

Práctica 6

Simulación y Control de Robots

6.1.-Introducción

En los últimos años han aparecido numerosos proyectos de investigación relacionados con disciplinas de simulación virtual. Una correcta simulación dinámica es necesaria como primer paso para la obtención de herramientas capaces de ser utilizadas para el análisis y diseño de robots.

Además de necesitar una metodología numéricamente correcta, es de gran importancia el disponer de unas herramientas de visualización que permitan comprobar los resultados obtenidos, de manera que la detección de errores sea lo más intuitiva posible. Siguiendo esta idea, en esta práctica se va a presentar una serie de ejemplos en los que se utilizan los modelos dinámicos presentados anteriormente con la finalidad de poder realizar un correcto control del robot estudiado.

En esta práctica se detalla la simulación de un robot efectuando una planificación del extremo del robot en línea recta entre dos posiciones del espacio cartesiano. Se va a realizar el ejemplo con el robot de 4 gdl, dejando como práctica para el alumno la realización de los mismos ejercicios para el robot de 6 gdl.

6.2.-Sintonizado de los motores.

Una vez que se han seleccionado los servoaccionamientos para cada una de las articulaciones del robot (práctica 4), el paso siguiente es realizar un sintonizado adecuado de los mismos con el fin de que satisfagan un determinado comportamiento. Para realizar la sintonización en primer lugar se debe especificar cuál es la estructura de control a utilizar, detallando los controladores a utilizar. A continuación, mediante una determinada técnica de control, se realizará el sintonizado de los motores ajustando los valores de los controladores especificados. Para efectuar el sintonizado de los motores haremos uso del modelo dinámico directo del robot.

6.2.1. Estructuras de control de robots.

Hay dos estructuras de control típicas utilizadas en los robots: control acoplado y control desacoplado.

- Control desacoplado

En el control desacoplado se considera que las articulaciones del robot están desacopladas, de modo que un par en un determinado actuador únicamente tendrá efecto sobre el movimiento de la articulación correspondiente. De esta forma existirá un controlador para cada articulación. La ventaja del control desacoplado radica en que el diseño del regulador más adecuado para cada articulación puede hacerse utilizando las técnicas más frecuentes de diseño.

- Control acoplado

En ocasiones, la suposición de que el robot es una serie de eslabones dinámicamente desacoplados, de forma que el movimiento de uno de ellos no afecta a los demás, no es siempre aceptable. Las técnicas de control acoplado consideran el modelo dinámico real del robot, haciendo uso del conocimiento del mismo para tratar de desacoplar el sistema.

Adoptaremos una estructura de control desacoplado, de manera que en cada articulación existe un regulador PID. Durante el sintonizado de una articulación, los actuadores del resto de articulaciones permanecen parados. En la figura 6.1 se muestra la estructura de control desacoplado típica de un robot de tres grados de libertad.

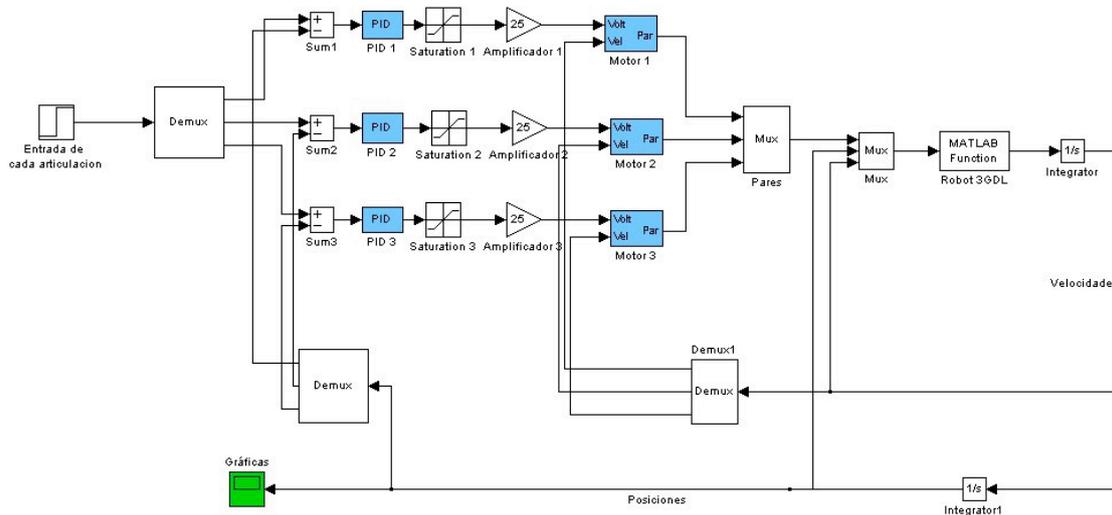


Figura 6.1. Estructura de control desacoplado de un robot de 3 grados de libertad. En este caso, la entrada a cada articulación es un escalón.

6.2.2. Técnicas de sintonizado.

El sintonizado de un robot consiste en ajustar cada uno de los controladores para que el robot satisfaga un comportamiento especificado. Para ello, en un esquema de control desacoplado, se deberá sintonizar de manera independiente cada articulación del robot. El sintonizado de una articulación consiste en ajustar los parámetros del regulador que actúa sobre ella para que su respuesta cumpla unas determinadas especificaciones.

