *Initial condition*. 73
3.1.3.5 Configuración bloque *Machine environment*. 74
3.1.4 TOOLBOX HEMERO 78
3.1.4.1 Comandos del toolbox Hemero. 79
CAPITULO IV 83
4. MODELAMIENTO DE UN BRAZO ROBOTICO CON 4 GRADOS DE LIBERTAD EN EL ENTORNO DE MATLAB PARA SOLADURA DE ARCO, Y DISEÑO DEL SISTEMA DE CONTROL AUTOMÁTICO 83
4.1 PLANTEAMIENTO DEL PROBLEMA 83
4.2 SOLUCIÓN DE PROBLEMA. 88
4.2.1 SELECCIÓN DE CONFIGURACIÓN DEL ROBOT. 88
4.2.1.1 Tipos de configuraciones. 88
4.2.1.2 Selección de la configuración de entre las alternativas antes mencionadas. 95
4.3 DIMENSIONAMIENTO 96

4.4 PROBLEMA CINEMÁTICO DIRECTO DETERMINACIÓN DE LOS PARAMETROS DE DENAVIT-HARTENBERG. 98
4.5 PROBLEMA CINEMÁTICO INVERSO-TOOLBOX HEMERO. 100
4.6 MODELO DINÁMICO DEL BRAZO ROBOT. 101
4.6.1 BASE 101
4.6.2 PRIMER ESLABÓN BRAZO. 102
4.6.3 SEGUNDO ESLABÓN BRAZO. 103
4.7 MODELADO EN SIMULINK 106
4.7.1 PRIMER PAR CINEMÁTICO DE REVOLUCIÓN (BASE): 109
4.7.2 PRIMER CUERPO O ESLABÓN (BASE GIRATORIA): 110
4.7.3 SEGUNDO PAR CINEMÁTICO DE ARTICULACIÓN: 111
4.7.4 SEGUNDO CUERPO O ESLABÓN: 112
4.7.5 TERCER PAR CINEMÁTICO DE ARTICULACIÓN: 113
4.7.6 TERCER CUERPO O ESLABÓN: 114
4.7.7 PAR CINEMÁTICO PRISMÁTICO: 115
4.7.8 CUARTO CUERPO O ESLABÓN: 116
4.7.9 CONFIGURACIÓN DE *JOINT SENSORS* (SENSORES): 117
4.7.9.1 Configuración de *Joint actuator* (Actuadores o motores PAP): 118
4.8 IMPLEMENTACIÓN DEL CONTROL AUTOMATICO 121
4.8.1 CONTROL DE TRAYECTORIA CORDÓN DE SOLDADURA (TRES PRIMEROS ESLABONES DEL ROBOT, LA BASE Y BRAZOS). 123
4.8.1.1 Ejemplo de ingreso de señal solución de cinemática inversa. 124
4.8.2 CONTROL DE STICK OUT (ALTURA DE ELECTRODO). 128
4.9 SIMULACIÓN. 131
4.9.1 PARÁMETROS DE SIMULACIÓN. 131
4.9.1.1 Tiempo de simulación. 131
4.9.1.2 Configuración de simulación. 132
4.9.2 EJECUCIÓN DE SIMULACIÓN. 133
4.10 ANÁLISIS DE RESULTADOS 135
4.10.1 CORDÓN DE SOLDADURA 135
4.10.2 TIEMPO DE SOLDADURA. 138
4.10.3 VELOCIDAD DE SOLDADURA. 138
4.10.4 STICK OUT. 138

4.11 PROBLEMA ADICIONAL. 138
4.11.1 PLANTEAMIENTO. 138
4.11.2 RESULTADOS. 140
4.12 CONCLUSIÓN 141
CAPITULO V 158
5. CONCLUSIONES Y RECOMENDACIONES 158
5.1 CONCLUSIONES 142
5.2 RECOMENDACIONES 143

INDICE DE FIGURAS

Figura 1.1 Diagrama de bloques de un sistema de control retroalimentado. 3
Figura 1.2 Diagrama de bloques de SCLA. 4
Figura 1.3 Diagrama de bloques PID. 9
Figura 1.4 Sistema de control adaptativo. 15
Figura 2.1 Robot industrial. 19
Figura 2.2 Clasificación de robots. 19
Figura 2.3 Estructura de un manipulador. 20
Figura 2.4 Caja de transmisión de un robot. 20
Figura 2.5 Robot móvil con actuadores eléctricos. 21
Figura 2.6 Sensores de contacto. 21
Figura 2.7 Efector final de robot con forma de mano. 22
Figura 2.8 Cilindro de simple efecto. 25
Figura 2.9 Actuador neumático de doble efecto. 25
Figura 2.10 Motor paso a paso. 26
Figura 2.11 Ejes que representan un sólido rígido que se traslada de una posición a otra. 30
Figura 2.12 Rotación de un sólido rígido en el espacio tridimensional. 32
Figura 2.13 La posición de la trama {B} relativa a la trama {A}. 33
Figura 2.14 Vector fijo en la trama {B} girando respecto a la trama {A}. 34
Figura 2.15 La velocidad lineal de un punto debido a una velocidad angular. 35
Figura 2.16 La velocidad de un vínculo i mediante los vectores $\boldsymbol{vi}$ y $\boldsymbol{wi}$ 37
Figura 2.17 Vectores de velocidad de los vínculos adyacentes. 38
Figura 2.18 Robot industrial muy extendido. 44
Figura 2.19 Parámetros de Denavit-Hartenberg. 46
Figura 2.20 El vector ${}^{A}P$ posiciona el elemento de volumen diferencial, *dv.* 52
Figura 3.1 Sistema mecánico masa-resorte amortiguador. 59
Figura 3.2 Representación del sistema mecánico de la figura anterior en Simulink. 59
Figura 3.3 Conjunto de librerías SimMechanics. 60
Figura 3.4 Librería cuerpos sólidos. 60
Figura 3.5 Librería de pares cinemáticas. 61
Figura 3.6 Librería de sensores y actuadores. 62
Figura 3.7 Movimiento del péndulo 63
Figura 3.8 Creación de nuevo modelo en Simulink. 64
Figura 3.9 Inserción bloques de cuerpos sólidos. 65
Figura 3.10 Casilla de *machine enviroment.* 66
Figura 3.11 Integración de un bloque de pares cinemáticos. 66
Figura 3.12 Integración de bloques de sensores y actuadores. 67
Figura 3.13 Determinación de entradas para sensores y actuadores. 68
Figura 3.14 Integración de un bloque de visualización de datos *Scope*. 68
Figura 3.15 Configuración del bloque de bancada. 69

Figura 3.16 Configuración del par cinemático. 70
Figura 3.17 Configuración de un eslabón. 70
Figura 3.18 Sistema dinámico péndulo simple. 71
Figura 3.19 Datos de masa e inercia del eslabón. 72
Figura 3.20 Configuración sistema de coordenadas relativas. 73
Figura 3.21 Configuración de bloque de condición inicial. 73
Figura 3.22 Configuración de bloque de entorno de máquina. 74
Figura 3.23 Menú simulación-configuración de parámetros. 75
Figura 3.24 Menu Simscape-SimMechanics. 75
Figura 3.25 Ejecución de simulación de modelo péndulo simple. 76
Figura 3.26 Vista de mecanismo péndulo simple en funcionamiento. 76
Figura 3.27 Señales de posición angular (grados) y velocidad angular (grados por segundo). 77
Figura 3.28 Toolbox Hemero. 79
Figura 4.1 Válvula tipo mariposa. 84
Figura 4.2 Ampliación de la válvula tipo mariposa. 84
Figura 4.3 Brida de acero. 85
Figura 4.4 Dimensiones de la brida de acero y disposición de cordón de soldadura. 85
Figura 4.5 Función cartesiana que representa el cordón de soldadura. 86
¡Error! Referencia de hipervínculo no válida.
Figura 4.7 Parámetros de soldadura. 88
Figura 4.8 Robot cartesiano. 89
Figura 4.9 Volumen de trabajo para una configuración cartesiana. 89
Figura 4.10 Robot cilíndrico. 90
Figura 4.11 Volumen de trabajo para una configuración cilíndrica. 90
Figura 4.12 Robot polar. 91
Figura 4.13 Volumen de trabajo para una configuración polar. 91
Figura 4.14 Robot angular. 92
Figura 4.15 Volumen de trabajo de una configuración angular. 93
Figura 4.16 Robot Scara. 94
Figura 4.17 Área de trabajo de configuración escara. 94
Figura 4.18 Configuración angular seleccionada del brazo robot. 95
¡Error! Referencia de hipervínculo no válida.
¡Error! Referencia de hipervínculo no válida.
Figura 4.21 Apertura máxima de robot. 97
¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.
Figura 4.22 Disposición de ejes para cada par cinemático. 98
Figura 4.23 Brazo robot en una posición aleatoria, vista lateral derecha. 99
Figura 4.24 Brazo robot en una posición aleatoria, vista lateral izquierda. 100
Figura 4.25 Robot soldador en posición similar al de la figura 4.24. 100
Figura 4.26 Base del robot. 102

Figura 4.27 Brazo del robot. 103
Figura 4.28 Antebrazo de robot. 103
Figura 4.29 Eslabones existentes en el robot soldador. 107
Figura 4.30 Pares cinemáticos en el robot soldador. 107
Figura 4.31 Integración de bloques para Brazo robot. 108
Figura 4.32 Sistema de ejes coordenados globales respecto a la base, en la figura se observa que el eje de rotación de la articulación es el eje *Z* por tanto la matriz de ejes de acción queda como [0 0 1]. 109
Figura 4.33 Configuración primer par cinemático. 109
Figura 4.34 Disposición de punto inicial y final de la base. 110
Figura 4.35 Configuración primer eslabón. 111
Figura 4.36 Ejes coordenados globales para segundo par cinemático, en la figura se observa que el eje de rotación de la articulación es el eje *Y* por tanto la matriz de ejes de acción queda como [0 1 0]. 111
Figura 4.37 Configuración segundo par cinemático. 112
Figura 4.38 Disposición del segundo eslabón. 112
Figura 4.39 Configuración segundo eslabón. 113
Figura 4.40 Ejes globales y tercer par cinemático, en la figura se observa que el eje de rotación de la articulación es el eje *Y* por tanto la matriz de ejes de acción queda como [0 1 0]. 113
Figura 4.41 Configuración tercer par cinemático. 114
Figura 4.42 Posición del antebrazo relativo a sistema de ejes coordenados. 114
Figura 4.43 Configuración tercer eslabón. 115
Figura 4.44 Ejes de último eslabón, en la figura se observa que el eje de traslación del par prismático es el eje *Z* por tanto la matriz de ejes de acción queda como [0 0 1]. 115
Figura 4.45 Configuración par prismático. 116
Figura 4.46 Posición de último eslabón. Hay que acotar que la posición del dentro de gravedad es respecto al sistema global de coordenadas y que las posiciones de CS1 y CS2 están respecto al centro de gravedad. 116
Figura 4.47 Configuración cuarto eslabón. 117
Figura 4.48 Configuración de sensores. 118
Figura 4.49 Configuración de actuadores. 118
¡Error! Referencia de hipervínculo no válida.
Figura 4.51 Configuración de constructor de señales. 122
Figura 4.52 Creación de una señal definida por el usuario. 122
Figura 4.53 Cuadro de dialogo donde se debe ingresar cada columna de la matriz q. 124
Figura 4.54 Datos de la primera columna de matriz solución siendo ingresados como señal de control. 125
Figura 4.55 Señal de velocidad angular de la base del robot. 126
Figura 4.56 Señal de aceleración angular. 126
Figura 4.57 Señal requerida por cada par cinemático. 127

Figura 4.58 Subsistema de señales. 128
Figura 4.59 Señal de control de stick out. 129
Figura 4.60 Regulación de ganancia para obtener la variación de stick out correcta. 129
¡Error! Referencia de hipervínculo no válida.
Figura 4.62 Tiempo y escala de tiempo de simulación. 131
Figura 4.63 Configuración de parámetros de simulación. 132
Figura 4.64 Opciones de configuración de parámetros de simulación. 132
Figura 4.65 Botón *Start simulation.* 133
Figura 4.66 Inicio de simulación. 133
Figura 4.67 Final de la simulación. 134
Figura 4.68 Secuencia de trabajo de robot 134
Figura 4.69 Secuencia que muestra la trayectoria que realiza el robot. 135
Figura 4.70 Sensor de movimiento en el eslabón base. Medición de posición angular (radianes) y velocidad angular (radianes por segundo). 136
Figura 4.71 Sensor de movimiento en el Brazo. Medición de posición angular (grados) y velocidad angular (grados por segundo). 136
Figura 4.72 Sensor de movimiento en el antebrazo. Medición de posición angular (grados) y velocidad angular (grados por segundo). 137
Figura 4.73 Sensor de movimiento en el efector final. Medición de posición lineal (metros) y velocidad lineal (metros por segundo). Aquí se observa la variación de stick out. 137
Figura 4.74 Válvula tipo mariposa. 138
¡Error! Referencia de hipervínculo no válida.
¡Error! Referencia de hipervínculo no válida.
Figura 4.77 Movimiento del robot durante la simulación. Claramente se observa el cordón de soldadura que traza el robot desde la posición inicial hasta la posición final. 140

CAPITULO I

CONTROL AUTOMATICO

1.1 INTRODUCCIÓN AL CONTROL AUTOMATICO.

El control automático es la rama de la ingeniería que juega un papel fundamental en los procesos de manufactura e industriales modernos. El control se ha vuelto esencial en operaciones en las industrias en procesos. Esto se debe a que el control automático brinda un medio para lograr el funcionamiento óptimo de sistemas dinámicos, mejorar la calidad y abaratar los costos de producción, expandir el ritmo de la producción, liberar la complejidad de algunas rutinas y tareas manuales repetitivas.

1.2 DEFINICIONES.

1.2.1 PLANTAS.

Una planta es un proceso que se va a modelar y posteriormente controlar, cuyo objetivo es realizar una operación determinada. Se ha de designar como planta a cualquier objeto físico que ha de ser controlado.

1.2.2 PROCESOS.

Se define “proceso” como una operación o desarrollo natural, progresivamente continuo, caracterizado por una serie de cambios graduales que llegan a determinado resultado final; o una operación artificial o voluntaria, progresivamente continua que consiste en una serie de acciones controladas o movimientos dirigidos sistemáticamente hacia determinado resultado o fin.[1]

[1] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 3.

1.2.3 SISTEMAS

Un sistema es una combinación de componentes que actúan conjuntamente y cumplen determinado objetivo. Un sistema no está limitado a los objetivos físicos. El concepto sistema puede ser aplicado a fenómenos abstractos y dinámicos, como los de la economía. Por lo tanto hay que interpretar el término sistema como referido a sistemas físicos, biológicos, económicos, etc.

1.2.4 PERTURBACIONES

Una perturbación es una señal que tiende a afectar adversamente el valor de la salida de un sistema. Si la perturbación se genera dentro del sistema, se la denomina interna, mientras que una perturbación externa se genera fuera del sistema y constituye una entrada.

1.2.5 CONTROL DE REALIMENTACIÓN

Control de realimentación es una operación que, en presencia de perturbaciones tiende a reducir la diferencia entre la salida y la entrada de referencia de un sistema (o un estado deseado, arbitrariamente variado) y que lo hace sobre la base de esta diferencia. Aquí solamente se consideran perturbaciones a las no previsibles (es decir a las desconocidas de antemano), pues para las que pueden ser predichas o conocidas siempre se puede incluir una compensación dentro del sistema de modo que sean innecesarias las mediciones.[2]

1.2.6 SISTEMAS DE CONTROL REALIMENTADO

Sistema de control realimentado es aquel que tiende a tener una relación preestablecida entre la salida y la entrada de referencia, comparando ambas y utilizando la diferencia como parámetro de control.

---

[2] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 3.

Es de notar que los sistemas de control realimentado no están limitados al campo de ingeniería, sino que se los puede encontrar en campos ajenos a la misma, como la biología y economía.

1.3 CONTROL DE LAZO CERRADO Y LAZO ABIERTO

Se va a definir ambos sistemas (cerrado y abierto) para luego hacer comparaciones entre ambos.

1.3.1 SISTEMAS DE CONTROL DE LAZO CERRADO

Un sistema de control de lazo cerrado es en el cual la señal de salida tiene efecto directo sobre la acción de control como indica la figura 1.1.

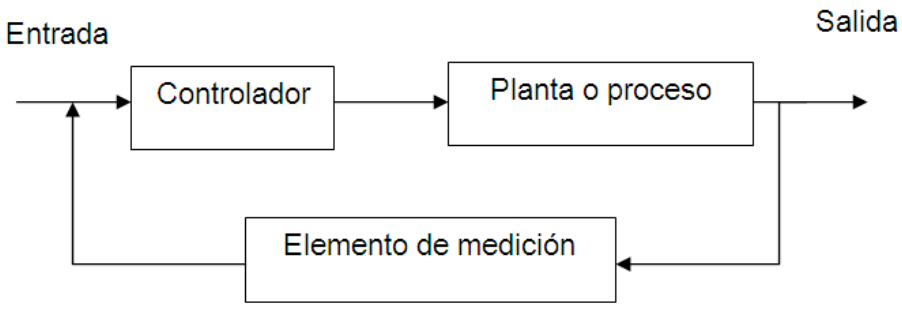


Figura 1.1 Diagrama de bloques de un sistema de control retroalimentado.[3]

La señal de error actuante, que es la diferencia entre la señal de entrada y de realimentación, entra al detector o control de manera de reducir el error y llevar la salida del sistema al valor deseado. En otras palabras, el término <<lazo cerrado>> implica el uso de acción de realimentación para reducir el error del sistema.[4]

1.3.2 SISTEMAS DE CONTROL DE LAZO ABIERTO

Los sistemas de control de lazo abierto son aquellos en los que la señal de salida no tiene efecto alguno sobre la acción de control. Es de decir en un sistema

[3] Fuente: propia
[4] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 4.

de control de lazo abierto la señal no se mide ni se realimenta para comparación con la entrada.

En este sistema la señal de salida no se compara con la referencia. Por tanto para cada entrada de referencia corresponde una condición de operación fijada, de tal manera la exactitud del sistema depende de la calibración. En presencia de perturbaciones el sistema de control de lazo abierto no cumple con su tarea asignada. Estos sistemas solamente pueden ser empleados cuando se conocen bien las señales de entrada y salida, y si no existieran perturbaciones externas ni internas.

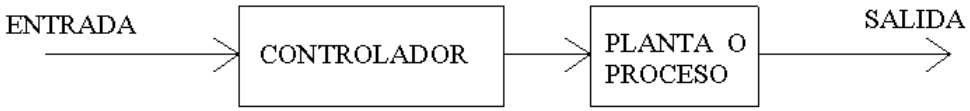


Figura 1.2 Diagrama de bloques de SCLA.[5]

1.3.3 COMPARACIÓN ENTRE LOS SISTEMAS DE LAZO CERRADO Y LAZO ABIERTO

Una clara ventaja del uso de sistemas de lazo cerrado es que la realimentación hace al sistema, en su respuesta, relativamente insensible a perturbaciones externas y a variaciones internas de parámetros del sistema.[6]

De este modo es posible utilizar componentes relativamente económicos e inexactos y lograr la exactitud requerida en determinada planta; mientras sería imposible en el caso de lazo abierto.

---

[5] Fuente: propia

[6] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 6.

Por otro lado en los sistemas de lazo cerrado la estabilidad constituye un problema de importancia por la tendencia a sobre corregir errores, que puede producir oscilaciones de amplitud constante o variable.

1.4 MODELOS MATEMATICOS DE SISTEMAS FISICOS.

Muchos sistemas ya sean mecánicos, eléctricos, hidráulicos, etc., pueden ser caracterizados por ecuaciones diferenciales. Se puede obtener la respuesta de un sistema dinámico a una entrada (o función excitadora), si se resuelven esas ecuaciones diferenciales. Para obtener las ecuaciones se utilizan las leyes físicas que gobiernan un sistema particular, por ejemplo las leyes de Newton para sistemas mecánicos, las leyes de Kirchhoff para sistemas eléctricos, etc.

1.4.1 MODELOS MATEMÁTICOS.

La descripción matemática de las características dinámicas de un sistema se denomina modelo matemático. El primer paso en el análisis de un sistema dinámico, es elaborar su modelo. Hay que tener presente siempre que desarrollar un modelo matemático es la parte más importante de todo análisis.

Los modelos pueden tomar muchas formas distintas. Según el sistema particular que se trate y las circunstancias, una representación matemática puede ser más adecuada que otras representaciones.

Una vez obtenido el modelo matemático de un sistema se pueden usar diversas herramientas analíticas y computacionales con el objeto de su adecuado análisis y síntesis. [7]

1.4.2 SISTEMAS LINEALES.

Los sistemas lineales son aquellos en los que las ecuaciones del modelo son lineales. Una ecuación diferencial es lineal si los coeficientes son constantes o funciones únicamente de la variable independiente. La propiedad más importante de los sistemas lineales es que se les puede aplicar el principio de

[7] Ogata K., Ingeniería de Control Moderna; Capítulo 4; 2da edición; pág. 75.

superposición. El principio de superposición especifica que la respuesta producida por la aplicación simultánea de dos funciones excitadoras distintas, es la suma de las dos respuestas individuales. Por tanto, para sistemas lineales se puede calcular la respuesta a diversas entradas, tratando una entrada por vez y añadiendo o sumando los resultados. Este principio es el que nos permite resolver ecuaciones diferenciales lineales complicadas, con soluciones simples. [8]

1.4.3 SISTEMAS NO LINEALES.

Son aquellos representados por ecuaciones no lineales. Ejemplo de ecuaciones no lineales son:

$$y = a \sin x \qquad \text{Ec. (2.1)}$$

$$y = x^2 \qquad \text{Ec. (2.2)}$$

$$z = x^2 + y^2 \qquad \text{Ec. (2.3)}$$

Siendo *z* función no lineal de *x* y *y*.

Una ecuación diferencial recibe el nombre de no lineal si no es lineal. Ejemplo de ecuaciones diferenciales no lineales:

$$\frac{d^2x}{dt^2} + \left(\frac{dx}{dt}\right)^2 + x = A \sin wt \qquad \text{Ec. (2.4)}$$

$$\frac{d^2x}{dt^2} + (x^2 - 1)\frac{dx}{dt} + x = 0 \qquad \text{Ec. (2.5)}$$

$$\frac{d^2x}{dt^2} + \frac{dx}{dt} + x + x^3 = 0 \qquad \text{Ec. (2.6)}$$

Aunque algunas relaciones físicas son frecuentemente representadas por ecuaciones lineales, en la mayor parte de los casos realmente las relaciones no son muy lineales.

---

[8] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 76.

De hecho, un estudio cuidadoso de los sistemas físicos indica que aún los denominados <<sistemas lineales>> son realmente lineales en restringidos rangos de operación. En la práctica, muchos sistemas electromecánicos, hidráulicos, neumáticos, etc., involucran relaciones no lineales entre las variables. Por ejemplo, la salida de un componente puede saturarse para niveles elevados de señal de entrada. Puede haber una franja o zona muerta que afecta las señales pequeñas (la franja muerta de un componente es un pequeño rango de variaciones de entrada a las cuales el componente no es sensible).

En algunos componentes puede haber no linealidades cuadráticas, por ejemplo los amortiguadores utilizados en sistemas físicos pueden ser lineales para operaciones de baja velocidad, pero volverse no lineales en velocidades altas y la fuerza de amortiguación volverse proporcional al cuadrado de la velocidad de funcionamiento.

La característica más importante de los sistemas no lineales es que no es aplicable el principio de superposición. En general, los procedimientos para hallar soluciones de problemas que involucran sistemas no lineales, son extremadamente complicados. Debido a esta dificultad matemática, inherente a los problemas no lineales, a menudo se encuentra necesario introducir sistemas lineales <<equivalentes>> en reemplazo de los no lineales. Estos sistemas lineales equivalentes son validos solamente para restringidos rangos de operación. [9]

1.5 ALGEBRA DE BLOQUES

1.5.1 DIAGRAMA DE BLOQUES

Un sistema de control consiste en cierta cantidad de componentes, con el objeto de mostrar las funciones cumplidas por cada componente, en ingeniería de control se acostumbra a usar un diagrama denominado <<diagrama de bloques>>.

---

[9] Ogata K., Ingeniería de Control Moderna; Capítulo 1; 2da edición; pág. 77.

Un diagrama de bloques de un sistema es una representación gráfica de las funciones realizadas por cada componente y del flujo de las señales.

Un diagrama así indica las interrelaciones entre componentes. A diferencia de una representación matemática, puramente abstracta, un diagrama de bloques tiene la ventaja de indicar en forma más realista el flujo de señales del sistema real.

En un diagrama de bloques todas las variables del sistema son enlazadas entre sí mediante bloques funcionales. El <<bloque funcional>>, o simplemente <<bloque>> es un símbolo de la operación matemática que el bloque produce a la salida, sobre la señal que tiene a la entrada. Las funciones de transferencia de los componentes, generalmente se colocan en los bloques correspondientes, que están conectados por flechas para indicar la dirección del flujo de señales. Nótese que la señal puede pasar solamente en el sentido de las flechas. Así un diagrama de bloques de un sistema de control muestra explícitamente una propiedad o característica unilateral.[10]

1.6 CONTROL PID (Proporcional-Integral-Derivativo)

La familia de controladores PID son robustos (estables) en muchas aplicaciones y son los que más se utilizan en la industria. La estructura de un controlador PID es simple, debido a eso se limita el rango de plantas donde pueden controlar en forma satisfactoria (existe un grupo de plantas inestables que no pueden ser estabilizadas con ningún miembro de la familia PID).

[10] Ogata K., Ingeniería de Control Moderna; Capítulo 4; 2da edición; pág. 94.

1.6.1 ESTRUCTURA DEL PID

Consideremos un lazo de control de una entrada y una salida (SISO) de un grado de libertad:

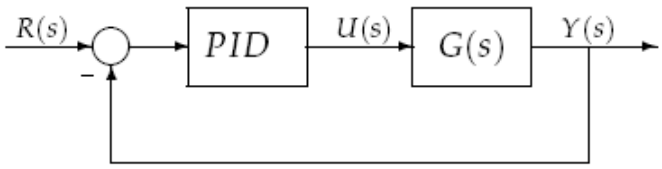


Figura 1.3 Diagrama de bloques PID.[11]

Los miembros de la familia de controladores PID, incluyen tres acciones: proporcional (P), Integral (I) y derivativa (D). Estos controladores son los denominados P, I, PI, PD y PID.

1.6.2 P: ACCIÓN DE CONTROL PROPORCIONAL

Da una salida del controlador que es proporcional al error, es decir: $\mathbf{u(t) = K_p e(t)}$,

Que descrita desde su función transferencia queda:

$$C_p(s) = K_p(1) \qquad \text{Ec. (2.7)}$$

$$u(t) = K_p e(t) \qquad \text{Ec. (2.8)}$$

Donde Kp es una ganancia proporcional ajustable. Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y error en régimen permanente (off-set).

[11] Fuente propia.

1.6.3 I: ACCIÓN DE CONTROL INTEGRAL

Da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

$$u(t) = K_i \int_0^t e(\tau)d\tau \qquad \text{Ec. (2.9)}$$

$$C_i(s) = \frac{K_i}{s} \qquad \text{Ec. (2.10)}$$

La señal de control $u(t)$ tiene un valor diferente de cero cuando la señal de error $e(t)$ es cero, por lo que se concluye que dada una referencia constante, o perturbaciones, el error en régimen permanente es cero.

1.6.4 PI: ACCIÓN DE CONTROL PROPORCIONAL-INTEGRAL

Se define mediante

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau)d\tau \qquad \text{Ec. (2.11)}$$

Donde Ti se denomina tiempo integral y es quien ajusta la acción integral. La función de transferencia resulta:

$$C_{PI}(s) = K_p(1 + \frac{1}{T_i s}) \qquad \text{Ec. (2.12)}$$

Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error pequeño positivo siempre nos dará una acción de control creciente, y si fuera negativo la señal de control seria decreciente. Este razonamiento sencillo nos muestra que el error en régimen permanente será siempre cero.

Muchos controladores industriales tienen solo acción PI. Se puede demostrar que un control PI es adecuado para todos los procesos donde la dinámica es esencialmente de primer orden.

1.6.5 PD: ACCIÓN DE CONTROL PROPORCIONAL-DERIVATIVA

Se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \qquad \text{Ec. (2.13)}$$

Donde Td es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante que amplifica las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que solo es eficaz durante períodos transitorios. La función transferencia de un controlador PD resulta:

$$C_{PD}(s) = K_p + sK_pT_d \qquad \text{Ec. (2.14)}$$

Cuando una acción de control derivativa se agrega a un controlador proporcional, permite obtener un controlador de alta sensibilidad, es decir que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Aunque el control derivativo no afecta en forma directa al error en estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia K, lo cual provoca una mejora en la precisión en estado estable.

1.6.6 PID: ACCIÓN DE CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVA

Esta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(\tau)d\tau + K_p T_d \frac{de(t)}{dt} \qquad \text{Ec. (2.15)}$$

$$C_{PID}(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \qquad \text{Ec. (2.16)}$$

Desde una perspectiva moderna, un controlador PID es simplemente un controlador de hasta segundo orden, conteniendo un integrador.

Descubrimientos empíricos demuestran que la estructura del PID por lo general tiene la suficiente flexibilidad como para alcanzar excelentes resultados en muchas aplicaciones. El término básico es el término proporcional, P, que genera una actuación de control correctivo proporcional al error. El término integral, I, genera una corrección proporcional a la integral del error. Esto nos asegura que si aplicamos un esfuerzo de control suficiente, el error de seguimiento se reduce a cero. El término derivativo, D, genera una acción de control proporcional al cambio de rango del error. Esto tiende a tener un efecto estabilizante pero por lo general genera actuaciones de control grandes.

Los diferentes métodos de sintonización de los parámetros de un controlador PID, van de acuerdo a la estructura que se utilice del mismo.

1.7 CONTROL AUTOMATICO EN MANIUPULADORES ROBOTICOS.

1.7.1 TÉCNICA DE PAR COMPUTADO.

1.7.1.1 Sistemas lineales y no lineales.

En el estudio de manipuladores robóticos se consideran a estos como sistemas no lineales y variantes en el tiempo. Cuando las linealidades no son severas, puede utilizarse la linealización local para derivar modelos lineales que son aproximaciones de las ecuaciones no lineales en ubicaciones adyacentes a un punto de operación. El problema de los manipuladores es que no se adaptan bien a este método, ya que los manipuladores se mueven constantemente a lo largo de regiones de su espacio de trabajo tan ampliamente separadas que no puede encontrarse una linealización válida para todas las regiones.

En un sistema no lineal se podría decir que los polos se mueven a lo largo de plano real-imaginario como una función de la posición del bloque.

Por tanto no se pueden seleccionar ganancias fijas que mantuvieran a los polos en posiciones deseadas. Como una opción podríamos considerar las

ganancias variables en el tiempo, es decir que varíen como una posición del bloque para que de esta forma siempre este críticamente amortiguado.

En esencia se calcularía K_p de tal manera que el efecto no lineal se cancelará con un término de la ley de control. Este esquema de control se llama ley de control de linealización.[12]

1.7.1.2 Sistemas de control multientradas y multisalidas.

El problema de controlar un manipulador es MIMO (múltiples entradas y múltiples salidas), es decir tenemos un vector de posiciones, velocidades y aceleraciones de la articulación deseadas, y la ley de control debe calcular un vector de señales de articulación-actuador. Nuestro esquema básico en el que se particiona la ley de control en una porción basada en el modelo y una posición del servo aun puede aplicarse, pero ahora aparece en una forma matricial-vectorial. La ley de control toma la forma:

$$F = \alpha F' + \beta \qquad \text{Ec. (2.17)}$$

En donde, para un sistema de n grados de libertad, F (salidas), F' (entradas) y β (parámetros de control) son vectores de $n \times 1$ (n= grados de libertad) y α es una matriz de $n \times n$. Observe que la matriz α no es necesariamente diagonal, sino que se elige para desacoplar las n ecuaciones de movimiento. Si α y β se seleccionan correctamente, entonces desde la entrada, F', el sistema aparece como si fuera n masas unitarias independientes. Por esta razón, en el caso multidimensional la porción basada en el modelo de la ley de control se llama ley de control de linealización y desacoplamiento. La ley de servo se convierte automáticamente en:[13]

$$F' = \ddot{X}_d + K_v \dot{E} + K_p E \qquad \text{Ec. (2.18)}$$

Donde K_v y K_p ahora son matrices n x n, que se seleccionan generalmente para ser diagonales con ganancias constantes en la diagonal.

---

[12] CRAIG J., Robótica; Capítulo 10; 3ª edición; pág. 291.
[13] CRAIG J., Robótica; Capítulo 10; 3ª edición; pág. 295.

E y $\dot{E}$ son vectores $n \times 1$ de los errores en posición y velocidad, respectivamente.[14]

1.7.2 CONTROL ADAPTATIVO.

1.7.2.1 Sistemas de control adaptivo.

Un sistema de control adaptivo es aquel que continua y automáticamente mide las características dinámicas de la planta, las compara con las características deseadas y usa la diferencia para variar parámetros ajustables del sistema o para generar una señal de accionamiento de modo que se pueda mantener el funcionamiento óptimo con independencia de las variaciones ambientales; alternativamente este sistema puede medir su propio funcionamiento de acuerdo con un índice de comportamiento dado y modificar de ser necesario sus propios parámetros para mantener el funcionamiento óptimo con independencia de los cambios ambientales. El comportamiento óptimo de un sistema se logra encontrando la condición para la cual un sistema determinado funciona de la mejor forma posible, en control adaptivo esto se define por medio del índice de comportamiento que justamente es el que da al sistema, el comportamiento óptimo. El índice de comportamiento indica el máximo o el mínimo del comportamiento óptimo del sistema por lo que sí su valor es cero se puede utilizar como una señal de error del adaptivo. La reducción de costos de funcionamiento es un parámetro muy utilizado a la hora de la determinación del índice de comportamiento, sin embargo también se toman en cuenta otras condiciones como la confiabilidad, la aplicabilidad y la selectividad. Como ya se ha mencionado el control adaptivo pretende dar un funcionamiento óptimo a la planta, sin embargo, en muchas ocasiones el conocimiento que se tiene de las características de la misma es muy escaso, por lo que realizar solamente una identificación inicial puede no ser suficiente.

[14] CRAIG J., Robótica; Capítulo 10; 3ª edición; pág. 295.

En estos casos la determinación del máximo o mínimo del índice de comportamiento se debe realizar periódicamente con un intervalo de tiempo que tome en cuenta la velocidad con que se modifican los parámetros del sistema.

La figura 1.4 que se muestra a continuación es el diagrama de bloques que define un sistema adaptivo.

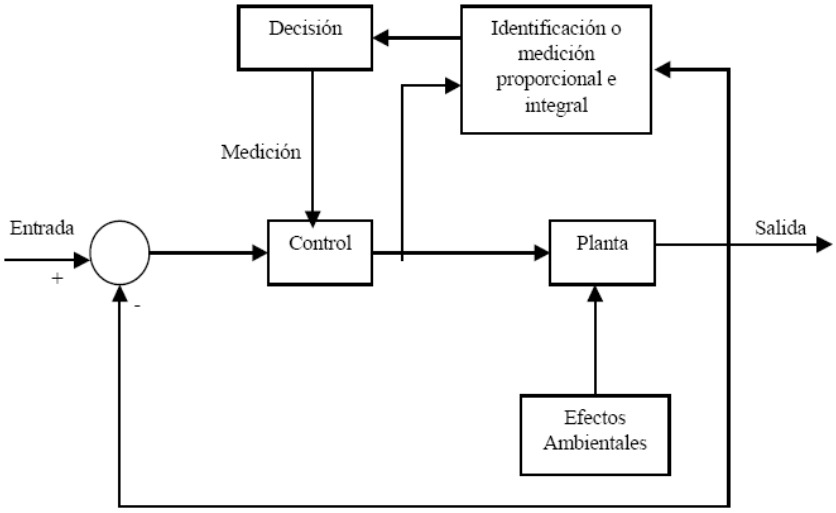


Figura 1.4 Sistema de control adaptativo.[15]

En este sistema se puede identificar la planta y se observa como el índice de comportamiento es medido varias veces y su valor es comparado con el que se obtiene del controlador para finalmente tomar la decisión de cambiar o no la señal actuadora. Con fines explicativos, el proceso de un controlador adaptativo puede verse separando dos escalas de tiempo que evolucionan a diferente velocidad. Los cambios en los parámetros del controlador ocurren según la escala lenta. A su vez, la dinámica del lazo ordinario de realimentación transcurre según la escala rápida. Existen dos grandes grupos de controladores adaptivos, los Controladores Adaptivos con Modelo de Referencia (MRAC) y los Reguladores Autoajustables (STR).

[15] http://www2.eie.ucr.ac.cr/~valfaro/docs/monografias/0201/ucr.ie431.trabajo.2002.01.grupo07.pdf

Estos últimos son sistemas de lazo abierto y modifican los parámetros del sistema a partir de una tabla previamente calculada para distintos puntos de funcionamiento, en función de una variable auxiliar. Los MRAC por su parte, logran un comportamiento de lazo cerrado dado por un modelo de referencia ante una entrada dada. Son tres las partes principales que se distinguen en el sistema de control adaptivo:

a. Identificación de las características dinámicas de la planta:

La identificación de las características dinámicas de la planta se debe realizar sin afectar el funcionamiento normal de la planta, en forma periódica y con el fin de analizar en cada resultado la respuesta del sistema. Una forma de realizar esta identificación es aplicando una señal a la planta, la cual puede ser sus entradas normales ya que estas no pueden producir salidas no deseadas o resultados confusos, pero hay que tener en cuenta que esta señal debe contar con características adecuadas de ancho de banda, amplitud, etc. para obtener una correcta identificación. El tiempo en que tarda la identificación no debe ser muy grande ya que pueden cambiar los parámetros de la planta en el transcurso de una identificación por lo que se hace imposible conocer totalmente el comportamiento de la planta. Si ya son conocidos los parámetros de la planta, solamente es necesario obtener el índice de comportamiento, que indica que parámetros deben ser variados, para tener la identificación completa. Existen sistemas en los que la adaptabilidad depende únicamente del índice de comportamiento a estos sistemas se les conoce como optimizantes.

b. Toma de decisión basada en la identificación de la planta.

Luego de haber realizado la identificación completa de la planta se debe comparar los resultados obtenidos con los óptimos, que indican el comportamiento que debe seguir la planta, para tomar la decisión de cuales parámetros deben modificarse y la forma en que se deben cambiar.

c. Modificación basada en la decisión hecha.

Con la toma de la decisión se deben cambiar algunos parámetros para modificar el funcionamiento de la planta entre ellos se encuentra la señal de entrada que puede ser cambiada ajustando los parámetros del controlador para así compensar las variaciones de la planta (modificación de parámetro del control), o bien, sintetizando la señal de control óptima que se basa en la función de transferencia, el índice de comportamiento y la respuesta transitoria deseada (síntesis de la señal de control).

1.7.3 CONTROL POR APRENDIZAJE.

Quizás es uno de los métodos más utilizados para el control de robots desde hace años. Consiste en “enseñar al manipulador” a realizar una tarea específica guiándolo manualmente mediante un dispositivo de aprendizaje que le ayuda a memorizar las posiciones por las que pasa el efector final y la posición específica de cada par cinemático. De esta forma el robot puede realizar los mismos movimientos por si solo con una gran precisión.

Una de las ventajas de este método es la simplificación de todo el proceso hasta llegar a obtener los movimientos para realizar una tarea como se verá más adelante en el capítulo de cinemática de robots. Este tipo de control es de lazo abierto ya que se le da la solución previamente al robot y este solo la ejecuta sin control en tiempo real.

CAPITULO II

TEORÍA DE MAQUINAS ORIENTADA A ROBÓTICA

2.1 INTRODUCCIÓN A LA ROBÓTICA.

La Robótica es la ciencia y la tecnología de los robots. Se ocupa del diseño, manufactura y aplicaciones de los robots. La robótica combina diversas disciplinas como son: la mecánica, la electrónica, la informática, la inteligencia artificial y la ingeniería de control. Otras áreas importantes en robótica son el álgebra, los autómatas programables y las máquinas de estados.

2.2 DEFINICIÓN DE ROBOT INDUSTRIAL.

Por robot industrial de manipulación se entiende a una máquina de manipulación automática reprogramable y multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento.[16]

[16] Apuntes fundamentos de robótica, UNED.

Figura 2.1 Robot industrial.[17]

2.2.1 CLASIFICACIÓN DE ROBOTS.

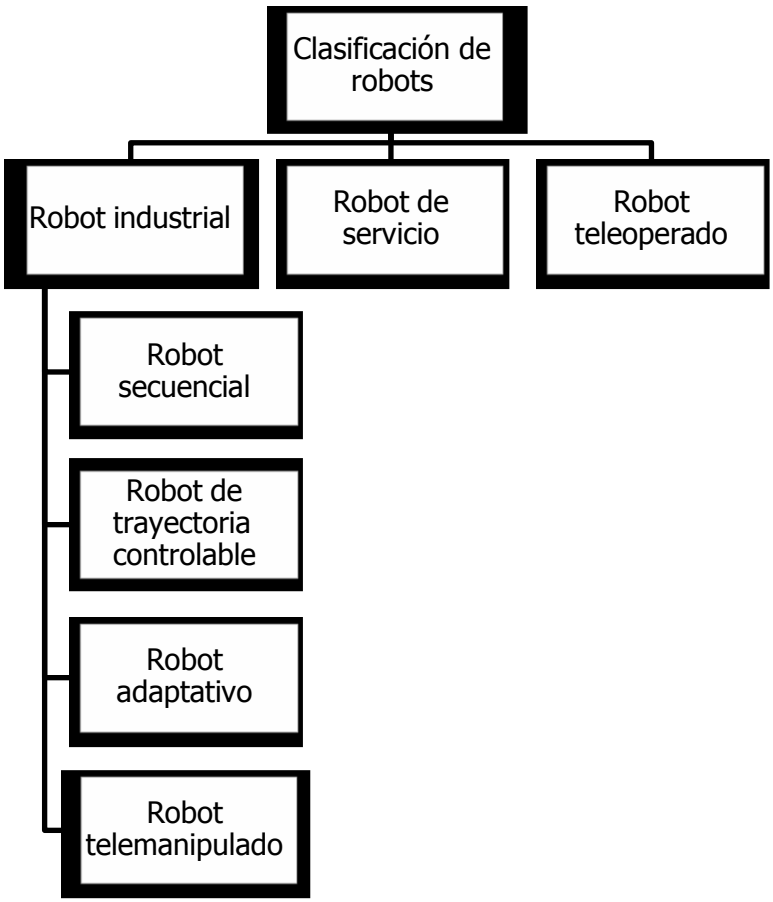


Figura 2.2 Clasificación de robots.[18]

[17] http://www.idpsa.com/images/img_galeria_polimorficos.jpg

2.2.2 ARQUITECTURA DE ROBOTS

Un Robot está formado por los siguientes elementos:

Estructura mecánica (eslabones + articulaciones)

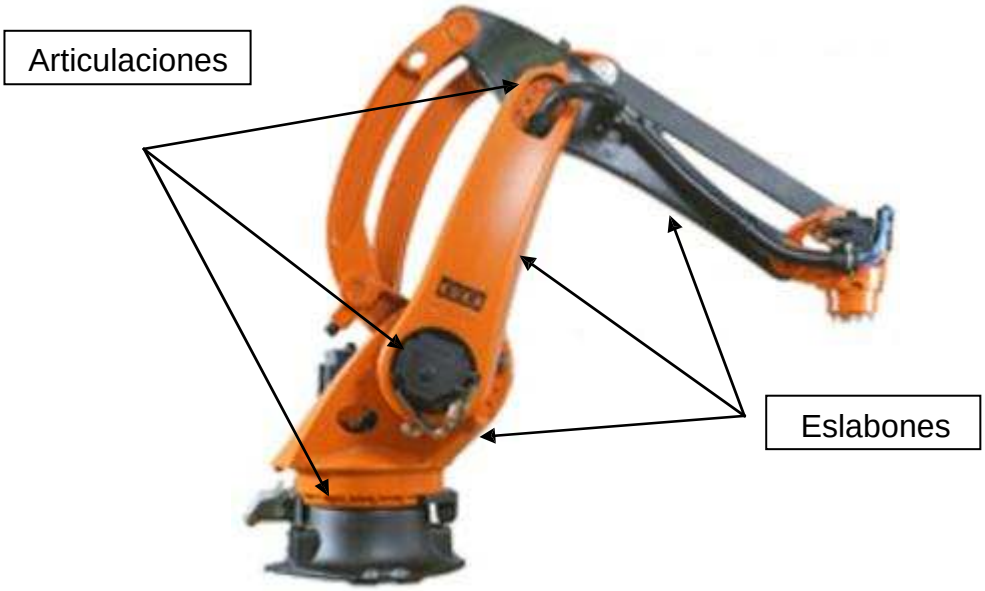


Figura 2.3 Estructura de un manipulador.[19]

Transmisiones, (reductores o accionamiento directo).

