

Configuración y Programación del Controlador

Difuso

Una vez que el Robot UDLAP está totalmente construido en el ambiente de simulación y es posible observar su representación grafica, el siguiente y último paso del proceso será diseñar y programar los controladores difusos que generarán el movimiento de las articulaciones. La programación y configuración del controlador se realizó utilizando *Fuzzy Logic Toolbox*, herramienta que permitió elegir el método de inferencia, el método de defusificación, editar las variables y los conjuntos difusos, así como las reglas que describirán el comportamiento de los controladores.

6.1 Definición de las Variables

Se consideraron diferentes variables para utilizarse en el controlador, pero basándose en los documentos [11] y [12] se decidió utilizar como variables el *Error*, generado por el ángulo deseado de la articulación y el ángulo actual de la articulación, y el *Cambio en el Error*, que es la *Velocidad* a la que se mueve la articulación. A la salida el

controlador debe generar el *Torque* que moverá a las articulaciones. El diagrama de la figura 6.1 muestra del control que se implementará en las articulaciones.

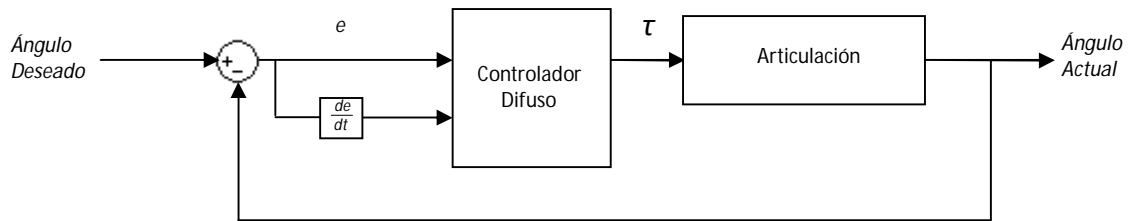


Figura 6.1: Diagrama a bloques del Control

Las variables de entrada *Error* y *Velocidad* y la variable de salida *Torque* que se utilizarán para controlar el movimiento de la articulación de la cintura, se conforman por cinco conjuntos difusos. Todas las variables cuentan con conjuntos difusos de forma triangular y trapezoidal. Las etiquetas lingüísticas que se asignaron a los conjuntos de las variables son cuando el error es: Muy Negativo, Negativo, Cero, Positivo y Muy Positivo. La figura 6.2 muestra los conjuntos difusos que conforman la variable de salida *Error*.

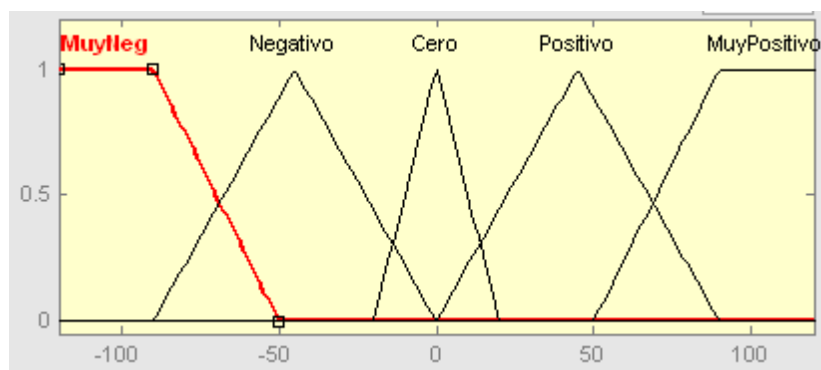


Figura 6.2: Conjuntos difusos de la variable Error

El rango de valores para esta variable se asignó de acuerdo a los límites de movimiento del área de trabajo la articulación de la cintura, que es de 120° .

Los conjuntos difusos de la variable *Velocidad*, así como las etiquetas lingüísticas que se les asignaron se pueden observar en la figura 6.3.