El controlador que utilizaremos en cada una de las articulaciones será un regulador PID. La función de transferencia de un regulador PID es la siguiente:

$$G_{PID}(s) = k_p \left(1 + \frac{1}{T_i s} + T_d s\right) = k_p \left(1 + \frac{k_i}{s} + k_d s\right) = P + \frac{I}{s} + Ds \quad (6.1)$$

Para realizar el ajuste de los parámetros de un regulador PID existen múltiples métodos. A continuación se van a estudiar dos métodos muy utilizados: el *manual* y el de *Ziegler-Nichols*.

- Método manual (prueba y error)

Este método propone una manera de ir modificando los parámetros del regulador para conseguir que la respuesta de la articulación cumple las especificaciones impuestas.

- 1- En primer lugar se aumenta la constante proporcional P hasta obtener el tiempo de cruce deseado (el menor posible sin tener una sobreoscilación exagerada).
- 2- A continuación se procede a aumentar la constante derivativa D para disminuir la sobreoscilación (esto incrementará el tiempo de cruce).
- 3- Por último se aumenta la constante integral I para eliminar el error que exista en régimen permanente (esto incrementará la sobreoscilación y disminuirá el tiempo de respuesta).

- Método de Ziegler-Nichols

Es un método experimental en el que es necesario que la respuesta de la articulación en bucle cerrado ante escalón sea periódica. Para ello solamente se introduce una ganancia proporcional que se va aumentando hasta conseguir la respuesta periódica. Si no se consigue que la respuesta sea periódica, no se puede aplicar este método.

Sea k_{cr} la ganancia proporcional crítica a la cual la respuesta del sistema se vuelve periódica y P_{cr} el período de la respuesta. En base a estos valores, el método de Ziegler-Nichols proporciona los valores del regulador que aseguran una respuesta aceptable. Los valores que proporciona este método se muestran en la tabla 6.1.

$$\begin{aligned} k_p &= 0.6 k_{cr} \\ T_i &= 0.5 P_{cr} \\ T_d &= 0.125 P_{cr} \end{aligned}$$

Tabla 6.1. Valores de los parámetros de un regulador PID proporcionados por el método de Ziegler-Nichols.

Hay que destacar que es posible que la respuesta obtenida aplicando los valores obtenidos por este método no satisfaga las especificaciones impuestas, por lo que será necesario modificar estos valores mediante prueba y error. La utilidad de este método es que nos proporciona un punto de partida para realizar el ajuste de los parámetros.

Ejemplo 6.1

Sintonizado del robot de cuatro grados de libertad.

Vamos a suponer que el esquema de control de este robot es desacoplado, por lo que cada articulación dispondrá de un regulador PID y durante el proceso de sintonizado de una articulación concreta, los actuadores del resto de articulaciones permanecerán apagados.

Para efectuar el sintonizado del robot hay que situarlo en la peor posición, introducir a la articulación que se desea ajustar un escalón y comprobar que la respuesta de la articulación cumple las especificaciones deseadas. La peor posición en la que se puede encontrar el robot viene dada por:

$$q = (0 \quad 1 \quad 1.2 \quad 0)$$

Para un funcionamiento satisfactorio del robot exigiremos que la respuesta de cada articulación cumpla las siguientes especificaciones:

- Tiempo de cruce: < 100 mseg
- Sobreoscilación: < 20 %
- Tiempo de establecimiento: < 200 mseg

La entrada a las articulaciones rotacionales (articulación 1 y 4) para realizar el sintonizado será un escalón de 0.08 rad. En las articulaciones prismáticas se introducirá un escalón de 100mm (0.1m) para sintonizar los reguladores. El escalón será positivo o negativo en función de la articulación que se desee sintonizar. El escalón siempre debe elegirse de modo que precise los máximos requerimientos de par. En el caso de las articulaciones 1 y 4 es independiente considerar el escalón positivo o negativo, ya que a efectos de requerimientos de par, no hay diferencia en girar la articulación en un sentido o en otro. En el caso de las articulaciones 2 y 3 el máximo requerimiento de par se produciría cuando la articulación se desplace en el sentido positivo del eje z. Sin embargo, puesto que la peor posición del robot se produce cuando la articulación 2 y 3 poseen su extensión máxima, no es posible introducir el escalón en el sentido positivo del eje z. Por esta razón, en las articulaciones 2 y 3 se ha introducido un escalón negativo.

Se ha desarrollado en Simulink® un esquema de control en bucle cerrado, para cada una de las articulaciones, que permite ajustar un regulador PID y comprobar si la respuesta del sistema ante una entrada escalón satisface las especificaciones impuestas.

Motor 1

Para realizar el sintonizado del motor 1 se ha desarrollado en Simulink® el modelo que se encuentra en el fichero *sintonizar1d4.mdl*. Este fichero contiene el esquema de control en bucle cerrado de la primera articulación que permite ajustar su regulador PID para que la respuesta satisfaga las especificaciones. Si se ejecuta desde el entorno de Matlab “*sintonizar1d4*” aparecerá en pantalla la aplicación para el sintonizado de la primera articulación, que se muestra en la figura 6.2.