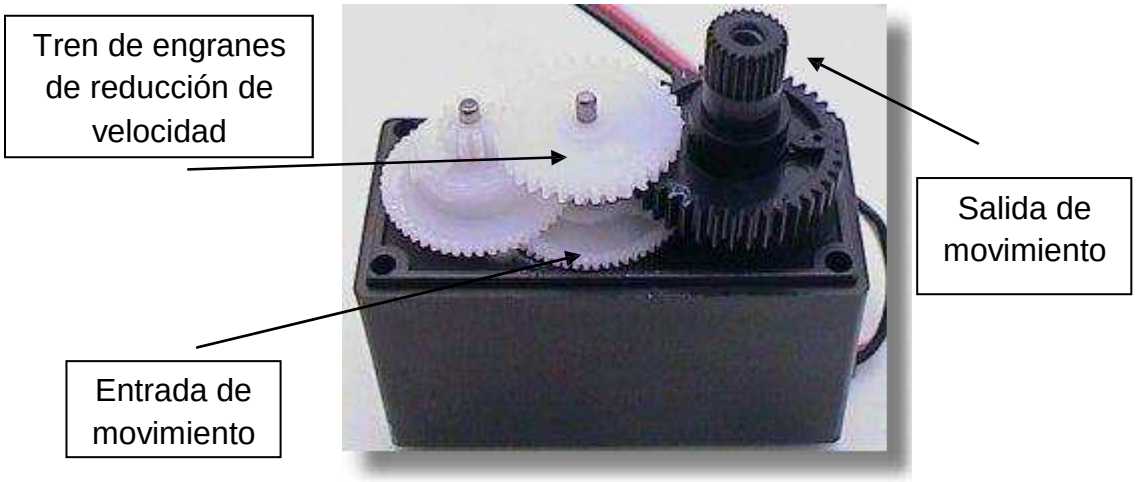


Figura 2.4 Caja de transmisión de un robot.[20]

---

[18] Apuntes fundamentos de robótica, UNED.
[19] http://www.logismarket.es/ip/otai-automatizaciones-con-robots-automatizaciones-con-robots-324714-FGR.jpg
[20] http://cfievalladolid2.net/tecno/cyr_01/robotica/sistema/images/motore13.jpg

Sistema de accionamiento (actuadores, neumáticos hidráulicos o eléctricos)

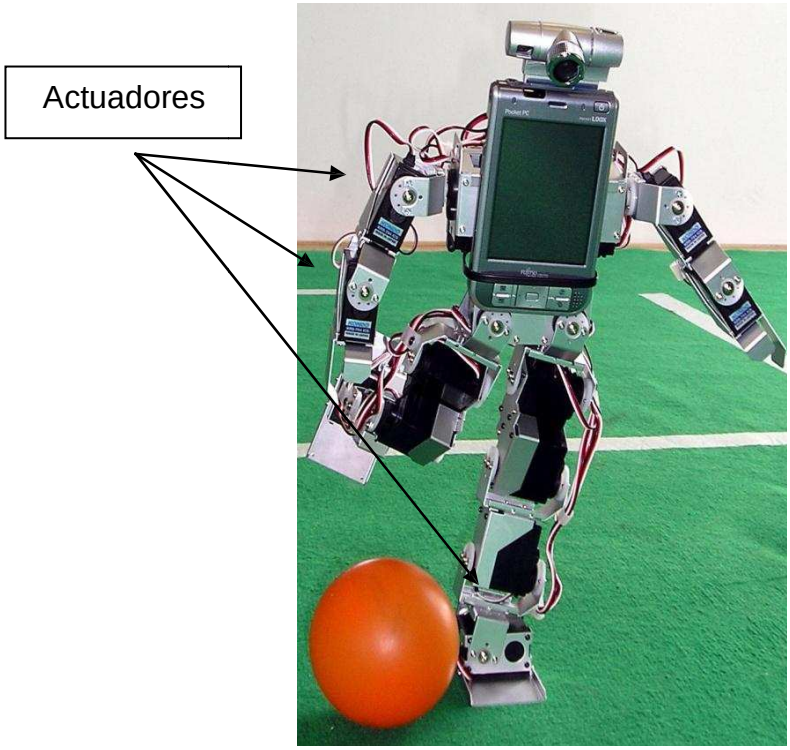


Figura 2.5 Robot móvil con actuadores eléctricos.[21]

Sistema sensorial (posición, velocidad y presencia)

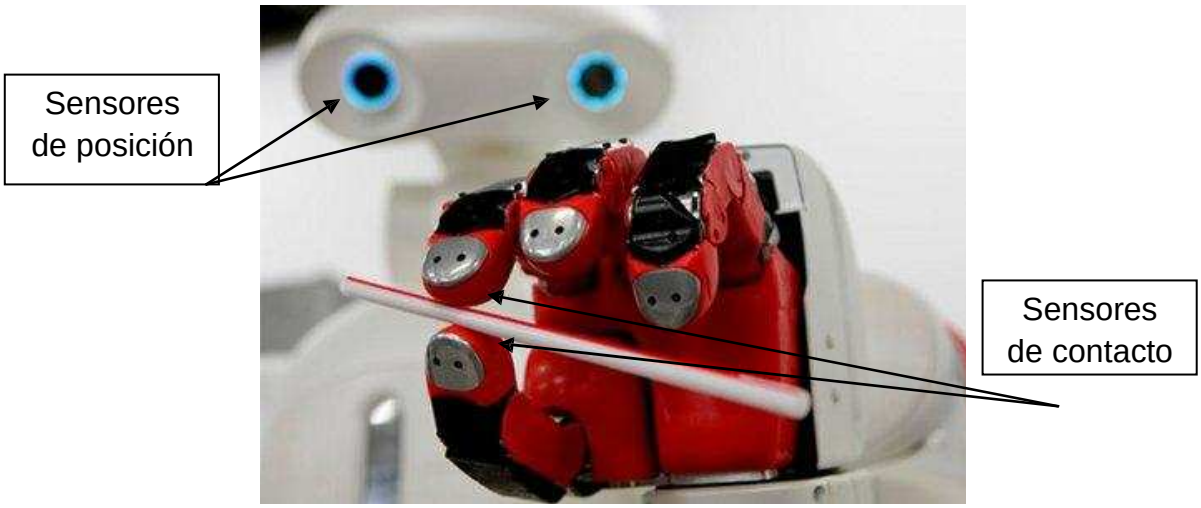


Figura 2.6 Sensores de contacto.[22]

[21] http://ounae.com/img/ounae/2007/08/robot.jpg
[22] http://img.dailymail.co.uk/i/pix/2007/11_04/robotR2711_468x562.jpg

Sistema de control

Elementos terminales

Figura 2.7 Efector final de robot con forma de mano.[23]

Estructura mecánica: Mecánicamente un robot está formado por una serie de elementos o eslabone unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos. Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior se denomina grado de libertad GDL. El número de GDL del robot viene dado por la suma de los GDL de cada articulación que lo componen. Las articulaciones utilizadas son únicamente la prismática y la de rotación, con un solo GDL cada una. Para posicionar y orientar un cuerpo en el espacio son necesarios 6 parámetros [3 de posición + 3 de orientación], es decir 6 GDL; pero en la práctica se utilizan 4 ò 5 GDL por ser suficientes. Otros casos requieren más de 6 GDL para tener acceso a todos los puntos. Cuando el número de GDL es mayor que los necesarios, se dice que el robot es redundante.

Transmisiones: Las transmisiones son los elementos encargados de transmitir el movimiento desde los actuadores hasta las articulaciones.

---

[23] http://ffodissahnerak.files.wordpress.com/2007/10/shadow-robot-hand.jpg

Dado que el robot mueve su extremo con aceleraciones elevadas, es de gran importancia reducir al máximo su momento de inercia, para ello, los actuadores están lo más cerca posible de la base del robot, lo que obliga a utilizar sistemas de transmisión que trasladen el movimiento hasta las articulaciones. También pueden ser utilizadas para convertir movimiento lineal en circular o viceversa. Características básicas de un buen sistema de transmisión:

Tamaño y peso reducidos.
Evitar holguras.
Deben tener gran rendimiento.
Ser capaz de soportar un funcionamiento continuo a un par elevado incluso a grandes distancias.

Las transmisiones más habituales son las que cuentan con movimiento circular tanto a la entrada como a la salida. (Engranajes, correas...).

Reductores: Son los encargados de adaptar el par y la velocidad de salida del actuador a los valores adecuados para el movimiento de los elementos del robot. A los reductores utilizados en robótica se les exigen unas condiciones de funcionamiento muy restrictivas por las altas prestaciones que se les exigen en cuanto a precisión y velocidad de posicionamiento.

Características:

Bajo peso y tamaño
Bajo rozamiento
Capaces de realizar una reducción elevada de velocidad en un único paso
Deben minimizar su momento de inercia
Tienen una velocidad máxima de entrada admisible
Deben soportar elevados pares puntuales. (Continuos arranques y paradas)
El juego angular debe ser lo menor posible (giro del eje de salida sin que gire el de entrada)

Alta rigidez torsional (par que hay que aplicar al eje de salida para que bloqueado el de entrada gire un ángulo unitario)

Accionamiento directo: En el accionamiento directo el eje del actuador se conecta directamente a la articulación, sin utilización de reductores intermedios, ya que éstos introducen defectos negativos como juego angular, rozamiento que impiden alcanzar la precisión y velocidad requeridas.

Ventajas:

Posicionamiento rápido y preciso pues evitan los rozamientos de transmisiones y reductores.

Mayor control del sistema a costa de una mayor complejidad.

Simplifican el sistema mecánico al eliminarse el reductor.

Inconvenientes:

Tipo de motor a emplear ya que necesitamos un par elevado a bajas revoluciones manteniendo la mayor rigidez posible, que encarecen el sistema. Suelen ser del tipo SCARA.

Actuadores: Generan el movimiento de los elementos del robot según las órdenes dadas por la unidad de control.

Actuadores neumáticos: La fuente de energía es aire a presión.

Actuadores neumáticos de cilindros neumáticos de simple efecto: se consigue el desplazamiento en un solo sentido, como consecuencia del empuje del aire a presión, mientras que en el otro sentido se desplaza por el efecto de un muelle recuperador.

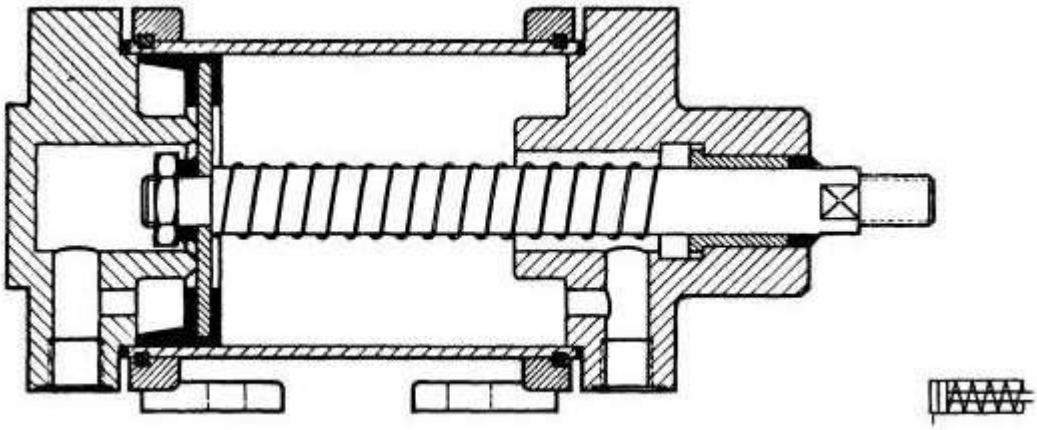


Figura 2.8 Cilindro de simple efecto.[24]

Actuadores neumáticos de cilindros de doble efecto: El aire empuja al émbolo en las dos direcciones, persiguiendo un posicionamiento en los extremos del mismo, y no un posicionamiento continuo (esto puede conseguirse mediante una válvula de distribución).

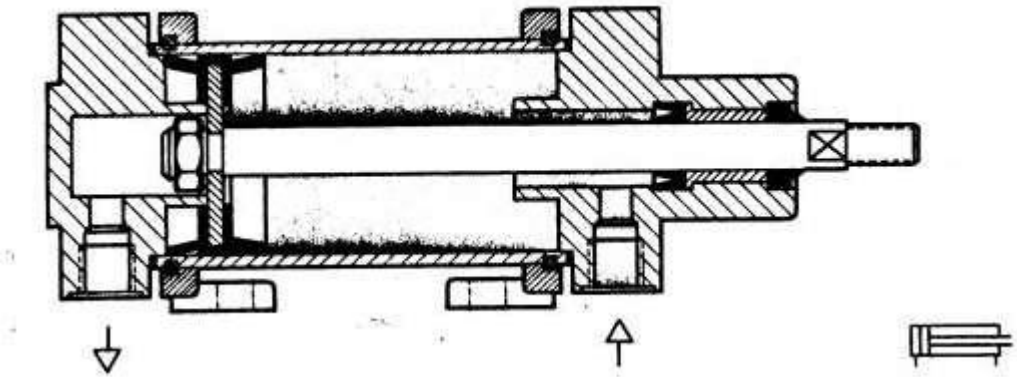


Figura 2.9 Actuador neumático de doble efecto.[25]

Motores paso a paso: son actuadores eléctricos.

Motores paso a paso de magnetización permanente: poseen una polarización magnética constante. El rotor gira para orientar sus polos respecto al estator.

[24] Apuntes fundamentos de robótica, UNED.
[25] Apuntes fundamentos de robótica, UNED.

Figura 2.10 Motor paso a paso.[26]

Motores paso a paso de reluctancia variable: el rotor está formado por un material ferromagnético dentado que tiende a orientarse con los polos bobinados de estator. Este motor sirve para realizar pasos más pequeños que los que se pueden lograr con el motor de magnetización permanente. Además el rotor de este motor es de baja inercia lo que le permite tener mejor respuesta dinámica durante su funcionamiento. Una desventaja de este motor es que tiene un torque menor que el de magnetización permanente.

Motores paso a paso híbridos: combinan los dos anteriores.

La señal de control son los trenes de pulsos que van actuando rotativamente sobre una serie de electroimanes dispuestos en el estator, por cada pulso recibido el rotor del motor gira un número determinado de grados. Para conseguir el giro del motor a un número determinado de grados, las bobinas del estator deben ser excitadas secuencialmente a una frecuencia que determina la velocidad de giro.

Ventajas:

Funcionamiento simple y exacto.
Pueden funcionar con velocidad variable.
Ligeros fiables y fáciles de controlar.

[26] http://www.directindustry.es/prod/hansen-corporation/motor-electrico-dc-con-iman-permanente-17404-362072.html

Inconvenientes:

El funcionamiento a bajas revoluciones no es suave.
Sobrecalentamiento a velocidades elevadas.
Potencia nominal baja.
Motores de corriente alterna.

Presentan una mayor dificultad de control que los motores DC. Sin embargo las mejoras introducidas en las máquinas síncronas hacen que se presenten como un claro competidor de los DC debido a:

No tienen escobillas
Usan convertidores estáticos que permiten variar la frecuencia con facilidad y precisión
Emplean microelectrónica que permite una gran capacidad de control El inductor se sitúa en el rotor y está constituido por imanes permanentes, mientras que el inducido, situado en el estator, está formado por tres devanados iguales desfasados 120º , y se alimenta de tensión trifásica.

La velocidad de giro depende de la frecuencia de la tensión que alimenta el inducido, ésta frecuencia se controla a través de un convertidor de frecuencia. Dispone de unos sensores de posición para evitar la pérdida de sincronismo, manteniendo en todo momento el ángulo entre rotor y estator (autopilotados).

Ventajas sobre los DC:

No presentan problemas de mantenimiento por no tener escobillas
Tienen una gran evacuación del calor por estar el bobinado pegado a la carcasa desarrollan potencias mayores

Inconvenientes:

Presentan una mayor dificultad de control que los motores DC.

Sensores internos: Para conseguir que un robot realice su tarea con precisión, velocidad e inteligencia, es necesario que disponga de información de su estado (sensores internos) y del estado de su entorno (sensores externos).

Sensores de posición: Codificadores angulares de posición (encoders).

Sensores de velocidad: La información de la velocidad de movimiento de cada actuador sé realimenta a un bucle de control analógico implementado en el propio accionador del elemento motor. El captador utilizado es una tacogeneratriz que proporciona una tensión proporcional a la velocidad de giro.

2.3 DESCRIPCIONES ESPACIALES Y TRANSFORMACIONES

MATRICES DE TRANSFORMACION

2.3.1 FORMULACIÓN MATRICIAL DE LAS TRANSFORMACIONES

Un robot está diseñado para manipular piezas y realizar operaciones diversas, dentro de una determinada región de trabajo. Esta manipulación se reduce a giros y traslaciones realizados en el espacio tridimensional; además, el robot puede estar dotado de una visión, en cuyo caso debe ser capaz de reconocer una misma pieza a diferentes distancias y en distintas posiciones. El enfoque matricial de estos problemas presenta grandes ventajas, puesto que las transformaciones de giro, traslación, cambio de ejes, etc., se plantean mediante el producto de una matriz y un vector, con lo que se obtiene una formulación muy sencilla. Además, es muy importante que estas transformaciones vengan definidas por un producto, ya que de este modo, se pueden aplicar transformaciones sucesivas sobre un mismo objeto, de forma que el conjunto de todas ellas pueda considerarse, a su vez, como una nueva transformación.

2.3.1.1 Matriz de traslación

Sea un sólido rígido en un espacio tridimensional; un punto P del mismo vendrá definido por tres coordenadas (x, y, z), que agrupamos en un vector $\boldsymbol{X_1}$, tal que:

$$\boldsymbol{X_1} = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \qquad \text{Ec. (2.1)}$$

Una traslación vendrá definida por un vector $\boldsymbol{D}$, de modo que, si se aplica a un sólido, el punto de posición $\boldsymbol{X_1}$ pasara a ocupar una nueva $\boldsymbol{X_2}$ definida por:

$$\boldsymbol{X_2} = \boldsymbol{X_1} + \boldsymbol{D} \qquad \text{Ec. (2.2)}$$

O en forma desarrollada:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{y_1} \\ \boldsymbol{z_1} \end{bmatrix} + \begin{bmatrix} \boldsymbol{d_x} \\ \boldsymbol{d_y} \\ \boldsymbol{d_z} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x_1} & \boldsymbol{d_x} \\ \boldsymbol{y_1} + \boldsymbol{d_y} \\ \boldsymbol{z_1} & \boldsymbol{d_z} \end{bmatrix} \qquad \text{Ec. (2.3)}$$

De esta forma, la matriz de transformación de traslación se representa, de forma muy sencilla, como suma de dos vectores; sin embargo, nos es posible encontrar una matriz de orden 3x3 que represente la traslación pero si existe una matriz 4x4, de forma que:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \boldsymbol{d_x} \\ 0 & 1 & 0 & \boldsymbol{d_y} \\ 0 & 0 & 1 & \boldsymbol{d_z} \\ 0 & 0 & 0 & \boldsymbol{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{y_1} \\ \boldsymbol{z_1} \\ 1 \end{bmatrix} \qquad \text{Ec. (2.4)}$$

La figura 2.11 muestra la traslación de un cuerpo rígido que representa la ec 2.4.

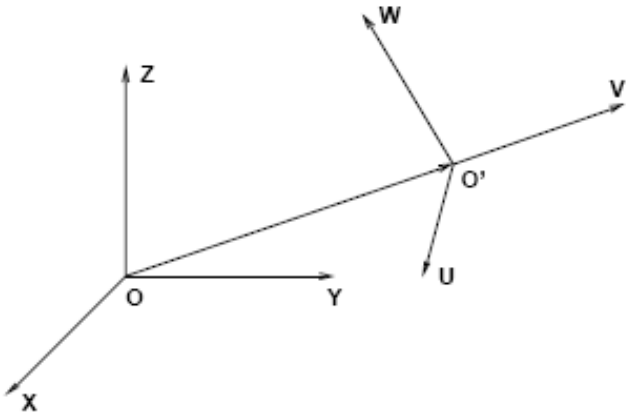


Figura 2.11 Ejes que representan un sólido rígido que se traslada de una posición a otra.

O en notación compacta:

$$[X_2] = [T_d][X_1] \qquad \text{Ec. (2.5)}$$

Donde $[T_d]$ es la "matriz de transformación".

En la expresión anterior se ha añadido un cuarto elemento al vector de coordenadas $\boldsymbol{X}$. Este modo de representar un vector se denomina de "coordenadas homogéneas".

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} cos\,\theta & -sen\,\theta \\ sen\,\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \qquad \text{Ec. (2.6)}$$

2.3.1.2 Matriz de rotación.

Las ecuaciones de rotación en dos dimensiones se escriben a continuación:

$$x_2 = x_1 \cos\theta - y_1 \operatorname{sen}\theta \qquad \text{Ec. (2.7)}$$

$$y_2 = x_1 \operatorname{sen}\theta + y_1 \cos\theta \qquad \text{Ec. (2.8)}$$

Que, en forma matricial utilizando coordenadas homogéneas es,

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\cos\theta} & -sen\,\theta & 0 \\ \boldsymbol{sen\,\theta} & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \qquad \text{Ec. (2.9)}$$

Que en notación compacta, se viene representado por

$$[X_2] = [R_\theta][X_1] \qquad \text{Ec. (2.10)}$$

En el caso de tres dimensiones, el problema es más complejo; el procedimiento a seguir depende de la forma en que la rotación venga definida. Cuando esta está determinada por tres rotaciones $\theta, \varphi, \emptyset$ alrededor de los ejes z, y, x respectivamente, cada una de ellas queda representada mediante las matrices:

$$[R_{\theta,z}] = \begin{bmatrix} \boldsymbol{\cos\theta} & -sen\,\theta & 0 & 0 \\ \boldsymbol{sen\,\theta} & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{Ec. (2.11)}$$

$$[R_{\varphi,y}] = \begin{bmatrix} \cos\varphi & 0 & sen\,\varphi & 0 \\ 0 & 1 & 0 & 0 \\ -sen\,\varphi & 0 & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{Ec. (2.12)}$$

$$[R_{\emptyset,x}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\emptyset & \boldsymbol{-sen\,\emptyset} & 0 \\ 0 & sen\,v & \boldsymbol{\cos\emptyset} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{Ec. (2.13)}$$

La rotación total se calcula mediante el producto matricial:

$$[R_{\theta,\varphi,\emptyset}] = \boldsymbol{[R_{\emptyset,x}][R_{\varphi,y}][R_{\theta,z}]} \qquad \text{Ec. (2.14)}$$

Es preciso observar que el producto de las tres matrices no tiene la propiedad conmutativa. Por tanto esta última ecuación permite encontrar la matriz

rotación en tres dimensiones calculables. Un ejemplo de rotación tridimensional de un cuerpo rígido o eslabón se muestra en la figura 2.12.

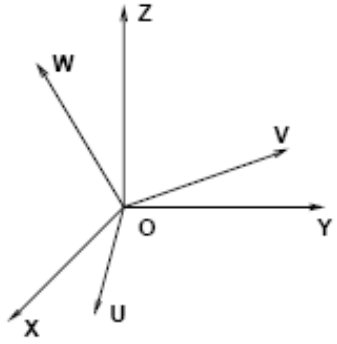


Figura 2.12 Rotación de un sólido rígido en el espacio tridimensional.

2.4 CINEMÁTICA DE ROBOTS

2.4.1 SÍNTESIS Y ANÁLISIS

En cualquier problema de ingeniería debe distinguirse entre la síntesis y el análisis. Es evidente la mayor dificultad que reviste la síntesis, ya que constituye una labor verdaderamente creadora; en el análisis, por el contrario, se trabaja sobre un sistema cuya geometría y topología son conocidas. Esta forma de diseño consiste, fundamentalmente, en crear un sistema de partida y analizarlo, corrigiéndolo en función de los resultados obtenidos y de los buscados, hasta que se obtenga una solución satisfactoria.

2.4.2 VELOCIDAD LINEAL Y ROTACIONAL DE CUERPOS RÍGIDOS

Se estudiará e investigará la descripción del movimiento de un cuerpo rígido, cuando menos en lo que respecta a la velocidad.

Para realizar el estudio del movimiento unimos un sistema de coordenadas a cualquier cuerpo que deseemos describir. De esta manera, el movimiento de los cuerpos rígidos puede estudiarse de manera equivalente al movimiento de las tramas relativas entre sí.

2.4.2.1 Velocidad lineal.

Considere una trama {B} unida a un cuerpo rígido. Deseamos describir el movimiento de ${}^{B}Q$ respecto a la trama {A}, como se muestra en la figura 2.13 podemos considerar que {A} es fija.

La trama {B} se ubica de manera relativa a {A} como lo describe un vector de posición ${}^{A}P_{BORG}$ y una matriz de rotación ${}^{A}_{B}R$. De momento, supondremos que la orientación ${}^{A}_{B}R$ no cambia con el tiempo; esto es, el movimiento del punto Q respecto a {A} se debe a que ${}^{A}P_{BORG}$ o ${}^{B}Q$ cambian con el tiempo. Es bastante simple resolver para la velocidad lineal del punto Q en términos de {A}. Sólo se expresan ambas componentes de la velocidad en términos de {A} y se suman:

$$ {}^{A}V_{Q} = {}^{A}V_{BORG} + {}^{A}_{B}R \ {}^{B}V_{Q} \qquad \text{Ec. (2.15)} $$

La ecuación (2.15) es solamente para el caso en el que la orientación relativa de {B} y {A} permanece constante.

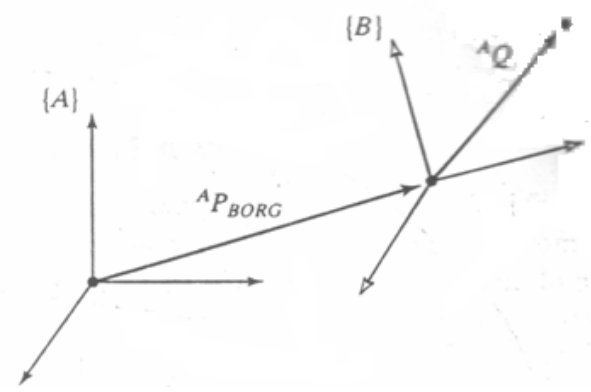


Figura 2.13 La posición de la trama {B} relativa a la trama {A}.

2.4.2.2 Velocidad de rotación.

Ahora consideremos dos tramas con orígenes coincidentes y velocidad relativa lineal cero: sus orígenes permanecerán coincidentes en todo momento.

Una o ambas podrían estar fijas a cuerpos rígidos pero, por claridad, los cuerpos rígidos no se muestran en la figura 2.14.

La orientación de la trama {B} con respecto a la trama {A} está cambiando en el tiempo. Como se indica en la figura 2.13, la velocidad de rotación de {B} relativa a {A} es descrita por un vector llamado ${}^{A}W_{B}$. También hemos indicado un vector ${}^{B}Q$ que localiza un punto fijo en {B}. Ahora consideremos la pregunta más importante, cómo cambia un vector con el tiempo según se ve desde {A} cuando está fijo en {B} y los sistemas están rotando.

Consideremos que el vector Q es constante viéndolo desde la trama {B}; esto es,

$${}^{B}V_{Q}=0 \qquad \text{Ec. (2.16)}$$

Aunque es constante de manera relativa a {B}, está claro que el punto Q tendrá una velocidad como se ve desde {A} debido a la velocidad rotacional ${}^{A}W_{B}$. Para conocer la velocidad del punto Q, usaremos una aproximación intuitiva. La figura 2.15 muestra dos fracciones de tiempo, mientras el vector Q gira alrededor de ${}^{A}W_{B}$. Esto es lo que un observador vería en {A}.

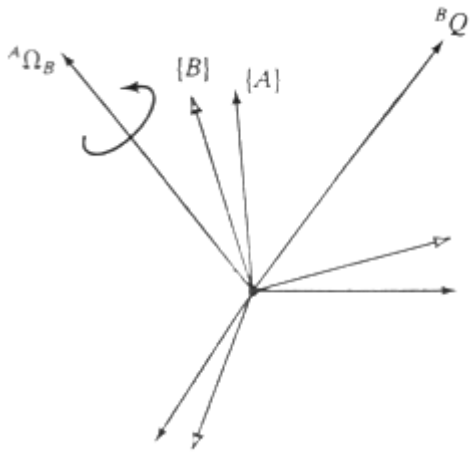


Figura 2.14 Vector fijo en la trama {B} girando respecto a la trama {A}.

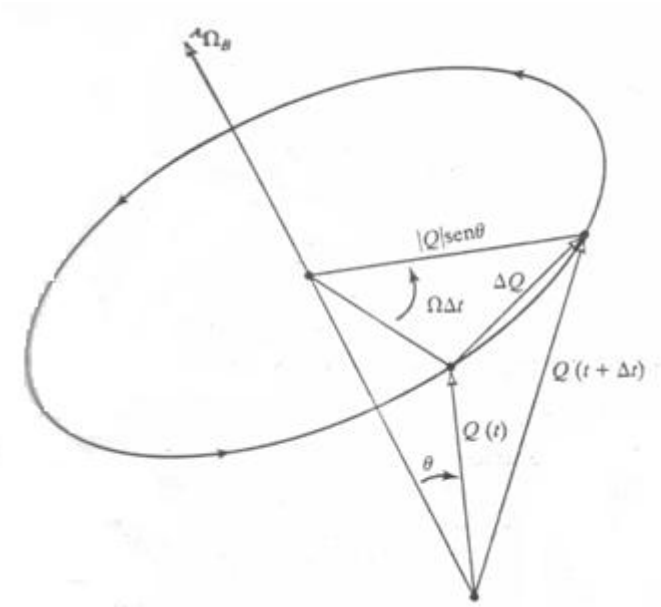


Figura 2.15 La velocidad lineal de un punto debido a una velocidad angular.

Al examinar la figura 2.15 podemos averiguar tanto la dirección como la magnitud del cambio en el vector, visto desde {A}. En primer lugar, está claro que el cambio diferencial en ^AQ debe ser perpendicular tanto a ^AW_B como a ^AQ .En segundo lugar, podemos ver gracias a la figura 2.15 que la magnitud del cambio diferencial es:

$$|\Delta Q| = \left(\left|{}^A Q\right| sin\,\theta\right)\left(\left|{}^A \Omega_B\right|\Delta t\right) \qquad \text{Ec. (2.17)}$$

Estas condiciones sobre magnitud y dirección sugieren inmediatamente el producto cruz de los vectores. Evidentemente, nuestras conclusiones sobre dirección y magnitud se satisfacen mediante la siguiente forma computacional:

$$^A V_Q = {}^A\Omega_B * {}^A Q \qquad \text{Ec. (2.18)}$$

En el caso general, el vector Q también podría estar cambiando respecto a la trama {B}, por lo que si sumamos este componente tenemos que:

$$^{A}V_{Q} = {}^{A}({}^{B}V_{Q}) + {}^{A}\Omega_{B} * {}^{A}Q \qquad \text{Ec. (2.19)}$$

Si utilizamos una matriz de rotación pará eliminar el doble superíndice y, teniendo en cuenta que la descripción de ^{A}Q en cualquier instante es $^{A}_{B}R^{B}$ Q terminamos con:

$$^{A}V_{Q} = {}^{A}_{B}R\,{}^{B}V_{Q} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q \qquad \text{Ec. (2.20)}$$

2.4.2.3 Velocidad lineal y de rotación simultáneas.

Podemos expandir de manera muy simple la ecuación (2.20) para el caso en el que los orígenes no son coincidentes: sumando la velocidad lineal del origen a la ecuación (2.20) para derivar la fórmula general para la velocidad de un vector fijo en la trama {B}, visto desde la trama {A}:

$$^{A}V_{Q} = {}^{A}V_{BORG} + {}^{A}_{B}R\,{}^{B}V_{Q} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q \qquad \text{Ec. (2.21)}$$

La ecuación (2.21) es el resultado final para la derivada de un vector en una trama móvil, visto desde una trama estacionaria.

2.4.3 MOVIMIENTO DE LOS VÍNCULOS DE UN ROBOT.

Al considerar los movimientos de los vínculos de un robot, siempre utilizaremos la trama de vínculo {0} como nuestra trama de referencia. Por lo tanto, y. es la velocidad lineal del origen de la trama del vínculo {i}, y w_i es la velocidad angular de la trama del vínculo (i}.

En cualquier instante, cada uno de los vínculos de un robot en movimiento tiene cierta velocidad lineal y angular. La figura 2.16 indica estos vectores para el vínculo i. En este caso se indica que están escritos en la trama {i}.

2.4.3.1 Propagación de velocidad vínculo a vínculo.

Ahora consideremos el problema de calcular las velocidades lineal y angular de los vínculos de un robot. Un manipulador es una cadena de cuerpos, cada uno capaz de moverse en relación a sus cuerpos adyacentes. Debido a esta estructura podemos calcular la velocidad de cada vínculo en orden, empezando desde la base. La velocidad del vínculo i + 1 será la del vínculo i más cualquier nuevo componente de velocidad que se haya agregado por la articulación i + 1.

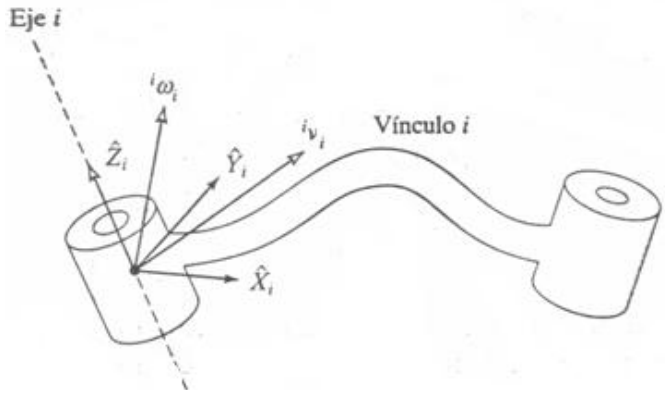


Figura 2.16 La velocidad de un vínculo i mediante los vectores $\boldsymbol{v_i}$ y $\boldsymbol{w_i}$

Según lo que se indica en la figura 2.16, vamos a pensar ahora en cada vínculo del mecanismo como si fuera un cuerpo rígido con vectores de velocidad lineal y angular que describen su movimiento. Además, expresaremos estas velocidades respecto a la trama del vínculo en sí, en vez de hacerlo respecto al sistema de coordenadas base. La figura 2.17 muestra los vínculos i e i + 1, junto con sus vectores de velocidad definidos en las tramas de los vínculos. [27]

Pueden agregarse velocidades de rotación cuando ambos vectores ω se escriben respecto a la misma trama. Por lo tanto, la velocidad angular del vínculo i + 1 es la misma que la del vínculo i más un nuevo componente producido por la

[27] CRAIG J., Robótica; Capítulo 5; 3ª edición; pág. 144.

velocidad de rotación en la articulación i + 1. Esto puede escribirse en términos de la trama {i} como:[28]

$$ {}^{i}\omega_{i+1} = {}^{i}\omega_{i} + {}^{i}_{i+1}R\dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \qquad \text{Ec. (2.22)} $$

Observe que

$$ \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} = {}^{i+1}\begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix} \qquad \text{Ec. (2.23)} $$

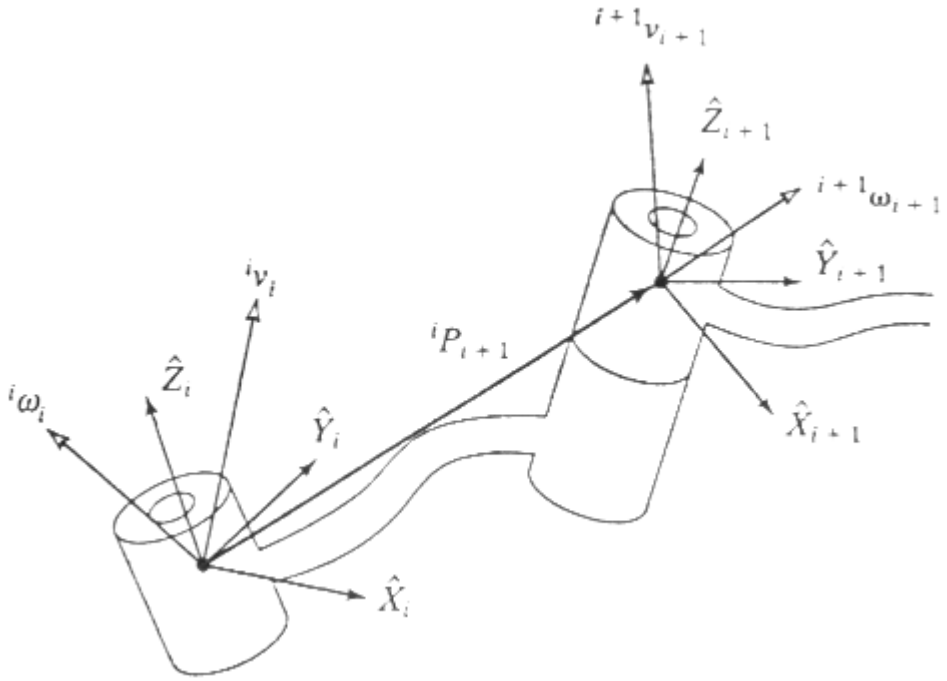


Figura 2.17 Vectores de velocidad de los vínculos adyacentes.

Hemos utilizado la matriz de rotación que relaciona a las tramas {i} e {i+1} para representar el componente en rotación agregado, debido al movimiento de la articulación en la trama {i}. La matriz de rotación gira el eje de rotación de la articulación i+ 1 hacia su descripción en la trama {i}. Por lo que pueden sumarse los dos componentes de velocidad angular.

[28] CRAIG J., Robótica; Capítulo 5; 3ª edición; pág. 146.

Si multiplicamos previamente ambos lados de la ecuación (2.22) por ${}^{i+1}R$, podemos encontrar la descripción de la velocidad angular del vínculo i + 1 respecto a la trama {i+1}:

$$ {}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R \ {}^{i}\omega_{i} + \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \qquad \text{Ec. (2.24)} $$

La velocidad lineal del origen de la trama {i+1} es la misma que la del origen de la trama {i} más un nuevo componente producido por la velocidad de rotación del vínculo i. Esta es exactamente la situación descrita por la ecuación (2.21), en donde desaparece un término debido a que ${}^{i}P_{i+1}$ es constante en la trama {i}. Por lo tanto, tenemos que:

$$ {}^{i}v_{i+1} = {}^{i}v_{i} + {}^{i}\omega_{i} * {}^{i}P_{i+1} \qquad \text{Ec. (2.25)} $$

Si multiplicamos previamente ambos lados por ${}^{i+1}_{i}R$, calculamos:

$$ {}^{i+1}v_{i+1} = {}^{i+1}_{i}R \ ({}^{i}v_{i} + {}^{i}\omega_{i} * {}^{i}P_{i+1}) \qquad \text{Ec. (2.26)} $$

Las relaciones correspondientes para el caso en el que la articulación i + 1 es prismática son:

$$ {}^{i}\omega_{i+1} = {}^{i+1}_{i}R \ {}^{i}\omega_{i} \qquad \text{Ec. (2.27)} $$

$$ {}^{i+1}v_{i+1} = {}^{i+1}_{i}R\big({}^{i}v_{i} + {}^{i}\omega_{i} * {}^{i}P_{i+1}\big) + \dot{d}_{i+1} {}^{i+1}\hat{Z}_{i+1} \qquad \text{Ec. (2.28)} $$

Si aplicamos estas ecuaciones exitosamente de vínculo en vínculo, podremos entonces calcular ${}^{N}\omega_{N}$ y ${}^{N}v_{N}$ velocidades de rotación y lineal del último vínculo. Observe que las velocidades resultantes se expresan en términos de la trama {N}. Esto resulta útil, como veremos posteriormente. Si las velocidades se desean en términos del sistema de coordenadas base, pueden girarse hacia coordenadas base mediante la multiplicación con ${}^{O}_{N}R$.

Cabe recordar que la velocidad lineal se asocia con un punto, pero la velocidad angular está asociada con un cuerpo. Por lo tanto, aquí el término "velocidad de un vínculo" indica la velocidad lineal del origen de la trama del vínculo y la velocidad de rotación del mismo.[29]

2.4.4 JACOBIANOS.

El Jacobiano es una forma multidimensional de la derivada. Por ejemplo, suponga que tenemos seis funciones, cada una de las cuales es una función de seis variables independientes:

$$y_1 = f_1(x_1, x_2, x_3, x_4, x_5, x_6)$$

$$y_2 = f_2(x_1, x_2, x_3, x_4, x_5, x_6)$$

……

$$y_6 = f_6(x_1, x_2, x_3, x_4, x_5, x_6)$$

También podríamos usar flotación vectorial para escribir estas ecuaciones:

$$Y = F(X) \qquad \text{Ec. (2.29)}$$

Ahora, si deseamos calcular los diferenciales de y en función de los diferenciales de x_j , simplemente utilizamos la regla de cadena para calcular y obtener:[30]

$$\delta y_1 = \frac{\partial f_1}{\partial x_1}\delta x_1 + \frac{\partial f_1}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_1}{\partial x_6}\delta x_6$$

$$\delta y_2 = \frac{\partial f_2}{\partial x_1}\delta x_1 + \frac{\partial f_2}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_2}{\partial x_6}\delta x_6$$

……

[29] CRAIG J., Robótica; Capítulo 5; 3ª edición; pág. 146.
[30] CRAIG J., Robótica; Capítulo 5; 3ª edición; pág. 149.

$$\delta y_6 = \frac{\partial f_6}{\partial x_1}\delta x_1 + \frac{\partial f_6}{\partial x_2}\delta x_2 + \cdots + \frac{\partial f_6}{\partial x_6}\delta x_6$$

Lo cual podría escribirse con más simplicidad en notación vectorial:

$$\delta Y = \frac{\partial F}{\partial X_1}\delta X \qquad \text{Ec. (2.30)}$$

La matriz de 6 x 6 de derivadas parciales en la ecuación (2.30) es lo que llamamos el Jacobiano, J. Observe, que si las funciones $f(x_1)$ a $f(x_6)$ son no lineales, entonces las derivadas parciales son una función de las x_i., por lo que podemos usar la notación:

$$\delta Y = J(X)\delta X \qquad \text{Ec. (2.31)}$$

Al dividir ambos lados por el elemento tiempo, podemos pensar en el Jacobiano como una asignación o mapeo de las velocidades en X a las velocidades en Y:

$$\dot{Y} = J(X)\dot{X} \qquad \text{Ec. (2.32)}$$

En cualquier instante específico, X tiene cierto valor y $J(X)$ es una transformación lineal. En cada nuevo instante de tiempo X ha cambiado y por lo tanto, también ha cambiado la transformación lineal. Los Jacobianos son transformaciones lineales que varían en el tiempo.[31]

[31] CRAIG J., Robótica; Capítulo 5; 3ª edición; pág. 149.

En el campo de la robótica, generalmente usamos Jacobianos que relacionan velocidades de articulaciones con velocidades cartesianas en la punta del brazo; por ejemplo:

$$^0v = {}^0J(\Theta)\dot{\Theta} \qquad \text{Ec. (2.33)}$$

En donde Θ es el vector de ángulos de articulación del manipulador y v es un vector de velocidades cartesianas. En la ecuación (2.33) hemos agregado un superíndice a la izquierda a nuestra notación Jacobiana para indicar en qué trama se expresa la velocidad cartesiana resultante. Algunas veces el superíndice se omite cuando la trama es obvia o cuando no es importante para el desarrollo. Tenga en cuenta que, para cualquier configuración dada del manipulador, las proporciones de articulación están relacionadas con la velocidad de la punta en modo lineal, aunque ésta es solamente una relación instantánea: en el siguiente instante, el Jacobiano ha cambiado ligeramente. Para el caso general de un robot con seis articulaciones el Jacobiano es de 6 x 6, Θ es de 6 x 1 y ^0v es de 6 x 1. Este vector de velocidad cartesiana de 6 x 1 es el vector de velocidad lineal de 3 x 1 y el vector de velocidad de rotación de 3 x 1 apilados:

$$^0v = \begin{bmatrix} {}^0v \\ {}^0\omega \end{bmatrix} \qquad \text{Ec. (2.34)}$$

Pueden definirse Jacobianos de cualquier dimensión (incluyendo los no cuadrados). El número de filas es igual al número de grados de libertad en el espacio cartesiano que se esté considerando. El número de columnas en un Jacobiano es igual al número de articulaciones del manipulador. Por ejemplo, al tratar con un brazo planar no hay razón para que el Jacobiano tenga más de tres filas, aunque para los manipuladores planares redundantes podrían haber un número arbitrariamente grande de columnas (una para cada articulación).