Figura 6.3: Conjuntos difusos de la variable *Velocidad*

El rango de valores para la variable *Velocidad* se obtuvo de una simulación del Robot UDLAP en SimMechanics. Los valores se obtuvieron de una simulación en la que se conectó un *Scope* a un bloque sensor el cual midió la velocidad a la que se movía la articulación con un torque conocido. El *Scope* permitió observar que la articulación se movía a una velocidad máxima de 50 grados por segundo cuando se le aplicaba un torque de 2 N·m.

Los conjuntos que conforman la variable de salida *Torque* y las etiquetas lingüísticas que se les asignaron se muestran en la figura 6.4.

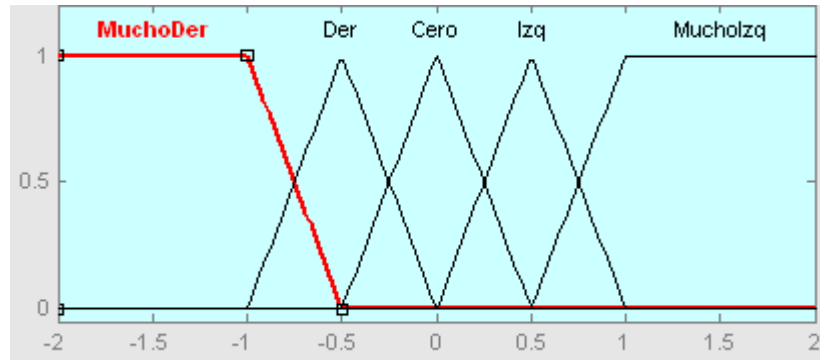


Figura 6.4: Conjuntos difusos de la variable Torque

Los valores del rango que se utilizaron se aproximan a los de las especificaciones del Robot UDLAP. Para el caso de la articulación de la cintura es necesario un torque estático de 1.2 N·m.

6.2 Definición de la base de Conocimiento

Una vez que se han descrito las variables de entrada con sus respectivos conjuntos difusos es necesario establecer las reglas que conformarán la base de conocimiento que regirán el comportamiento del controlador. En esta parte se recurrió al conocimiento experto de los directores de este trabajo, uno experto en el área de control y otro experto en el área de lógica difusa. Con su ayuda se creó una tabla de inferencia con las etiquetas lingüísticas de los conjuntos difusos de las variables de entrada y salida. La tabla 6.1 muestra la tabla de inferencia que se generó para la articulación de la cintura.

Tabla 6.1: Tabla de inferencia para la articulación de la cintura

| Torque | | Velocidad | | | | |
|--------|--------------|------------|-----------|-----------|-----------|------------|
| | | Rapido Der | Der | Cero | Izq | Rápido Izq |
| ERROR | Muy Neg | Cero | Der | Mucho Der | Mucho Der | Mucho Der |
| | Negativo | Der | Cero | Der | Mucho Der | Mucho Der |
| | Cero | Mucho Izq | Izq | Cero | Der | Mucho Der |
| | Positivo | Mucho Izq | Mucho Izq | Izq | Cero | Izq |
| | Muy Positivo | Mucho Izq | Mucho Izq | Mucho Izq | Izq | Cero |

La tabla representa las reglas que se utilizarán en el proceso de inferencia y que regirán el comportamiento del controlador de la articulación de la cintura.

En el proceso de fusificación los valores de entrada que ingresen al controlador se compararán con sus respectivos conjuntos difusos en las variables para determinar el conjunto al que pertenece. Posteriormente se calculará el grado de membresía o pertenencia del valor de entrada en el conjunto difuso al que pertenezca [9].

La inferencia es el proceso mediante el cual se genera un mapeo para asignar a una entrada una salida utilizando lógica difusa. El proceso de inferencia provee las bases para la toma de decisiones del sistema. Este proceso involucra la utilización de funciones de membresía y las reglas generadas en la base de conocimiento [9].

Existen dos tipos de métodos de inferencia que pueden ser implementados en el Fuzzy Logic Toolbox: el método Mamdani y el método Sugeno. El método de inferencia

Mamdani es el que se utiliza con mayor frecuencia y fue el primero en utilizarse en los sistemas de control difuso.