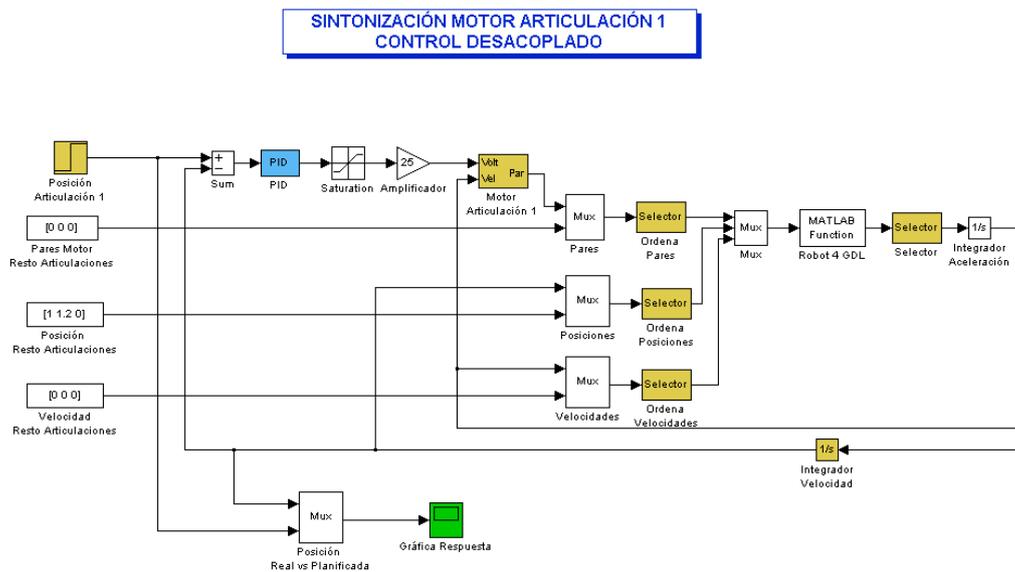


Figura 6.2. Herramienta desarrollada en Simulink para realizar el sintonizado de la primera articulación.

Para obtener la aceleración de cada articulación se ha desarrollado en Matlab® la función *dindir4gdl.m*; que calcula la aceleración de cada articulación cuando el robot transporta una determinada carga en función de la posición y velocidad actual de la articulación y de su fuerza/par de entrada. Esta función hace uso de la función *walkerorin4.m*, explicada en la práctica 3. El código fuente de *dindir4gdl.m* se muestra a continuación:

```
% DINDIR4GDL Dinámica directa de un robot de 4GDL.
% QPP = DINDIR4GDL(ENTRADA) calcula el vector 4x1 de aceleración de
% cada articulación utilizando el tercer método de Walker y Orin.
% ENTRADA(1:4) representa el par de entrada a cada articulación.
% ENTRADA(5:8) es la posición de cada articulación. ENTRADA(9:12)
% es la velocidad de cada articulación.
%
% Ver también WALKERORIN4.
```

```
function qpp = dindir4gdl(entrada)
```

```
tau = entrada(1:4); % Par de entrada a cada articulación
q = entrada(5:8); % Posición de cada articulación
qp = entrada(9:12); % Velocidad de cada articulación
```

```
% Parámetros de la carga
masaext = 10;
ineraciaext = [0.0167 0 0;0 0.0167 0;0 0 0.0167];
```

```
% Se convierten los pares de la articulación 2 y 3 en fuerzas.
n = 0.85; % Eficiencia husillo
p = 0.025; % Paso del husillo (mm)
tau(2:3) = 2*pi*tau(2:3)/p;
```

```
% Se calcula la aceleración utilizando el método de Walker y Orin.
```

En la función *dindir4gdl.m* el par proporcionado por los motores 2 y 3 se convierte a fuerza, ya que las articulaciones 2 y 3 son prismáticas, utilizando la siguiente ecuación:

$$F = \frac{2\pi n\tau}{p}$$

donde τ es el par (Nm), p es el paso del husillo (m) y n es la eficiencia del husillo.

Las entradas al esquema de control son:

- El escalón de posición que se aplica a la primera articulación. El escalón es de 0.08 rad y, puesto que para esta articulación es independiente que sea positivo o negativo, se ha considerado positivo. El valor de la posición de esta primera articulación en la peor configuración del robot es 0 rad, por lo tanto el escalón será de 0 rad a 0.08 rad.
- Los pares del resto de motores. Puesto que los actuadores de las articulaciones que no están siendo sintonizadas son apagadas, los pares del resto de motores son nulos.
- La posición del resto de articulaciones en su peor configuración.
- La velocidad de las restantes articulaciones. La velocidad de las restantes articulaciones siempre será cero, puesto que sus actuadores están apagados.

El bloque *PID* representa a un regulador convencional PID con la siguiente función de transferencia:

$$G_{PID}(s) = P + \frac{I}{s} + Ds$$

Asignando los valores adecuados a P, I y D se conseguirá que la respuesta del sistema sea la deseada.

En el bloque correspondiente al motor de la articulación 1 se deben colocar los datos de los parámetros del motor que se seleccionó en la practica 4 para esa articulación. Estos datos se muestran en la tabla 6.2.

Articulación	Nombre Motor	R(Ω)	L(mH)	K _T (Nm/A)	K _V (V/rad/s)	I _{máx} (A)
1	DA23GGB	0.8	0.93	0.058	0.058	18.5

Tabla 6.2. Valores de los parámetros del motor utilizado para la primera articulación.

Los bloques *Ordena Pares*, *Ordena Posiciones* y *Ordena Velocidades*, simplemente indican el orden del vector de pares, posiciones y velocidades del robot. En este caso los vectores poseen el orden adecuado, ya que se está sintonizando la primera articulación, de manera que el orden que indicaremos en los tres selectores será [1 2 3 4]. Sin embargo, veremos que esto no es así para el sintonizado del resto de articulaciones.