2.4.4.1 Singularidades.

Si tenemos una transformación lineal que relacione la velocidad de articulación con la velocidad cartesiana, una pregunta razonable sería: ¿es invertible esta matriz? Es decir, ¿es no singular? Si la matriz es no singular, entonces podemos invertirla para calcular las velocidades de articulación a partir de las velocidades cartesianas dadas:

$$\dot{\Theta} = J^{-1}(\Theta)v \qquad \text{Ec. (2.37)}$$

Ésta es una importante relación. Por ejemplo, suponga que deseamos que la mano del robot se mueva con un cierto vector de velocidad en espacio cartesiano. Utilizando la ecuación (2.37) calcularíamos las proporciones de articulación necesarias en cada instante a lo largo de la ruta. La pregunta real de invertibilidad sería si es invertible el Jacobiano para todos los valores de Q y si no lo es, entonces, en dónde no es invertible.

La mayoría de los manipuladores tiene valores de Q para los que el Jacobiano se vuelve singular. Dichas ubicaciones se llaman singularidades del mecanismo, o simplemente singularidades. Todos los manipuladores tienen singularidades en los límites o frontera de su espacio de trabajo.

2.4.4.2 Clasificación de las singularidades.

2.4.4.2.1 Singularidades en la frontera del espacio de trabajo.

Ocurren cuando el manipulador está completamente extendido o doblado sobre sí mismo, en tal forma que el efector final se encuentra en o muy cerca de los límites del espacio de trabajo. A continuación en la figura 2.18 se muestra un robot totalmente extendido verticalmente, se puede decir que se encuentra en una singularidad.

Figura 2.18 Robot industrial muy extendido.[32]

2.4.4.2.2 Singularidades dentro del espacio de trabajo.

Ocurren lejos de los límites del espacio de trabajo; generalmente se producen por el alineamiento de dos o más ejes de articulación.

Cuando un manipulador se encuentra en una configuración singular, ha perdido uno o más grados de libertad (viéndolo desde el espacio cartesiano). Esto significa que hay cierta dirección (o sub espacio) en el espacio cartesiano sobre la cual es imposible mover la mano del robot, sin importar qué proporciones de

[32] http://www.interempresas.net/FotosBD/R0-118782.jpg

articulación estén seleccionadas. Es obvio que esto ocurre en la frontera del espacio de trabajo de los robots.

2.5 CINEMÁTICA DIRECTA DE ROBOTS.

El objetivo de la cinemática directa de un manipulador es en base a transformaciones homogéneas matriciales y algebra vectorial, encontrar la posición en el espacio del efector final del robot o cualquier vínculo de este.

2.5.1 PARÁMETROS DE DENAVIT-HARTENBERG.

Denavit-Hartenberg propusieron en 1955 un método matricial que permite establecer de manera sistemática un sistema de coordenadas ligado a cada eslabón i de una cadena articulada, pudiéndose determinar a continuación las ecuaciones cinemáticas de la cadena completa.

Según la representación Denavit-Hartenberg, escogiendo adecuadamente los sistemas de coordenadas asociados para cada eslabón, será posible pasar de uno al siguiente mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón. Estas transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permitan relacionar el sistema de referencia del elemento i con el sistema del elemento i-1. Las transformaciones en cuestión son las siguientes:

Rotación alrededor del eje Z_{i-1} un ángulo θ_1.
Traslación a lo largo de Z_{i-1} una distancia di; vector di (0,0,di).
Traslación a lo largo de X_i una distancia ai; vector ai (0,0,ai).
Rotación alrededor del eje X_i, un ángulo α_1.

En la figura 2.19 se puede observar dos eslabones (Links) consecutivos los cuales se han relacionado mediante los parámetros de Denavit-Hartemberg.

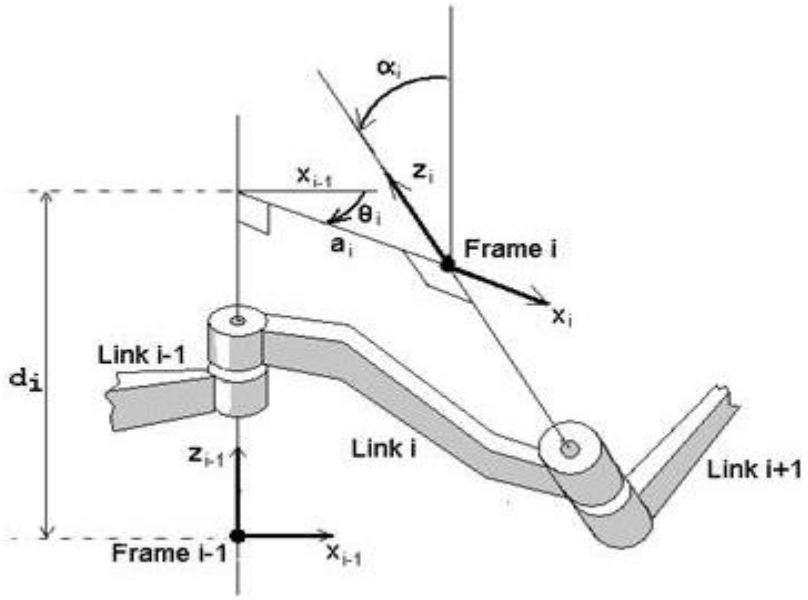


Figura 2.19 Parámetros de Denavit-Hartenberg.[33]

Los cuatro parámetros de Denavit-Hartenberg ($\boldsymbol{\theta_1}$, $\mathbf{d_i}$, $\mathbf{a_i}$, $\boldsymbol{\alpha_1}$) dependen únicamente de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente.

$\boldsymbol{\theta_1}$: Es el ángulo que forman los ejes (Frame en la figura 2.19) X_{i-1} y X_i medido en un plano perpendicular al eje Z_{i-1}, utilizando la regla de la mano derecha. Se trata de un parámetro variable en articulaciones giratorias.

$\mathbf{d_i}$: Es la distancia a lo largo del eje Z_{i-1} desde el origen del sistema de coordenadas (i-1)- esimo hasta la intersección del eje Z_{i-1} con el eje X_i. Se trata de un parámetro variable en articulaciones prismáticas.

$\mathbf{a_i}$: Es a la distancia a lo largo del eje X_i que va desde la intersección del eje Z_{i-1} con el eje X_i hasta el origen del sistema i-esimo, en el caso de articulaciones

[33] http://www.fauskes.net/media/nb/threedill/img/denavitparam.pdf

giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes Z_{i-1} y Z_i.

α_1: Es el ángulo de separación del eje Z_{i-1} y el eje Z_i, medido en un plano perpendicular al eje X_i, utilizando la regla de la mano derecha.

Una vez obtenido los parámetros de Denavit-Hartenberg procedemos a construir las matrices de transformación para cada parámetro, cuyo producto sucesivo de estas cuatro matrices representa la transformación homogénea de cada eslabón:[34]

$$ {}^{i-1}A_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Ec. (2.38)

$$ {}^{i-1}A_i = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Ec. (2.39)

2.6 CINEMÁTICA INVERSA DE ROBOTS

El objetivo del problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $\Theta = (\theta_1, \theta_2, ..., \theta_n)$ para que su extremo se posicione y oriente según una determinada localización espacial. Así como es posible abordar el problema cinemático directo de una manera sistemática a partir de la utilización de matrices de transformación homogéneas, e independientemente de la configuración del robot, no ocurre lo mismo con el problema cinemático inverso, siendo el procedimiento de obtención de las ecuaciones fuertemente dependiente de la configuración del robot.

---

[34] http://www.fauskes.net/media/nb/threedill/img/denavitparam.pdf

El problema cinemático inverso presenta fuertes no linealidades durante su solución, súmese a esto que pueden haber muchas soluciones para un mismo planteamiento, además este problema se agrava mientras más grados de libertad tiene el robot. Así para nuestro robot pueden existir hasta 8 soluciones para una posición específica del efector final.

Se han desarrollado varias herramientas informáticas para la solución de este problema. Entre estas tenemos el Toolbox Hemero para ser utilizado en la plataforma de Simulink, el cual está orientado para resolver específicamente problemas de cinemática inversa y robótica en general, este se explicará en detalle más adelante.

2.7 DINAMICA DE ROBOTS

En el estudio de las aceleraciones se consideran las fuerzas requeridas para producir movimiento utilizando las ecuaciones de movimiento para un manipulador las cuales describen la manera en que se produce el movimiento del manipulador debido a los momentos de torsión aplicados por los actuadores, o a fuerzas externas aplicadas al manipulador.

Hay dos problemas que se deben resolver que están relacionados con la dinámica de un manipulador. En el primer problema se nos proporciona un punto de trayectoria Θ, $\dot{\Theta}$ y $\ddot{\Theta}$, y deseamos encontrar el vector requerido de momentos de torsión de articulación. Esta formulación de la dinámica es útil para el problema de control del manipulador. El segundo problema es calcular cómo se moverá el mecanismo bajo la aplicación de un conjunto de momentos de torsión de articulación. Esto es, dado un vector de momento de torsión τ, calcular el movimiento resultante del manipulador, Θ, $\dot{\Theta}$ y $\ddot{\Theta}$. Esto es útil para simular el manipulador.

2.7.1 ACELERACIÓN DE UN CUERPO RÍGIDO

En cualquier instante, los vectores de velocidad lineal y angular tienen derivadas llamadas aceleraciones lineal y angular, respectivamente. Esto es:

$$ {}^{B}\dot{V}_{Q} = \frac{d}{dt}\,{}^{B}V_{Q} = \lim_{\Delta t \to 0} \frac{{}^{B}V_{Q}(t+\Delta t) - {}^{B}V_{Q}(t)}{\Delta t} \qquad \text{Ec. (2.40)} $$

$$ {}^{A}\dot{\Omega}_{B} = \frac{d}{dt}\,{}^{A}\Omega_{B} = \lim_{\Delta t \to 0} \frac{{}^{A}\Omega_{B}(t+\Delta t) - {}^{A}\Omega_{B}(t)}{\Delta t} \qquad \text{Ec. (2.41)} $$

Al igual que con las velocidades, cuando se sabe que la trama de referencia de la diferenciación es cierta trama de referencia universal, por ejemplo {U}, utilizamos la notación:

$$ \dot{v} = \dot{V}_{Aorg} \qquad \text{Ec. (2.42)} $$

$$ \dot{\omega}_{A} = {}^{U}\dot{\Omega}_{A} \qquad \text{Ec. (2.43)} $$

2.7.1.1 Aceleración lineal.

Empezaremos reformulando la ecuación que describe la velocidad de un vector ${}^{B}Q$ visto desde la trama {A} cuando los orígenes son coincidentes:

$$ {}^{A}V_{Q} = {}^{A}R\,{}^{B}V_{Q} + {}^{A}\Omega_{B} * {}^{A}R\,{}^{B}Q \qquad \text{Ec. (2.44)} $$

El lado izquierdo de esta ecuación describe cómo cambia ${}^{A}Q$ en el tiempo. Debido a que los orígenes son coincidentes, podríamos reescribir esta ecuación como:[35]

$$ \frac{d}{dt}\left({}^{A}_{B}R\,{}^{B}V_{Q}\right) = {}^{A}_{B}R\,{}^{B}V_{Q} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q \qquad \text{Ec. (2.45)} $$

---

[35] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 165.

Esta forma de la ecuación será útil cuando se derive la ecuación de la aceleración correspondiente.

Al diferenciar la ecuación (2.45) podemos derivar expresiones para la aceleración de ${}^{B}Q$ visto desde {A} cuando coinciden los orígenes de {A} y {B}:

$$ {}^{B}\dot{V}_{Q} = \frac{d}{dt}\left({}^{A}_{B}R\,{}^{B}V_{Q}\right) + {}^{A}\dot{\Omega}_{B} * {}^{A}_{B}R\,{}^{B}Q + {}^{A}\Omega_{B} * \frac{d}{dt}\left({}^{A}_{B}R\,{}^{B}Q\right) \qquad \text{Ec. (2.46)} $$

Ahora aplicamos la ecuación (2.45) dos veces; una al primer término y otra al último término. El lado derecho de la ecuación (2.46) se convierte en:

$$ {}^{A}_{B}R\,{}^{B}\dot{V}_{Q} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}V_{\zeta} + {}^{A}\dot{\Omega}_{B} * {}^{A}_{B}R\,{}^{B}Q + {}^{A}\Omega_{B}\left({}^{A}_{B}R\,{}^{B}V_{\zeta} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q\right) $$

Ec. (2.47)

Si combinamos dos términos obtenemos

$$ {}^{A}_{B}R\,{}^{B}\dot{V}_{\zeta} + 2\,{}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}V_{\zeta} + {}^{A}\dot{\Omega}_{B} * {}^{A}_{B}R\,{}^{B}Q + {}^{A}\Omega_{B}\left({}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q\right) \qquad \text{Ec. (2.48)} $$

Por último, para generalizar en el caso en que los orígenes no sean coincidentes, agregamos un término que proporciona la aceleración lineal del origen de {B}, lo que produce la siguiente fórmula general para el lado derecho de la ecuación:

$$ {}^{A}\dot{V}_{BORG} + {}^{A}_{B}R\,{}^{B}\dot{V}_{Q} + 2\,{}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}V_{\zeta} + {}^{A}\dot{\Omega}_{B} * {}^{A}_{B}R\,{}^{B}Q + {}^{A}\Omega_{B}\left({}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q\right) $$

Ec. (2.49)

Un caso particular que vale la pena mencionar es cuando ${}^{B}Q$ es constante, o: [36]

$$ {}^{B}V_{n} = 0 \qquad \text{Ec. (2.50)} $$

[36] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 166.

En este caso, la ecuación (2.49) se simplifica a:

$$ {}^{B}\dot{V}_{Q} = {}^{A}\dot{V}_{Borg} + {}^{A}\Omega_{B}\left({}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}Q\right) + {}^{A}\dot{\Omega}_{B} * {}^{A}_{B}R\,{}^{B}Q \qquad \text{Ec. (2.51)} $$

Se utiliza esta última ecuación para calcular la aceleración lineal de los vínculos de un manipulador con articulaciones giratorias. Cuando exista una articulación prismática se utilizará la forma más general de la ecuación (3.49).[37]

2.7.1.2 Aceleración angular.

Considere el caso en el que {B} está girando de manera relativa a {A} con ${}^{A}\Omega_{B}$, *y* {C} está girando de manera relativa a {B} con ${}^{B}\Omega_{C}$, Para calcular ${}^{A}\Omega_{C}$ sumamos los vectores en la trama {A}:

$$ {}^{A}\Omega_{C} = {}^{A}\Omega_{B}\omega + {}^{A}_{B}R\,{}^{A}\Omega_{B} \qquad \text{Ec. (2.52)} $$

Al diferenciar obtenemos:

$$ {}^{A}\dot{\Omega}_{C} = {}^{A}\dot{\Omega}_{B}\omega + \frac{a}{dt}\left({}^{A}_{B}R\,{}^{A}\Omega_{B}\right) \qquad \text{Ec. (2.53)} $$

Ahora, si aplicamos la ecuación (2.45) al último término de la ecuación (2.53), obtenemos

$$ {}^{A}\dot{\Omega}_{C} = {}^{A}\dot{\Omega}_{B} + {}^{A}_{B}R\,{}^{B}\dot{\Omega}_{C} + {}^{A}\Omega_{B} * {}^{A}_{B}R\,{}^{B}\Omega_{C} \qquad \text{Ec. (2.54)} $$

Utilizaremos este resultado para calcular la aceleración angular de los vínculos de un manipulador.

---

[37] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 167.

2.7.2 DISTRIBUCIÓN DE LA MASA

A menudo hablamos de la masa de un cuerpo rígido en sistemas con un solo grado de libertad; en el caso del movimiento giratorio sobre un solo eje es muy común el término momento de inercia. Para un cuerpo rígido que tiene la libertad de moverse en tres dimensiones, hay un número infinito de posibles ejes de rotación. En el caso de la rotación sobre un eje arbitrario necesitamos una forma completa de caracterizar la distribución de la masa de un cuerpo rígido.

Ahora definiremos un conjunto de cantidades que proporcionan información sobre la distribución de la masa de un cuerpo rígido con respecto a una trama de referencia. La figura 2.20 muestra un cuerpo rígido con una trama adscrita. Pueden definirse los tensores de inercia en relación con cualquier trama, pero siempre consideraremos el caso de un tensor de inercia definido para una trama unida al cuerpo rígido. Cuando sea importante indicaremos, con un superíndice a la izquierda, la trama de referencia de un tensor de inercia dado.

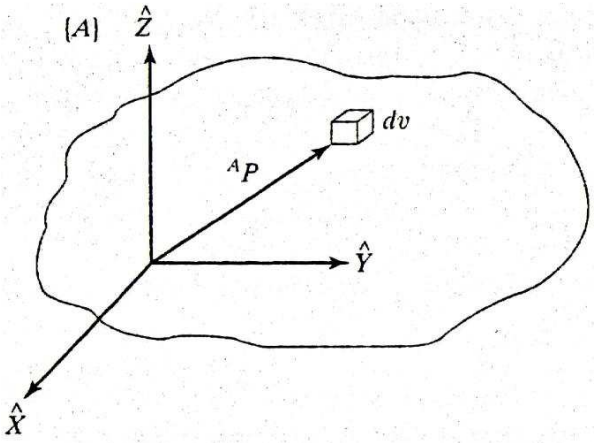


Figura 2.20 El vector ^{A}P posiciona el elemento de volumen diferencial, *dv*.

El tensor de inercia relativo a la trama {A} se expresa en forma matricial como la siguiente matriz de 3 x 3:

$$ {}^{A}I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \qquad \text{Ec. (2.55)} $$

En donde los elementos escalares están dados por:

$$ I_{xx} = \iiint_V (y^2 + z^2)\rho dv \qquad \text{Ec. (2.56)} $$

$$ I_{yy} = \iiint_V (x^2 + z^2)\rho dv \qquad \text{Ec. (2.57)} $$

$$ I_{zz} = \iiint_V (x^2 + y^2)\rho dv \qquad \text{Ec. (2.58)} $$

$$ I_{xy} = \iiint_V xy\rho dv \qquad \text{Ec. (2.59)} $$

$$ I_{xz} = \iiint_V xz\rho dv \qquad \text{Ec. (2.60)} $$

$$ I_{yz} = \iiint_V yz\rho dv \qquad \text{Ec. (2.61)} $$

Aquí, el cuerpo rígido está compuesto de elementos de volumen diferencial, *dv,* que contienen material de densidad ρ. Cada elemento de volumen está ubicado con un vector, ${}^{A}P=[x\ y\ z]^{T}$, como se muestra en la figura 2.20.

Los elementos I_{xx} , I_{yy} y I_{zz} se llaman momentos de inercia de masas. Observe que, en cada caso estamos integrando los elementos de la masa, *pdv,* multiplicados por los cuadrados de las distancias perpendiculares al eje correspondiente. Los elementos con índices mezclados se llaman productos de inercia de masas. Este conjunto de seis cantidades independientes dependerá, para un cuerpo dado, de la posición y orientación de la trama en la que estén definidas. Si podemos elegir la orientación de la trama de referencia, es posible hacer que los productos de inercia sean cero. Cuando se alinean así los ejes de la trama de referencia se llaman ejes principales, y los momentos de masas correspondientes son los momentos principales de inercia.

El tensor de inercia es una función de la ubicación y la orientación de la trama de referencia. El teorema de ejes paralelos es una manera de calcular

cómo cambia el tensor de inercia bajo las *traslaciones* del sistema de coordenadas de referencia. El teorema de ejes paralelos relaciona el tensor de inercia en una trama cuyo origen está en el centro de masa, con el tensor de inercia con respecto a otra trama de referencia. Cuando {*C*} se encuentra en el centro de masas del cuerpo y {*A*} es una trama trasladada arbitrariamente, el teorema puede definirse como:[38]

$$ {}^{A}I_{zz} = {}^{C}I_{zz} + m(x_c^2 + y_c^2) \qquad \text{Ec. (2.62)} $$

$$ {}^{A}I_{xy} = {}^{C}I_{xy} - m x_c \, y_c \qquad \text{Ec. (2.63)} $$

En donde $P_c = [x_c, y_c, z_c]^T$ ubica el centro de masas relativo a {*A*}; los momentos y productos de inercia restantes se calculan a partir de permutaciones de *x, y y z* en la ecuación (2.62). El teorema puede definirse en forma vectorial-matricial como:

$$ {}^{A}I = {}^{C}I + m[P_c^T P_c I_3 - P_c P_c^T] \qquad \text{Ec. (2.64)} $$

En donde I_3 es la matriz de identidad de 3 x 3.

2.7.3 FORMULACIÓN LAGRANGIANA DE LA DINÁMICA DE MANIPULADORES

Una alternativa para el método de Newton-Euler, es la formulación dinámica lagrangiana. Podría decirse que la formulación de Newton-Euler es un método de "balance de fuerzas" dinámicas, y que la formulación lagrangiana es un método de "balance de energías" de la dinámica. Desde luego, para el mismo manipulador, ambos métodos proporcionarán las mismas ecuaciones de movimiento.

La energía cinética del *i*-ésimo vínculo, que llamaremos k_i, puede expresarse como:

[38] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 168.

$$k_i = \frac{1}{2} m_i v_{ci}^T + \frac{1}{2} {}^i\omega_i^T {}^{Ci}I \, {}^i\omega_i \qquad \text{Ec. (2.65)}$$

En donde el primer término es la energía cinética debida a la velocidad lineal del centro de masas del vínculo, y el segundo término es la energía cinética debida a la velocidad angular del vínculo. La energía cinética total del manipulador es la suma de la energía cinética en los vínculos individuales; esto es:

$$k = \sum_{i=1}^{n} k_i \qquad \text{Ec. (2.66)}$$

Las v_{ci} y ${}^i\omega_j$ en la ecuación (2.65) son funciones de Θ y de $\dot{\Theta}$, por lo que podemos ver que la energía cinética de un manipulador puede describirse mediante una fórmula escalar como una función de la posición de articulación y la velocidad, $k(\Theta, \dot{\Theta})$. De hecho, la energía cinética de un manipulador se da mediante:

$$k(\theta, \dot{\theta}) = \frac{1}{2} \dot{\theta}^T M(\theta) \dot{\theta} \qquad \text{Ec. (2.67)}$$

En donde M(Θ) es la matriz de masas de *n* x *n* del manipulador en cuestión. Una expresión con la forma de la ecuación (2.67) se conoce como forma cuadrática, ya que cuando se expande, la ecuación escalar resultante será cuadrática y estará compuesta únicamente de términos $\dot{\Theta}_i$. Además, como la energía cinética total debe ser siempre positiva, la matriz de masas del manipulador debe ser una matriz definida positiva. Este tipo de matrices tienen la propiedad de que su forma cuadrática siempre es un escalar positivo. La ecuación (2.67) puede verse como análoga a la expresión familiar para la energía cinética de una masa puntual: [39]

$$k = \frac{1}{2} m v^2 \qquad \text{Ec. (2.68)}$$

[39] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 182.

El hecho de que la matriz de masas de un manipulador deba ser definida positiva, es análogo al hecho de que una masa escalar sea siempre un número positivo.

La energía potencial del *i*-ésimo vínculo, que llamaremos u_i puede expresarse como:

$$u_i = -m_i \; {}^0g^T \; {}^0P_{Ci} + u_{ref_i} \qquad \text{Ec. (2.69)}$$

En donde °g es el vector de gravedad de 3 x 1; °P_{Ci} es el vector que ubica al centro de masas del *i*-ésimo vínculo, y u_{refi} es una constante que se elige de manera que el valor mínimo de u_i sea cero. La energía potencial total almacenada en el manipulador es la suma de la energía potencial en los vínculos individuales; esto es:

$$u = \sum_{i=1}^{n} u_i \qquad \text{Ec. (2.70)}$$

Como las °P_C en la ecuación (2.69) son funciones de θ, podemos ver que la energía potencial de un manipulador puede describirse mediante una fórmula escalar como una función de la posición de articulación, $u(\Theta)$.

La formulación dinámica Lagrangiana provee los medios para derivar las ecuaciones de movimiento a partir de una función escalar llamada Lagrangiano, que se define como la diferencia entre la energía cinética y potencial de un sistema mecánico. En nuestra notación, el Lagrangiano de un manipulador es: [40]

$$\mathcal{L}(\theta, \dot{\theta}) = k(\theta, \dot{\theta}) - u(\theta) \qquad \text{Ec. (2.71)}$$

Y las ecuaciones de movimiento para el manipulador se dan entonces así:

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\theta}} - \frac{\partial \mathcal{L}}{\partial \theta} = \tau \qquad \text{Ec. (2.72)}$$

[40] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 183.

En donde τ es el vector *n x 1* de momentos de torsión de los actuadores. En el caso de un manipulador, esta ecuación se convierte en:[41]

$$\frac{d}{dt}\frac{\partial k}{\partial \dot{\Theta}} - \frac{\partial k}{\partial \Theta} + \frac{\partial u}{\partial \Theta} = \tau \qquad \text{Ec. (2.73)}$$

[41] CRAIG J., Robótica; Capítulo 6; 3ª edición; pág. 183.

CAPITULO III

PROGRAMACIÓN Y USO DEL ENTORNO DE MATLAB Y SIMULINK

3.1 INTRODUCCION AL SIMULINK

3.1.1 GENERALIDADES DE SIMULINK

Simulink es un software de interfaz gráfica que permite modelar, simular y analizar sistemas dinámicos. Lo que permite elaborar un modelo en primera parte y luego observar como este se comporta.

Con la ayuda de Simulink se puede construir modelos fácilmente desde cero o con modelos ya existentes. Además Simulink permite trabajar con sistemas lineales y no lineales, permitiéndonos implementar sistemas de control automático a estos.

Actualmente el uso de esta plataforma computacional se extiende a científicos e ingenieros de todo el mundo y en distintas aéreas como son:

Tecnología aeroespacial y de defensa.
Automatización
Comunicaciones
Electrónica y procesamiento de señales
Medicina e instrumentación

Se trata de un entorno gráfico donde un sistema se representa como una interconexión de bloques elementales, cada bloque lleva asociado un modelo matemático que representa su relación entrada/salida.

Un ejemplo de representación de un sistema mediante Simulink sería el siguiente:

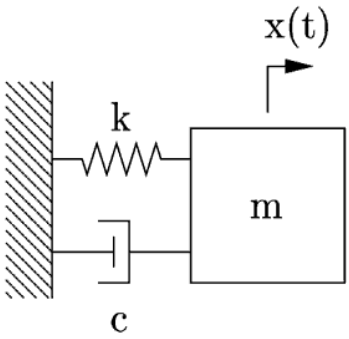


Figura 3.1 Sistema mecánico masa-resorte amortiguador.

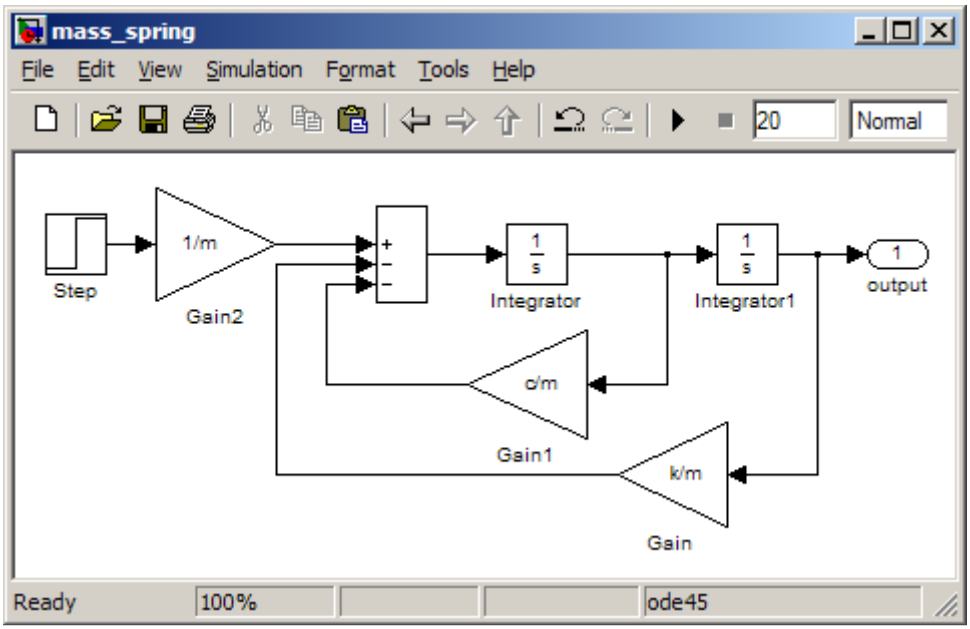


Figura 3.2 Representación del sistema mecánico de la figura anterior en Simulink.

3.1.2 SIMMECHANICS

El toolbox SimMechanics está especialmente destinado para el modelamiento de sistemas dinámicos y mecanismos como también sensores, actuadores, cuerpos o eslabones, pares cinemáticos, elementos de fuerza y añadir restricciones de movimiento entre elementos, etc. En fin todo lo necesario

para modelar mecanismos, simular su funcionamiento cinemático y dinámico. A continuación se muestra las librerías de SimMechanics:

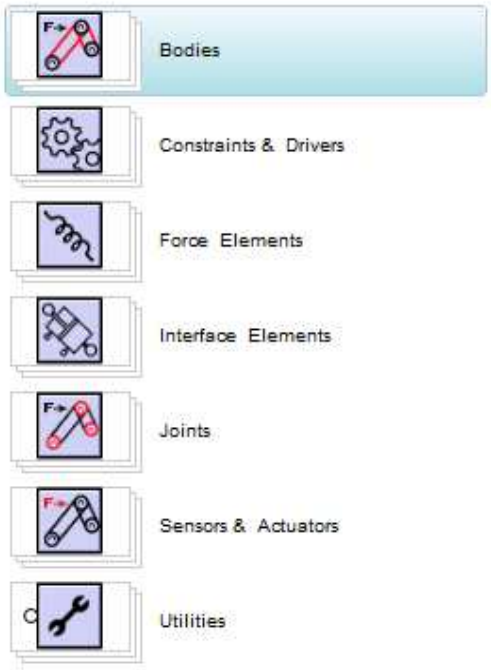


Figura 3.3 Conjunto de librerías SimMechanics.

A continuación se explica los bloques que serán utilizados.

3.1.2.1 Cuerpos (*Bodies*).

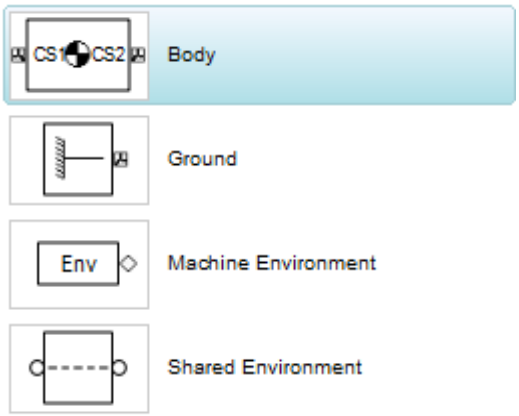


Figura 3.4 Librería cuerpos sólidos.

Esta librería contiene 4 tipos de bloque de los que ocuparemos los 3 primeros.

Body: este bloque al ser insertado cumple la función de un eslabón cualquiera con propiedades de masa, inercia y forma geométrica.

Ground: representa la bancada de un mecanismo en otras palabras es el eslabón fijo.

Machine environment: este bloque nos permite preestablecer las condiciones externas a las que la maquina o mecanismo se someterá durante su funcionamiento, por ejemplo la gravedad.

3.1.2.2 Uniones, juntas o pares cinemáticos (*Joints*).

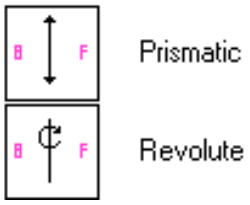


Figura 3.5 Librería de pares cinemáticas.

De esta librería se utilizará para el brazo robot los siguientes bloques:

Prismatic: este bloque corresponde a un par prismático, se ubicará entre dos bloques de eslabones para restringir el movimiento a prismático entre ambos cuerpos.

Revolute: par cinemático de revolución, cumple la función de articulación.

3.1.2.3 Sensores y actuadores (*Sensors and actuators*).

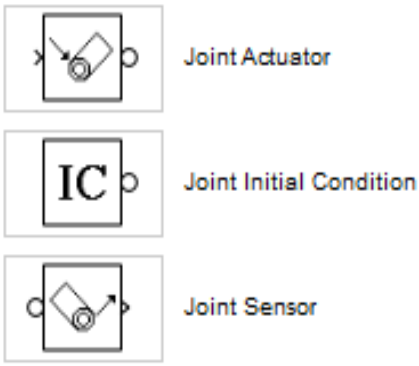


Figura 3.6 Librería de sensores y actuadores.

De la librería de sensores y actuadores usaremos los siguientes:

Joint actuator: este bloque funciona como un motor o actuador el cual proporcionará movimiento a un eslabón, se conecta al par cinemático que contiene a los dos eslabones y el movimiento se transmitirá al eslabón que se encuentra a la salida del par cinemático.

Joint sensor: este el bloque que funciona como sensor de movimiento o llamado en robótica como *encoder*, pueden medir velocidad, aceleración y posición lineal y angular. Se coloca en el par cinemático conectado a los dos eslabones que se quiere medir.

Joint initial condition: establece la posición inicial de un eslabón con la que debe arrancar en tiempo de simulación.

3.1.3 ELABORACIÓN DE UN MODELO DINÁMICO SENCILLO: EL PÉNDULO SIMPLE.

A continuación se mostrará el procedimiento para simular un sistema dinámico a partir de un sistema físico en este caso un péndulo simple. En la siguiente figura se muestra el sistema físico a simular.

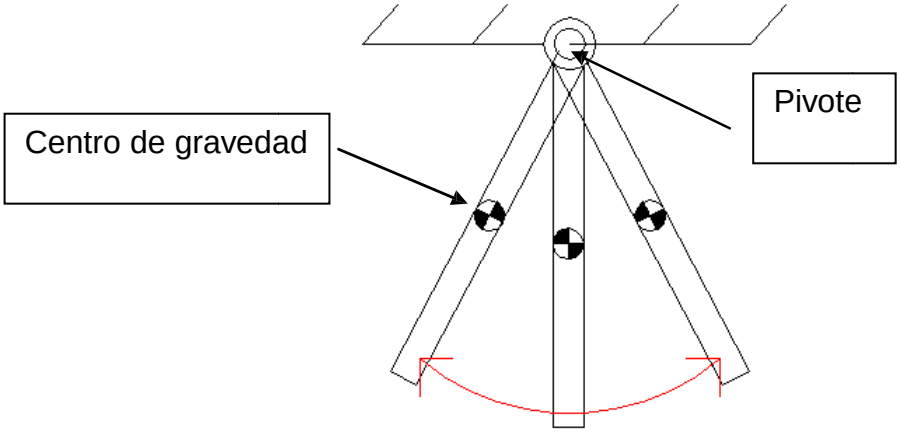


Figura 3.7 Movimiento del péndulo

Para elaborar este modelo en Simulink se deberá en primer lugar crear el diagrama de bloques correspondiente que representara cada parte de este sistema físico como son:

Bancada, base o tierra.
Articulación.
Eslabón o péndulo.

Al ser este un sistema dinámico se debe establecer los parámetros físicos para que Simulink pueda simular el comportamiento cinemático y dinámico de este. Para lo cual se define parámetros del péndulo como son:

Masa.
Posición del centro de gravedad.
Dimensiones.
Matriz de distribución de masas (inercias).

Durante la simulación se requerirá medir la posición, velocidad y aceleración angular que adquiera el péndulo en función del tiempo y visualizarla ya que como es lógico este oscilará respecto a la articulación. Para esto añadiremos al modelo:

Un sensor de movimiento.

Un visualizador de señales.

Antes de arrancar la simulación es conveniente establecer las condiciones iniciales del estado cinemático del péndulo para lo cual se define:

Posición angular inicial.

Velocidad angular inicial.

Esta serie de parámetros se definirán en el transcurso de la elaboración del modelo físico en Simulink.

Elaboración del modelo físico en Simulink:

Como primer paso se localiza el toolbox *SimMechanics* y acudimos a *new model* para crear un nuevo modelo.

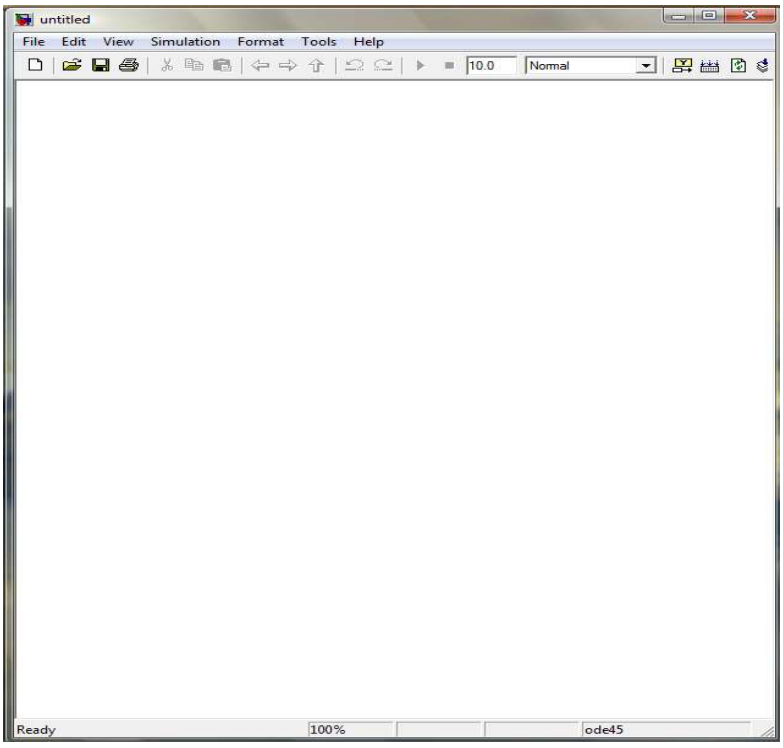


Figura 3.8 Creación de nuevo modelo en Simulink.

Se procede a insertar los elementos haciendo click sin soltar en uno de los bloques y arrastrando hasta llevarlos a la ventana del nuevo modelo a los siguientes elementos: *Ground*, *Machine enviroment* y *Body* desde el submenú *Bodies*, como se indica en la figura 3.9.

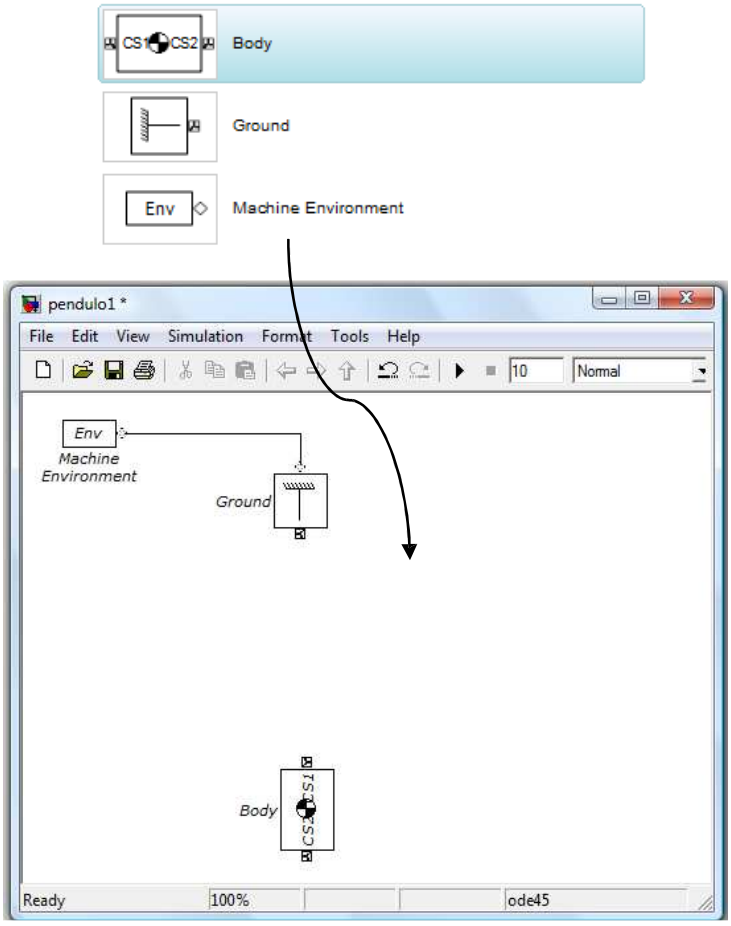


Figura 3.9 Inserción bloques de cuerpos sólidos.

Ahora se conecta los bloques, para esto se hace doble click en *Ground* y se hace visto en la casilla de *machine enviroment* como indica la figura 3.10 para que aparezca el puerto de entrada para este elemento y se hace click en aceptar.

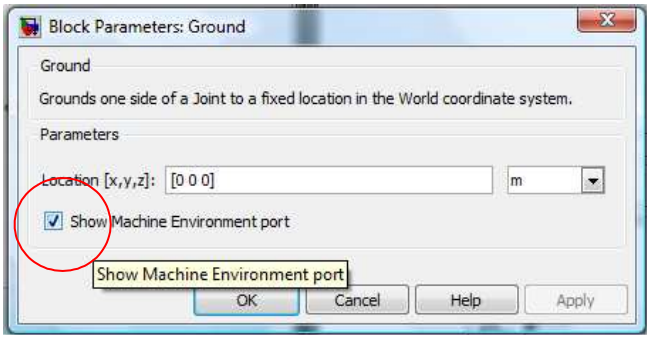


Figura 3.10 Casilla de *machine enviroment.*

Ahora se inserta el bloque *Revolute* desde el submenú *Joints* y se procede a conectar los bloques como se muestra en la figura 3.11:

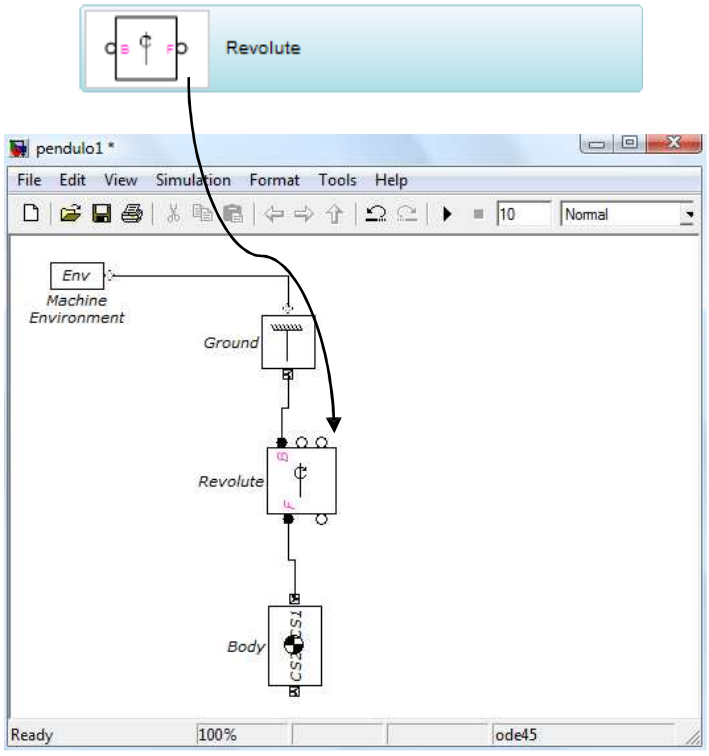


Figura 3.11 Integración de un bloque de pares cinemáticos.

Desde el submenú *Sensors and actuators* se inserta los siguientes bloques, como indica la figura 3.12:

Joint sensor.

Joint initial condition.