El método Mamdani utiliza los grados de membresía de las variables de entrada que se generaron en la fusificación y se aplican sobre las reglas de la base de conocimiento para determinar su grado de membresía en los conjuntos difusos de la variable de salida. Con los grados de membresía de la variable de salida se genera un polígono que refleja los valores de membresía a lo largo de los conjuntos difusos de la variable de salida [6].

Por último se realiza la defusificación mediante el método del Centroide, cuya implementación es muy práctica y resulta muy eficiente. Este método también conocido como Centro de Área (COA) calcula el centro de gravedad del polígono que se generó en la inferencia.

Las variables de entrada y de salida que se usaron para controlar el movimiento de las demás articulaciones son similares a las que se usaron para la articulación de la cintura, así como la forma y distribución de sus conjuntos difusos. Lo que difiere en las variables es el rango de los valores de los conjuntos difusos. En el caso de la variable *Error* los rangos son diferentes porque las articulaciones tienen un rango de giro diferente. También las reglas que conforman la base del conocimiento son diferentes para cada articulación. En el apéndice A se pueden observar las variables y las reglas de la base de conocimiento de las demás articulaciones.

6.3 Configuración del controlador difuso en Fuzzy Logic Toolbox

Una vez que se han definido las variables de entrada y salida, definido las reglas de la base de conocimiento que se utilizarán en la inferencia, así como el método de inferencia y defusificación que se utilizarán, solo queda introducir ésta información en el sistema de inferencia difuso (FIS). La interfaz gráfica con la que cuenta resulta muy amigable y fácil de usar.

El primer paso es definir el método de inferencia que utilizará el FIS y crear las variables que se definieron anteriormente como se muestra en la figura 6.5.