Puesto que estamos ajustando la primera articulación, la única aceleración que nos interesa es la primera, por lo que a continuación del bloque *Robot 4 GDL*, que corresponde a la dinámica directa del robot, utilizamos un selector para elegir la aceleración de la primera articulación. A continuación, se emplea un bloque integrador para obtener la velocidad de la articulación. En este bloque especificamos que la condición inicial es cero, ya que la velocidad inicial de esta articulación es de 0 rad/s. Para calcular la posición de la articulación se integra la velocidad utilizando otro bloque integrador. En este integrador la condición inicial será cero, puesto que la posición inicial de la articulación es 0 rad.

Antes de realizar la simulación del modelo para determinar si la respuesta del sistema es la adecuada, es necesario configurar los parámetros de la simulación con los valores que se muestran en la figura 6.3.

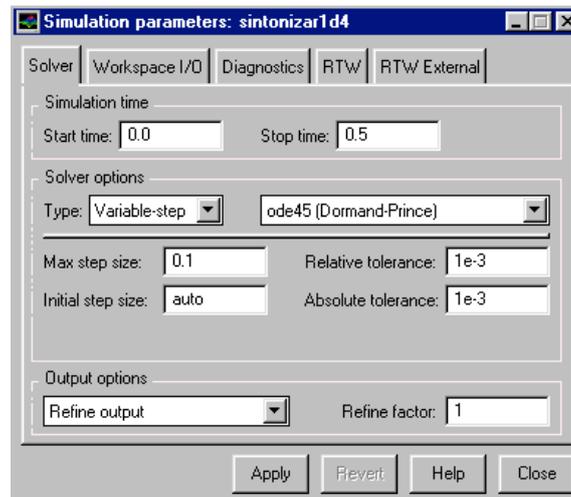


Figura 6.3. Configuración de los parámetros de la simulación.

Cuando finalice la simulación pulsando dos veces con el ratón sobre el bloque *Gráfica Respuesta* aparece la posición de la articulación y el escalón introducido.

En el caso de esta articulación, no ha sido posible realizar el sintonizado que cumpla las especificaciones establecidas. Esto se debe a que el motor elegido responde más lento de lo esperado. Para solucionar este problema se va a seleccionar un motor que proporciona un par mayor. Las características de este motor se muestran en la tabla 6.3.

Articulación	Nombre Motor	R(Ω)	L(mH)	K _T (Nm/A)	K _V (V/rad/s)	I _{máx} (A)
1	DA23JBB	0.39	0.65	0.054	0.054	25.3

Tabla 6.3. Valores de los parámetros del motor utilizado para la primera articulación.

Con el motor seleccionado ya es posible realizar el sintonizado de forma que se satisfagan las especificaciones. Un posible regulador que proporciona que la respuesta de la primera articulación se ajuste a las especificaciones establecidas es el siguiente:

$$G_{PID}(s) = 90 + 1.6s$$

En la figura 6.4 se muestra como varía la posición de la articulación 1 ante el escalón introducido. Como se puede apreciar en la gráfica, con el regulador escogido se satisfacen los requerimientos impuestos.

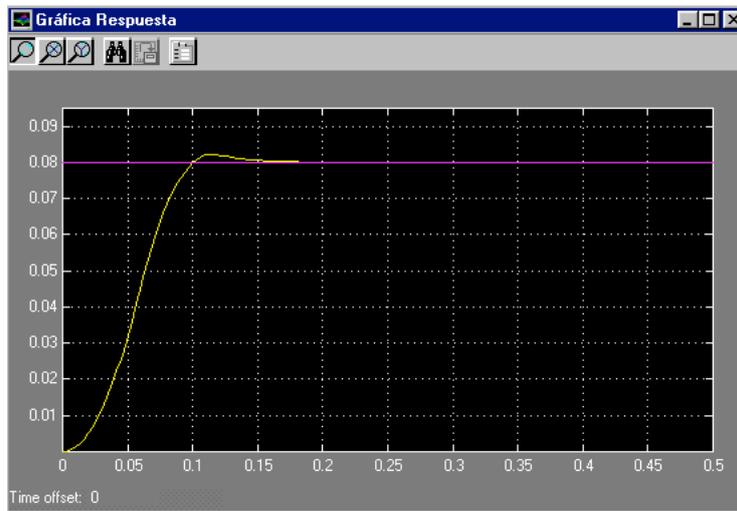


Figura 6.4. Posición de la articulación 1 ante entrada en escalón.

Motor 2

Para realizar el sintonizado del motor 2 se utiliza el modelo implementado en Simulink que se encuentra en el fichero *sintonizar2d4.mdl*. Este modelo contiene el esquema de control en bucle cerrado de la segunda articulación que permite ajustar su regulador PID para que la respuesta satisfaga las especificaciones.

Las entradas al esquema de control son:

- El escalón de posición que se aplica a la segunda articulación. Este escalón es de 100 mm. El valor de la posición de esta articulación en la peor configuración del robot es 1 m. El mayor requerimiento de par se produce cuando el escalón es positivo. Sin embargo como la articulación posee su extensión máxima, el escalón no es realizable físicamente, por lo que se introducirá un escalón negativo. Por lo tanto el escalón será de 1 m a 0.9 m.
- Los pares del resto de motores. Al igual que en el caso anterior, los pares del resto de motores son nulos.
- La posición del resto de articulaciones en su peor configuración.
- La velocidad de las restantes articulaciones. Como en el caso anterior, la velocidad de las restantes articulaciones es cero.

En el bloque correspondiente al motor de la articulación 2 se deben colocar los datos de los parámetros del motor que se seleccionó en la práctica 4 para esa articulación.