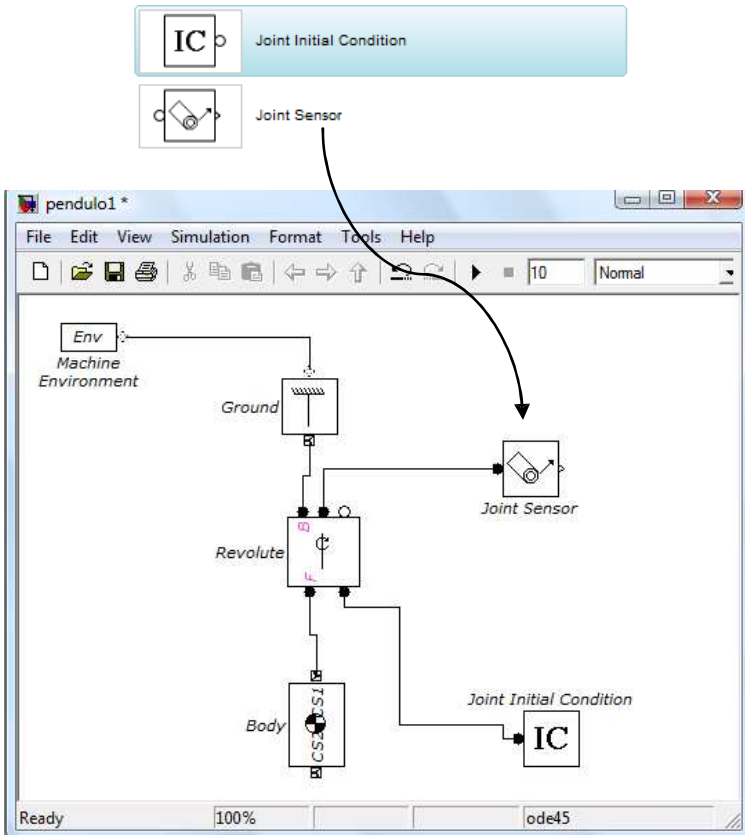


Figura 3.12 Integración de bloques de sensores y actuadores.

Ahora se hace doble click en el bloque *revolute* y en el campo *sensors* and *actuators* se ingresa el número 2 (uno para la condición inicial y otro para el sensor del par) como indica la figura 3.13.

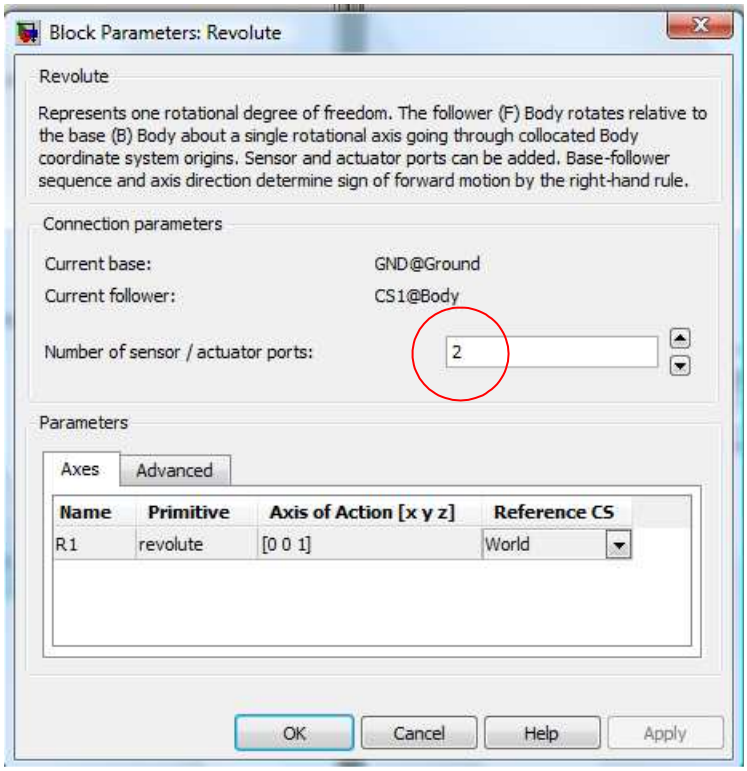


Figura 3.13 Determinación de entradas para sensores y actuadores.

Por último insertamos un *element scope* desde *sinks*:

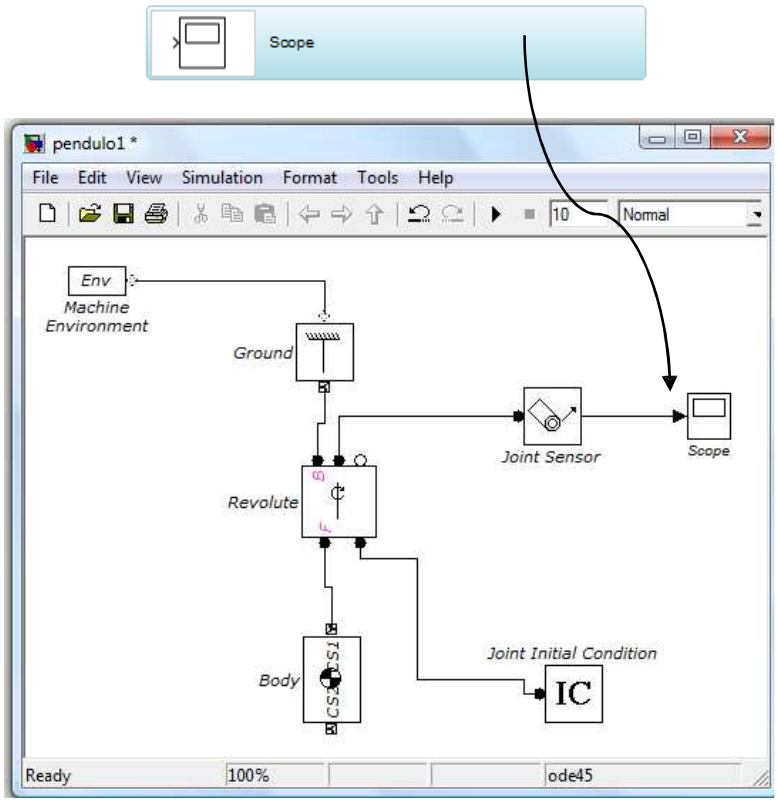


Figura 3.14 Integración de un bloque de visualización de datos *Scope*.

Una vez completado el conjunto de bloques del sistema dinámico procedemos a configurar cada bloque con los parámetros físicos antes mencionados para que nuestro sistema responda correctamente.

3.1.3.1 Configuración del bloque de bancada *Ground*.

En elemento *Ground* debemos especificar cuál es la ubicación espacial de este como referencia para todo el mecanismo que se conectará a este.

Establecemos la posición de este en el espacio, para lo cual hacemos doble clic en el bloque *Ground* y llenamos el campo *Location* con la matriz [0 0 0] lo que indica que la bancada se encuentra en el inicio de los ejes de coordenadas *X Y Z.* Hecho esto cerramos la ventana presionando el botón OK.

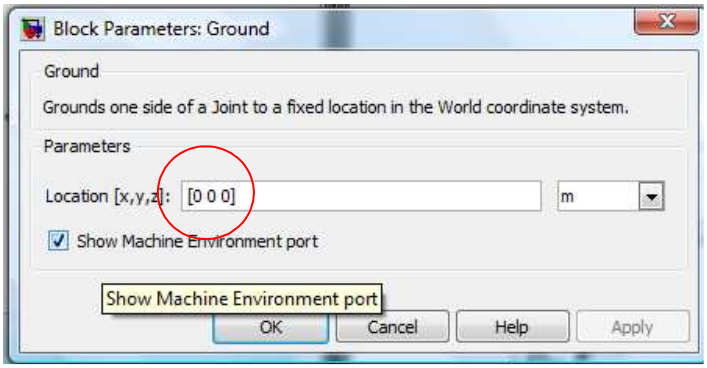


Figura 3.15 Configuración del bloque de bancada.

3.1.3.2 Configuración bloque *Revolute*.

En el par de revolución debemos ingresar la matriz de rotación que indica rotación unidireccional del cuerpo con el que se enlazara.

Hacemos doble clic en el bloque *Revolute* y en la ventana que aparece muestra la dirección en la que el par de articulación mediante una matriz fila. Se debe asegurar que el eje que corresponda al de rotación tenga el número 1 y el resto de ejes tenga 0. En este caso la matriz es [0 0 1] para establecer que el eje de rotación es el eje *Z*.

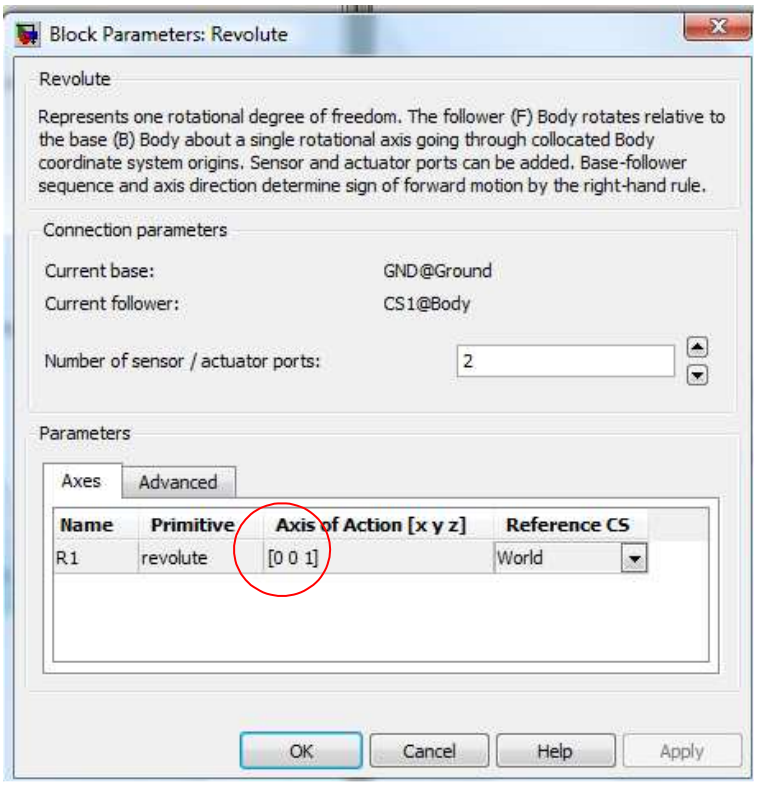


Figura 3.16 Configuración del par cinemático.

3.1.3.3 Configuración elemento *Body*.

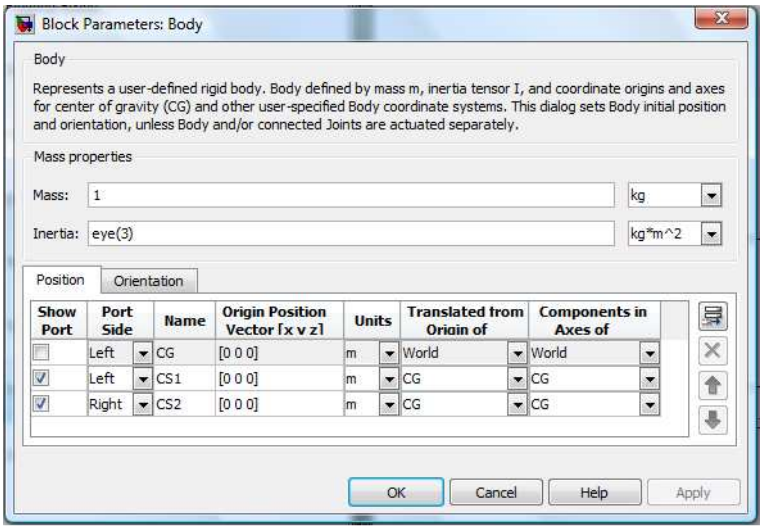


Figura 3.17 Configuración de un eslabón.

El elemento *body* contendrá los datos que definen completamente al cuerpo para la simulación dinámica y cinemática posterior como son masa y

matriz de distribución de masa del cuerpo con sus respectivas unidades. Además de las coordenadas de los puntos donde se unen el par cinemático y este cuerpo, el centro de gravedad del cuerpo y el punto con el que se une a otros cuerpos o pares cinemáticos. Este bloque es el eslabón de nuestro mecanismo en la siguiente figura se muestra físicamente la disposición del eslabón.

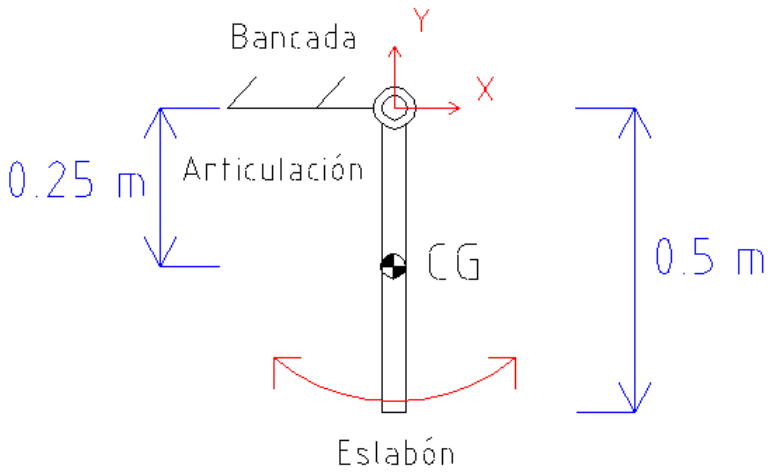


Figura 3.18 Sistema dinámico péndulo simple.

Para modelar el péndulo consideramos a este como un cilindro de longitud 50 cm, 1 cm de diámetro una masa de 5 Kg y la siguiente la matriz de distribución de masas o tensor de inercia:

$$I = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}$$

Se ingresa esta matriz de distribución de masa en el campo *Inertia* como se indica en la figura 3.19.

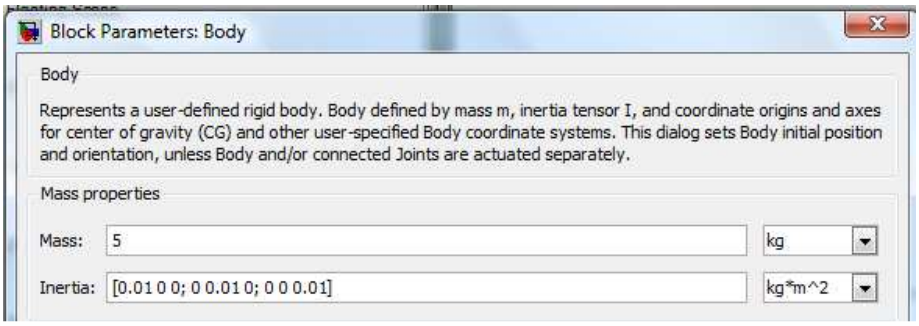


Figura 3.19 Datos de masa e inercia del eslabón.

Ahora necesitamos establecer las dimensiones exteriores del eslabón especificando donde empieza y dónde termina, para lo cual en el campo CG (centro de gravedad) ingresamos la matriz de posición del centro de gravedad respecto al origen de coordenadas *X Y Z*, esta matriz es [0 -0.25 0] que indica que el centro de gravedad se encuentra precisamente en la mitad del eslabón ya que este tiene una longitud de 50 cm. Tomar en cuenta que el signo negativo en el eje *Y* indica que el eslabón se encuentra hacia debajo de la bancada al igual que un péndulo y el número 0.25 se debe a que las unidades de este campo se han dejado por defecto en metros. En el campo *CS1* se establece la posición del inicio del eslabón respecto al origen de coordenadas *X Y Z*, esta matriz es [0,0,0] lo que indica que en inicio del eslabón coincide con el origen de las coordenadas *X Y Z* y el bloque de bancada que también coincide con el origen de coordenadas.

En el campo *CS2* se define en donde termina el eslabón respecto al sistema global de coordenadas, debido a que el eslabón está orientado hacia abajo respecto a la bancada y la longitud del eslabón es de 50 cm se concluye que esta matriz es [0 -0.5 0]. Estas posiciones se ingresan como indica la figura 3.20.

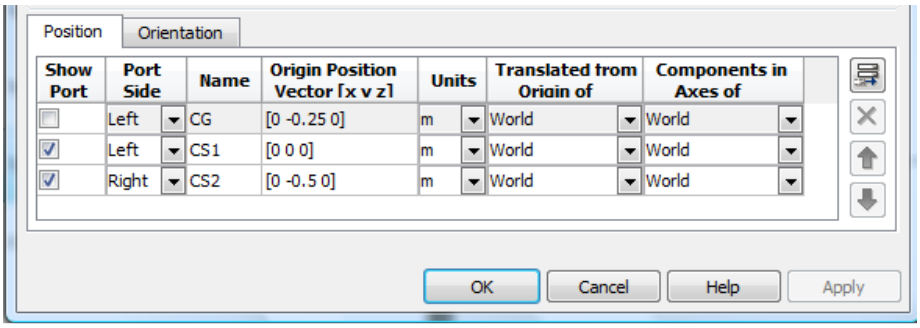


Figura 3.20 Configuración sistema de coordenadas relativas.

3.1.3.4 Configuración elemento *Initial condition*.

Al ejecutar la simulación del sistema dinámico péndulo simple es conveniente definir que este parta de una posición inicial para lo cual sirve el bloque *Initial condition* el cual establece la posición angular inicial del eslabón al iniciar la simulación.

Especificamos una posición angular inicial de 45 grados como indica la figura 3.21.

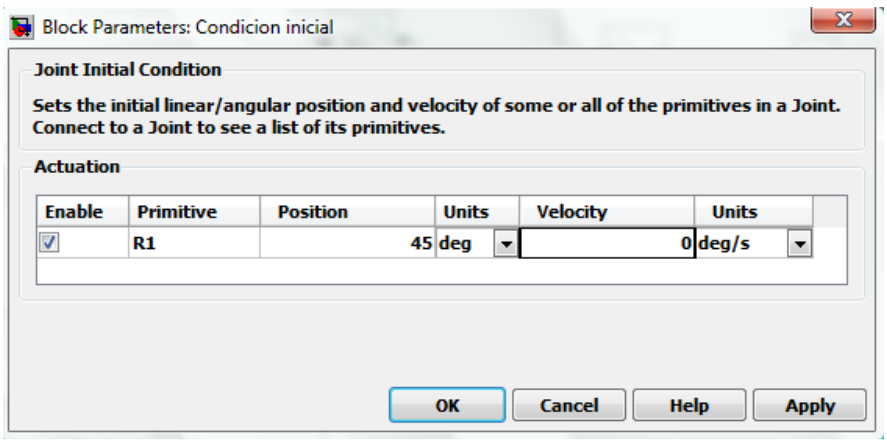


Figura 3.21 Configuración de bloque de condición inicial.

3.1.3.5 Configuración bloque *Machine environment*.

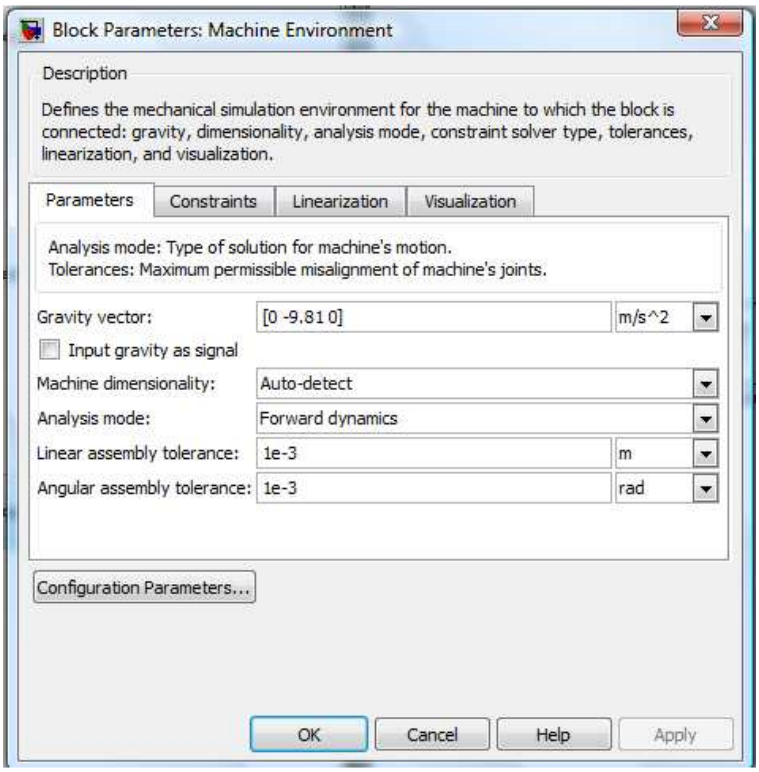


Figura 3.22 Configuración de bloque de entorno de máquina.

Este bloque permite establecer las condiciones externas a las que el sistema dinámico será sometido durante su funcionamiento como por ejemplo la fuerza de gravedad (dirección y magnitud).

Dada la disposición de nuestro mecanismo debemos establecer la gravedad en dirección de eje Y negativo (ver figura 3.22).

Para finalizar la construcción se debe determinar si durante la simulación se requiere visualizar el mecanismo en movimiento. Para esto nos dirigimos al menú de comandos de la ventana del modelo actual de Simulink y se hace click en *Simulation* y se selecciona la opción *Configuration parameters* como se indica en la figura 3.23.

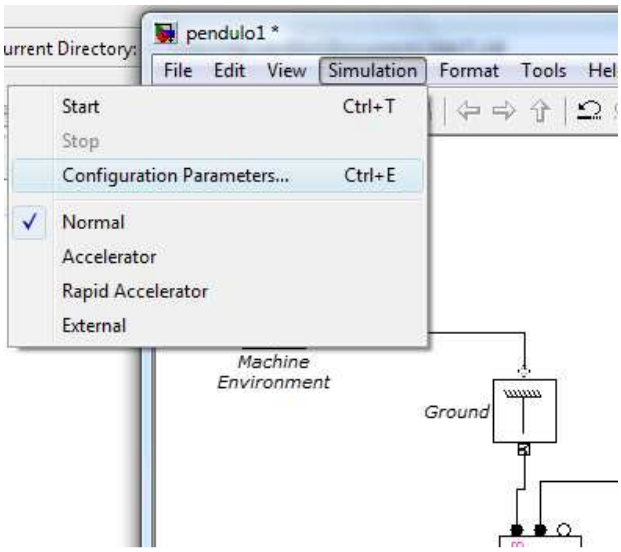


Figura 3.23 Menú simulación-configuración de parámetros.

Como resultado se abre una ventana como se indica en la figura 3.24 donde al final del menú de opciones se encuentra *Simscape-SimMechanics* donde se hace click, luego en el lado derecho de la ventana se muestran las opciones de configuración y hacemos visto en el recuadro que corresponde a la opción *Display machines after updating diagram* para poder visualizar el mecanismo durante la simulación. Para terminar pulsamos OK.

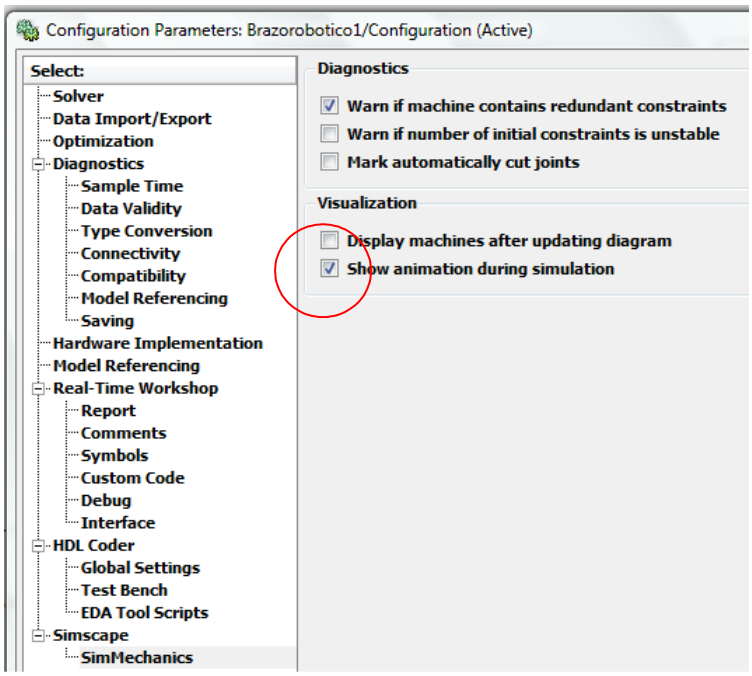


Figura 3.24 Menu Simscape-SimMechanics.

En este punto ha finalizado la construcción del modelo dinámico en *Simulink*. A continuación ejecutamos la simulación para esto presionamos el botón *Run* que se encuentra en la parte superior de la ventana que contiene el modelo como indica la figura 3.25 y hacemos doble clic en el bloque *Scope* para observar la señal de salida, al arrancar la simulación se mostraran las ventanas de las figuras 3.26 y 3.27:

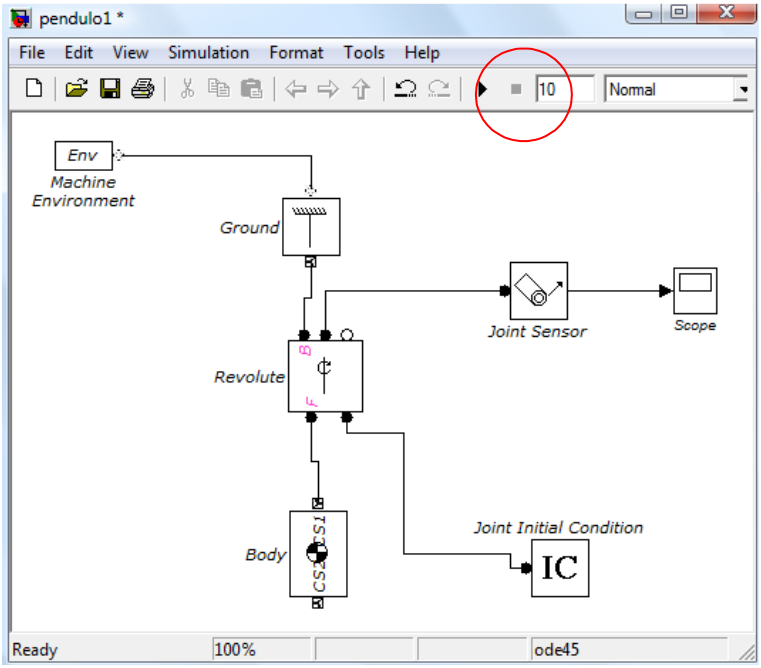


Figura 3.25 Ejecución de simulación de modelo péndulo simple.

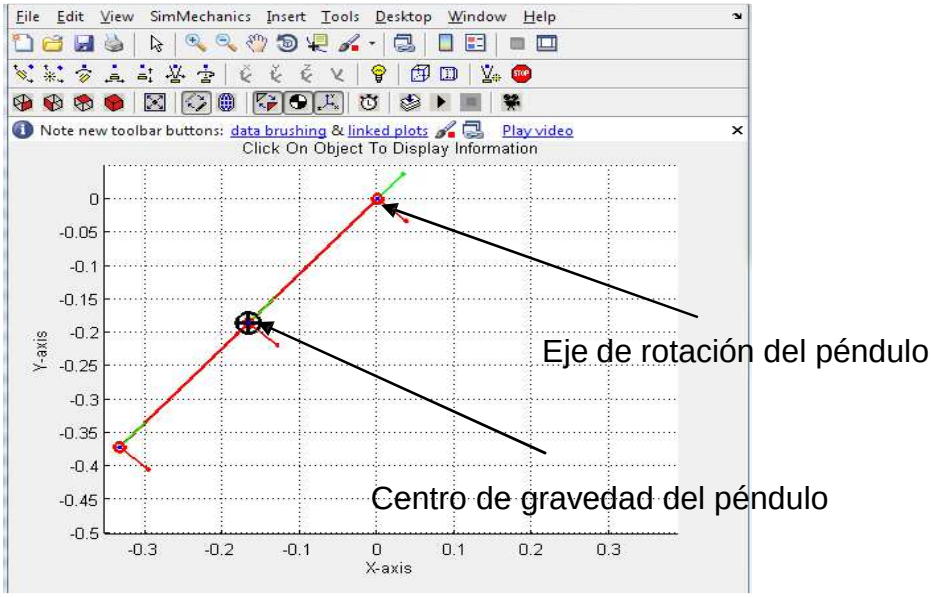


Figura 3.26 Vista de mecanismo péndulo simple en funcionamiento.

La figura 3.26 muestra una pantalla de la simulación del péndulo simple oscilante a velocidad angular constante debido a que se despreciaron fuerzas no conservativas. El cuerpo del péndulo está representado por la línea roja, tanto el eje de rotación como el centro de gravedad se han identificado en la figura. El tiempo de simulación será el mismo que se configuró desde la ventana del modelo de Simulink que fueron 10 segundos, es decir durante 10 segundos se observará el movimiento de oscilación del péndulo. Se puede notar que el eje de rotación del péndulo tiene como posición el punto *X*=0, *Y*=0. Para comprender mejor el movimiento del sistema dinámico se hace doble click en el bloque de sensor y se observará la ventana de la figura 3.27.

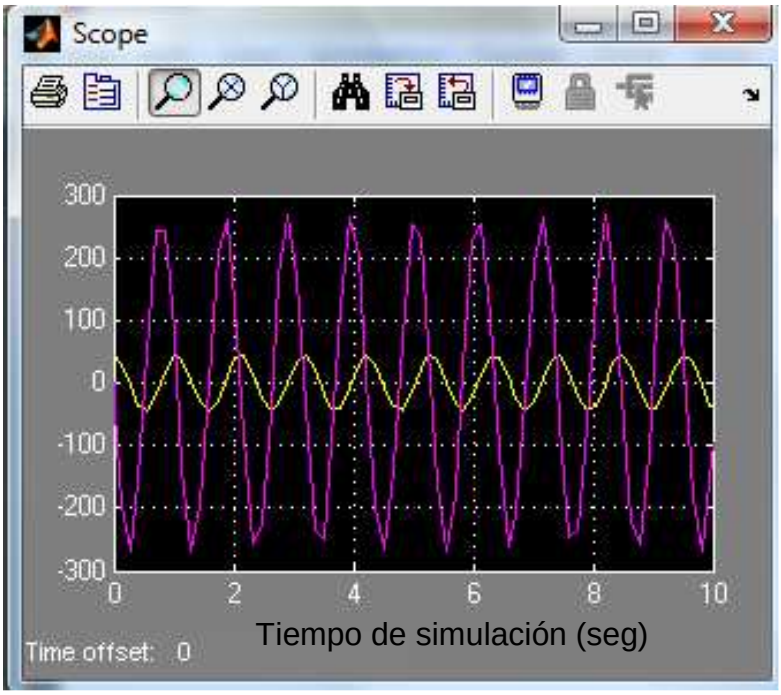


Figura 3.27 Señales de posición angular (grados) y velocidad angular (grados por segundo).

La figura 3.27 representa los resultados de posición (señal amarilla) y velocidad angular (señal púrpura) medidos por el sensor colocado en el par cinemático de revolución. Este es un gráfico de ángulo (grados) vs Tiempo de simulación (seg). Como se puede observar la señal de posición angular (color amarillo) oscila entre 45 grados positivos (cuando el péndulo esta a la derecha) y

45 grados negativos (péndulo hacia la izquierda), considerando que cero grados es cuando el péndulo está completamente vertical.

La señal de velocidad angular (color púrpura) varía entre 240º por segundo y -240º por segundo, la máxima velocidad absoluta se da cuando la posición del péndulo es totalmente vertical (señal amarilla en cero) ya que en este punto empieza a desacelerar el péndulo. De igual manera cuando la velocidad es cero la posición tendrá su máximo valor ya que este será el punto máximo que alcanzará el péndulo y empezará a moverse en sentido contrario. De esta manera se relacionan la posición y velocidad angular del péndulo según la figura 3 .27.

3.1.4 TOOLBOX HEMERO

El toolbox *Hemero* que se encuentra disponible en la dirección electrónica: http://grvc.us.es/hemero/descargar.html, dicho toolbox será descomprimido y copiado a la carpeta *Toolboxes* donde se encuentre instalado el Matlab, luego se agrega la extensión de su ubicación en la opción *Set path* para que sea reconocido por el programa al arrancar, por último se reinicia el programa y el toolbox *Hemero* será automáticamente detectado. Para empezar su uso se debe digitar *Hemero* en el *Comand window* de Matlab. Para que el toolbox *Hemero* pueda funcionar se debe contar con una versión de Matlab 5.2 o posteriores.

El Toolbox *Hemero* es una poderosa herramienta para la solución de complejos problemas que se presentan durante el diseño de un robot tanto de base fija como móvil. Tiene entre sus posibilidades la capacidad de almacenar los parámetros de Denavit-Hartenberg para su posterior análisis u operaciones y la capacidad de resolver la cinemática inversa de un manipulador en base a sus parámetros Denavit-Hartenberg que hayan sido definidos previamente y una trayectoria definida por el usuario.

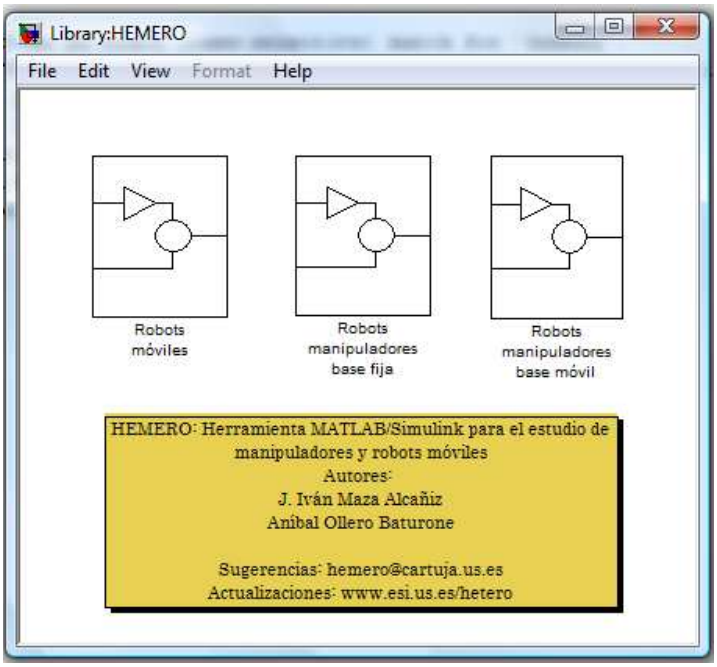


Figura 3.28 Toolbox Hemero.

3.1.4.1 Comandos del toolbox Hemero.

Comando dh: el propósito de este comando es contener los parámetros de denavit-Hartemberg del manipulador. Tendrá el formato de una matriz. Habrá una fila por cada enlace o par cinemático que tenga el manipulador, el siguiente esquema indica el formato:

$$\begin{bmatrix} \dots & \dots & \dots & \dots & \dots \\ \alpha_{i-1} & a_{i-1} & \theta_i & d_i & \sigma_i \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

Donde $\boldsymbol{\alpha_{i-1}, a_{i-1}, \theta_i, d_i}$: son los parámetros de Denavit-Hartenberg anteriormente definidos, y σ_i indicará el tipo de articulación (0 si es de rotación o cualquier otro numero si por el contrario es prismática). Entonces la matriz quedaría algo como:

$$\begin{bmatrix} \alpha_0 & a_0 & \theta_1 & d_1 & \sigma_1 \\ \alpha_1 & a_1 & \theta_2 & d_2 & \sigma_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \alpha_{i-1} & a_{i-1} & \theta_i & d_i & \sigma_i \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \alpha_{n-1} & a_{n-1} & \theta_n & d_n & \sigma_n \end{bmatrix}$$

Es decir una matriz nx5, donde el subíndice i indica el i-ésimo par cinemático del manipulador.

Los valores de ángulos deberán ser ingresados en radianes y las longitudes en cualquier unidad.

Comando ikine: su propósito es calcular la cinemática inversa del manipulador. La función ikine devuelve los valores de las variables articulares necesarias para que el eslabón final del manipulador tenga la posición y orientación de la transformación T. La solución que arroje la función ikine no es única en general y es posible que para una misma orientación haya varias soluciones como se ha visto anteriormente. La solución obtenida dependerá también de los valores iniciales de posición y orientación dados al manipulador.

Si el valor T ingresado es una transformación homogénea, entonces la solución será un vector fila con las variables articulares para dicha posición y orientación. En caso de que T sea una trayectoria de transformaciones homogéneas que indiquen un conjunto de *n* posiciones y orientaciones, la solución será una matriz de *n x m* donde n será cada posición y orientación en el espacio que debe ocupar el efector final del manipulador y m será la variable articular relativa a cada par cinemático que tenga el manipulador.

A la matriz de valores articulares iniciales que se debe ingresar se le puede dar valores arbitrarios o simplemente valores nulos.

Para el caso de manipuladores con menos de 6 grados de libertad, el efector final no podrá alcanzar ciertos puntos de posición y orientación, esto lleva a una no convergencia de la función ikine. Para esto se debe especificar un vector de pesos M cuyos elementos serán 0 para los grados de libertad que en

cartesianas estén restringidos y 1 si fuera otro el caso. Los elementos de M corresponde a traslaciones en los ejes X, Y y Z y rotación en los ejes X, Y y Z.

La expresión *ilimit* representa el número máximo de iteraciones que se realizaran en búsqueda de una solución.

La expresión *stol* será el error máximo admisible entre los valores buscados y los valores encontrados en cuanto a transformaciones homogéneas.

La sintaxis para el uso de este comando será como sigue:

$$q = ikine(dh, stol, ilimit, q_0, M)$$

La matriz o vector q almacenará la solución.

Comando plotbot: su propósito es representar gráficamente el manipulador, es decir construye una representación gráfica del manipulador a partir de los parámetros cinemáticos contenidos en el comando *dh* y el vector de variables articulares que se proporciona.

El parámetro opt es una cadena que especifica las opciones de representación que se deseen de entre las siguientes:

d Establece que la representación se muestre en 2D (sobre el plano XY).

f Dibujar los cuadros de referencia asociados a cada articulación.

l No borrar el robot en cada paso si se trata de una animación.

r Repetir la animación 50 veces.

w Dibujar el sistema de referencia asociado al efector final.

Si no se usa el parámetro opt, se hace la representación convencional.

La sintaxis para el uso de este comando será por ejemplo:

$$plotbot(dh, q, 'dfw')$$

Para mayor información de los comandos disponibles se recomienda revisar el manual del Toolbox Hemero que viene incluido en el fichero descargable, este se compone de 8 capítulos que explican en detalle todo lo que es posible hacer con esta herramienta en idioma español.

CAPITULO IV

MODELAMIENTO DE UN BRAZO ROBOTICO CON 4 GRADOS DE LIBERTAD EN EL ENTORNO DE MATLAB PARA SOLADURA DE ARCO, Y DISEÑO DEL SISTEMA DE CONTROL AUTOMÁTICO.

4.1 PLANTEAMIENTO DEL PROBLEMA

Se requiere modelar un sistema brazo robot para que cumpla una tarea de soldadura controlada a través del ordenador.

El sistema brazo robot consiste en: su configuración mecánica, el dimensionamiento del modelo físico del robot, el modelo cinemático directo e inverso, el modelo dinámico y el sistema de control del robot.

La tarea de soldadura consiste en la elaboración de un cordón de suelda representado por la trayectoria que realizará virtualmente el robot en el ordenador.

Tarea específica:

Realizar un cordón de soldadura para unir una brida de acero a una sección de tubería de acero para construir una válvula tipo mariposa como indica la figura 4.1.

Figura 4.1 Válvula tipo mariposa.[42]

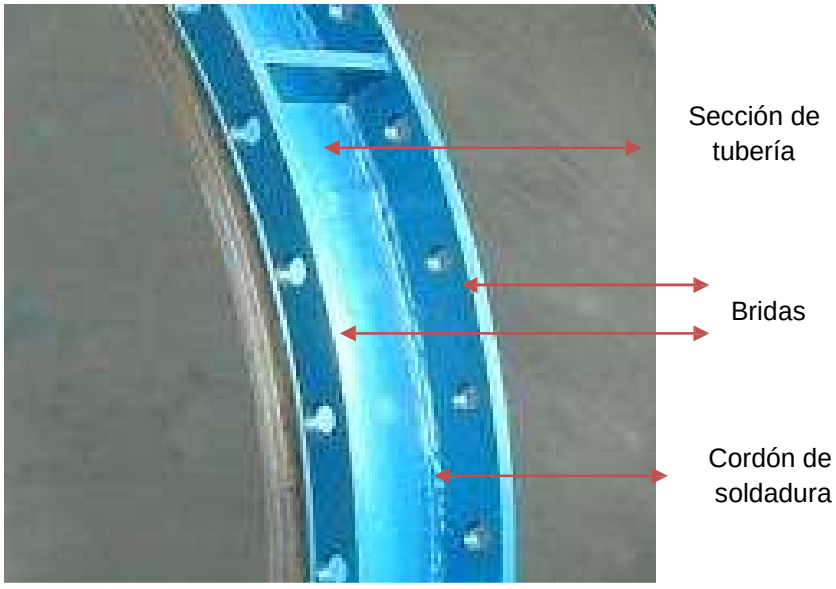


Figura 4.2 Ampliación de la válvula tipo mariposa.

Como se ve en la figura 4.2 la válvula está compuesta por dos bridas soldadas a una sección de tubería por una soldadura de arco. El diseño de esta

[42] http://www.made-in-china.com/image/2f0j00qMwEPZpgnDbeM/Flange-Type-Butterfly-Valve.jpg

brida se basa en la norma API 609 y la inspección posterior a la soldadura en la norma API 598.

Dimensiones de la brida:

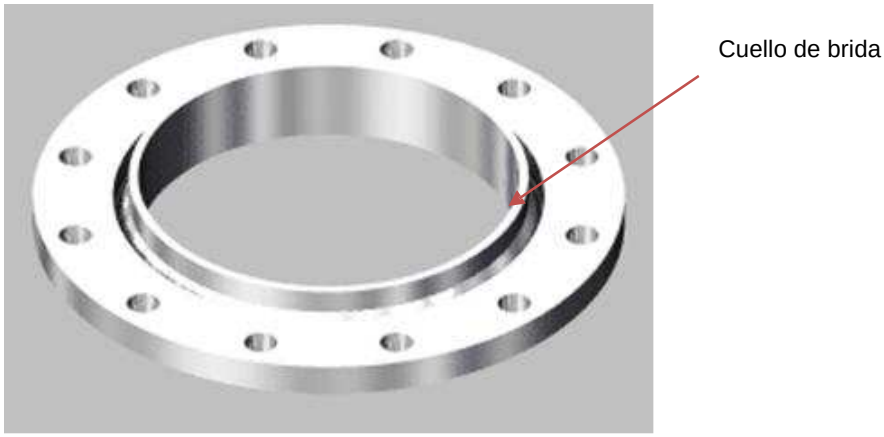


Figura 4.3 Brida de acero.

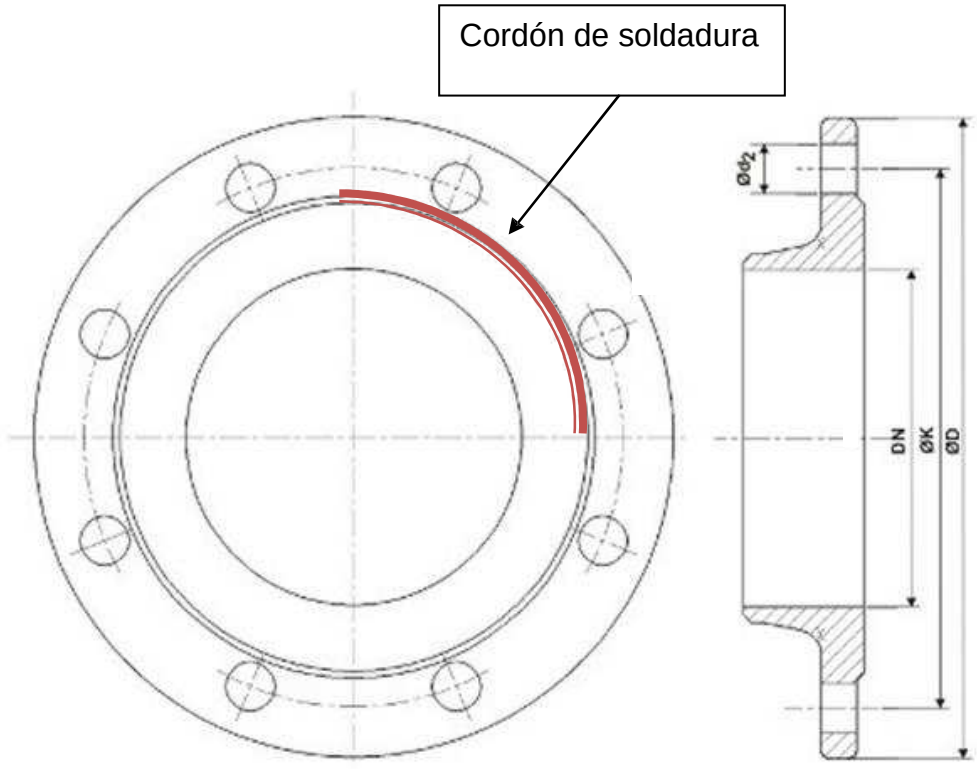


Figura 4.4 Dimensiones de la brida de acero y disposición de cordón de soldadura.

$$\emptyset d_2 = 5\ cm\ ,\quad DN = 200\ cm,\quad \emptyset K = 230 cm,\qquad \emptyset D = 250\ cm$$

Tomando en cuenta que entre cada brida y la sección de tubería se ubican 4 nervios de rigidez distribuidos equidistantemente, la circunferencia completa que representa el cordón de soldadura queda dividida en 4 partes iguales por lo que solo es necesario simular una de las cuatro partes es decir un arco de radio 100 cm.