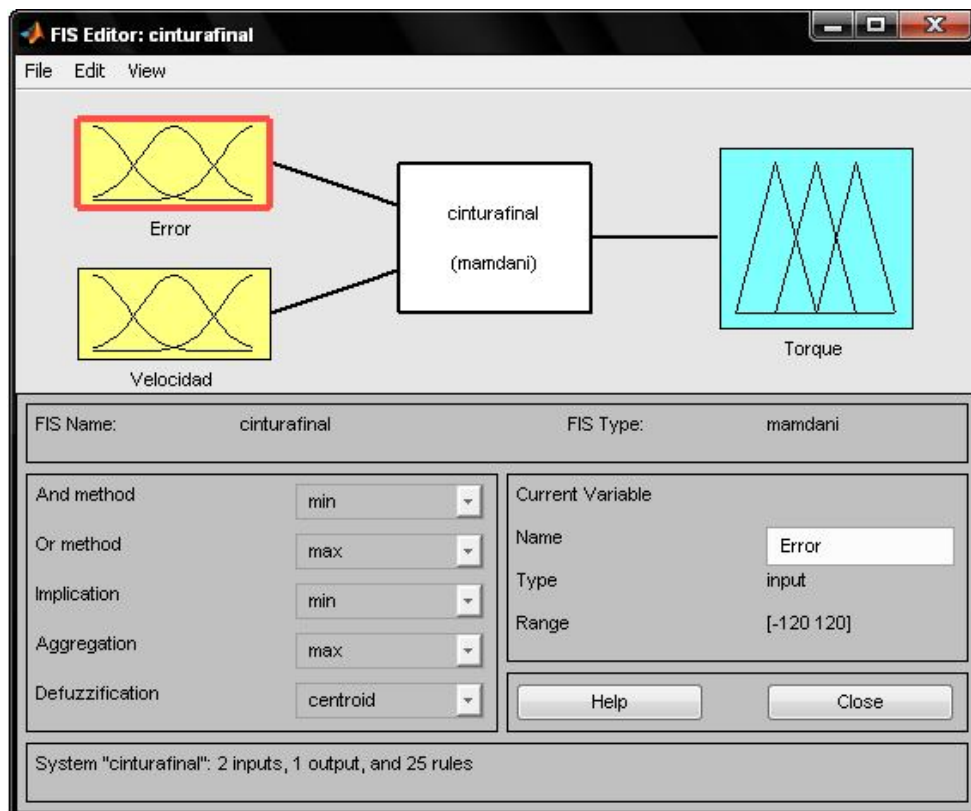


Figura 6.5: Definición de las variables en la ventana FIS Editor

La ventana de edición del FIS muestra las variables de entrada y salida así como el método de inferencia que se eligió, el cual corresponde al método Mamdani. En la misma ventana se selecciona el método de defusificación que se utilizará, este será el método del Centroide.

El segundo paso es definir los conjuntos difusos de las variables de entrada y salida. La figura 6.6 muestra la ventana donde se definen los conjuntos difusos de las variables.

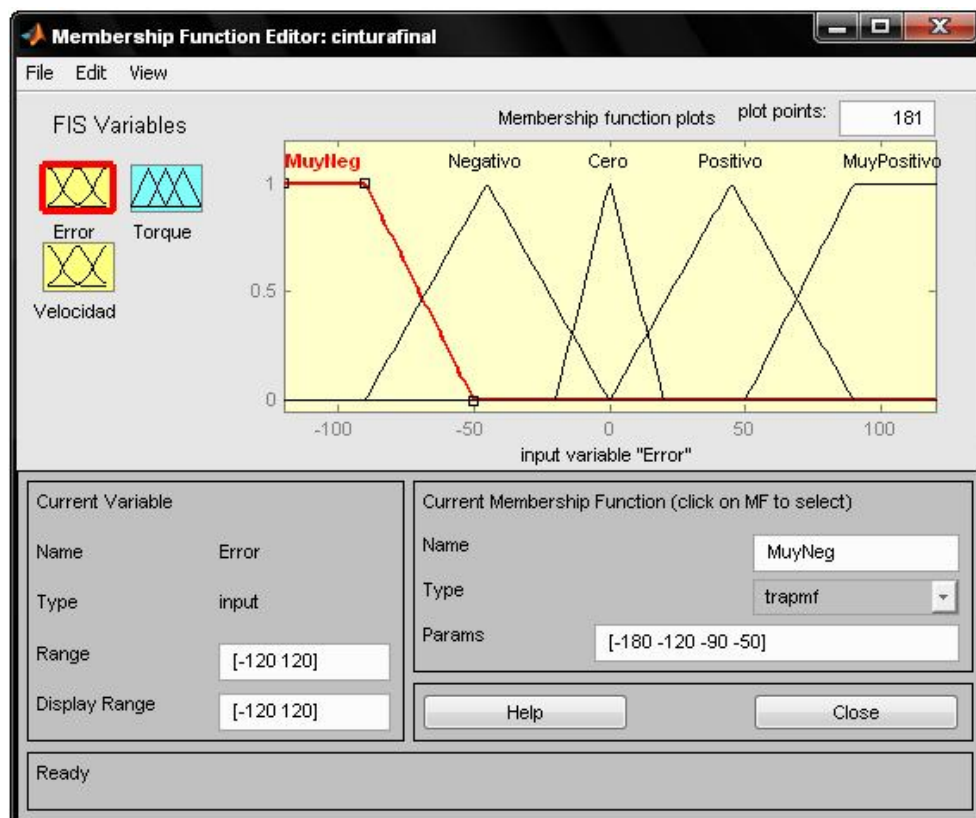


Figura 6.6: Definición de los conjuntos difusos de la variable Error en la ventana FIS Editor

En esta ventana se puede definir la forma de los conjuntos difusos la cual puede ser triangular, trapezoidal, gaussiana, etc. También se puede definir el rango y número de

conjuntos, así como los parámetros que marcarán los límites de los conjuntos, y las etiquetas lingüísticas que se les asignaron.