Articulación	Nombre Motor	R(Ω)	L(mH)	K_T (Nm/A)	K_V (V/rad/s)	$I_{m\acute{a}x}$ (A)
2	DA34HBB	1.6	1.56	0.176	0.176	23.7

En los bloques *Ordena Pares*, *Ordena Posiciones* y *Ordena Velocidades* se indica el orden del vector de pares, posiciones y velocidades del robot. El orden que indicaremos en los tres selectores será [2 1 3 4].

Puesto que se está ajustando la segunda articulación, a continuación del bloque *Robot 4 GDL*, se indicará en el selector que se elige la aceleración de la segunda articulación. En el bloque *Integrador Aceleración* especificamos que la condición inicial es cero, ya que la velocidad inicial de esta articulación es de 0 m/s. En el bloque *Integrador Velocidad* se indica que la condición inicial es 1, puesto que la posición inicial de la articulación es 1 m.

Un posible regulador que proporciona que la respuesta de la segunda articulación se ajuste a las especificaciones establecidas es el siguiente:

$$G_{PID}(s) = 300 + 6s$$

Si se realiza la simulación del modelo se obtiene la gráfica que muestra como varía la posición de la articulación 2 ante el escalón introducido. Como se puede apreciar en la gráfica, al igual que en el caso anterior, con el regulador escogido se satisfacen los requerimientos impuestos. Ver figura 6.5.

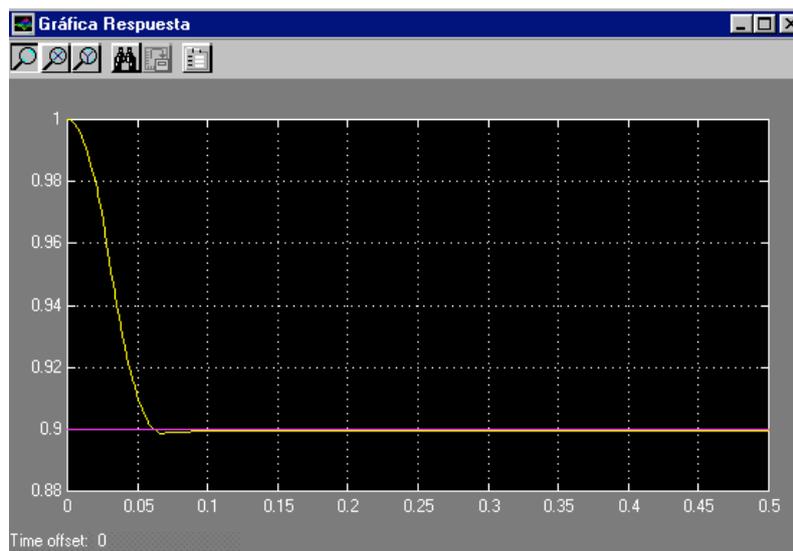


Figura 6.5. Posición de la articulación 2 ante entrada en escalón.

Motor 3

Para realizar el sintonizado del motor 3 se utiliza el modelo implementado en Simulink que se encuentra en el fichero *sintonizar3d4.mdl*. Este modelo contiene el esquema de control en bucle cerrado de la tercera articulación que permite ajustar su regulador PID para que la respuesta satisfaga las especificaciones.

Las entradas al esquema de control son:

- El escalón de posición que se aplica a la tercera articulación. Este escalón es de 100 mm. El valor de la posición de esta articulación en la peor configuración del robot es

1.2 m. Al igual que en el caso anterior, el mayor requerimiento de par se produce cuando el escalón es positivo. Sin embargo como la articulación posee su extensión máxima, el escalón no es realizable físicamente, por lo que se introducirá un escalón negativo. Por lo tanto el escalón será de 1.2 m a 1.1 m.

- Los pares del resto de motores. Al igual que en los casos anteriores, los pares del resto de motores son nulos.
- La posición del resto de articulaciones en su peor configuración.
- La velocidad de las restantes articulaciones. Como en los casos anteriores, la velocidad de las restantes articulaciones es cero.

En el bloque correspondiente al motor de la articulación 3 se deben colocar los datos de los parámetros del motor que se seleccionó en la práctica 4 para esa articulación.

Articulación	Nombre Motor	R(Ω)	L(mH)	K _T (Nm/A)	K _V (V/rad/s)	I _{máx} (A)
3	DA23GBB	0.8	0.93	0.058	0.058	18.5

En los bloques *Ordena Pares*, *Ordena Posiciones* y *Ordena Velocidades* se indica el orden del vector de pares, posiciones y velocidades del robot. El orden que indicaremos en los tres selectores será [2 3 1 4].

Puesto que se está ajustando la tercera articulación, a continuación del bloque *Robot 4 GDL*, se indicará en el selector que se elige la aceleración de la tercera articulación. En el bloque *Integrador Aceleración* especificamos que la condición inicial es cero, ya que la velocidad inicial de esta articulación es de 0 m/s. En el bloque *Integrador Velocidad* se indica que la condición inicial es 1.2, puesto que la posición inicial de la articulación es 1.2 m.

Un posible regulador que proporciona que la respuesta de la segunda articulación se ajuste a las especificaciones establecidas es el siguiente:

$$G_{PID}(s) = 200 + 5.7s$$

Si se realiza la simulación del modelo se obtiene la gráfica que muestra como varía la posición de la articulación 3 ante el escalón introducido. Como se puede apreciar en la gráfica, al igual que en el caso anterior, con el regulador escogido se satisfacen los requerimientos impuestos. Ver figura 6.6.