Para llevar a cabo la simulación es necesario representar matemáticamente el arco de circunferencia mediante una función real, de tal manera el cordón de soldadura queda representado por la siguiente función:

$$f(x) = \sqrt[2]{10000 - x^2} \qquad \text{ec.(4.1)}$$

Definiéndose el dominio de *x* dentro de los límites:

$0 \leq x \leq 100$, unidades en centímetros.

Tarea de soldadura desde que el robot se posiciona en el punto de inicio hasta que termina de soldar.

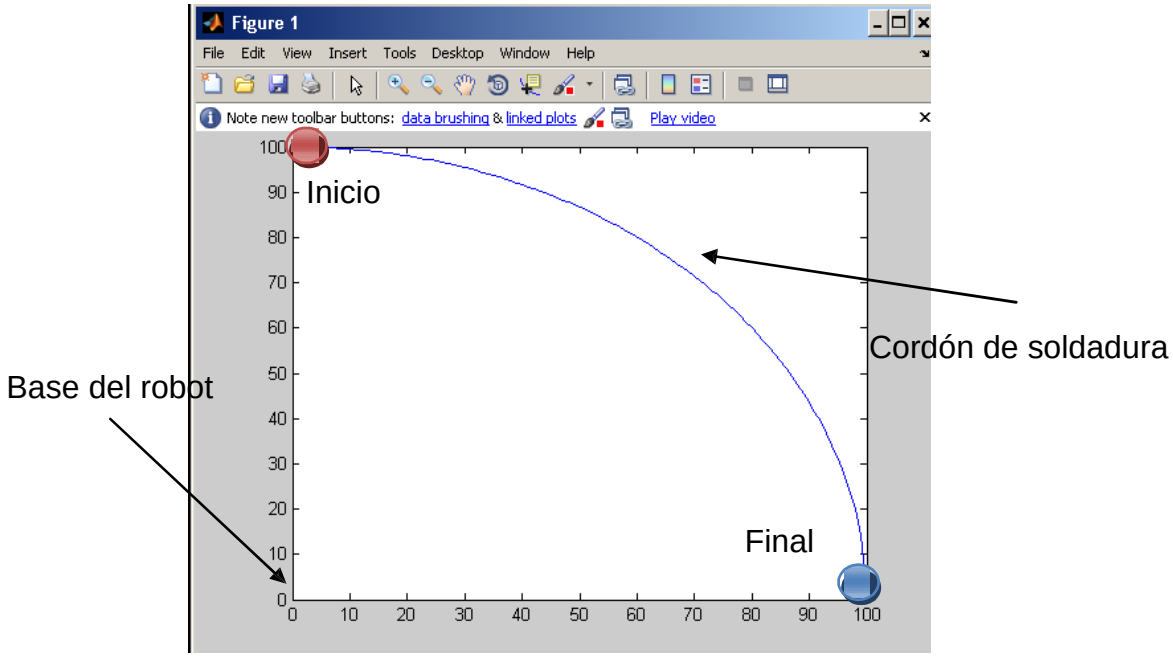


Figura 4.5 Función cartesiana que representa el cordón de soldadura.

La figura 4.5 muestra la función cartesiana antes mencionada que representara al cordón de soldadura.

Como se puede observar en la figura anterior el cordón de soldadura corresponde a la cuarta parte de una circunferencia entonces se puede calcular su longitud así:

$$longitud\ cordón = 2 * \pi * \frac{radio}{4} = 2 * \pi * \frac{100}{4} cm = 157.1\ cm$$

El plano de trabajo corresponde a la superficie donde se asentará la base del robot y que contiene la trayectoria a ser realizada. La figura 4.6 muestra el plano de trabajo:

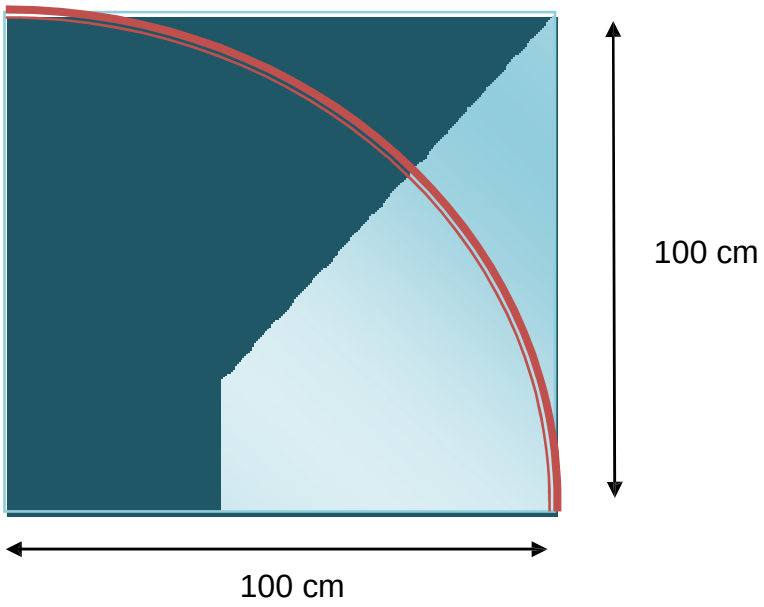


Figura 4.6 Área de trabajo.

Considerando que este es un trabajo de investigación se han establecido los siguientes parámetros los cuales se resumen en la figura 4.7:

Cordón a realizar	• Cordón de soldadura cuya trayectoria está definida por la función real: $\sqrt{(10000-x^2)}$, cuyos ejes cartesianos de origen se considera la base del robot.
Longitud de cordón de soldadura	• 157.1 cm
Velocidad de soldadura	• V=16 cm/seg (constante)
Stick out (altura de electrodo)	• Variación de 25mm a 50mm

Figura 4.7 Parámetros de soldadura.

4.2 SOLUCIÓN DE PROBLEMA.

4.2.1 SELECCIÓN DE CONFIGURACIÓN DEL ROBOT.

Cuando se habla de la configuración de un robot, se habla de la forma física que se le ha dado al brazo del robot.

Se empieza por definir la configuración mecánica del robot. El robot debe poseer los grados de libertad y movilidad necesaria para cumplir con el objetivo que se plantea. Para un robot existen las siguientes configuraciones posibles:

4.2.1.1 Tipos de configuraciones.

El brazo del manipulador puede presentar cuatro configuraciones clásicas: cartesiana, cilíndrica, polar y angular.

Configuración cartesiana:

La configuración tiene tres articulaciones prismáticas (3D o estructura PPP por sus tres pares prismáticos). Esta configuración es bastante usual en

estructuras industriales, tales como pórticos, empleadas para el transporte de cargas voluminosas. La especificación de posición de un punto se efectúa mediante las coordenadas cartesianas (x, y, z). Los valores que deben tomar las variables articulares corresponden directamente a las coordenadas que toma el extremo del brazo. Esta configuración no resulta adecuada para acceder a puntos situados en espacios relativamente cerrados y su volumen de trabajo es pequeño cuando se compara con el que puede obtenerse con otras configuraciones.

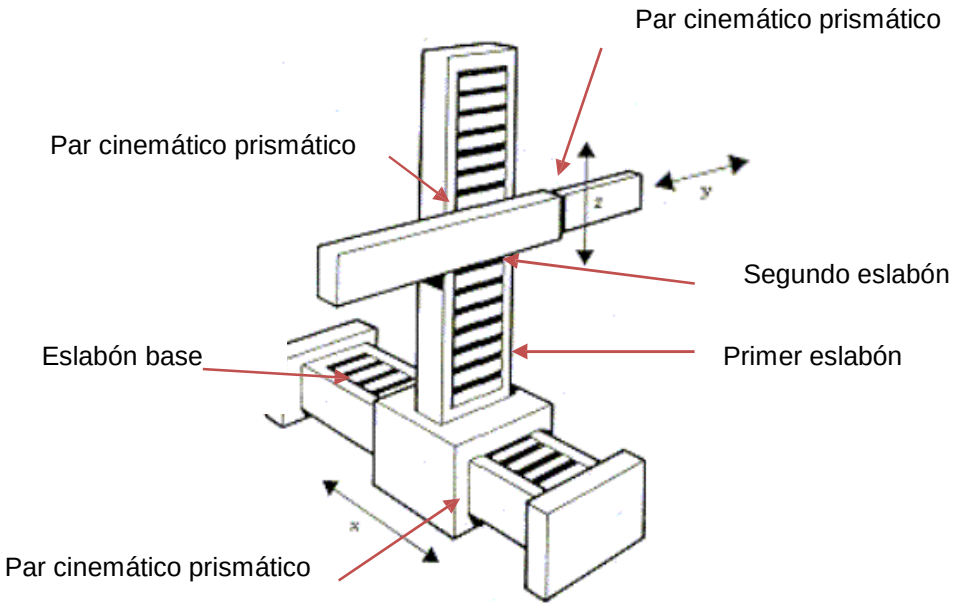


Figura 4.8 Robot cartesiano.

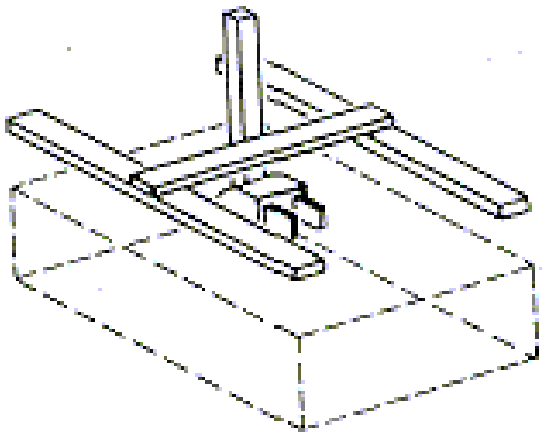


Figura 4.9 Volumen de trabajo para una configuración cartesiana.

Configuración cilíndrica:

Esta configuración tiene dos articulaciones prismáticas y una de rotación (2D, 1G). La primera articulación es normalmente de rotación (estructura RPP, un par rotacional-dos pares prismáticos). La posición se especifica de forma natural en coordenadas cilíndricas. Esta configuración puedes ser de interés en una célula flexible, con el robot situado en el centro de la célula sirviendo a diversas máquinas dispuestas radialmente a su alrededor. El volumen de trabajo de esta estructura RPP (o de la PRP), suponiendo un radio de giro de 360 grados y un rango de desplazamiento de L, es el de un toro de sección cuadrada de radio interior L y radio exterior 2L. Se demuestra que el volumen resultante es $3\pi L^3$.

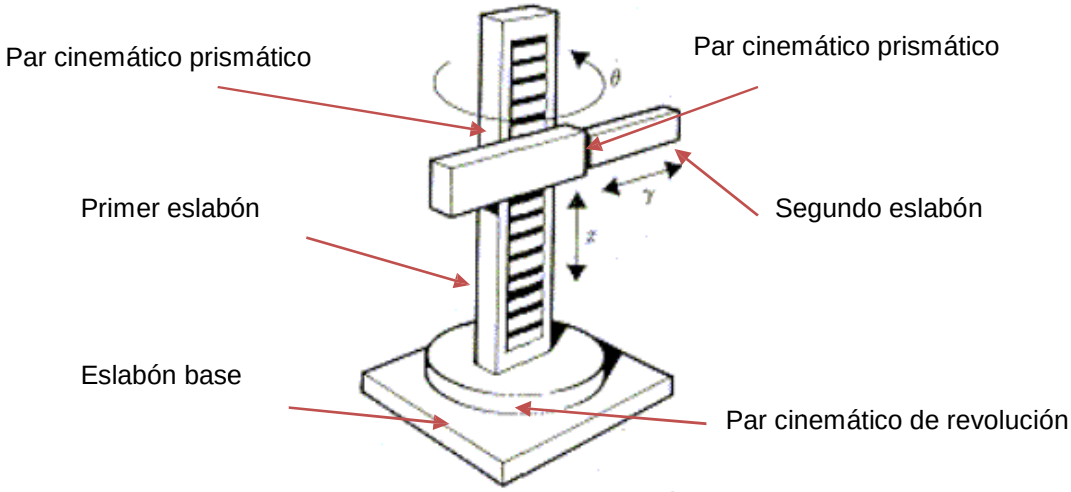


Figura 4.10 Robot cilíndrico.

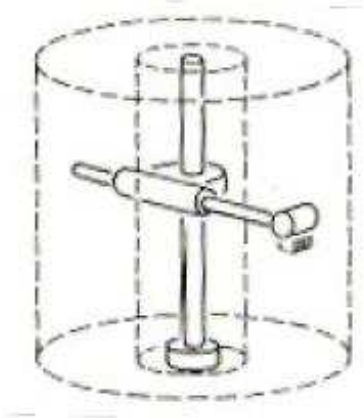


Figura 4.11 Volumen de trabajo para una configuración cilíndrica.

Configuración polar:

Está configuración se caracteriza por dos articulaciones de rotación y una prismática (2G, 1D o estructura RRP por dos pares rotacionales y uno prismático). En este caso las variables articulares expresan la posición del extremo del tercer enlace en coordenadas polares. En un manipulador con tres enlaces de longitud L, el volumen de trabajo de esta estructura, suponiendo un radio de giro de 360 grados y un rango de desplazamiento de L, es el que existe entre una esfera de radio 2L y otra concéntrica de radio L. Por consiguiente el volumen es $(28/3)\pi L^3$.

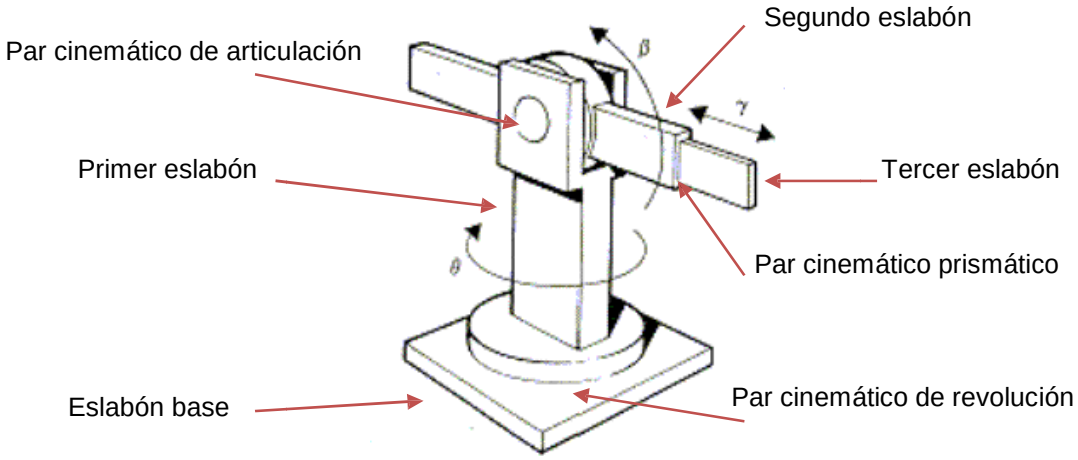


Figura 4.12 Robot polar.

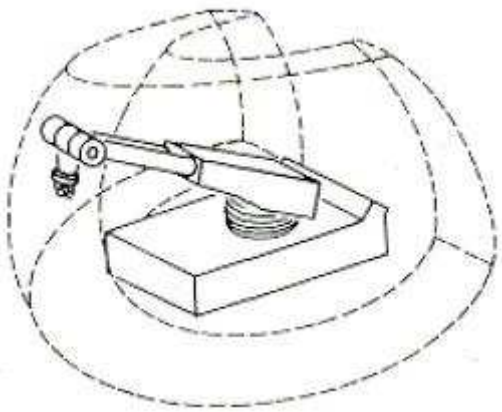


Figura 4.13 Volumen de trabajo para una configuración polar.

Configuración angular (o de brazo articulado):

Esta configuración es una estructura con tres articulaciones de rotación (3G o RRR). La posición del extremo final se especifica de forma natural en coordenadas angulares. La estructura tiene un mejor acceso a espacios cerrados y es fácil desde el punto de vista constructivo. Es muy empleada en robots manipuladores industriales, especialmente en tareas de manipulación que tengan una cierta complejidad. La configuración angular es la más utilizada en educación y actividades de investigación y desarrollo. En esta estructura es posible conseguir un gran volumen de trabajo. Si la longitud de sus tres enlaces es de L, suponiendo un radio de giro de 360 grados, el volumen de trabajo sería el de una esfera de radio 2L, es decir $(32/3)\pi L^3$.

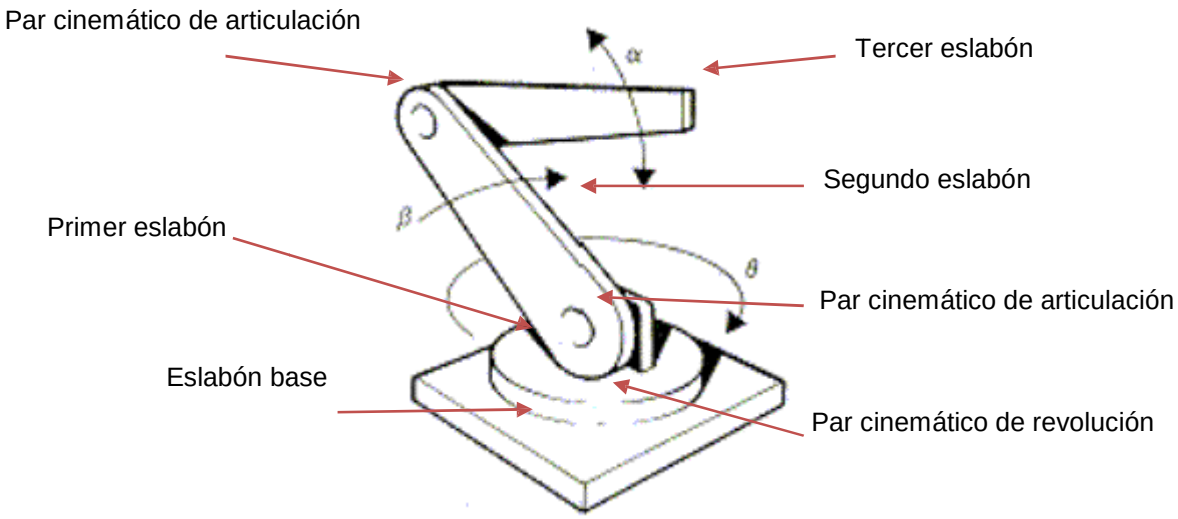


Figura 4.14 Robot angular.

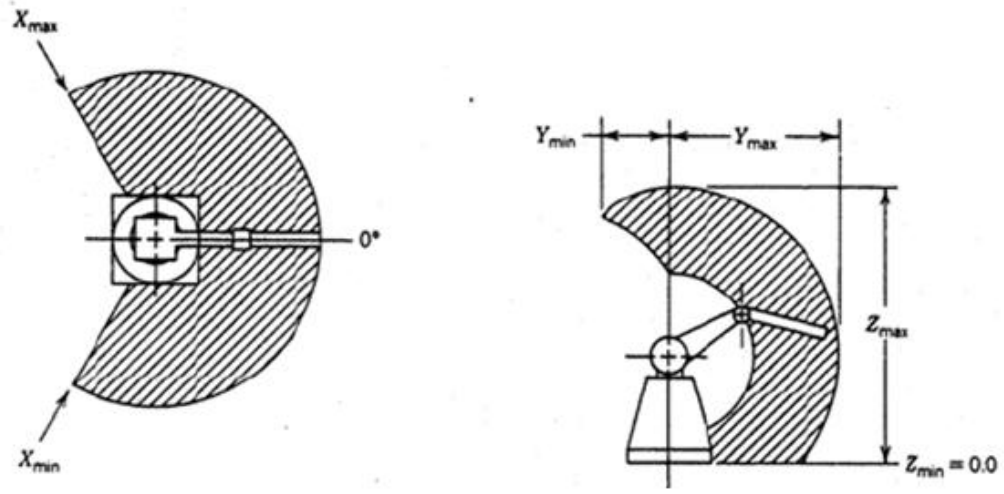


Figura 4.15 Volumen de trabajo de una configuración angular.

Además de las cuatro configuraciones clásicas mencionadas, existen otras configuraciones llamadas no clásicas.

El ejemplo más común de una configuración no clásica lo representa el robot tipo SCARA, cuyas siglas significan: Selective apliance arm robot for asembly.

Este brazo puede realizar movimientos horizontales de mayor alcance debido a sus dos articulaciones rotacionales. El robot de configuración SCARA también puede hacer un movimiento lineal (mediante su tercera articulación). Este tipo de robot sería muy factible en la soldadura bidimensional horizontal de no ser por la limitante que la propia forma del robot impone ya que en la práctica el acceso a la superficie a soldar no siempre es directo. Ejemplo de un robot SCARA en la figura 4.15:

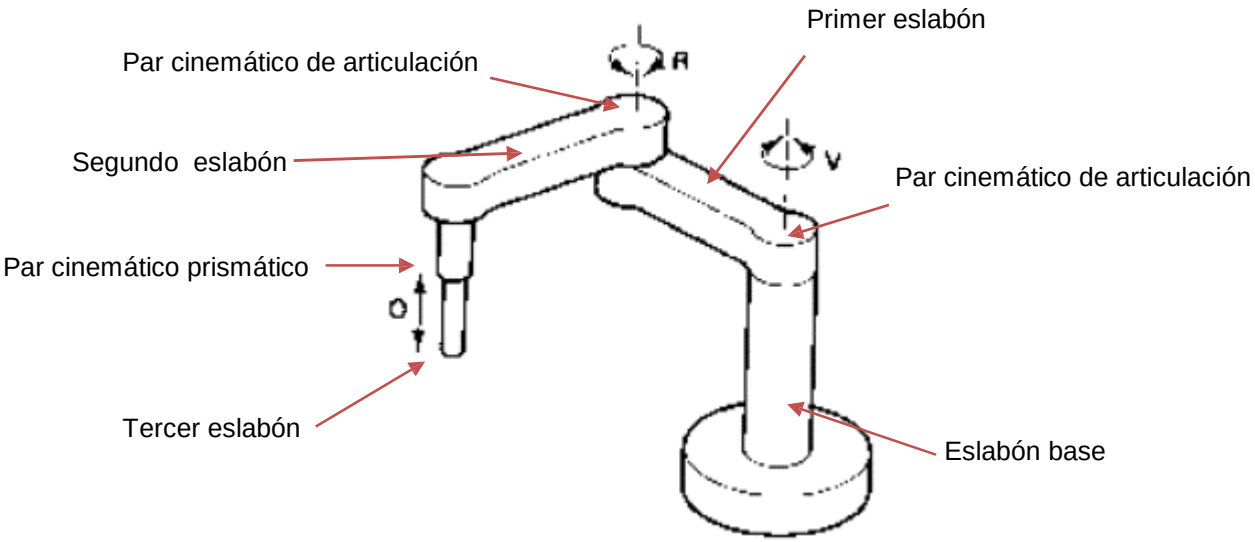


Figura 4.16 Robot Scara.

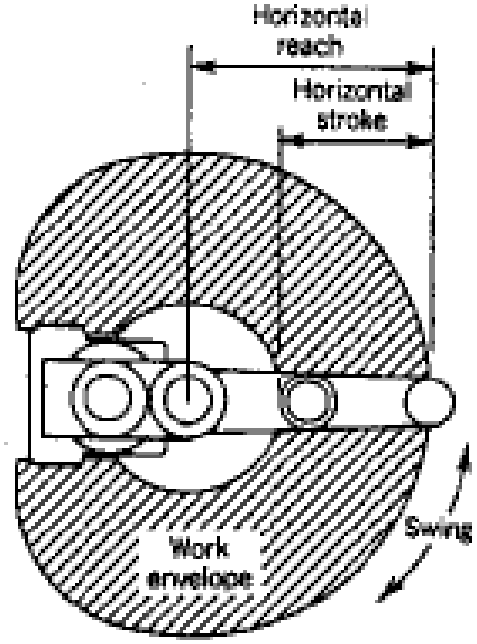


Figura 4.17 Área de trabajo de configuración escara.

4.2.1.2 Selección de la configuración de entre las alternativas antes mencionadas.

De las alternativas anteriores la más factible es la configuración angular ya que cuenta con una gran movilidad adecuada para una tarea de soldadura. En conclusión se elije la configuración angular.

Es necesario que a la configuración ya seleccionada se añada un par prismático que en la práctica permitirá controlar la distancia entre el electrodo y el cordón de soldadura (stick out).

A continuación se muestra un esquema general de cómo se comporta:

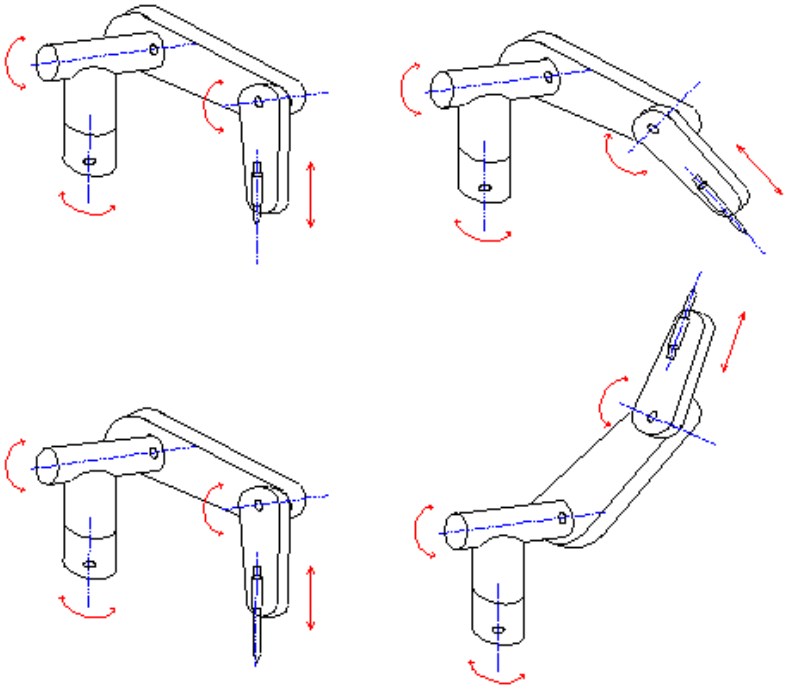


Figura 4.18 Configuración angular seleccionada del brazo robot.

4.3 DIMENSIONAMIENTO

Se debe tomar en cuenta que para el desarrollo de la función real que representa el cordón de soldadura, el origen del sistema de coordenadas (*x,y*) es la base del robot.

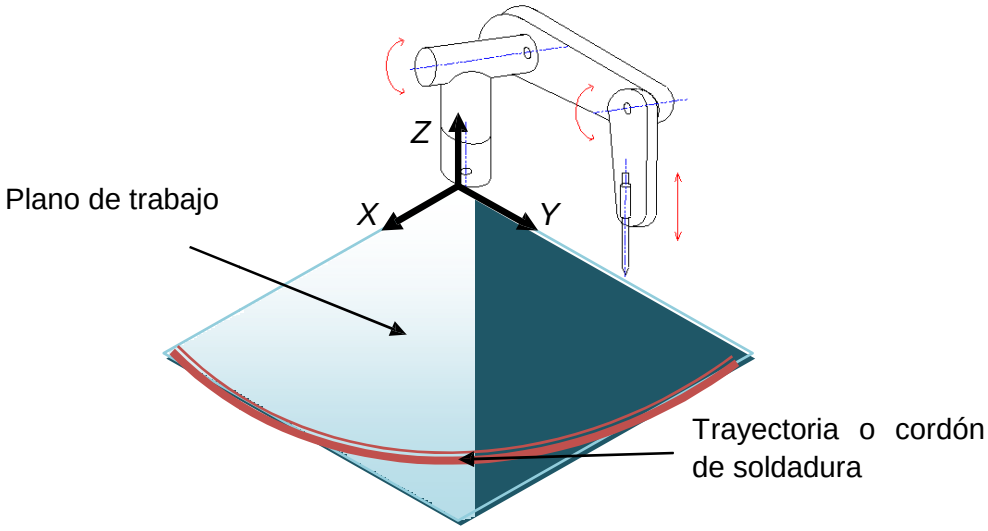


Figura 4.19 Disposición física brazo-área de trabajo y trayectoria a ser realizada.

El esquema indica la posición relativa de la base del brazo soldador (origen del sistema de coordenadas) respecto al área de trabajo planteada. Se debe calcular las dimensiones de los eslabones del mecanismo para que sean lo suficientemente grandes para cubrir el área de trabajo sin exceder esta innecesariamente.

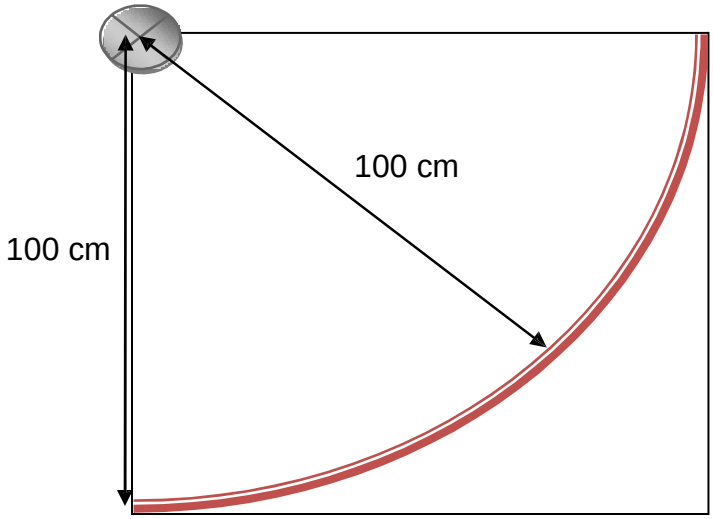


Figura 4.20 Vista aérea de la posición del área de trabajo relativa al sistema de coordenadas globales (base del robot).

La máxima distancia que deberá alcanzar el robot será constante en todo momento, calculamos la longitud de los eslabones del robot en base a un ángulo de abatimiento o apertura y considerando que ambos eslabones tienen igual longitud.

Como se indicó en la sección 2.4.4.2 durante el diseño del sistema mecánico se debe evitar infringir en singularidades dentro del espacio de trabajo del robot, en otras palabras evitar que el manipulador llegue a un abatimiento total es decir igual a 180º. Con este criterio se establece un ángulo máximo de abatimiento $\alpha_{max} < 180°$.

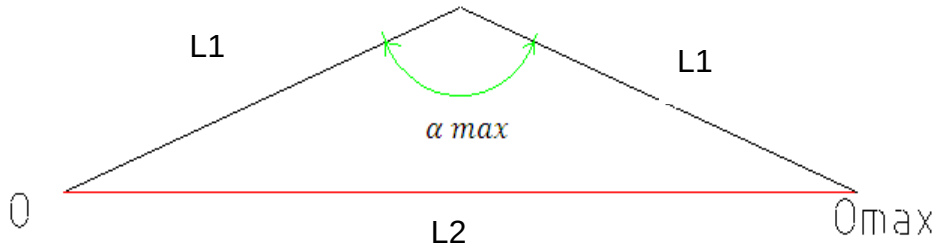


Figura 4.21 Apertura máxima de robot.

$$L_2{}^2 = L_1^2 + L_1^2 - 2L_1L_1 \cos \alpha_{max} \qquad \text{Ec. (4.1)}$$

$$2{,}5^2 = 2L_1^2 - 2L_1 L_1 \cos \alpha_{max}$$

$$L_1 = \frac{2{,}5^2}{2(1 - \cos \alpha_{max})} =$$

Establecemos α_{max}:

Para $\alpha_{max} = 165°$ $L_1 = 0.63m$

4.4 PROBLEMA CINEMÁTICO DIRECTO DETERMINACIÓN DE LOS PARAMETROS DE DENAVIT-HARTENBERG.

Observando la figura 4.22 y recurriendo a los criterios anteriormente mencionados en la sección 2.5, se procede a determinar los parámetros de Denavit-Hartenberg.

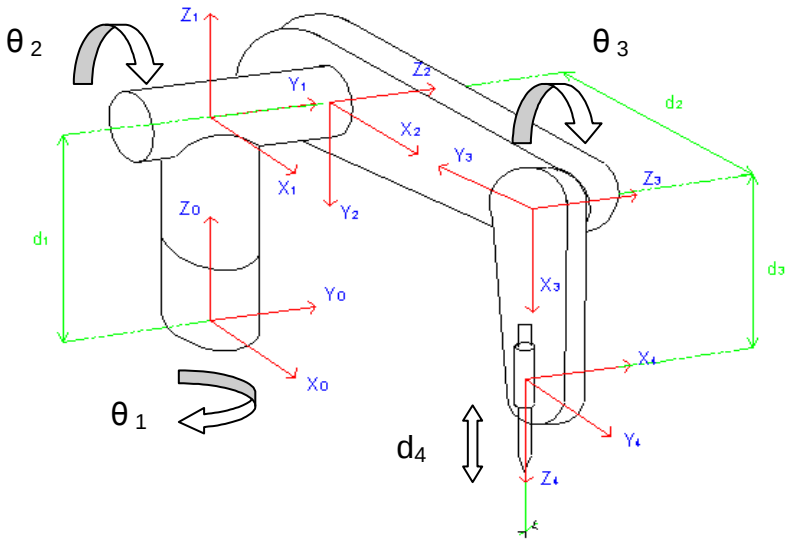


Figura 4.22 Disposición de ejes para cada par cinemático.

	Angulo α: ángulo entre ejes Z en un plano normal al eje X.	Distancia a: distancia hasta la intersección entre ejes X y Z, a lo largo de eje X.	Distancia d: distancia hasta la intersección entre ejes X y Z, a lo largo de eje Z.	Angulo θ: ángulo entre ejes X en un plano normal al eje Z.
Primer par (revolución)	0°	0	d_1	θ_1
Segundo par(revolución)	-90°	0	0	θ_2
Tercer par(revolución)	0°	d_2	0	θ_3
Cuarto par (prismático)	-90°	d_3	d_4	90°

Para comprobar su correcta elaboración de los parámetros de Denavit – Hartemberg procedemos a elaborar una rutina que muestre el robot en una posición aleatoria, esta rutina se adjunta en los anexos y al ser ejecutada el resultado es el que se muestra en las figuras a continuación.

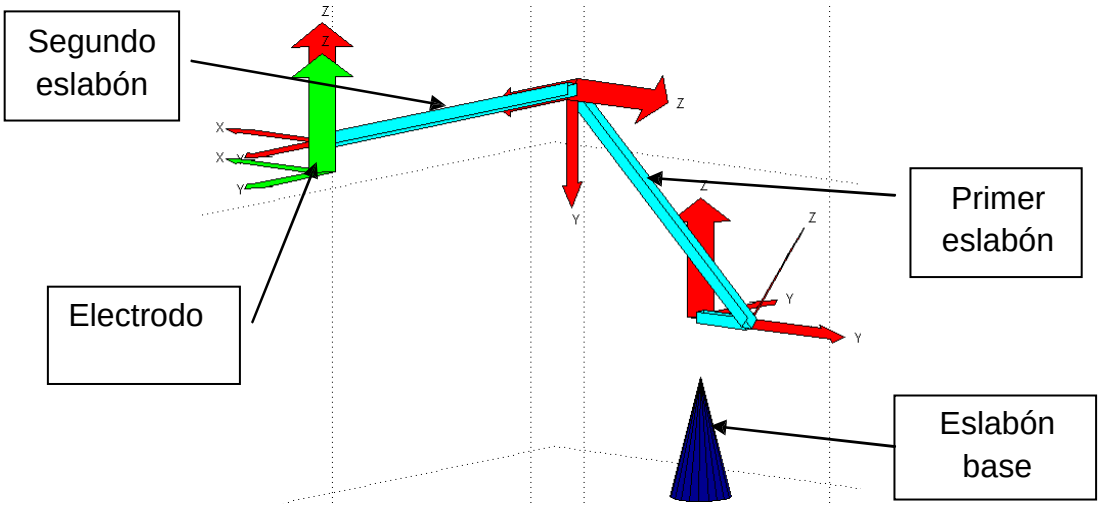


Figura 4.23 Brazo robot en una posición aleatoria, vista lateral derecha.

Como se observa en la figura el eslabón base del robot está representado por un cuerpo cónico de color azul, en cada par cinemático se ubica un sistema

de ejes coordenados al igual que en la figura 4.22. En la figura 4.24 se muestra una perspectiva diferente del robot donde se identifica cada par cinemático, esta figura se puede contrastar con la figura 4.25 para entender mejor la figura 4.23.

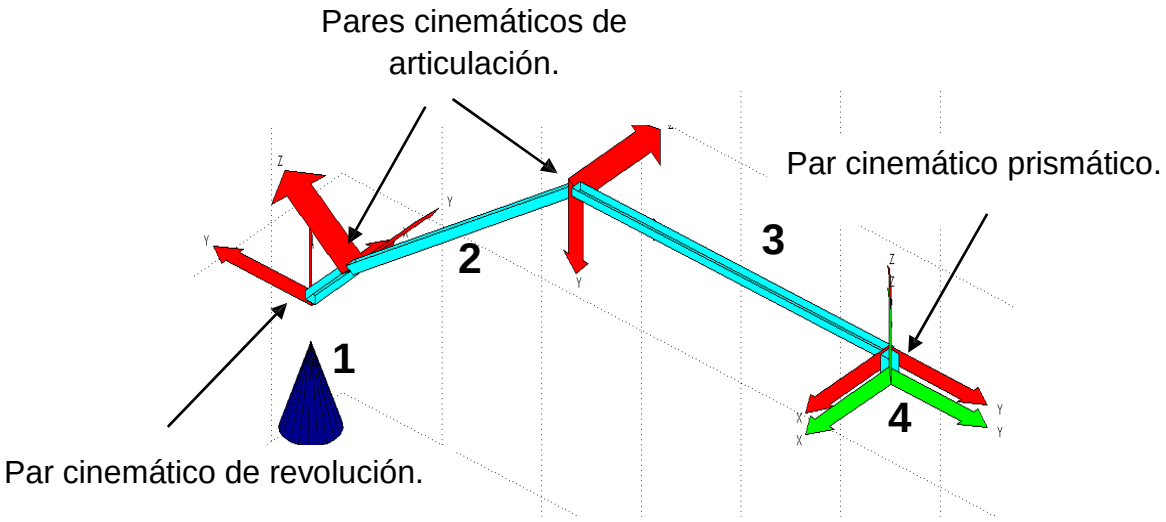


Figura 4.24 Brazo robot en una posición aleatoria, vista lateral izquierda.

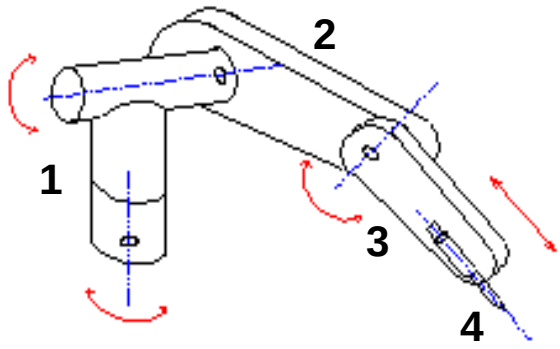


Figura 4.25 Robot soldador en posición similar al de la figura 4.24.

4.5 PROBLEMA CINEMÁTICO INVERSO-TOOLBOX HEMERO.

Como se explicó anteriormente en la sección 2.6 la solución del problema cinemático inverso para un robot de 4 grados de libertad es demasiado complicada para resolver manualmente.

Para solucionar el problema cinemático inverso recurriremos al uso del Toolbox Hemero de Matlab. En un archivo .m se almacena el algoritmo que

permitirá encontrar una solución a la cinemática inversa de nuestro manipulador a partir de los parámetros de Denavit-Hartenberg para cualquier trayectoria. Esta rutina y su explicación se incluyen en anexos. Hay que tomar en cuenta que la solución de la cinemática inversa se utilizará para elaborar el control automático del robot de tal manera esta no debe perderse de vista.

4.6 MODELO DINÁMICO DEL BRAZO ROBOT.

Para resolver el modelo dinámico del robot mediante la ecuación de movimiento de Lagrange debemos construir un diagrama de cuerpo libre del robot:

Descomponemos cada elemento que conforma el robot calculando su energía cinética (traslacional y rotacional) y energía potencial gravitacional:

4.6.1 BASE

Energía cinética:

Se calcula considerando la masa del cuerpo en cuestión (base), el radio de giro que se mide desde el eje de rotación (línea punteada en la figura 4.26) hasta la superficie exterior del cuerpo y la velocidad angular del eslabón base.

$$K_b = \frac{1}{2} m_b l^2 \dot{\theta}_1^2 \qquad \text{Ec. (4.3)}$$

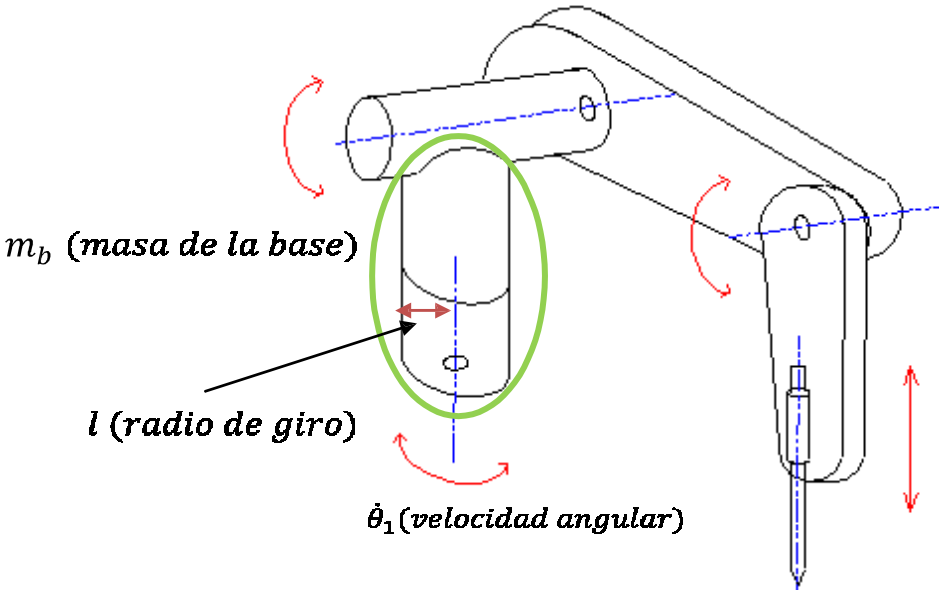


Figura 4.26 Base del robot.

La energía cinética traslacional no existe en la base ya que esta solo rota y no se traslada.[43]

4.6.2 PRIMER ESLABÓN BRAZO.

Energía cinética:

$$K_1 = \frac{1}{8} m_1 a^2 \dot{\theta}_2^2 + \frac{1}{16} m_1 a^2 \dot{\theta}_1^2 + \frac{1}{16} m_1 a^2 \cos 2\theta_2 \, \dot{\theta}_1^2 \qquad \text{Ec. (4.4)}$$

[43] CARACTERIZACIÓN DE UN ROBOT MANIPULADOR ARTICULADO, ABDALÁ CASTILLO SALOMÓN Y ÑECO CABERTA RAÚL DEL CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO, ESPAÑA.

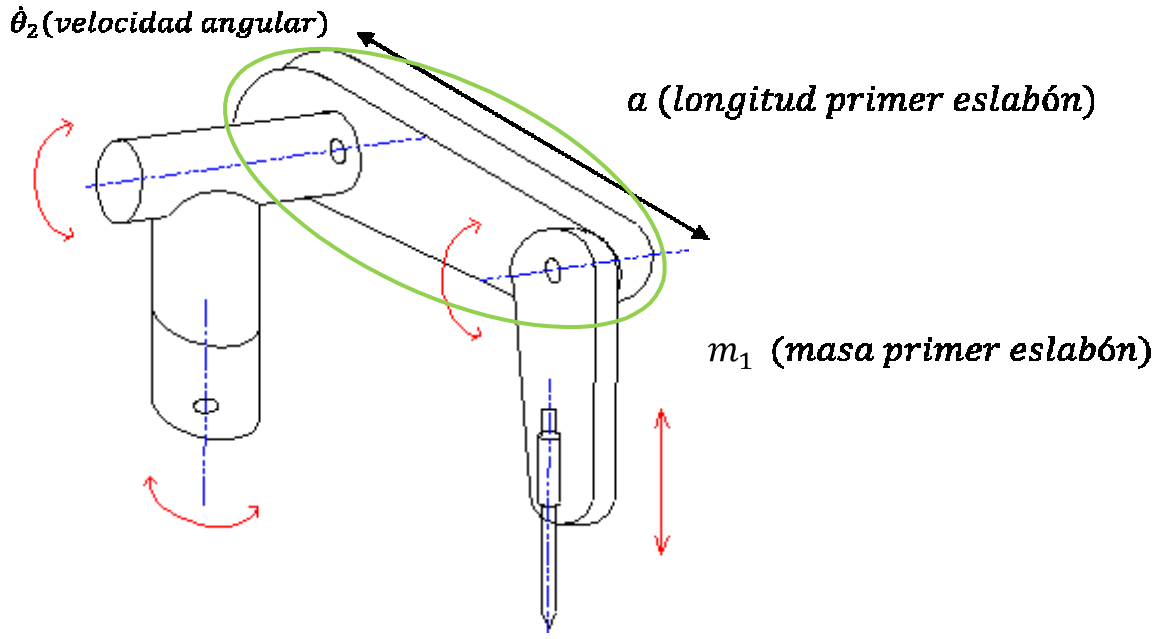


Figura 4.27 Brazo del robot.