Por último es necesario definir las reglas que harán funcionar al sistema. La edición de las reglas se realiza en la ventana que se muestra en la figura 6.7.

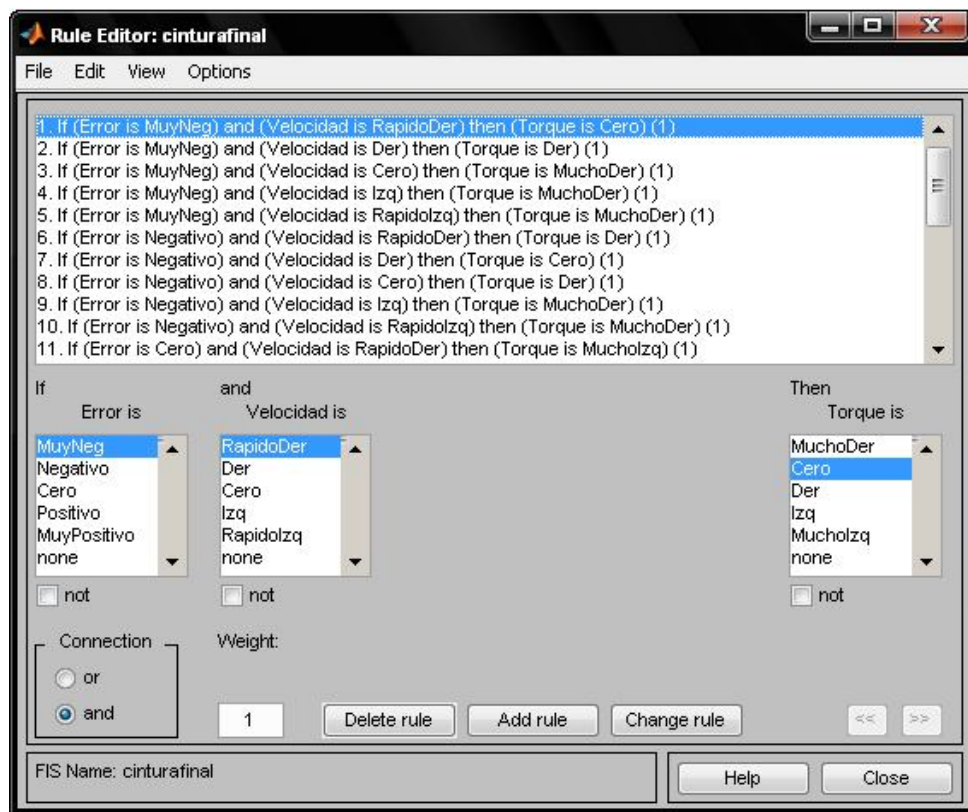


Figura 6.7: Ventana Rule Editor

En la ventana de edición de las reglas se presentan las variables con los conjuntos difusos que se definieron anteriormente. Ayudándose de la tabla de inferencia de la base de conocimiento se selecciona un conjunto de cada variable junto a las condiciones que los relacionan y se van generando una por una las reglas.

Una vez finalizada la configuración del controlador es posible observar las reglas que se generaron en el editor de reglas y como se activan cuando se ingresan valores de entrada. En la figura 6.8 se muestra la ventana de las reglas.