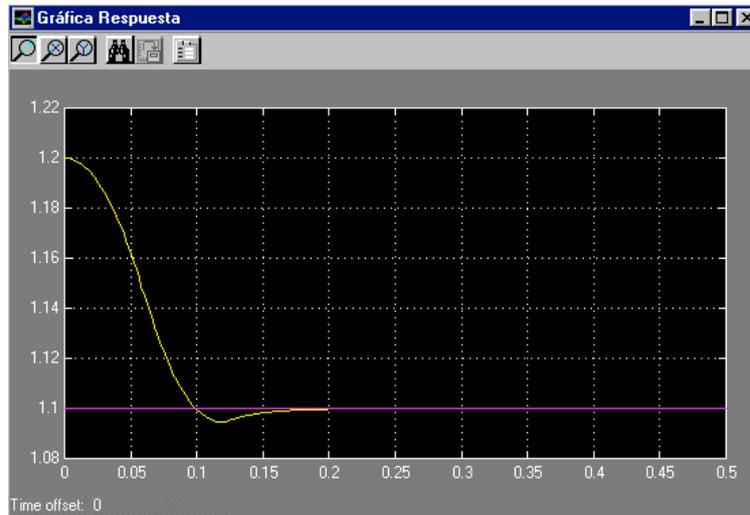


Figura 6.6. Posición de la articulación 3 ante entrada en escalón.

Motor 4

Para realizar el sintonizado del motor 4 se utiliza el modelo que se encuentra en el fichero *sintonizar4d4.mdl*. Este modelo contiene el esquema de control en bucle cerrado de la cuarta articulación que permite ajustar su regulador PID para que la respuesta cumpla las especificaciones.

Las entradas al esquema de control son:

- El escalón de posición que se aplica a la cuarta articulación. El valor de la posición de esta articulación en la peor configuración del robot es 0 rad. Para esta articulación es independiente que el escalón sea positivo o negativo. Consideraremos que el escalón es de 0 rad a 0.08 rad.
- Los pares del resto de motores. Al igual que en los casos anteriores, los pares del resto de motores son nulos.
- La posición del resto de articulaciones en su peor configuración.
- La velocidad de las restantes articulaciones. Como en los casos anteriores, la velocidad de las restantes articulaciones es cero.

En el bloque correspondiente al motor de la articulación 4 se deben colocar los datos de los parámetros del motor que se seleccionó en la práctica 4 para esa articulación.

Articulación	Nombre Motor	R(Ω)	L(mH)	K_T (Nm/A)	K_V (V/rad/s)	$I_{m\acute{a}x}$ (A)
4	DB17CDB	6.9	1.28	0.035	0.035	3.6

En los bloques *Ordena Pares*, *Ordena Posiciones* y *Ordena Velocidades* se indica el orden del vector de pares, posiciones y velocidades del robot. El orden que indicaremos en los tres selectores será [2 3 4 1].

Puesto que se está ajustando la cuarta articulación, a continuación del bloque *Robot 4 GDL*, se indicará en el selector que se elige la aceleración de la cuarta articulación. En el bloque *Integrador Aceleración* especificamos que la condición inicial es cero, ya que la velocidad inicial de esta articulación es de 0 rad/s. En el bloque *Integrador Velocidad*

se indica que la condición inicial es 0, puesto que la posición inicial de la articulación es 0 rad.

Un posible regulador que proporciona que la respuesta de la cuarta articulación se ajuste a las especificaciones establecidas es el siguiente:

$$G_{PID}(s) = 100$$

Si se realiza la simulación del modelo se obtiene la gráfica que muestra como varía la posición de la articulación 4 ante el escalón introducido. Como se puede apreciar en la gráfica, al igual que en los casos anteriores, con el regulador escogido se satisfacen los requerimientos impuestos. Ver figura 6.7.

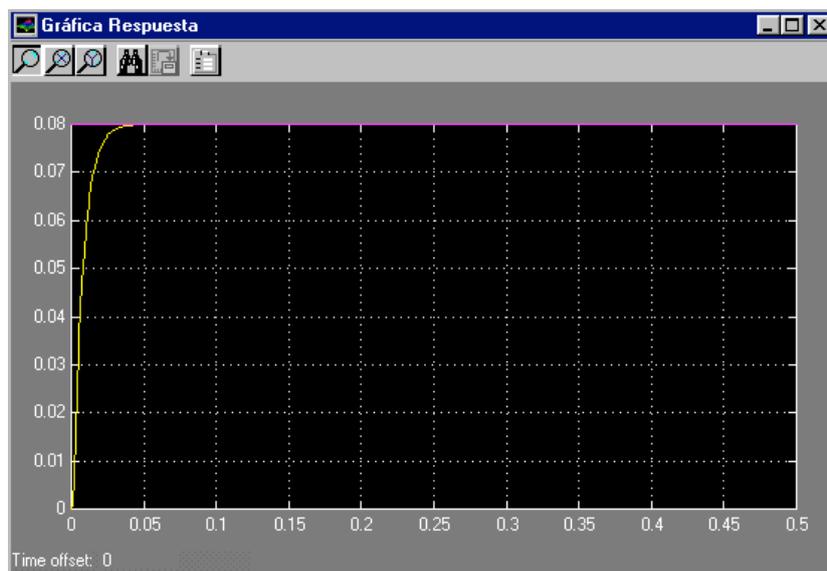


Figura 6.7. Posición de la articulación 4 ante entrada en escalón.