Energía potencial:

$$P_1 = m_1 g \frac{a}{2} \sin\theta_2 \qquad \text{Ec. (4.5)}^{44}$$

4.6.3 SEGUNDO ESLABÓN BRAZO.

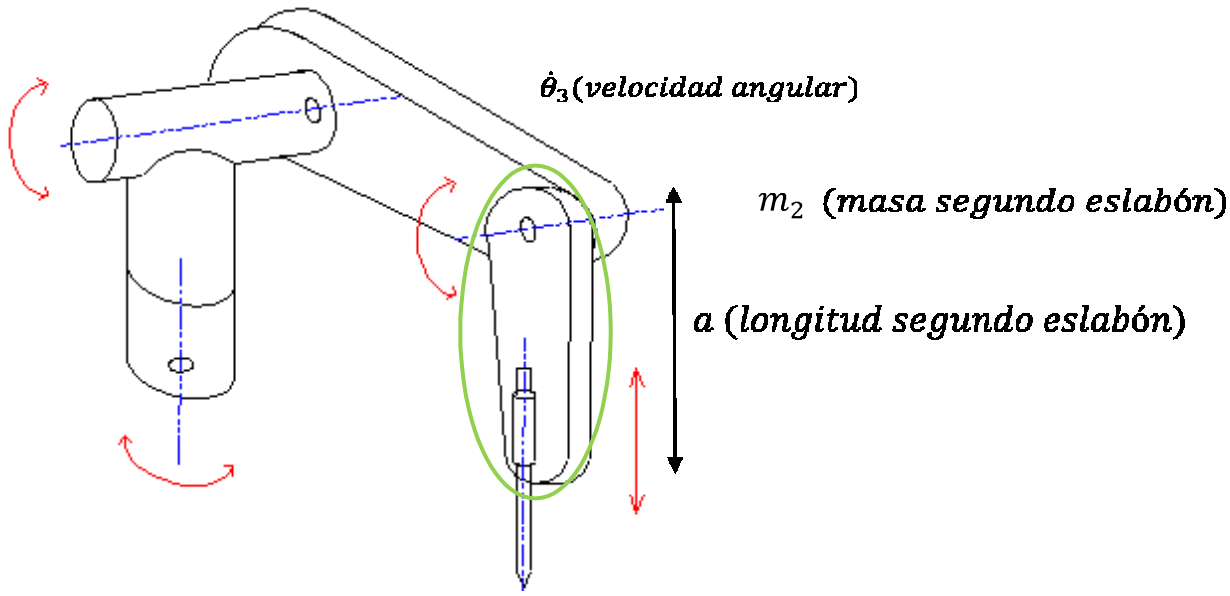


Figura 4.28 Antebrazo de robot.

[44] CARACTERIZACIÓN DE UN ROBOT MANIPULADOR ARTICULADO, ABDALÁ CASTILLO SALOMÓN Y ÑECO CABERTA RAÚL DEL CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO, ESPAÑA.

$$K_2 = \frac{1}{2} m_2 V_2^2 + \frac{1}{2} I_2 \dot{\theta}^2 \qquad \text{Ec. (4.6)}$$

Donde *V* es la velocidad que el primer eslabón transmite al segundo cuando el primero se mueve, esta velocidad tiene 2 componentes una en *X* otra en *Y*. A continuación se define *V*:[45]

$$V^2 = \dot{X}^2 + \dot{Y}^2 \qquad \text{Ec. (4.7)}$$

$$X_2 = a\cos\theta_2 + \frac{1}{2} L \cos(\theta_2 + \theta_3) \qquad \text{Ec. (4.8)}$$

$$Y_2 = a\sin\theta_2 + \frac{1}{2} L \sin(\theta_2 + \theta_3) \qquad \text{Ec. (4.9)}$$

$$I_2 = m_2 X_2^2 \qquad \text{Ec. (4.10)}$$

$$V_2^2 = a^2\dot{\theta}_2^2 + ab\left(\dot{\theta}_2 + \dot{\theta}_2\dot{\theta}_3\right)\cos\theta_3 + \frac{1}{4} b^2(\dot{\theta}_2 + \dot{\theta}_3)^2 \qquad \text{Ec. (4.11)}$$

$$P_2 = m_2 g a \sin\theta_2 + \frac{1}{2} m_2 g b \sin(\theta_2 + \theta_3) \qquad \text{Ec. (4.12)}$$

Calculamos el Lagrangiano haciendo la sumatoria de energías cinéticas y restando la sumatoria de energías potenciales gravitacionales:

$$\mathcal{L} = \sum K - \sum P \qquad \text{Ec. (4.13)}$$

$$\mathcal{L} = \frac{1}{2} m_b l^2 \dot{\theta}_1^2 + \frac{1}{8} m_1 a^2 \dot{\theta}_2^2 + \frac{1}{16} m_1 a^2 \dot{\theta}_1^2 + \frac{1}{16} m_1 a^2 \cos 2\theta_2 \, \dot{\theta}_1^2 + \frac{1}{2} m_2 \left(a^2\dot{\theta}_2^2 + ab\left(\dot{\theta}_2 + \dot{\theta}_2\dot{\theta}_3\right)\cos\theta_3 + \frac{1}{4} b^2\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2\right) + \frac{1}{2} m_2 X_2 \dot{\theta}^2{}_1 + m_1 g \frac{a}{2} \sin\theta_2 + m_2 g a \sin\theta_2 + \frac{1}{2} m_2 g b \sin(\theta_2 + \theta_3)$$

Ec. (4.14)

[45] CARACTERIZACIÓN DE UN ROBOT MANIPULADOR ARTICULADO, ABDALÁ CASTILLO SALOMÓN Y ÑECO CABERTA RAÚL DEL CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO, ESPAÑA.

$$\mathcal{L} = \frac{1}{2}m_b l^2\dot{\theta}_1^2 + \frac{1}{8}m_1 a^2\dot{\theta}_2^2 + \frac{1}{16}m_1 a^2\dot{\theta}_1^2 + \frac{1}{16}m_1 a^2\cos 2\theta_2\,\dot{\theta}_1^2 + \frac{1}{2}m_2 a^2\dot{\theta}_2^2 + \frac{1}{2}m_2 ab\left(\dot{\theta}_2 + \dot{\theta}_2\dot{\theta}_3\right)\cos\theta_3 + \frac{1}{8}m_2 b^2\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2 + \frac{1}{2}m_2\dot{\theta}_2^2 a\cos\theta_2 + \frac{1}{2}m_2\dot{\theta}_2^2 2\frac{1}{2}l\ \cos(\theta_2 + \theta_3) + \frac{1}{2}m_1 ga\sin\theta_2 + m_2 ga\sin\theta_2 + \frac{1}{2}m_2 gb\sin(\theta_2 + \theta_3)$$

Ec. (4.15)

A continuación procedemos a derivar como indica la siguiente ecuación:

$$\tau = \frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{\Theta}}\right) - \frac{\partial\mathcal{L}}{\partial\Theta} \qquad \text{Ec. (4.16)}$$

$$\Theta = \begin{bmatrix}\theta_1\\ \theta_2\\ \theta_3\end{bmatrix} \qquad \text{Ec. (4.17)}$$

$$\dot{\Theta} = \begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\end{bmatrix} \qquad \text{Ec. (4.18)}$$

$$\frac{\partial\mathcal{L}}{\partial\dot{\Theta}} = \begin{bmatrix} m_b l^2\dot{\theta}_1 + \frac{1}{8}m_1 a^2\dot{\theta}_1 + \frac{1}{8}m_1 a^2\cos 2\theta_2\,\dot{\theta}_1 \\ m_2 a^2\dot{\theta}_2 + \frac{1}{2}m_2 ab\left(1 + \dot{\theta}_3\right)\cos\theta_3 + \frac{1}{4}m_2 b^2\left(\dot{\theta}_2 + \dot{\theta}_3\right) + m_2 a\cos\theta_2\dot{\theta}_2 + \frac{1}{2}m_2\dot{\theta}_2 b\ \cos(\theta_2 + \theta_3) + \frac{1}{4}m_1 a^2\dot{\theta}_2 \\ \frac{1}{2}m_2 ab\left(\dot{\theta}_2 + \dot{\theta}_2\dot{\theta}_3\right)\cos\theta_3\dot{\theta}_2 + \frac{1}{4}m_2 b^2\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2 \end{bmatrix}$$

Ec. (4.19)

$$\frac{\partial\mathcal{L}}{\partial\Theta} = \begin{bmatrix} 0 \\ -\frac{1}{2}m_2 a\sin\theta_2\,\dot{\theta}_1^2 - \frac{1}{4}m_2\dot{\theta}_2^2 b\ \sin(\theta_2 + \theta_3) + \frac{1}{2}m_1 ga\cos\theta_2 + m_2 ga\cos\theta_2 + \frac{1}{2}m_2 gb\cos(\theta_2 + \theta_3) \\ -\frac{1}{4}m_2\dot{\theta}_2^2 l\ \sin(\theta_2 + \theta_3) + \frac{1}{2}m_2 gb\cos(\theta_2 + \theta_3) \end{bmatrix}$$

Ec. (4.20)

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\Theta}}\right)=\begin{bmatrix} m_b l^2\ddot{\theta}_1+\frac{1}{8}m_1a^2\ddot{\theta}_1-\frac{1}{8}m_1a^2\sin 2\theta_2\,\dot{\theta}_2+\frac{1}{8}m_1a^2\cos 2\theta_2\,\ddot{\theta}_1 \\ m_2a^2\ddot{\theta}_2-\frac{1}{2}m_2\boldsymbol{ab}\sin\theta_3\dot{\theta}_3+\frac{1}{2}m_2\boldsymbol{ab}\cos\theta_3\ddot{\theta}_3+\frac{1}{4}m_2b^2\left(\ddot{\theta}_2+\ddot{\theta}_3\right)-m_2\boldsymbol{a}\sin\theta_2\dot{\theta}_2{}^2+ \\ m_2\boldsymbol{a}\cos\theta_2\ddot{\theta}_2-\frac{1}{2}m_2\boldsymbol{l}\ \sin(\theta_2+\theta_3)\left(\dot{\theta}_2+\dot{\theta}_3\right)+\frac{1}{2}m_2\boldsymbol{l}\ \cos(\theta_2+\theta_3)\ddot{\theta}_2+\frac{1}{4}m_1a^2\ddot{\theta}_2 \\ \frac{1}{2}m_2ab\left[\left(\ddot{\theta}_2+\ddot{\theta}_2\dot{\theta}_3+\dot{\theta}_2\ddot{\theta}_3\right)\cos\theta_3\,\dot{\theta}_2+\left(\dot{\theta}_2+\dot{\theta}_2\dot{\theta}_3\right)(-\sin\theta_3\,\dot{\theta}_3\dot{\theta}_2+\cos\theta_3\,\ddot{\theta}_2)\right]+ \\ \frac{1}{4}m_2b^2(\ddot{\theta}_2+\ddot{\theta}_3) \end{bmatrix}$$

Ec. (4.21)

El modelo dinámico completo debe expresarse como indica la Ec. (4.16) para lo cual agrupamos la Ec. (4.20) y la Ec. (4.21) obteniéndose lo siguiente:

$$\tau=\begin{bmatrix} m_b l^2\ddot{\theta}_1+\frac{1}{8}m_1a^2\ddot{\theta}_1-\frac{1}{4}m_1a^2\sin 2\theta_2\,\dot{\theta}_2+\frac{1}{8}m_1a^2\cos 2\theta_2\,\ddot{\theta}_1 \\ m_2a^2\ddot{\theta}_2-\frac{1}{2}m_2\boldsymbol{ab}\sin\theta_3\dot{\theta}_3+\frac{1}{2}m_2\boldsymbol{ab}\cos\theta_3\ddot{\theta}_3+\frac{1}{4}m_2b^2\left(\ddot{\theta}_2+\ddot{\theta}_3\right)-m_2\boldsymbol{a}\sin\theta_2\dot{\theta}_2{}^2+ \\ m_2\boldsymbol{a}\cos\theta_2\ddot{\theta}_2-\frac{1}{2}m_2\boldsymbol{l}\ \sin(\theta_2+\theta_3)\left(\dot{\theta}_2+\dot{\theta}_3\right)+\frac{1}{2}m_2\boldsymbol{l}\ \cos(\theta_2+\theta_3)\ddot{\theta}_2+\frac{1}{4}m_1a^2\ddot{\theta}_2+ \\ \frac{1}{2}m_2\boldsymbol{a}\sin\theta_2\dot{\theta}_2{}^2-\frac{1}{2}b\,m_2\sin(\theta_2+\theta_3)\dot{\theta}_2{}^2+\frac{1}{2}m_1\boldsymbol{g}\boldsymbol{a}\cos\theta_2+m_2\boldsymbol{g}\boldsymbol{a}\cos\theta_2+ \\ \frac{1}{2}m_2gb\cos(\theta_2+\theta_3) \\ \frac{1}{2}m_2ab\left[\left(\ddot{\theta}_2+\ddot{\theta}_2\dot{\theta}_3+\dot{\theta}_2\ddot{\theta}_3\right)\cos\theta_3\,\dot{\theta}_2+\left(\dot{\theta}_2+\dot{\theta}_2\dot{\theta}_3\right)(-\sin\theta_3\,\dot{\theta}_3\dot{\theta}_2+\cos\theta_3\,\ddot{\theta}_2)\right]+ \\ \frac{1}{4}m_2b^2\left(\ddot{\theta}_2+\ddot{\theta}_3\right)-\frac{1}{4}b\,m_2\sin(\theta_2+\theta_3)\dot{\theta}_2{}^2-\frac{1}{2}m_2\boldsymbol{g}\boldsymbol{b}\cos(\theta_2+\theta_3) \end{bmatrix}$$

Ec. (4.22)

4.7 MODELADO EN SIMULINK

Como primer paso procedemos a construir el diagrama de bloques del sistema brazo robótico para lo cual ejecutamos el Simulink y procedemos a crear un nuevo archivo. A continuación vamos integrando los bloques que necesitamos:

Según se ve en las figuras 4.29 y 4.30 el robot está compuesto de 4 eslabones o cuerpos que son eslabón base (1),eslabón brazo (2), eslabón antebrazo (3), eslabón efector final (4); 4 pares cinemáticos: 1 de revolución (5), 2 articulaciones (6 y 7)y 1 prismático (8), 4 actuadores (análogos a motores de paso) es decir uno por cada par cinemático, 4 sensores de movimiento o encoders, una referencia o eslabón fijo al que llamaremos tierra o bancada y

opcionalmente podemos integrar un bloque de condición inicial. Se considera que cada actuador está integrado con un sensor y estos a su respectivo par cinemático donde funcionarán.

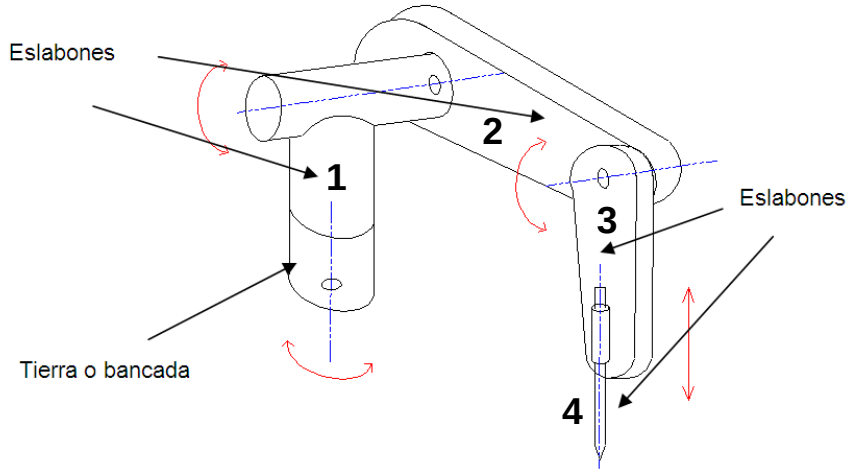


Figura 4.29 Eslabones existentes en el robot soldador.

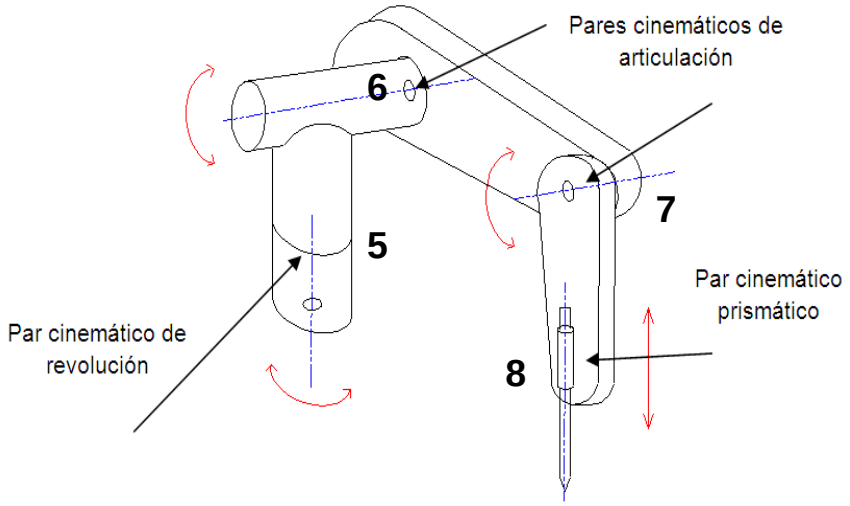


Figura 4.30 Pares cinemáticos en el robot soldador.

Dicho esto procedemos a integrar los bloques mencionados. Tal como se realizó en la sección 3.1.4.4 se acude a la librería SimMechanics de Simulink en busca de los siguientes bloques:

Body: estos bloques representarán a los 4 eslabones por tanto se requieren 4 bloques.

Joint actuator: que representan a los actuadores o motores paso a paso, se requieren uno por cada par cinemático es decir 4.

Joint sensor: que representan a los sensores de movimiento o encoders, se requieren uno por cada par cinemático es decir 4.

Revolute: son los pares cinemáticos de rotación (1) y los pares de articulación (2), en total hay 3.

Prismatic: representa el par cinemático prismático, hay uno solo.

Machine environment: este bloque establece las condiciones externas gravitatorias a las que la máquina estará sometida.

Ground: este es el bloque de bancada o eslabón fijo.

En total son requeridos 18 bloques como se acabo de explicar más un bloque de condición inicial, finalmente se requieren 19 bloques para representar al robot físicamente.

De esta manera ingresamos los bloques que se acaban de describir en el nuevo modelo de Simulink.

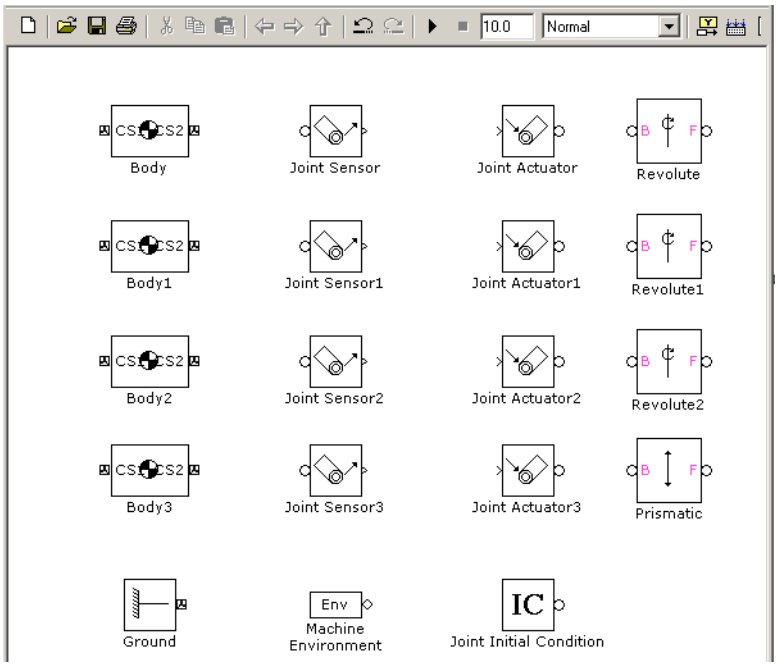


Figura 4.31 Integración de bloques para Brazo robot.

Ahora procedemos a configurar cada bloque, empezando con el primero:

4.7.1 PRIMER PAR CINEMÁTICO DE REVOLUCIÓN (BASE):

Se especifica el número de sensores y actuadores que se van a conectar que para el caso es 3 (1 sensor, 1 actuador y un bloque de condición inicial). Especificamos la dirección en que va a girar la base respecto al sistema global de coordenadas es decir el eje z.

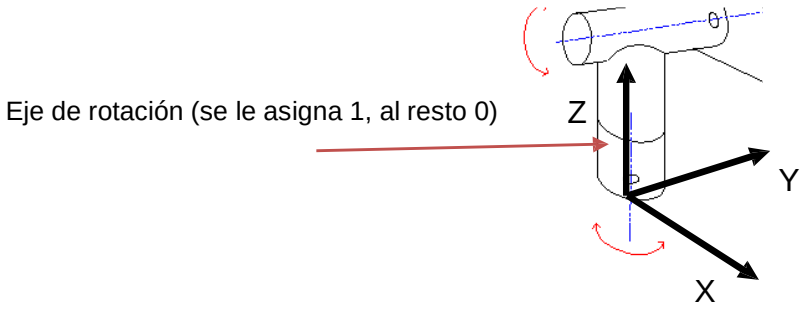


Figura 4.32 Sistema de ejes coordenados globales respecto a la base, en la figura se observa que el eje de rotación de la articulación es el eje *Z* por tanto la matriz de ejes de acción queda como [0 0 1].

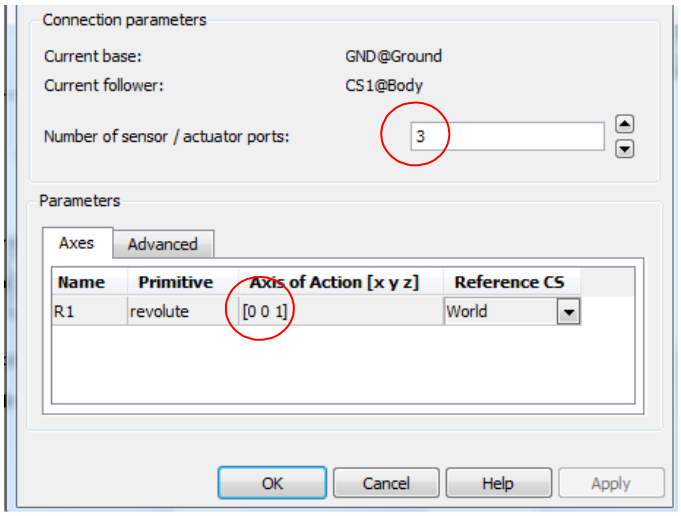


Figura 4.33 Configuración primer par cinemático.

4.7.2 PRIMER CUERPO O ESLABÓN (BASE GIRATORIA):

Se define la masa de este cuerpo como *m1*, la unidad de masa quedará expresada en *Kg*.

Definimos el tensor de inercia o matriz de inercias del cuerpo como *Tensorinercia1* en unidades de Kg/m^2.

Para optimizar el tiempo durante las simulaciones se ha parametrizado estos datos de masa e inercia como:

m1

Tensorinercia1

Para que mediante otra rutina se puedan definir numéricamente en función de las dimensiones del robot, las cuales pueden alterarse fácilmente y se evita volver a configurar cada bloque.

Ahora se establece las coordenadas donde empieza (CS1), es decir [0 0 0] y termina el cuerpo (CS2) [0 0 0.1], es decir que la altura (eje *Z*) de este eslabón es 10 cm respecto al sistema de coordenadas globales. De igual manera se establece la posición del centro de gravedad del cuerpo (CG) en [0 0 0.05]. La unidad de distancia se establece en *m*.

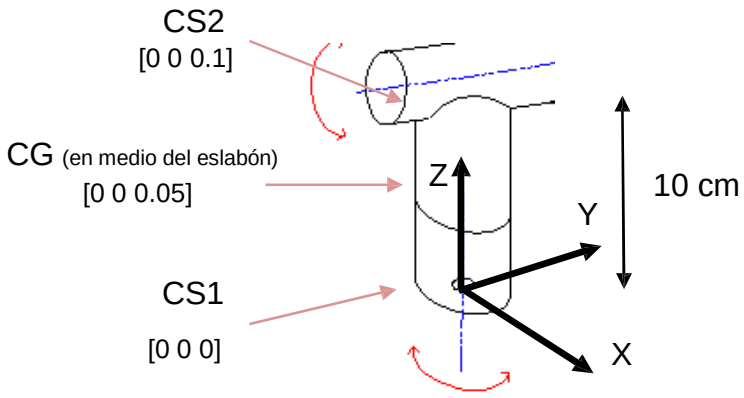


Figura 4.34 Disposición de punto inicial y final de la base.

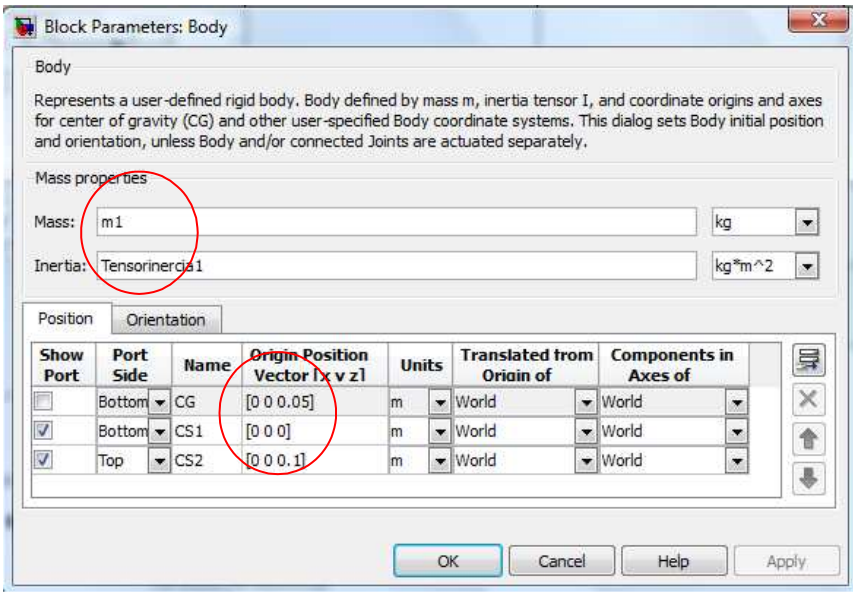


Figura 4.35 Configuración primer eslabón.

4.7.3 SEGUNDO PAR CINEMÁTICO DE ARTICULACIÓN:

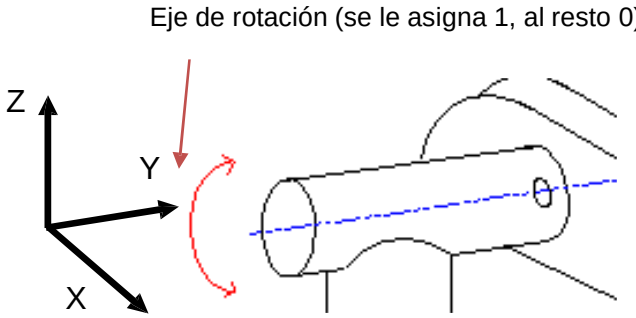


Figura 4.36 Ejes coordenados globales para segundo par cinemático, en la figura se observa que el eje de rotación de la articulación es el eje *Y* por tanto la matriz de ejes de acción queda como [0 1 0].

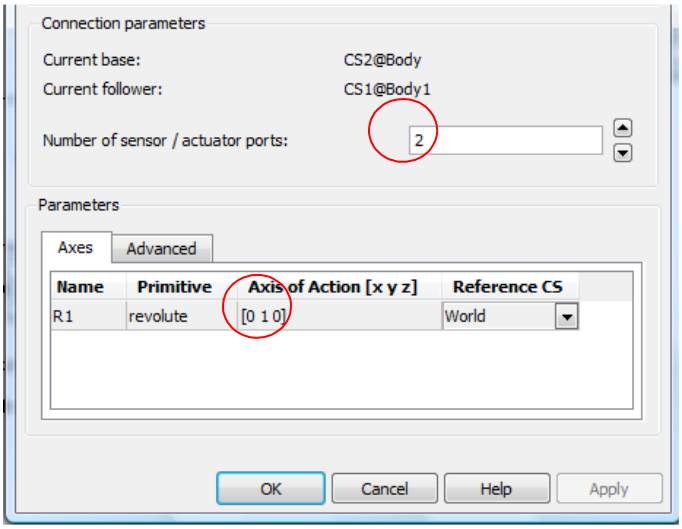


Figura 4.37 Configuración segundo par cinemático.

Igual que el primer par con la diferencia que el sentido de giro es el eje *Y* ya que el eje de rotación del primer par es perpendicular al eje de rotación del segundo par.

4.7.4 SEGUNDO CUERPO O ESLABÓN:

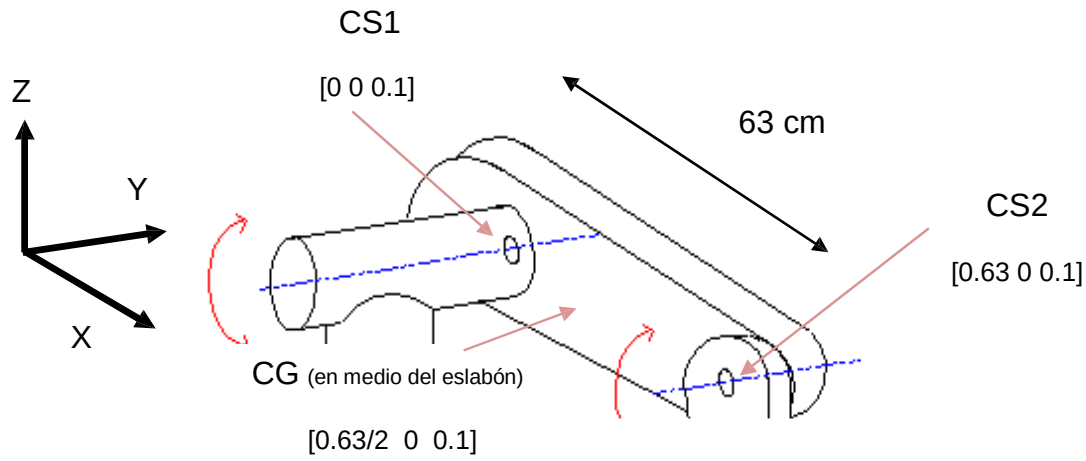


Figura 4.38 Disposición del segundo eslabón.

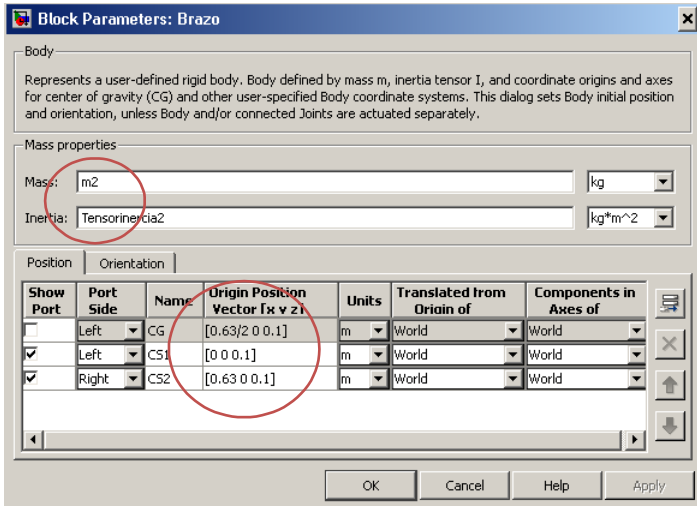


Figura 4.39 Configuración segundo eslabón.

Igual que el primer cuerpo dejamos parametrizando los datos de masa e inercia con las variables:

m2

Tensorinercia2

Se define su posición en el espacio tomando en cuenta su posición relativa el cuerpo anterior y tomando en cuenta las dimensiones establecidas en la sección 4.2.2.

4.7.5 TERCER PAR CINEMÁTICO DE ARTICULACIÓN:

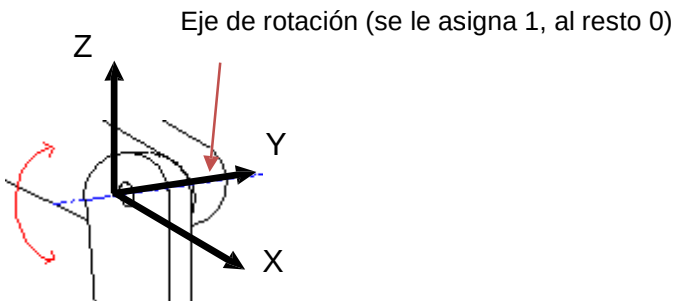


Figura 4.40 Ejes globales y tercer par cinemático, en la figura se observa que el eje de rotación de la articulación es el eje *Y* por tanto la matriz de ejes de acción queda como [0 1 0].

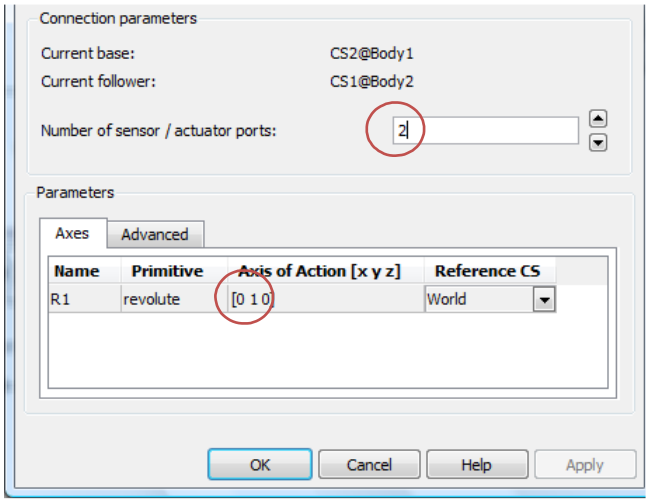


Figura 4.41 Configuración tercer par cinemático.

Este bloque se configura igual al anterior bloque de revolución.

4.7.6 TERCER CUERPO O ESLABÓN:

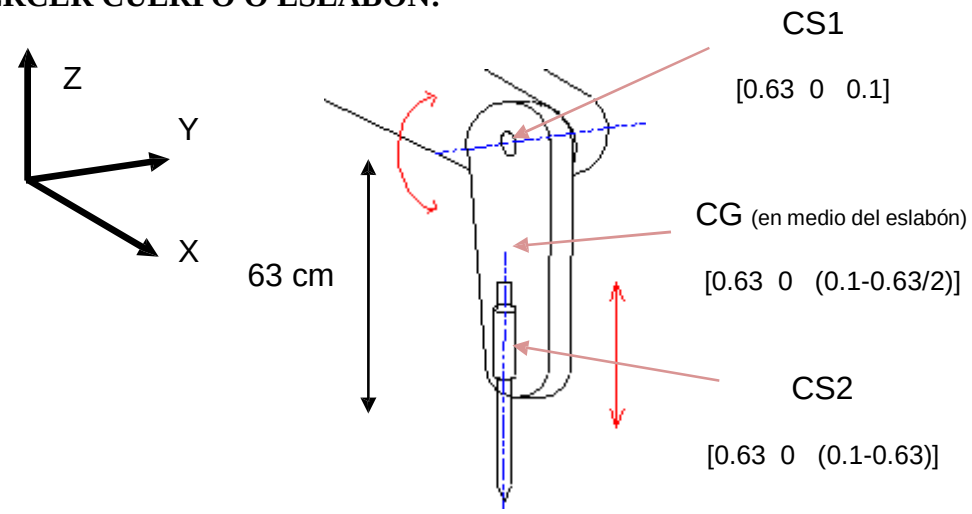


Figura 4.42 Posición del antebrazo relativo a sistema de ejes coordenados.

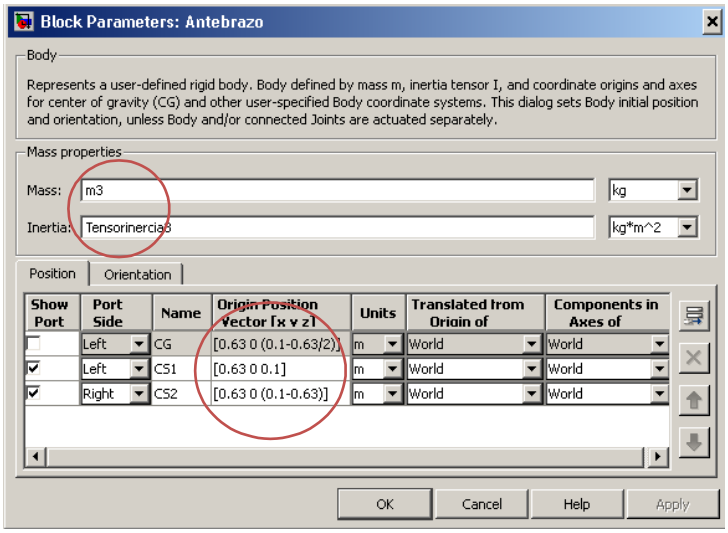


Figura 4.43 Configuración tercer eslabón.

Igual que el primer cuerpo dejamos parametrizando los datos de masa e inercia con las variables:

M3

Tensorinercia2

Se define su posición en el espacio tomando en cuenta su posición relativa el cuerpo anterior y tomando en cuenta las dimensiones establecidas en la sección 4.3.

4.7.7 PAR CINEMÁTICO PRISMÁTICO:

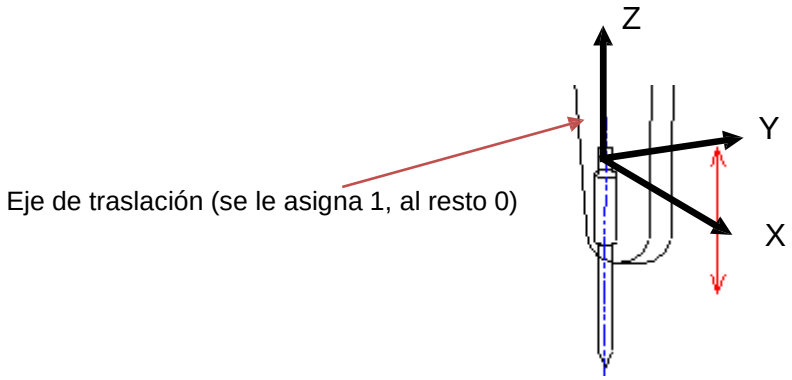


Figura 4.44 Ejes de último eslabón, en la figura se observa que el eje de traslación del par prismático es el eje *Z* por tanto la matriz de ejes de acción queda como [0 0 1].

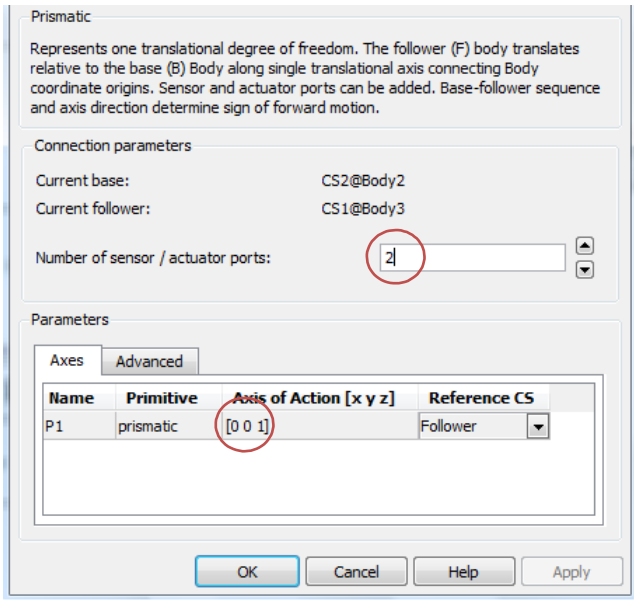


Figura 4.45 Configuración par prismático.

En el par prismático debemos especificar la dirección en que se da el movimiento lineal de traslación que es el eje *Z* como indica la figura 4.44.

4.7.8 CUARTO CUERPO O ESLABÓN:

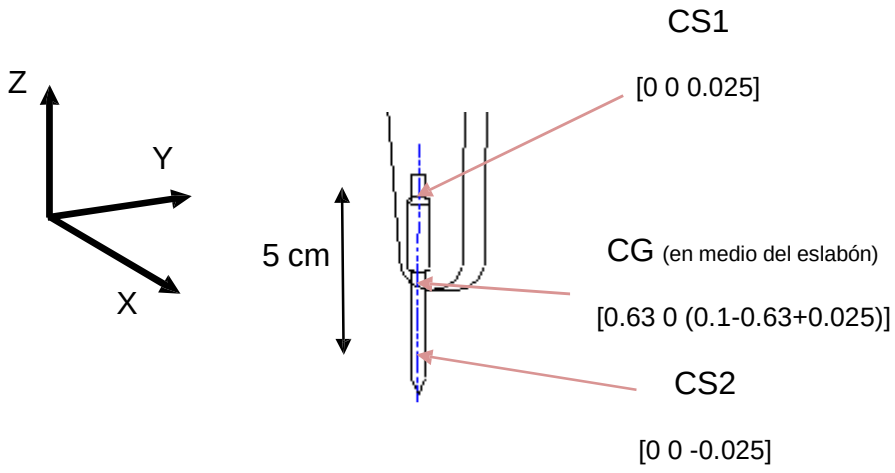


Figura 4.46 Posición de último eslabón. Hay que acotar que la posición del dentro de gravedad es respecto al sistema global de coordenadas y que las posiciones de CS1 y CS2 están respecto al centro de gravedad.

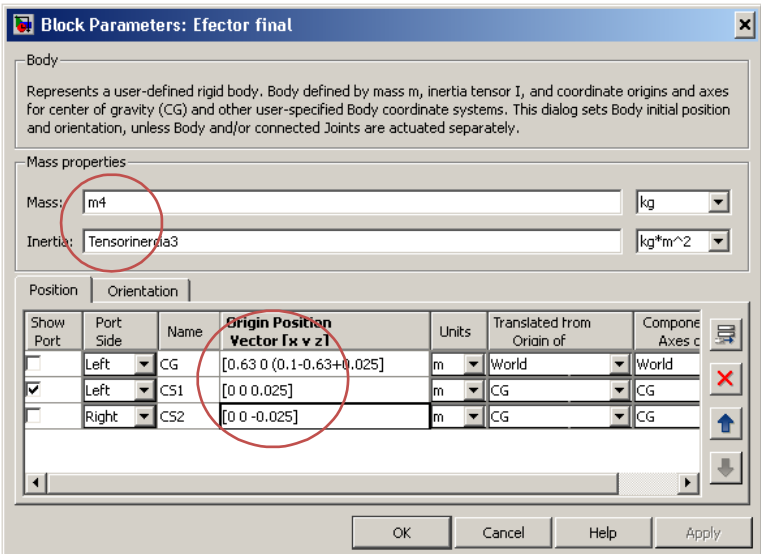


Figura 4.47 Configuración cuarto eslabón.

La característica principal de este cuerpo es que las dimensiones y masa de este es mucho menor ya que es solo un dispositivo que aleja o acerca el electrodo de la junta.

4.7.9 CONFIGURACIÓN DE *JOINT SENSORS* (SENSORES):

Los sensores son 4 en total y se conectarán uno con cada par cinemático y se configuraran de la siguiente manera:

Se coloca visto en las casillas de posición angular (*Angle*) y velocidad angular (*Angular velocity*).

Se elije como unidades *rad* y *rad/s* respectivamente como se indica en la figura 4.48.