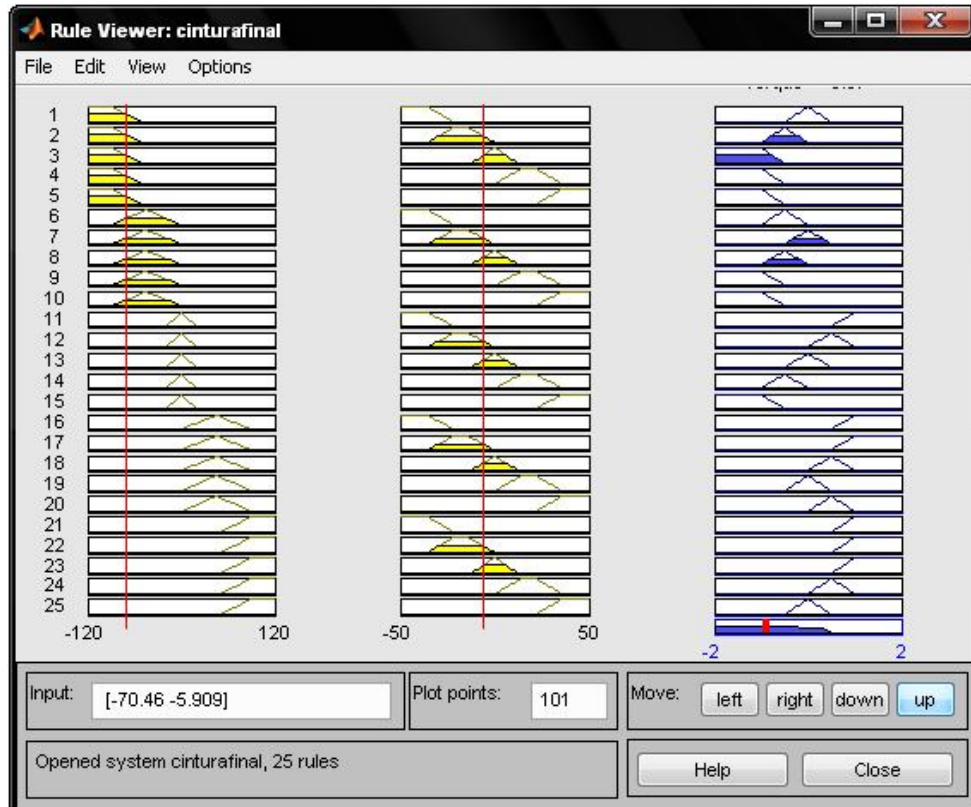


Figura 6.8: Ventana Rule Viewer

En la ventana se observan las variables de entrada así como los conjuntos que las conforman en color amarillo. Las líneas rojas representan los valores de entrada que se ingresan al controlador y se pueden observar los grados de membresía a los que corresponden dentro del conjunto difuso. La variable de salida con sus conjuntos difusos en color azul muestran las reglas que se cumplieron y en la parte inferior de la columna se

puede apreciar el polígono generado por la inferencia. El valor de salida que produce la defusificación es un valor *crisp* que se puede observar en la parte superior de la columna.

Para poder trabajar con el FIS de la articulación de la cintura, en el ambiente de Simulink, es necesario exportarlo al *Workspace* de trabajo de MatLab para que pueda ser utilizado como una variable. El FIS de la articulación de la cintura se exportará con el nombre *cinturafinal* y en bloque *Fuzzy Logic Controller* se ingresará el nombre con el que se exportó el FIS al espacio de trabajo. Los FIS de las demás articulaciones se exportarán con los nombres: *hombrofinal*, *codofinal* y *muñecafinal*. El apéndice B contiene la configuración para las demás articulaciones del Robot UDLAP.

6.4 Trayectorias de Movimientos

Una vez que todas las partes necesarias para efectuar la simulación del Robot UDLAP están listas es necesario especificar las trayectorias que deberá seguir. Primero se generaron trayectorias de prueba, que serán leídas por un generador, a partir de una tabla de datos, la cual contiene la posición que deben seguir las articulaciones en un tiempo dado. Para generar la primera trayectoria se utilizó Excel. En la tabla se especifica que las articulaciones deben moverse de un grado en un grado hasta llegar a cierto punto cada 0.1 segundos. Con esto lo que se genera es el perfil de la trayectoria que debe seguir el Robot UDLAP que se contruyó en SimMechanics.

Una vez que se generó la trayectoria es necesario guardarla en un archivo de tipo *M-file* con la extensión *.mat* para que el bloque generador pueda leerla. Una vez que el archivo *.mat*, con la trayectoria, se debe importar al espacio de trabajo de MatLab para que pueda ser usado como una variable. Para hacerlo, se ingresa el nombre del archivo *.mat* al generador para poder iniciar la simulación.