6.3.- PRACTICA. Simulación de robots.

Una vez que se han seleccionado los servoaccionamientos del robot y se ha efectuado el sintonizado, ya puede realizarse una simulación completa del funcionamiento del robot.

Ejemplo 6.2

Simulación del robot de 4 GDL.

Se ha realizado la simulación del robot de 4 GDL para que realice una planificación de su extremo en línea recta entre dos puntos del espacio cartesiano. La planificación realizada únicamente es en posición, de manera que el robot parte de la posición inicial con velocidad nula. Para realizar la planificación se ha dividido la trayectoria en un número determinado de intervalos, de forma que en cada iteración de la simulación, se proporciona la posición que debe alcanzar el extremo del robot. Para realizar la

simulación se ha desarrollado en Simulink el modelo que se encuentra en el fichero *simulador4.mdl*. Ver figura 6.8.

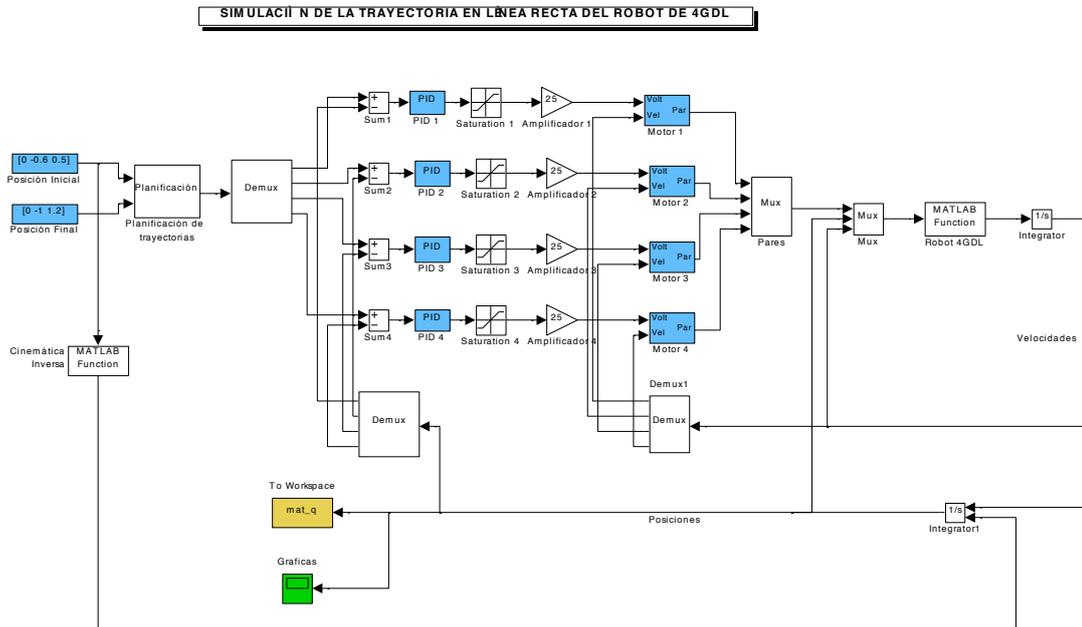


Figura 6.8. Simulador del robot de 4 GDL.

Como puede apreciarse en la figura 6.8, para realizar la simulación se utiliza un esquema de control desacoplado, de forma que cada articulación posee un regulador PID. Los valores de los motores del robot que se utilizan para realizar la simulación son aquellos que se emplearon en el proceso de sintonizado mostrado en el ejemplo 6.5. Estos valores se recogen en la tabla 6.4. Así mismo los valores de los reguladores PID son los que se obtuvieron en el ejemplo 6.1, que se muestran en la tabla 6.5.

Articulación	Nombre Motor	R(Ω)	L(mH)	K_T (Nm/A)	K_V (V/rad/s)	$I_{m\acute{a}x}$ (A)
1	DA23JBB	0.39	0.65	0.054	0.054	25.3
2	DA34HBB	1.6	1.56	0.176	0.176	23.7
3	DA23GBB	0.8	0.93	0.058	0.058	18.5
4	DB17CDB	6.9	1.28	0.035	0.035	3.6

Tabla 6.4. Motores utilizados en el simulador del robot de 4 GDL.

Articulación	P	I	D
1	90	0	1.6
2	300	0	6
3	200	0	5.7
4	100	0	0

Tabla 6.5. Valores de los reguladores del robot de 4 GDL.

El bloque *Robot 4 GDL* representa la dinámica directa del robot de 4 GDL. Como se mostró en el ejemplo 6.3, para realizar la dinámica directa se ha desarrollado en Matlab la función *dindir4gdl.m*. Esta función calcula la aceleración de cada articulación cuando el robot transporta una determinada carga en función de la posición y velocidad actual de la articulación y de su fuerza/par de entrada.

El simulador efectúa una planificación de la trayectoria del extremo del robot en línea recta entre la posición cartesiana inicial y final especificadas. Para realizar la planificación de trayectorias se ha desarrollado el subsistema *Planificación*. Este subsistema se muestra en la figura 6.9.

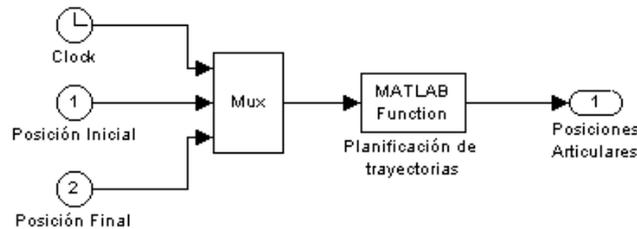


Figura 6.9. Subsistema de planificación de trayectorias.