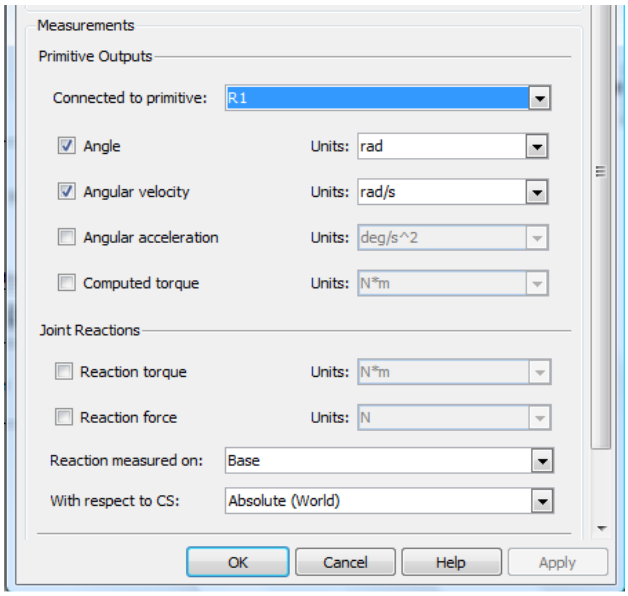


Figura 4.48 Configuración de sensores.

Para obtener los datos de posición y velocidad angular que los sensores han medido, junto cada sensor integraremos un bloque *scope* que servirá para visualizar los datos.

4.7.9.1 Configuración de *Joint actuator* (Actuadores o motores PAP):

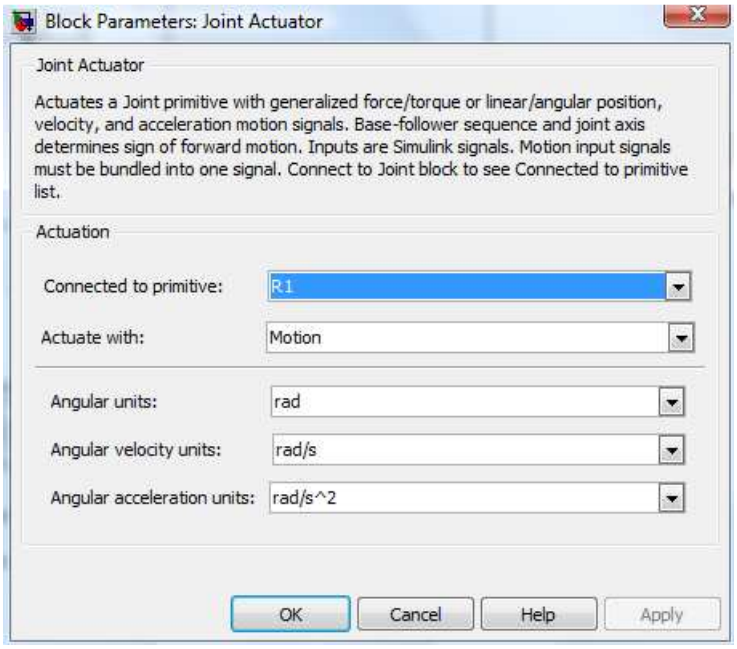


Figura 4.49 Configuración de actuadores.

En la casilla *Actuate with* se selecciona *Motion* para que se pueda ingresar la señal de movimiento que fue calculada mediante la cinemática inversa, y en la parte inferior seleccionamos como unidades *rad*.

Luego unimos los bloques para que adopte el orden y la lógica de nuestro brazo robot. Empezando por el bloque *Ground* que se conecta a un par cinemático (*revolute*) y al bloque *machine environment*, a continuación se conecta un bloque *Body* y a continuación de este otro par cinemático, de esta manera hasta conectar cada eslabón y cada par concatenadamente como muestra la figura 4.50.

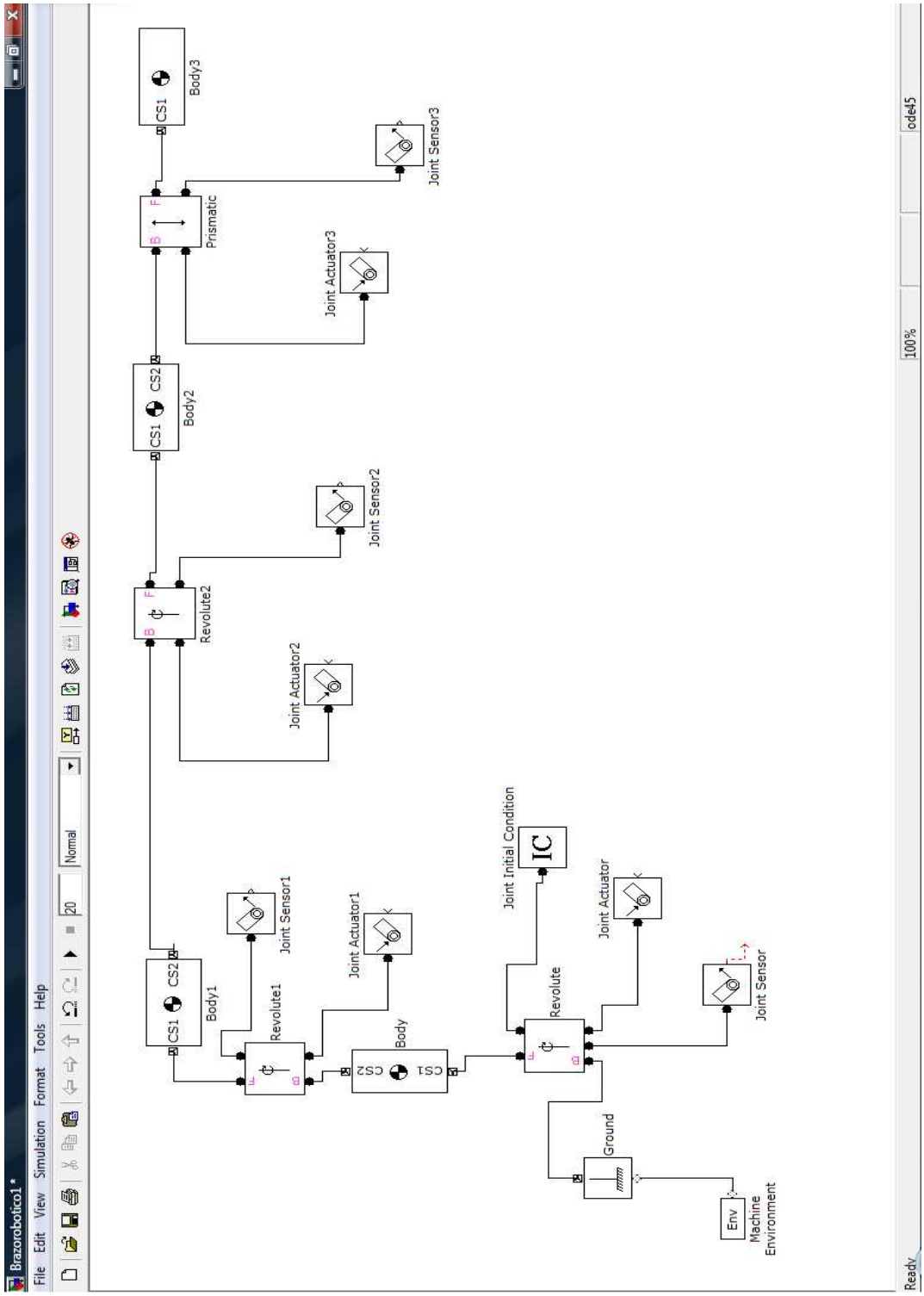


Figura 4.50 Unión de bloques Brazo-robot.

Con esto quedan configurados y ensamblados todos los bloques que representan al sistema dinámico del robot. Lo que sigue a continuación es implementar el control automático en el sistema dinámico ya elaborado.

4.8 IMPLEMENTACIÓN DEL CONTROL AUTOMATICO

Para implementar el control automático al sistema se emplea los siguientes bloques:

Signal builder: para crear la señal de entrada.
Mux: para agrupar la señal de entrada.
Gain: para amplificar la señal débil.

Como sistema de control automático hemos seleccionado un control de lazo abierto ya que al utilizar motores de paso se obtiene una gran precisión en el movimiento a desarrollar, para lo cual debemos ingresar la señal de control a cada actuador a través de un bloque *signal builder*.

El *signal builder* es un bloque que permite crear una señal cualquiera para ser utilizada en un sistema de control como señal de entrada. Esta señal puede ser sinusoidal, constante, escalonada, etc., es decir cualquier señal que se requiera. Para este caso en particular se requiere crear una señal con la solución de la cinemática inversa del robot soldador para la trayectoria que se definió desde un inicio en la rutina. A continuación se muestra el bloque y el cuadro de diálogo al hacer doble click en el mismo.

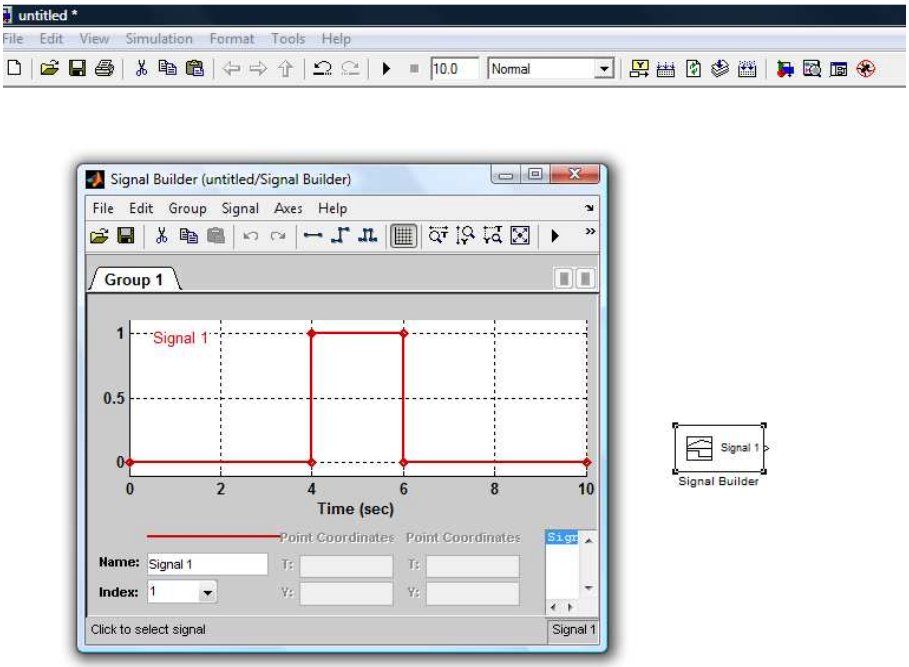


Figura 4.51 Configuración de constructor de señales.

A través de este bloque debemos ingresar la señal de entrada que se descompone en 3 señales: de posición, velocidad y aceleración angular. Esta señal proviene de la solución de la cinemática inversa obtenida anteriormente.

Entonces requerimos 3 señales por cada *signal builder* y 1 *signal builder* por cada actuador. Empezamos por añadir dos señales mas dirigiéndose al menú *Signal* y haciendo click en *new*, luego en *custom* que significa que se añadirá una señal establecida por el usuario tal como indica la figura 4.52.

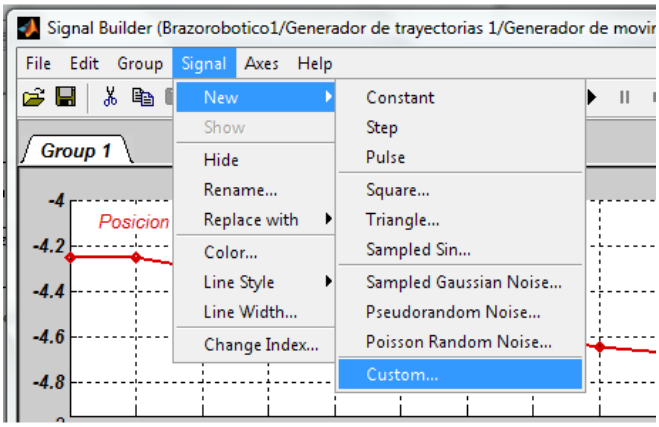


Figura 4.52 Creación de una señal definida por el usuario.

4.8.1 CONTROL DE TRAYECTORIA CORDÓN DE SOLDADURA (TRES PRIMEROS ESLABONES DEL ROBOT, LA BASE Y BRAZOS).

Para que el proceso de soldadura se dé correctamente y el cordón se realice uniformemente debemos asegurarnos de que el robot se mueva con velocidad constante (aceleración igual a cero).

La señal de control como se explico en la sección 4.5 se almacenó en la variable *q* como se especificó en la rutina de cinemática inversa detallada en anexos, esta matriz q también se encuentra en los anexos.

A continuación aparecerá el cuadro de dialogo de la figura 4.53 donde se debe ingresar los datos de la matriz q en donde se almacenó la solución de la cinemática inversa.

q =

15.3918	-7.5167	-17.1774
15.2917	-7.5167	-17.1779
15.1905	-7.5167	-17.1784
15.0872	-7.5166	-17.1789
14.9803	-7.5166	-17.1794
14.8683	-7.5166	-17.1799
14.7484	-7.5166	-17.1804
14.6165	-7.5166	-17.1808
14.4646	-7.5166	-17.1813
14.2721	-7.5166	-17.1817
13.8211	-7.5166	-17.1820

Cada columna representa las posiciones angulares de los pares cinemáticos de rotación en el orden respectivo, estas se almacenarán en el cuadro de dialogo con los intervalos de tiempo que se deseen para obtener el movimiento de eslabones adecuado. Los tres primeros eslabones ya configurados controlarán la trayectoria que realizará el robot.

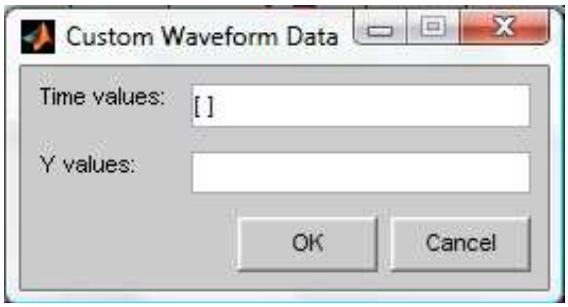


Figura 4.53 Cuadro de dialogo donde se debe ingresar cada columna de la matriz q.

4.8.1.1 Ejemplo de ingreso de señal solución de cinemática inversa.

Para empezar tomamos la primera columna de la matriz de cinemática inversa que se observo en la sección 4.8.1 que sería la siguiente:

Primera columna=

15.3918

15.2917

15.1905

15.0872

14.9803

14.8683

14.7484

14.6165

14.4646

14.2721

13.8211

Esta columna corresponde al primer eslabón del robot es decir la base giratoria del mismo. De tal manera que la base del robot girará acorde indique esta columna (unidades en radianes).

Primero añadimos una señal definida por el usuario como en la figura 4.52 y entonces procedemos a ingresar esta columna en el cuadro de dialogo de la figura 4.53 como se indica a continuación en la figura 4.54:

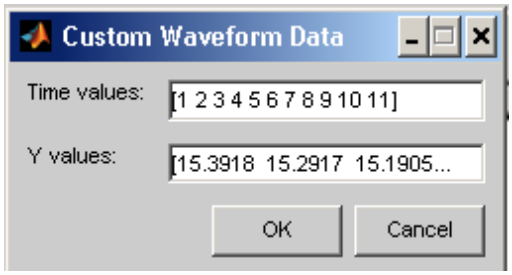


Figura 4.54 Datos de la primera columna de matriz solución siendo ingresados como señal de control.

Como se ve en la figura 4.54 en la casilla de *Y values* se debe ingresar cada valor de la columna separados por un espacio y todos estos dentro de corchetes tal como se ingresa una matriz fila en el *workspace* de Matlab y en la casilla de *Time values* se ingresar en pasos de a 1 cuantos valores son los que se ingresan en la casilla de *Y values* es decir 1, 2, 3…11 valores, de la misma manera se los ingresa separados con un espacio y dentro de corchetes como una matriz fila. De esta manera que definida la señal deposición angular de la base del robot.

Ahora debemos ingresar la señal de velocidad angular de la base del robot, considerando que se ha establecido una velocidad lineal de soldadura de 16 cm/seg en la sección 4.1 se debe calcular la velocidad angular.

$$velocidad\ angular = \frac{velocidad\ lineal}{radio\ de\ curvatura}$$

$$velocidad\ angular = \frac{16\ cm/seg}{100\ cm} = 0.16\ rad/seg$$

Ahora sabiendo que esta velocidad angular debe ser constante procedemos a ingresar esta señal:

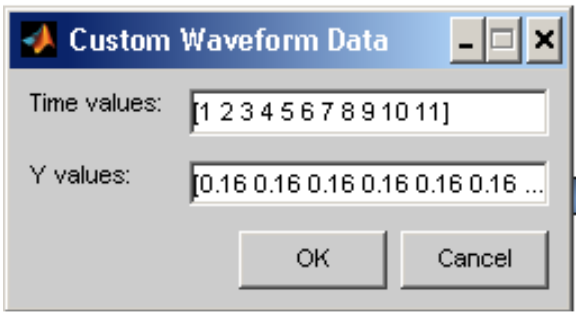


Figura 4.55 Señal de velocidad angular de la base del robot**.**

Como se ve en la figura 4.55 en la casilla de *Y values* se ingresa valores iguales a 0.16 que corresponden a la velocidad angular (11 valores) y en la casilla *Time values* números del 1 al 11 que indica que hay 11 valores en la matriz fila ingresada en la casilla inferior.

Finalmente para la señal de aceleración, conociendo que esta es nula, se procede a ingresar esta señal:

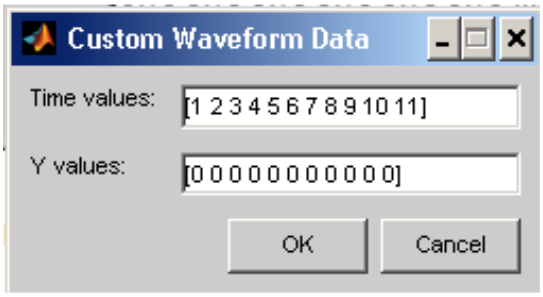


Figura 4.56 Señal de aceleración angular.

Como se observa en la figura 4.56, la casilla de *Y values* se muestran valores de cero (11 valores) y en la casilla de *Time values* pasos del 1 al 11 que indica que en la casilla de *Y values* hay 11 valores.

De esta manera quedan definidas las señales de control para el eslabón base del robot (posición, velocidad y aceleración).

Para identificar mejor las señales se les asigna los nombres respectivos como indica la figura 4.57.

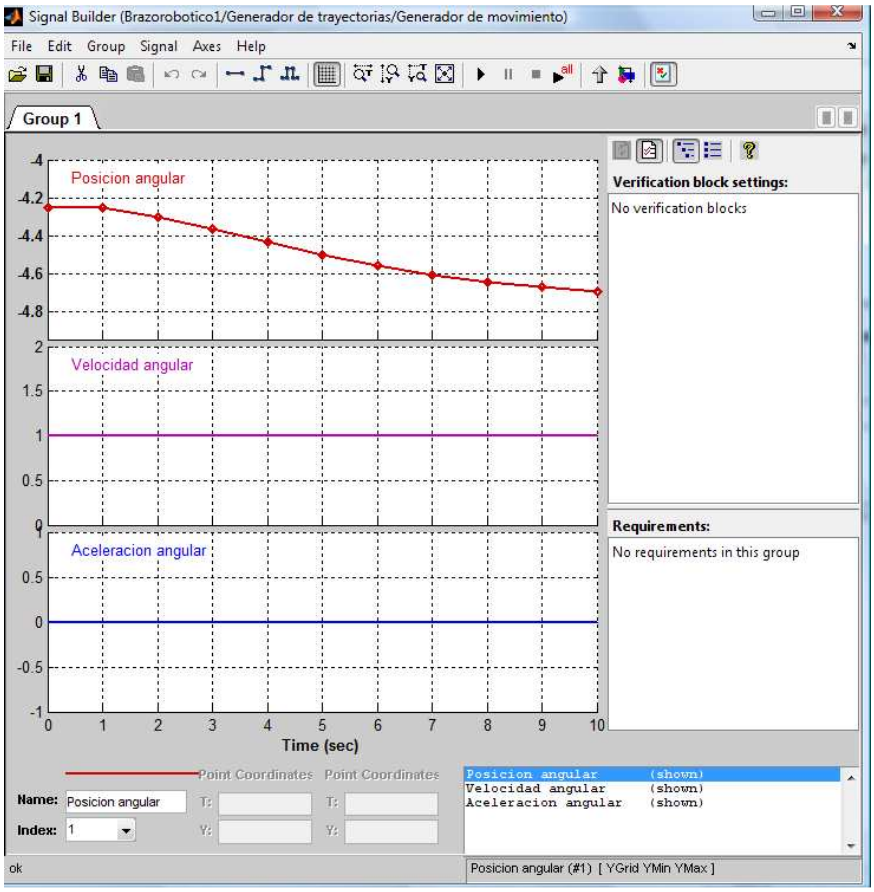


Figura 4.57 Señal requerida por cada par cinemático.

Entonces integramos los bloques mux y gain de esta manera:

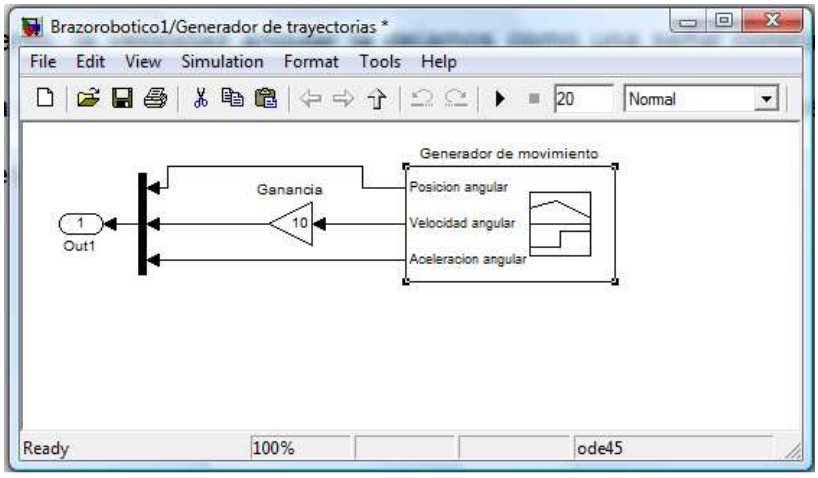


Figura 4.58 Subsistema de señales.

Hay que observar en la figura se ha establecido una ganancia a la salida de la señal de velocidad angular, esto permitirá variar la velocidad con que el eslabón base realice la tarea propuesta en la señal posición angular.

Para facilitar el trabajo agrupamos estos bloques en un subsistema. Para esto se selecciona los bloques que formarán un subsistema, se hace click derecho y click nuevamente en *Create subsystem*.

4.8.2 CONTROL DE STICK OUT (ALTURA DE ELECTRODO).

Para controlar el último eslabón (prismático) se requiere una señal independiente ya que en el proceso de soldadura este regula la separación entre el electrodo y el cordón de soldadura (stick out).

Se planteó inicialmente que este parámetro debe variar de 25mm a 50 mm uniformemente durante la soldadura, por tanto se deberá construir la señal adecuada para este propósito.

Tal como se realizó en la sección 4.8.1.1 se procede a añadir una señal definida por el usuario como se ve en la figura 4.52 y se procede a crear una señal de posición lineal que varié en 11 pasos de 25 mm a 50 mm. Luego una

señal de velocidad lineal constante para poder definirla después a través de una ganancia añadida posteriormente y una señal de aceleración lineal igual a cero.

Como se muestra la señal fue construida para que el stick out se desplace linealmente de 25 mm a 50 mm y con una velocidad constante.

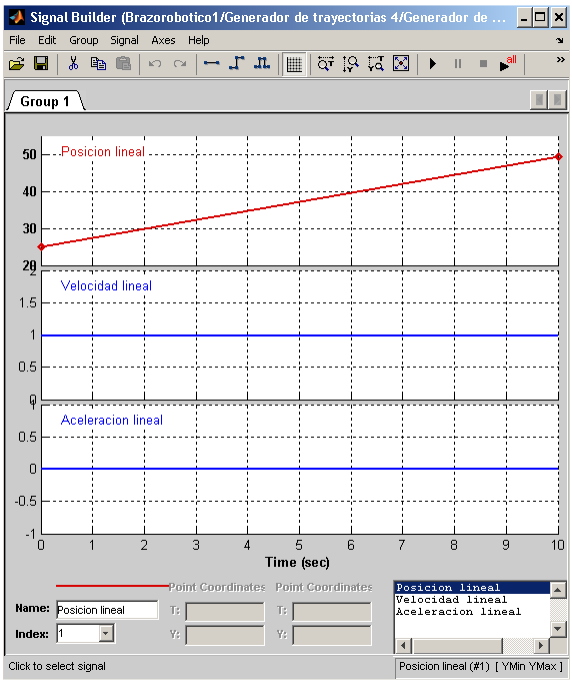


Figura 4.59 Señal de control de stick out.

Para lograr la variación de Stick out debemos regular la ganancia de la velocidad de 2.5 mm/seg.

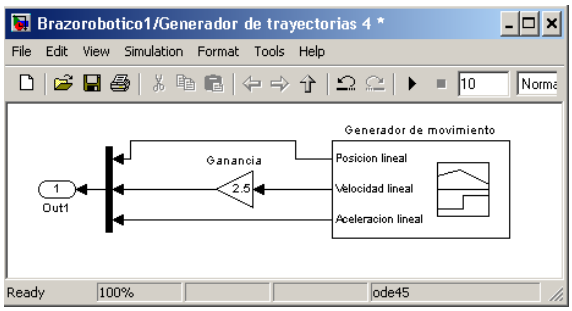


Figura 4.60 Regulación de ganancia para obtener la variación de stick out correcta.

De tal manera que el sistema queda construido e integrado como se indica en la figura 4.60 y 4.61:

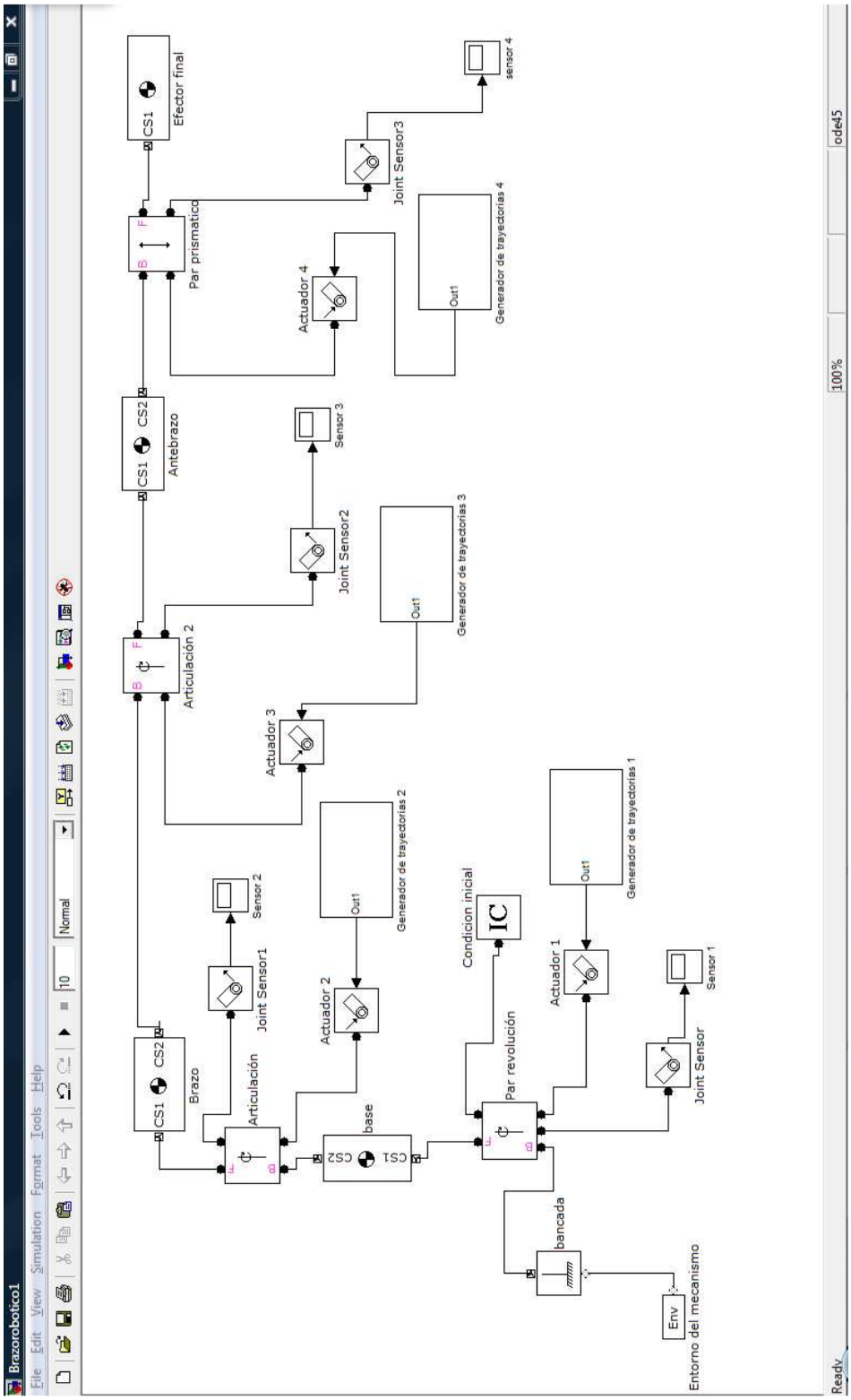


Figura 4.61 Sistema brazo-robot completo.

4.9 SIMULACIÓN.

Para poder llevar a cabo la simulación del robot se debe definir los valores de masa e inercia que quedaron parametrizados en los bloques de la figura 4.61, para esto se crea un archivo *.m* que almacene y grabe estos datos, este programa se incluye en los anexos.

Para ejecutar la simulación del brazo robot también es necesario ejecutar las rutinas que son necesarias para determinar valores de masas e inercias y cinemática inversa. Apenas se ejecuten las rutinas quedarán almacenados todos los valores de masas e inercias ya se podrá ejecutar la simulación final.

4.9.1 PARÁMETROS DE SIMULACIÓN.

Antes de ejecutar la simulación se debe establecer los parámetros con los que se llevará cabo la simulación. Estos son los siguientes:

4.9.1.1 Tiempo de simulación.

En la parte superior de la ventana de la figura 4.61 establecemos como tiempo de simulación 30 segundos el cual permitirá visualizar todo el proceso ya que este está estimado en realizarse en un tiempo menor a 15 segundos. Adicional a esto se establece una escala normal de tiempo ya que el tiempo de simulación no es tan extenso como para aplicar una aceleración a la simulación dinámica. Esto se muestra en la figura 4.62.

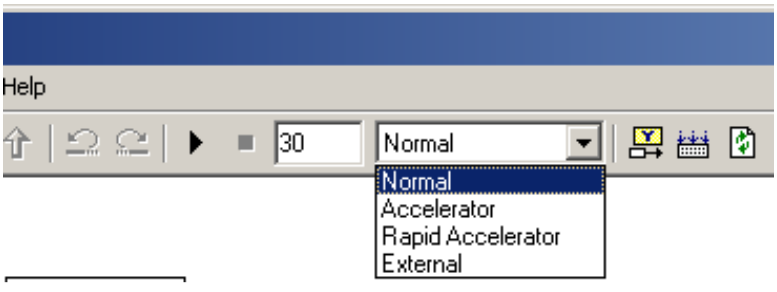


Figura 4.62 Tiempo y escala de tiempo de simulación.

4.9.1.2 Configuración de simulación.

Para llevar a cabo este paso se hace click en el botón *Simulation* en la parte superior de la ventana de la figura 4.61 y se hace click en la opción *Configuration Parameters* como se muestra en la figura 4.63.

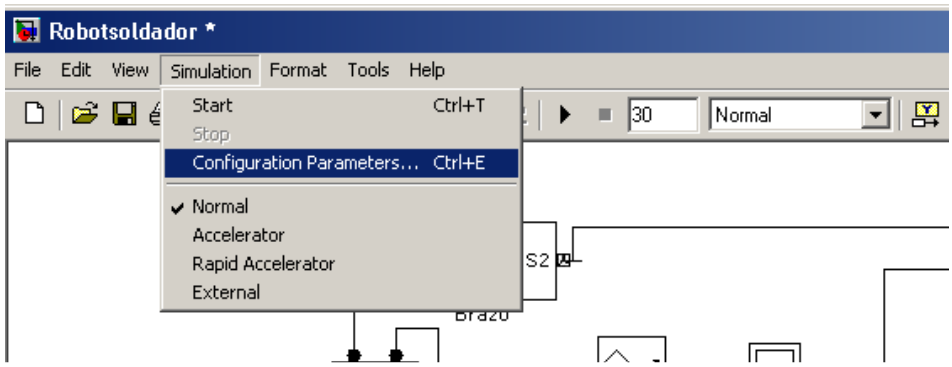


Figura 4.63 Configuración de parámetros de simulación.

A continuación aparece la ventana de la figura 4.64 donde no dirigimos a la opción *SimMechanics* y activamos la casilla *Show animation during simulation* que nos permitirá ver una animación del mecanismo durante la ejecución de la simulación.

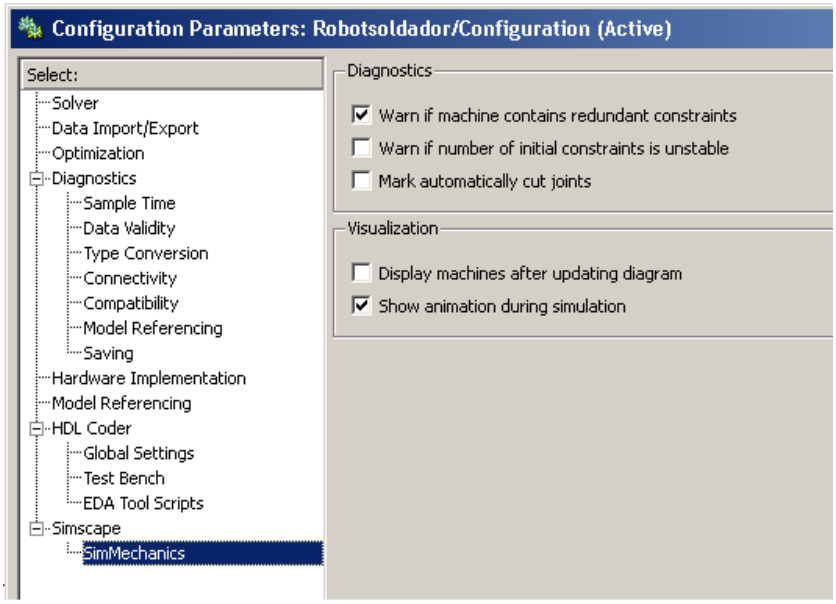


Figura 4.64 Opciones de configuración de parámetros de simulación.

De esta manera quedan establecidos los parámetros de simulación y se puede continuar ejecutando la misma.

4.9.2 EJECUCIÓN DE SIMULACIÓN.

Para ejecutar la simulación hacemos click en el botón *Start simulation* en la parte superior de la ventana de la figura 4.61 como se indica en la figura 4.65.

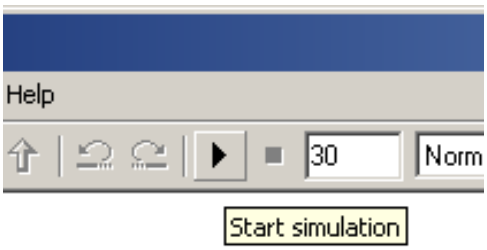


Figura 4.65 Botón *Start simulation.*

Al ejecutarse la simulación final del brazo robot se mostrará el funcionamiento de la máquina en la ventana de la figura 4.61.

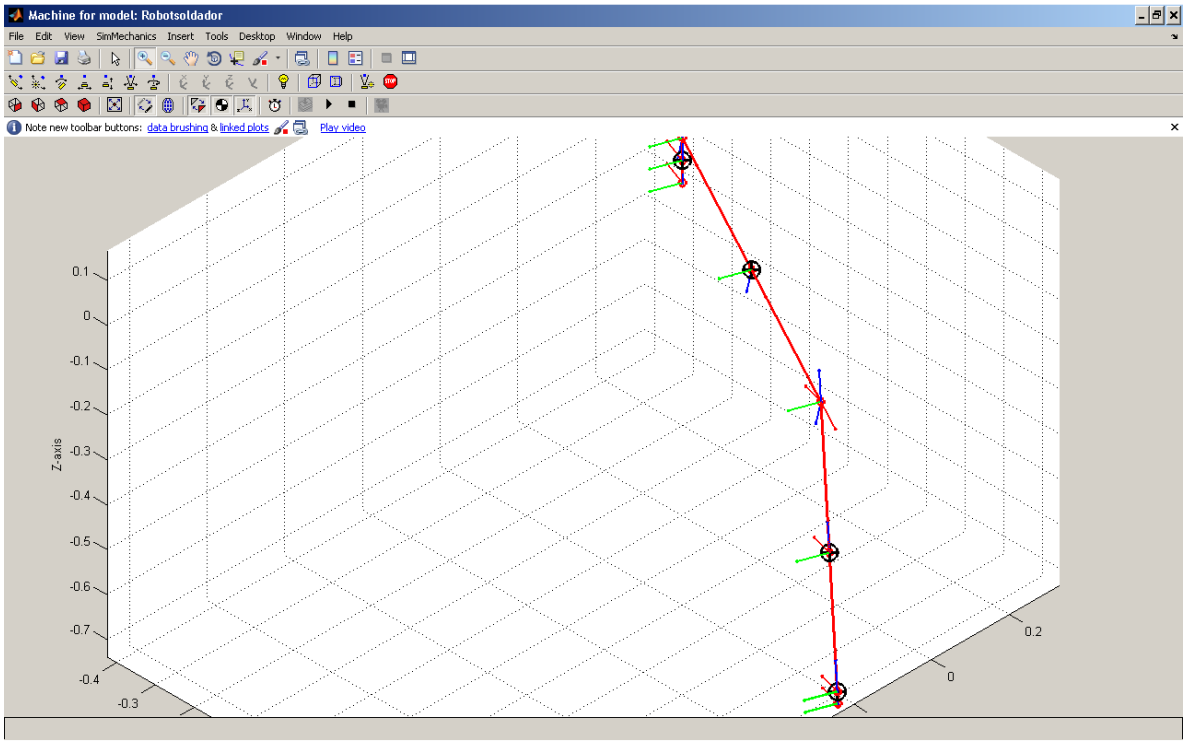


Figura 4.66 Inicio de simulación.

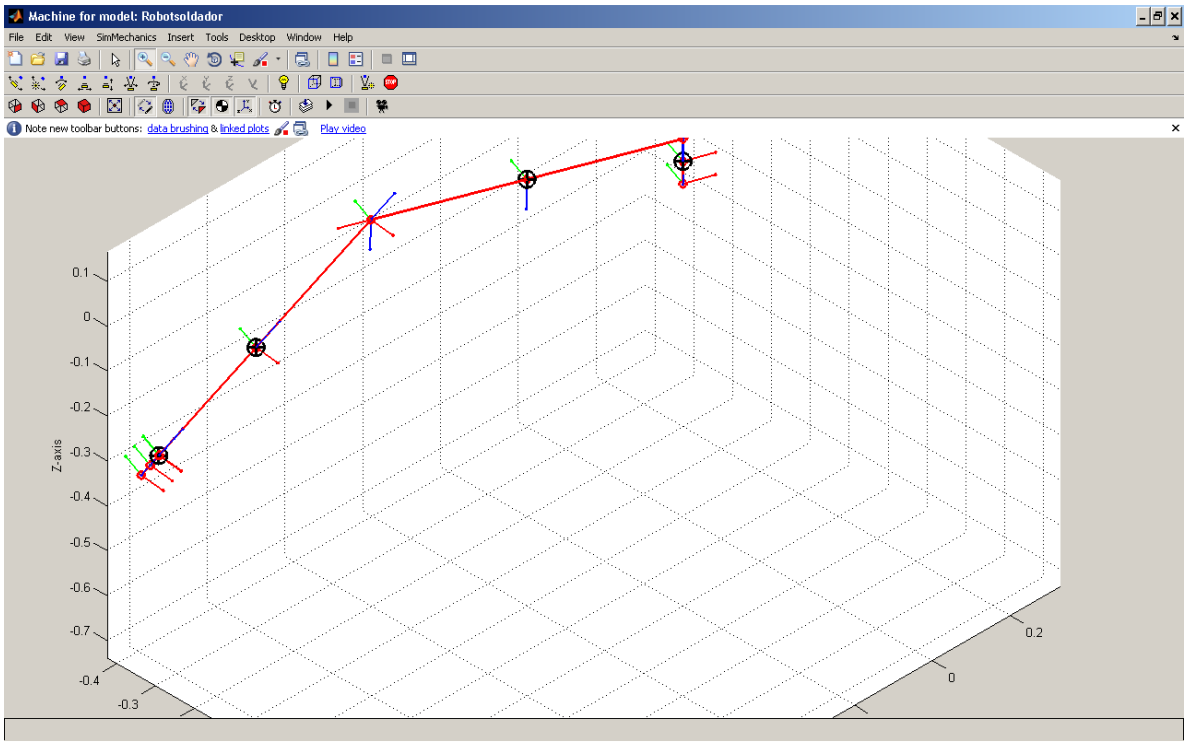


Figura 4.67 Final de la simulación.

Elaborando una secuencia cuadro a cuadro de lo que ocurre, obtenemos lo que muestra la figura 4.68 que muestra algunos puntos de la trayectoria de soldadura y su recorrido.

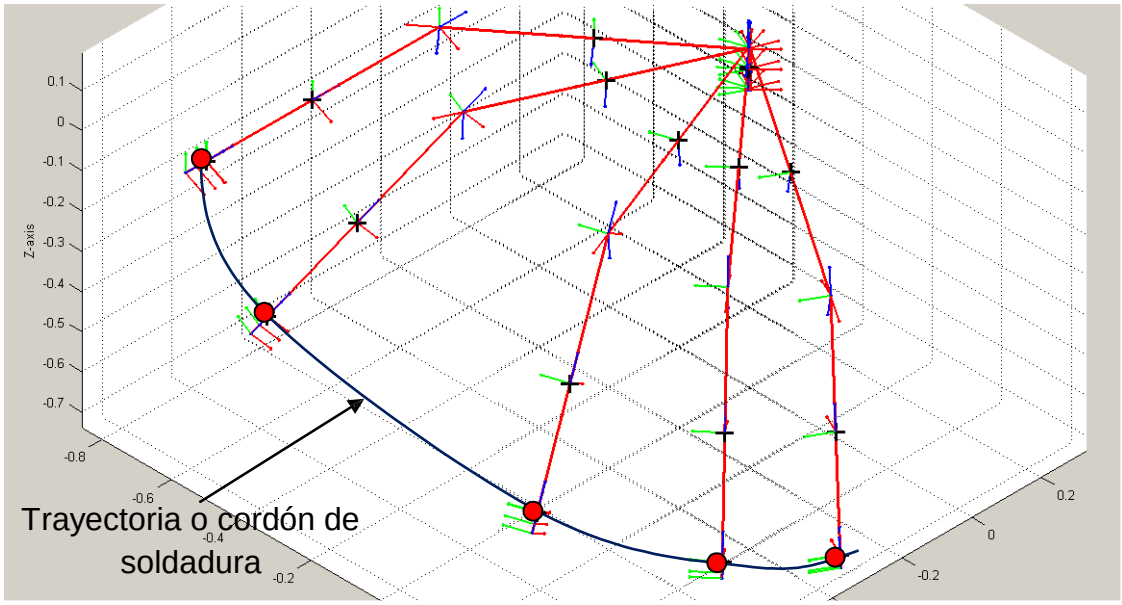


Figura 4.68 Secuencia de trabajo de robot

Durante la simulación se obtendrá los datos medidos de posición y velocidad angular alcanzada por cada eslabón que quedarán registrados en los sensores ubicados en cada par cinemático.

4.10 ANÁLISIS DE RESULTADOS

Es necesario contrastar los resultados de la simulación con los parámetros de soldadura que establecimos al inicio del problema.

4.10.1 CORDÓN DE SOLDADURA

En primer lugar el cordón de soldadura cumple con la trayectoria que se estableció, así muestra la figura 4.69 que corresponde a la solución de la cinemática inversa.