MatLab cuenta con una herramienta llamada *Robotics Toolbox*, que contiene diferentes funciones útiles para aplicaciones de robótica. *Robotics Toolbox* permite simular la cinemática y dinámica de manipuladores para uniones seriales, como el Robot UDLAP, y facilita la generación de trayectorias que un robot debe seguir.

Para generar una trayectoria se utiliza la función de MatLab *jtraj*, la cual genera puntos en un intervalo de tiempo. A esta función se le deben especificar tanto el punto *q0* que indica el punto inicial de la trayectoria como el punto *q1* que indica el punto final. Posteriormente se especifica el intervalo de tiempo donde se debe completar la trayectoria. Estos valores se envían a la función *jtraj* la cual generará los puntos de la trayectoria.

Las instrucciones para generar la trayectoria se ingresan en la ventana de comandos de MatLab como se muestra a continuación. Las instrucciones que se utilizaron y la trayectoria que se generó se muestran en la figura 6.9 y 6.10 respectivamente, donde la figura 6.9 muestra el código necesario para declarar la trayectoria y el intervalo de tiempo, mientras que la figura 6.11 muestra la gráfica de la trayectoria generada.

```

>> q0=0;
>> q1=60;
>> t=[0:0.1:10];
>> [q] = jtraj(q0, q1, t);
>> plot (t, q);
>> title ('Trayectoria de la Cintura');
>> xlabel ('tiempo');
>> ylabel ('angulo')

```

Figura 6.9: Instrucciones para generar la trayectoria

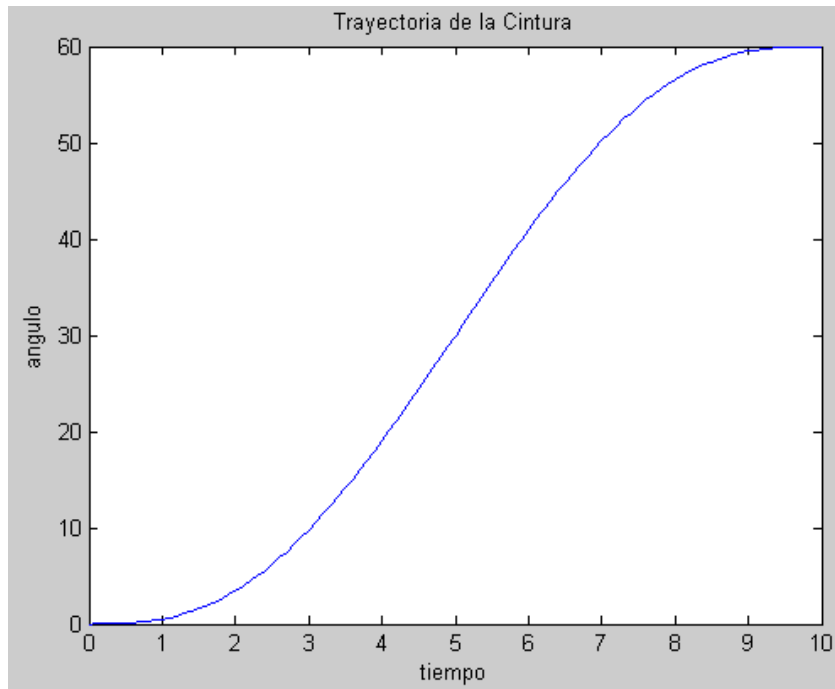


Figura 6.10: Gráfica de la trayectoria generada con Robotic Toolbox

En este capítulo se mostró la forma como se declaran y se configuran los controladores difusos que utilizaran las articulaciones del Robot UDLAP utilizando la herramienta Fuzzy Logic Toolbox. De la misma se mostró como se genera la trayectoria que deben seguir los controladores de las articulaciones. En el siguiente capítulo se presentarán las pruebas que se realizaron así como los resultados obtenidos de las simulaciones.