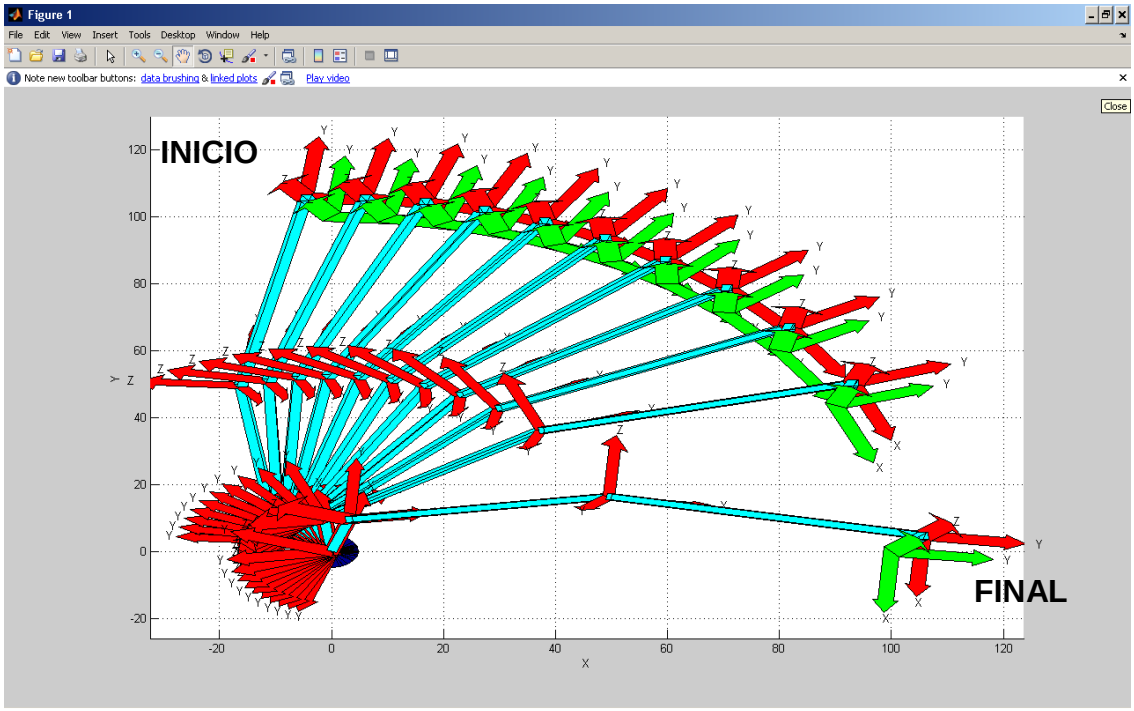


Figura 4.69 Secuencia que muestra la trayectoria que realiza el robot.

Esta es una vista aérea del robot en una secuencia de 11 pasos desde que arranca de la posición inicial hasta la final siguiendo la trayectoria deseada.

Para continuar el análisis de resultados se hace doble click en cada sensor del sistema brazo robot y se mostrarán las lecturas de posición y velocidad. Hay que tomaren cuenta que la señal en amarillo corresponde a la posición angular y la señal en purpura corresponde a la velocidad angular. Las interpretaciones de cada sensor se muestran a continuación de las figuras.

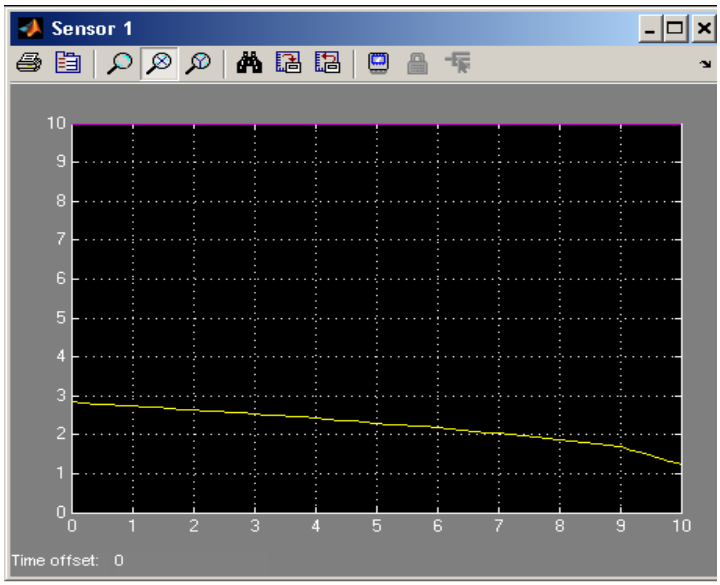


Figura 4.70 Sensor de movimiento en el eslabón base. Medición de posición angular (radianes) y velocidad angular (radianes por segundo)

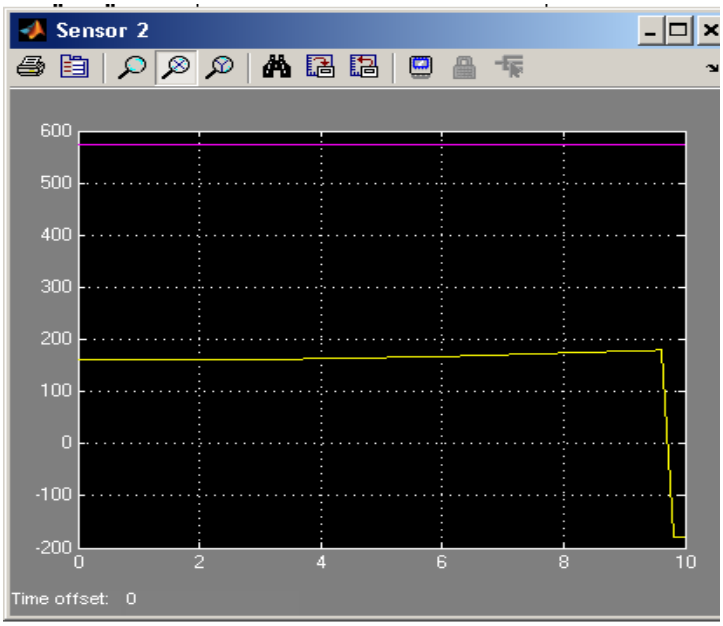


Figura 4.71 Sensor de movimiento en el Brazo. Medición de posición angular (grados) y velocidad angular (grados por segundo).

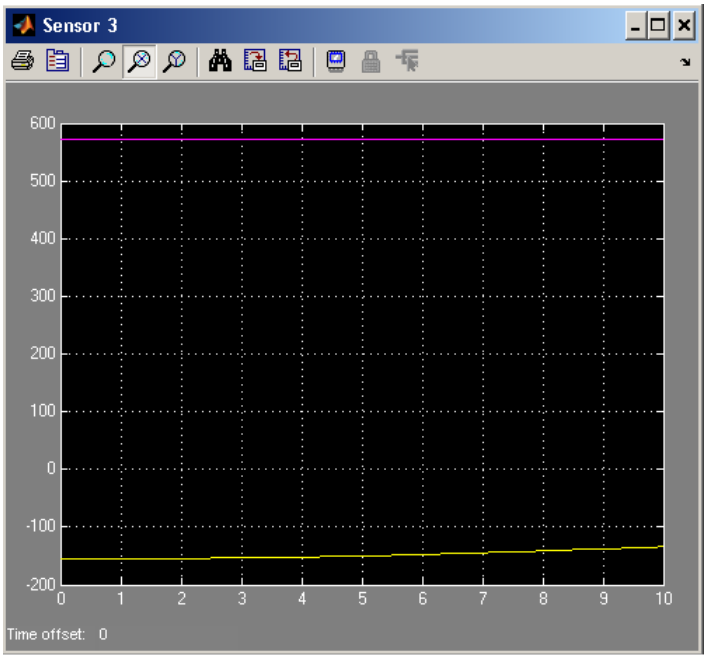


Figura 4.72 Sensor de movimiento en el antebrazo. Medición de posición angular (grados) y velocidad angular (grados por segundo).

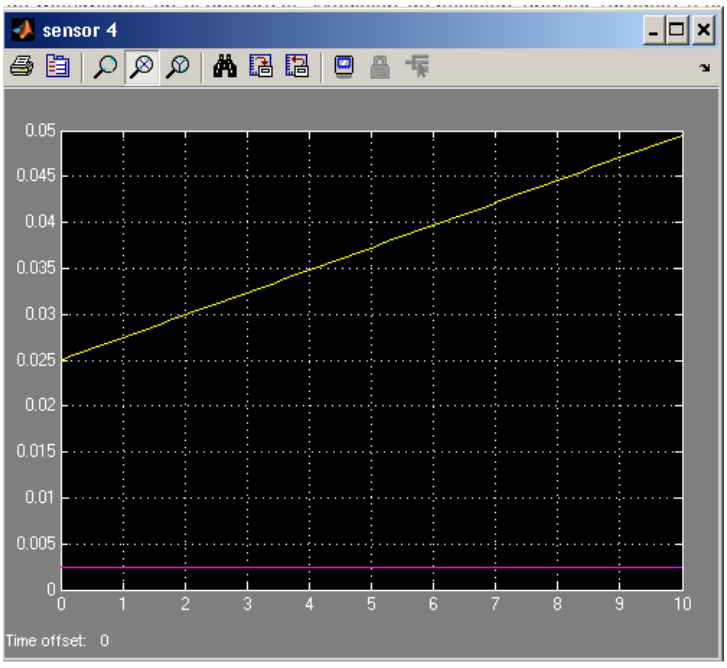


Figura 4.73 Sensor de movimiento en el efector final. Medición de posición lineal (metros) y velocidad lineal (metros por segundo). Aquí se observa la variación de stick out.

4.10.2 TIEMPO DE SOLDADURA.

Ahora en cuanto al tiempo de soldadura se cumple 11 como se puede observar en las lecturas de los sensores en las figuras de la 4.70 a 4.73.

4.10.3 VELOCIDAD DE SOLDADURA.

Como se puede observar en las lecturas de los sensores todas las velocidades angulares y de traslación (señales purpuras) son constantes, de esta manera el proceso de soldadura se lo lleva a cabo de manera uniforme.

4.10.4 STICK OUT.

La figura 4.73 representa la variación de posición de stick out y su posición, se observa que la velocidad con la que varia este es constante y su posición cambia de 25 mm 50 mm durante los 11 segundos que transcurre el proceso.

4.11 PROBLEMA ADICIONAL.

4.11.1 PLANTEAMIENTO.

Como problema adicional se plantea la elaboración del cordón de soldadura del nervio de la válvula tipo mariposa que refuerza la unión entre la brida a la sección de tubería. Para comprender la nueva tarea se debe observar la figura 4.74:

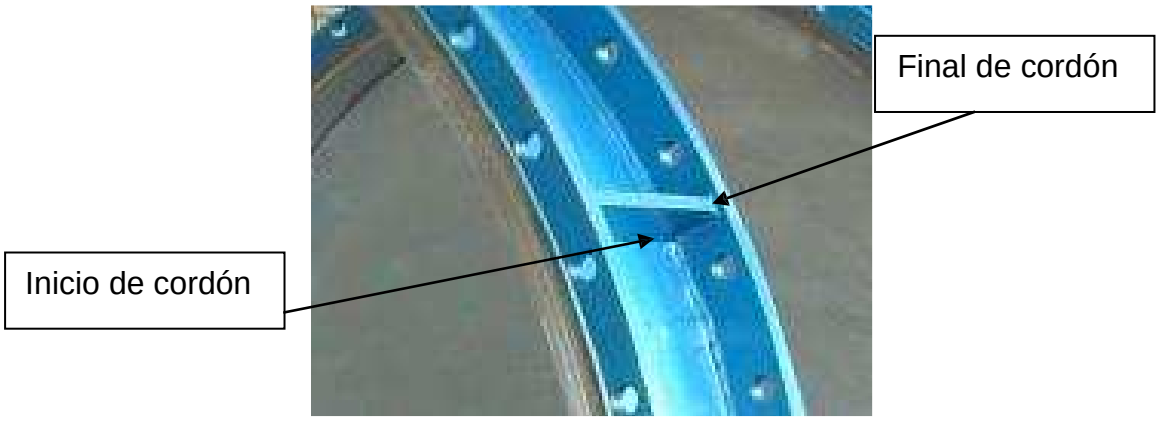


Figura 4.74 Válvula tipo mariposa.

En total son cuatro nervios que se ubican equidistantemente como indica la figura 4.75:

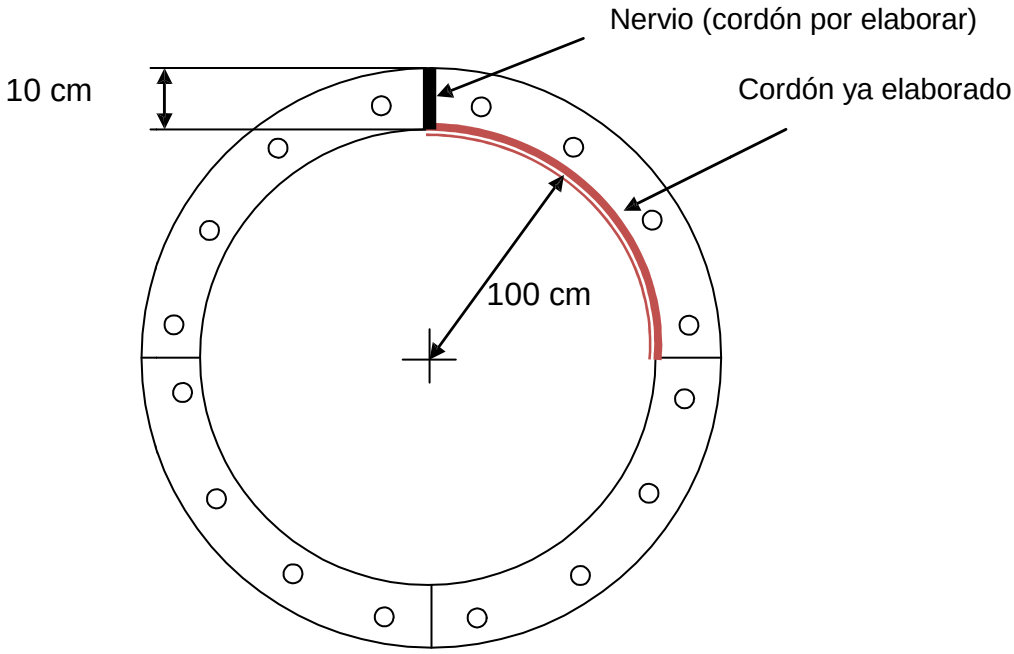


Figura 4.75 Cordones de soldadura.

Ya entendida la tarea de soldadura se procede a simular un solo cordón en representación de todos. Ahora en comparación del área de trabajo el cordón se dispone de la siguiente manera:

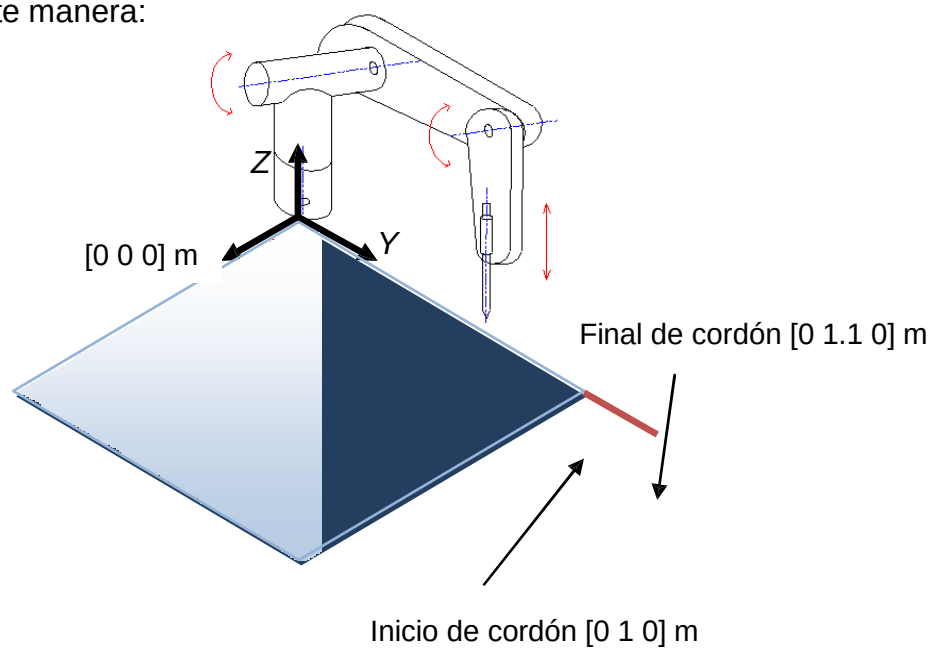


Figura 4.76 Posición del cordón respecto al robot.

En las secciones anteriores ya fue diseñado el robot que será empleado para realizar la tarea de soldadura (configuración mecánica, dimensionamiento y cinemática directa), ahora se requiere obtener la solución de la cinemática inversa para la nueva trayectoria que corresponde a una recta de 10 cm de longitud en la dirección del eje *Y*. Luego es necesario usar la solución de la cinemática inversa para crear la señal de control del robot. Los detalles de esta tarea de soldadura se ajuntan en anexos, a continuación se presenta los resultados de la simulación.

4.11.2 RESULTADOS.

Finalmente el cordón de soldadura se obtuvo correctamente según muestran las lecturas de los sensores ubicados en cada par cinemático y cuya interpretación se encuentran en los anexos. En la figura 4.77 se muestra la secuencia del movimiento del robot durante la simulación.

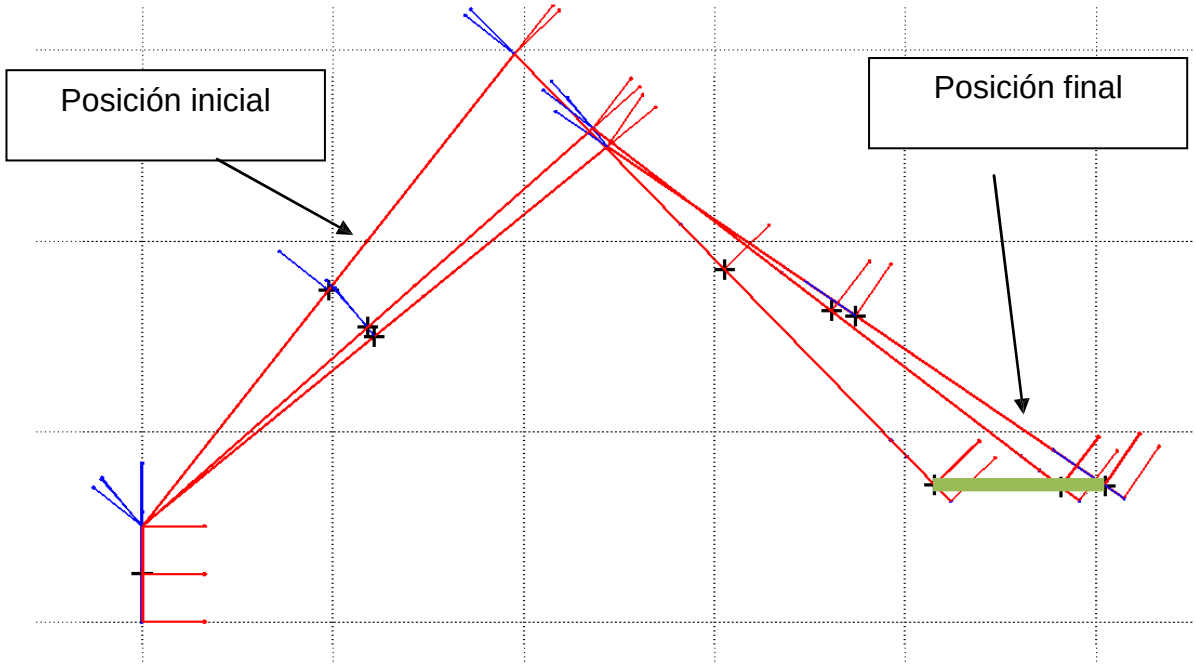


Figura 4.77 Movimiento del robot durante la simulación. Claramente se observa el cordón de soldadura que traza el robot desde la posición inicial hasta la posición final.

4.12 CONCLUSIÓN

La solución al problema planteado se empezó resolviendo por seleccionar que tipo de robot era el más idóneo para la tarea establecida, luego un dimensionamiento general del robot de esta manera poder calcular los parámetros de Denavit-Hartemberg y la cinemática inversa, a continuación se elaboró el modelo dinámico que servirá para dimensionar actuadores en caso de que se desee construir un prototipo del robot. Luego se empezó por crear el modelo de Simulink para de esta manera poder representarlo físicamente y poder controlarlo. Finalmente se crearon las señales de control para cada actuador y se ejecutó la simulación. Durante la ejecución se pudo observar cómo se comportó el robot físicamente durante el proceso de soldadura además de las lecturas de los sensores con las que se pudo corroborar que el proceso se llevo a cabo correctamente.

Habiéndose cumplido los parámetros de soldadura se puede contemplar la posibilidad de construir un prototipo para iniciar pruebas de funcionamiento.

Se puede variar las propiedades físicas del robot como masa e inercia desde la rutina correspondiente en cualquier momento, para esto se debe cambiar los datos en el programa de parámetros o si se requiere cambiar la trayectoria debemos hacerlo desde el programa de generación de trayectoria, luego de esto se procede a correr estas rutinas para que se actualice el sistema Brazo –robot con nuevos datos, en caso de haberse alterado la trayectoria se debe volver a cargar la matriz de solución de cinemática inversa en el constructor de señales respectivamente para cada eslabón, luego de realizado todo esto se puede volver a ejecutar la simulación del robot.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Los primeros capítulos de este trabajo comprende la base teórica para emprender en el estudio de la robótica desde el punto de vista de la mecánica.

El presente trabajo establece el camino previo a la construcción de un brazo robot para soldadura de arco en un próximo proyecto.

La simulación de la soldadura aquí planteada puede ser alternada fácilmente para otra trayectoria de soldadura que se desee y pueda físicamente realizarla el brazo robot.

Se ha demostrado la practicidad del uso de Simulink-SimMechanics para el estudio de la cinemática y dinámica de mecanismos.

Las rutinas creadas para la simulación pueden ser posteriormente aplicada para una construcción de un prototipo controlado con la plataforma Matlab en tiempo real.

El modelo dinámico del robot queda planteado para su eventual uso en un sistema de control avanzado.

Para recrear físicamente el sistema brazo robot se puede utilizar una interfaz conexión serial propia para Matlab que se puede encontrar fácilmente en el mercado.

El toolbox SimMechanics es una poderosa herramienta para trabajar en el área de la robótica al simplificar en gran medida el trabajo matemático.

5.2 RECOMENDACIONES

Implementar este tipo de software como herramienta para el estudio de las materias del pensum de Ingeniería Mecánica.

Reforzar los conocimientos de matemáticas avanzadas y algebra lineal ya que estas son herramientas muy importantes durante la investigación.

Difundir este proyecto para su posterior ejecución y uso por parte del laboratorio de automatización por ejemplo.

Ampliar el pensum de la Facultad en el campo de la robótica ya que esta rama de la ingeniería se vuelve cada vez más importante en la industria a nivel mundial.

Para mejorar la visualización final del mecanismo durante la simulación se puede emplear el uso de un software adicional de diseño grafico que permita mostrar más detalladamente el proceso de soldadura.

ANEXOS

ANEXO 1. VISUALIZACIÓN DEL ROBOT EN UNA POSICIÓN ALEATORIA.

La siguiente rutina muestra al robot en una posición estática para poder apreciarlo.

```
%Representación grafica del robot soldador%
%definimos los parámetros de Denavit Hartemberg del robot%
t1=pi/2; t2=pi/2; t3=pi/2;
a1=10; a3=63; a4=63; d4=10;
dh = [0     0   t1   10  0;
     -pi/4  10   t2   0  0;
      -pi/2     a3  t3   0  0;
     pi/2   a4 -pi/2 0  1;
      0     0    0   d4 1]
% inicial q0=[0 0 0] y una máscara [1 1 0 0 0 1]
q=[pi/2 -pi/2 pi/4 0 -5]
plotbot(dh,q,'lfdw')
```

ANEXO 2. RUTINA PARA ESTABLECER LOS PARÁMETROS DEL ROBOT.

La rutina que se presenta a continuación es una declaración de variables para estableces los valores de masa y medidas de cada elemento del robot para que de esta manera se pueda calcular las matrices de inercias respectivas de cada elemento del robot.

```
syms Ixx1 Iyy1 Izz1 Ixx2 Iyy2 Izz2 Ixx3 Iyy3 Izz3 m1 m2 m3 r
h a b c d L1 L2
Ixx1=(m1/12)*(3*r^2+h^2)
Iyy1=(m1/12)*(3*r^2+h^2)
Izz1=(m1*r^2)/2
Tensorinercia1=[Ixx1 0 0;0 Iyy1 0; 0 0 Izz1]
m1=10
r=.05
h=.1
Tensorinercia1=[Ixx1 0 0;0 Iyy1 0; 0 0 Izz1]
Izz1=(m1*r^2)/2
Iyy1=(m1/12)*(3*r^2+h^2)
Ixx1=(m1/12)*(3*r^2+h^2)
Tensorinercia1=[Ixx1 0 0;0 Iyy1 0; 0 0 Izz1]
Ixx2=(m2/12)*(a^2+L1^2)
Iyy2=(m2/12)*(b^2+L1^2)
Izz2=(m2/12)*(b^2+a^2)
a=.03
m2=25
b=.08
L1=0.65
Izz2=(m2/12)*(b^2+a^2)
Iyy2=(m2/12)*(b^2+L1^2)
Ixx2=(m2/12)*(a^2+L1^2)
Tensorinercia2=[Ixx2 0 0;0 Iyy2 0; 0 0 Izz2]
Ixx3=(m3/12)*(c^2+L2^2)
Iyy2=(m3/12)*(d^2+L2^2)
Izz2=(m3/12)*(c^2+d^2)
```

```
Iyy2=(m3/12)*(d^2+L2^2)
Izz3=(m3/12)*(c^2+d^2)
Iyy2=(m2/12)*(b^2+L1^2)
Izz2=(m2/12)*(b^2+a^2)
Ixx3=(m3/12)*(c^2+L2^2)
Izz2=(m2/12)*(b^2+a^2)
Iyy2=(m2/12)*(b^2+L1^2)
Ixx2=(m2/12)*(a^2+L1^2)
Tensorinercia2=[Ixx3 0 0;0 Iyy3 0; 0 0 Izz3]
Tensorinercia3=[Ixx3 0 0;0 Iyy3 0; 0 0 Izz3]
Tensorinercia2=[Ixx2 0 0;0 Iyy2 0; 0 0 Izz2]
m3=25
m3=20
c= .03
d=.05
L2=0.60
Tensorinercia3=[Ixx3 0 0;0 Iyy3 0; 0 0 Izz3]
Ixx3=(m3/12)*(c^2+L2^2)
Iyy3=(m3/12)*(d^2+L2^2)
Tensorinercia3=[Ixx3 0 0;0 Iyy3 0; 0 0 Izz3]
Izz3=(m3/12)*(c^2+d^2)
Tensorinercia3=[Ixx3 0 0;0 Iyy3 0; 0 0 Izz3]
```

ANEXO 3. RUTINA PARA ENCONTRAR LA CINEMATICA INVERSA DEL ROBOT.

```
%Representación grafica del robot soldador%
%definimos los parámetros de Denavit Hartemberg del robot%
t1=pi/2; t2=pi/2; t3=pi/2;
a1=10; a3=63; a4=63;
dh = [0     0   t1   10  0;
     -pi/4  10   t2   0  0;
      -pi/2     a3  t3   0  0;
     pi/2   a4 -pi/2 0  1;
      0     0    0   5 1]


% Se establecen la tolerancia y el número máximo de
iteraciones
stol=1e-6; ilimit=1000;
% Se introduce la trayectoria planteada
x=[0:10:100];
y=sqrt(10000-x.^2);
phi=zeros(1,length(x));
% Se crea la trayectoria de transformaciones


for k=1:length(x)
   TG(:,:,k) = [cos(phi(k)) -sin(phi(k)) 0 x(k);
                sin(phi(k)) cos(phi(k)) 0 y(k);
                0 0 1 0;
                0 0 0 1];
end
 % Se calcula el modelo inverso para cada uno de los puntos
de la trayectoria usando un vector
% inicial q0=[0 0 0] y una máscara [1 1 0 0 0 1]
q=ikine(dh,stol,ilimit,TG,[0 0 pi/4 0 -5],[1 1 1 0 0 1]);
 % inicial q0=[0 0 0] y una máscara [1 1 0 0 0 1]
 plotbot(dh,q,'lfdw')
```

En la primera parte de la rutina se ingresa una matriz de nombre *dh* que representa los parámetros de Denavit-Hartemberg obtenidos en la sección 4.2. Esto definirá al robot por completo para poder darle instrucciones más adelante.

Luego se define la tarea de soldadura, es decir la trayectoria que realizará el robot representada por una función real en este caso es la función:

$$y = \sqrt{10000 - x^2}$$

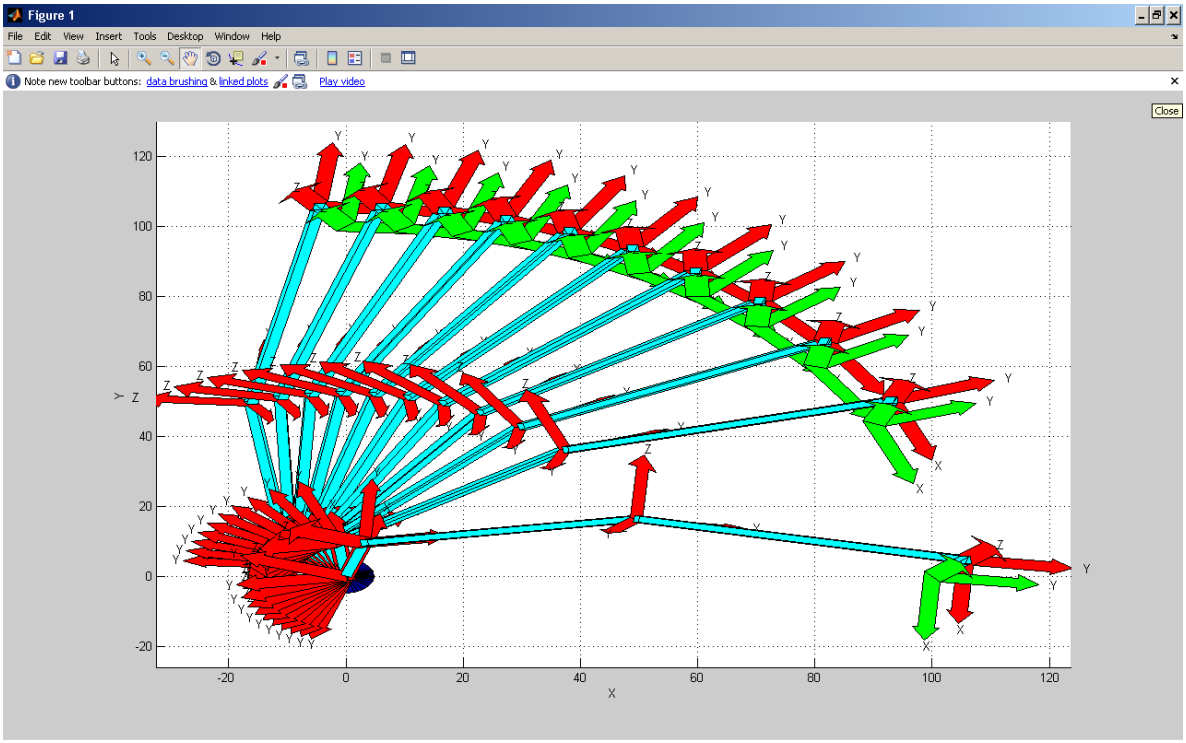
Se define una matriz *x* como dominio:

$$x = [0{:}\,100]$$

Tomando valores para cada 10, es decir 0, 10, 20, etc.

Después se crea una matriz de una columna y de numero de filas como numero de valores tenga el dominio de *x*. Para cada matriz se obtendrá la transformación lineal correspondiente. Por último mediante el comando *ikine* se resolverá la cinemática inversa representada por una matriz de una columna por cada grado de libertad y filas según el número de valores tenga el dominio de *x*. Esta matriz indicará como se mueve cada eslabón de acuerdo al resto de eslabones en cada paso. La matriz solución quedará almacenada en la variable q como se especifica en la rutina.

Al ejecutar esta rutina se obtendrá la siguiente figura que muestra al robot realizando la trayectoria planteada en la rutina.



Este es un algoritmo generalizado, ya que solo debemos cambiar la trayectoria (función real) para tener otra solución al problema cinemático inverso del robot.

También se mostrará la matriz solución en el *command window* de Matlab.

La matriz solución q es la siguiente:

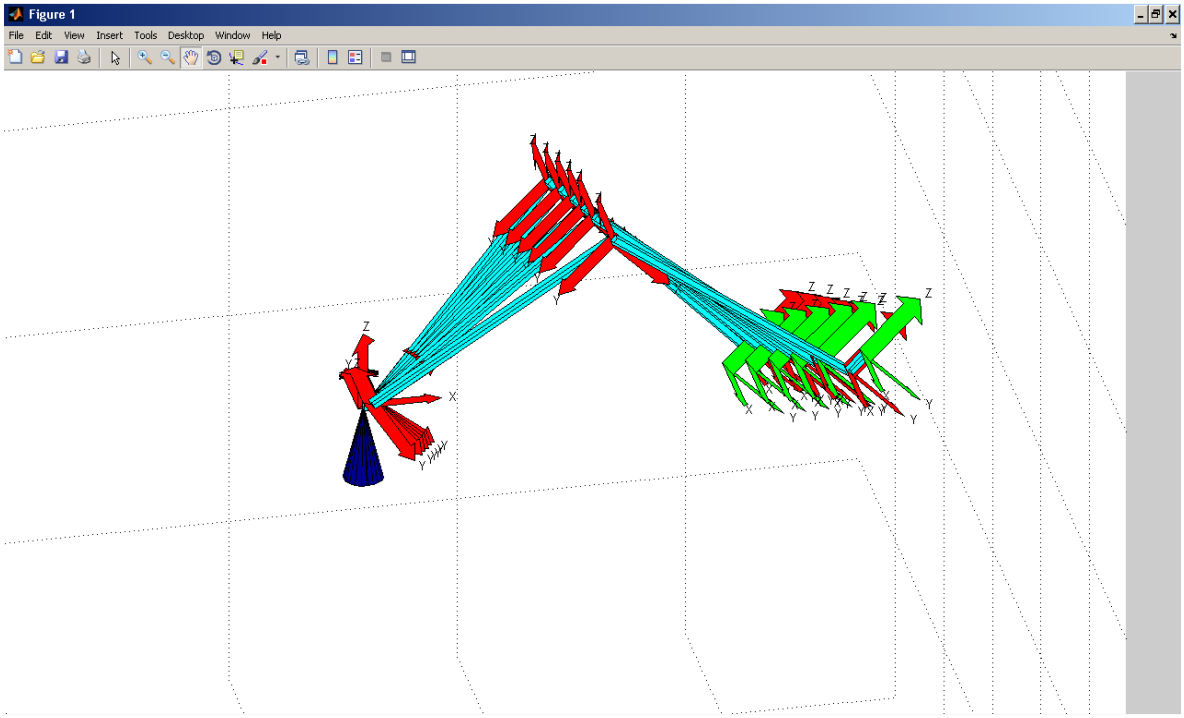
```
>> q

   q =

   15.3918   -7.5167    1.0026  -17.1774  -22.1774

   15.2917   -7.5167    1.0026  -17.1779  -22.1779

   15.1905   -7.5167    1.0026  -17.1784  -22.1784

   15.0872   -7.5166    1.0026  -17.1789  -22.1789

   14.9803   -7.5166    1.0026  -17.1794  -22.1794

   14.8683   -7.5166    1.0026  -17.1799  -22.1799

   14.7484   -7.5166    1.0026  -17.1804  -22.1804

   14.6165   -7.5166    1.0025  -17.1808  -22.1808
```

14.4646	-7.5166	1.0025	-17.1813	-22.1813
14.2721	-7.5166	1.0025	-17.1817	-22.1817
13.8211	-7.5166	1.0025	-17.1820	-22.1820

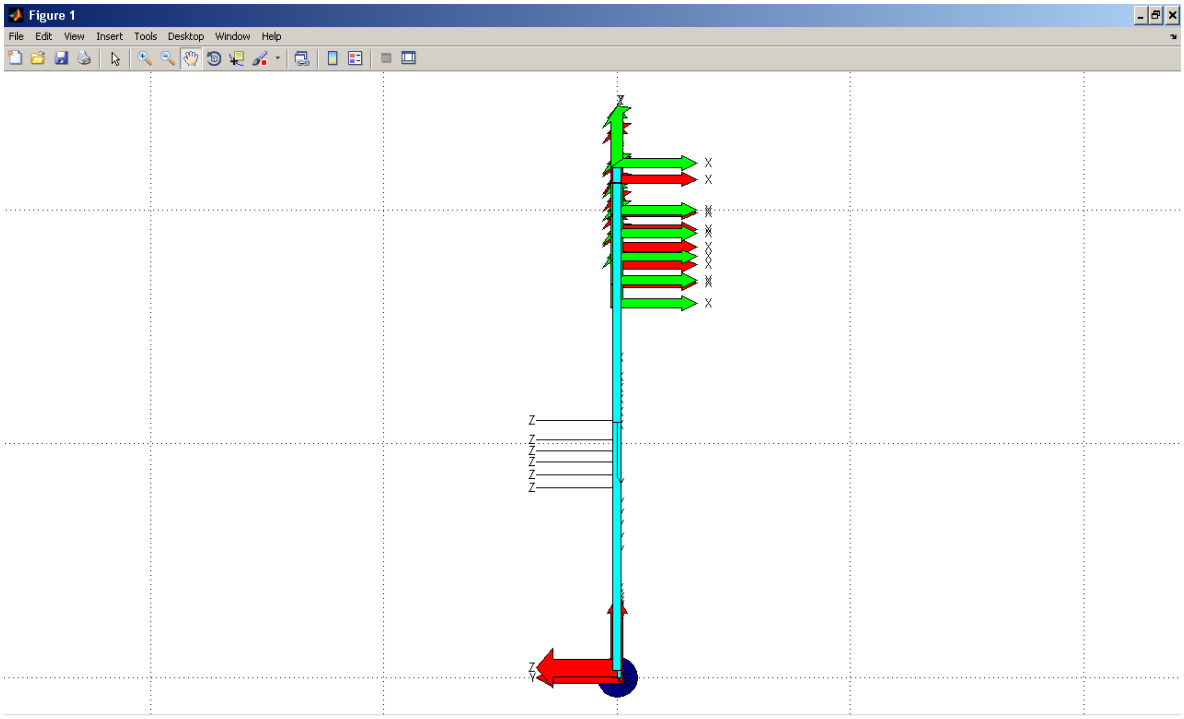
ANEXO 4. RUTINA PARA ENCONTRAR LA CINEMATICA INVERSA DE LA TAREA ADICIONAL.

```
%Representación grafica del robot soldador%
%definimos los parámetros de Denavit Hartemberg del robot%
t1=0; t2=0; t3=0;
a3=63; a4=63; d4=10;
dh = [0     0   t1  0  0;
     -pi/2  2   t2  0  0;
      0     a3  t3  0  0;
     pi/2   a4 -pi/2 0 1
     0      0    0  d4 1  ];

%%trayectoria de cordón en el nervio de la brida%%
% Se establecen la tolerancia y el número máximo de
iteraciones
stol=1e-6; ilimit=1000;
% Se introduce la trayectoria del cordón%
x=[0 0 0 0 0];
y=[80:5:100]
phi=zeros(1,length(x));
% Se crea la trayectoria de transformaciones

for k=1:length(x)
   TG(:,:,k) = [cos(phi(k)) -sin(phi(k)) 0 x(k);
                sin(phi(k)) cos(phi(k)) 0 y(k);
                0 0 1 0;
                0 0 0 1];
end

% Se calcula el modelo inverso para cada uno de los puntos de
la trayectoria usando un vector
% inicial q0=[0 0 0] y una máscara [1 1 0 0 0 1]
q=ikine(dh,stol,ilimit,TG,[0 0 pi/4 0 -5],[1 1 1 0 0 1]);

plotbot(dh,q,'lfdw')
```

La ejecución de la rutina anterior da como resultado la siguiente figura:



Que desde una vista superior se ve así:



A continuación se crea la señal de control utilizando la matriz solución de la cinemática inversa almacenada en q:

q =

7.8540	-13.4766	14.2620	-1.1776	-6.1776
7.8540	-13.4202	14.2056	0.4187	-4.5813
7.8540	-13.3641	14.1495	2.1151	-2.8849
7.8540	-13.3079	14.0933	3.9107	-1.0893
7.8540	-13.2517	14.0371	5.8060	0.8060

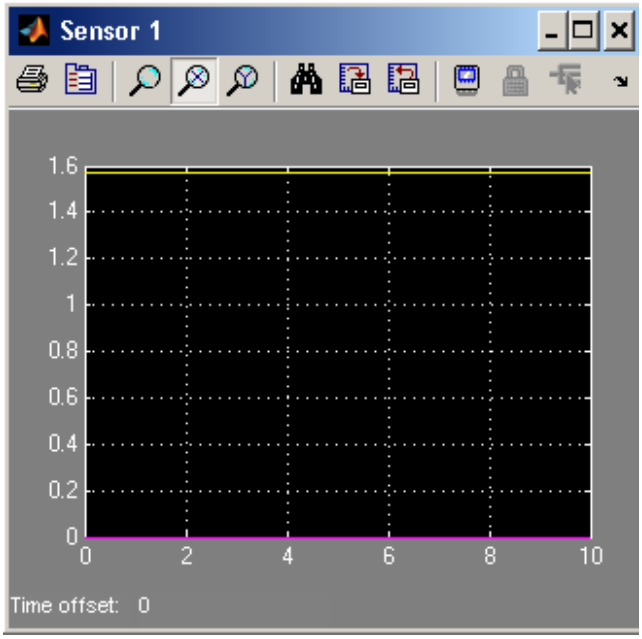
Para la simulación de esta tarea se ha variado los siguientes parámetros:

Stick out: invariante a 2.5 mm

Velocidad de soldadura: 4 cm/seg

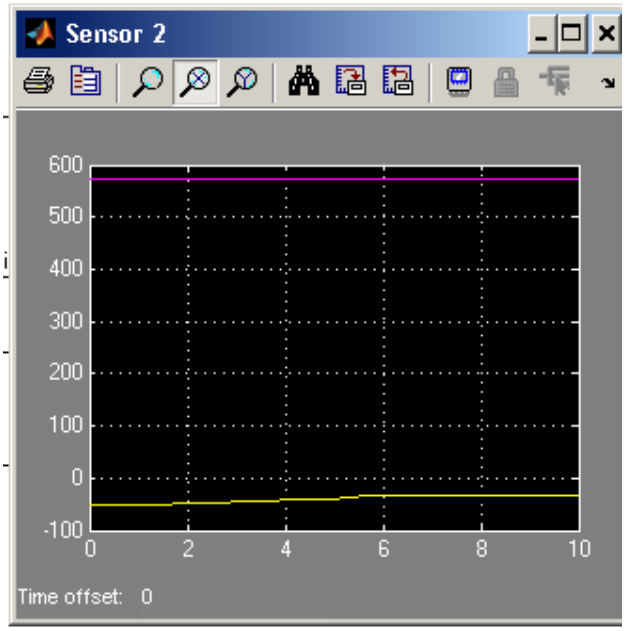
Durante la ejecución de la simulación se obtienen las siguientes lecturas de sensores.

Sensor de base:



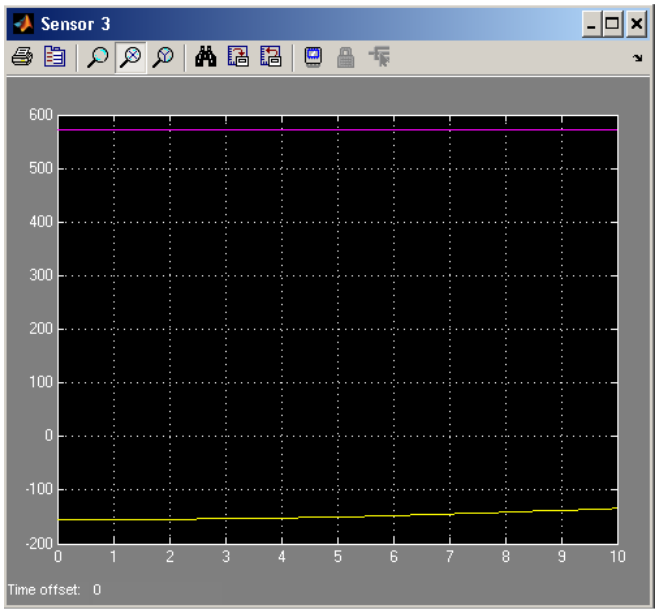
Donde se observa que la señal de posición angular (señal amarilla) permanece constante y la señal de velocidad angular (señal purpura) en cero como se esperaba.

Sensor Brazo:



Se observa en la figura una señal de velocidad constante y una de posición variable.

Sensor antebrazo:



Como se observa en la figura la señal de posición varía uniformemente y la velocidad se mantiene constante.