El subsistema de planificación es el encargado de suministrar al esquema de control de cada articulación la posición articular de cada articulación en cada instante de simulación. Para obtener las posiciones articulares se utiliza la función implementada en Matlab *planifica4.m*. El código fuente de esta función es el siguiente:

```
% PLANIFICA4 Planificación de trayectorias en línea recta de un robot de 4GDL
% Q = PLANIFICA4(ENTRADA) devuelve las coordenadas articulares
% correspondientes al instante actual de simulación en una planificación
% de trayectoria en línea recta entre dos puntos cartesianos. ENTRADA(1)
% representa el instante de tiempo actual (seg). ENTRADA(2:4) es la
% posición cartesiana inicial de la trayectoria. ENTRADA(5:7) es la
% posición cartesiana final de la trayectoria.
%
% Ver también CININV4GDL.

function q = planifica4(entrada)

t = entrada(1); % Instante actual de simulación (seg)
p1 = entrada(2:4); % Posición cartesiana inicial
p2 = entrada(5:7); % Posición cartesiana final

ts = 1; % Tiempo de simulación (seg)
intervalo = 1e-4; % Intervalo de integración (seg)

% Número de segmentos de la trayectoria
nseg = ts/intervalo;

% Cálculo del vector unitario
u = p2-p1;
mu = sqrt(u(1)^2+u(2)^2+u(3)^2);
u = (1/mu)*u;

% Cálculo de la distancia entre puntos
d = mu/nseg;

% Número de punto actual en la trayectoria (el inicial es 0)
i = t*nseg;

% Cálculo de la posición cartesiana actual de la mano del manipulador
ps = p1+(i*d)*u;

% Cálculo de las coordenadas articulares
q = cininv4gdl(ps);
```

Esta función proporciona las coordenadas articulares del robot del instante actual de simulación a partir del instante de tiempo y de la posición inicial y final expresadas en coordenadas cartesianas. La función realiza una planificación de trayectorias en línea recta entre la posición inicial y la final. Para ello divide la trayectoria en un número de segmentos que viene impuesto por los parámetros con los que se desea simular en Simulink:

$$n^{\circ} \text{segmentos} = \frac{\text{tiempo de simulación}}{\text{intervalo de integración}}$$

En función del instante de simulación calcula qué posición cartesiana debe tener el extremo del robot y, realizando la cinemática inversa del robot mediante la función *cininv4gdl.m*, calcula las variables articulares del robot. El código fuente de la función *cininv4gdl.m* es el siguiente:

```
% CININV4GDL Cinemática inversa de un robot de 4GDL.
% Q = CININV4GDL(ENTRADA) devuelve el vector 4x1 de coordenadas
% articulares que contiene la solución cinemática inversa.
% ENTRADA es un vector 3x1 que representa la posición expresada
% en coordenadas cartesianas.
%
% Ver también INVERSEKINEMATIC4.

function q = cininv4gdl(entrada)

% Posición cartesiana
p = entrada;

% Orientación deseada en el extremo del robot
n = [1 0 0]';
s = [0 0 -1]';
a = [0 1 0]';

% Matriz con la orientación y posición deseadas en el extremo del robot
T = [n s a entrada];

% Se calculan las coordenadas articulares
q = inversekinematic4(T);
```

La función *cininv4gdl.m* utiliza la función *inversekinematic4.m* obtenida en la práctica 2 para realizar la cinemática inversa del robot de 4 GDL. Como se puede apreciar en el código fuente presentado, se especifica la orientación deseada en el extremo del robot. Por lo tanto la orientación del extremo del robot permanecerá fija durante toda la planificación. Para que un robot pueda posicionarse en cualquier punto con cualquier orientación necesita un mínimo de 6 GDL. Este robot es de 4 GDL, por lo que existen trayectorias que no pueden realizarse si la orientación del extremo es fija. Para realizar cualquier trayectoria correctamente sería necesario modificar en cada momento la orientación del extremo. Es decir, habría que realizar una planificación de la posición del extremo, así como de su orientación. Sin embargo, como la única finalidad en este apartado es simular el comportamiento del robot de 4 GDL, la orientación del extremo permanecerá fija, de forma que se realizarán trayectorias en las que el extremo no cambie su orientación.

El bloque *Cinemática Inversa* calcula la cinemática inversa de la posición cartesiana inicial de la trayectoria mediante la función *cininv4gdl.m*. Este cálculo es necesario, ya que el integrador del modelo que obtiene las posiciones articulares del robot a partir de las velocidades, necesita como condición inicial las variables articulares correspondientes a la posición cartesiana inicial del extremo del robot.

Con la finalidad de poder efectuar una animación de la trayectoria del extremo del robot desde Matlab se guarda la posición de cada articulación en la matriz *mat_q*. Esta matriz posee una fila por cada instante de simulación y una columna por cada variable articular.

Para realizar la simulación de la trayectoria en primer lugar es necesario configurar los parámetros de la simulación. Los valores seleccionados de los parámetros son los que se muestran en la figura 6.10. Es muy importante destacar que el tipo de paso seleccionado del *Solver* debe ser fijo y además el valor del paso debe coincidir con el intervalo de integración especificado en la función *planifica4.m*. Así mismo, el tiempo de simulación deber ser idéntico al considerado en dicha función.

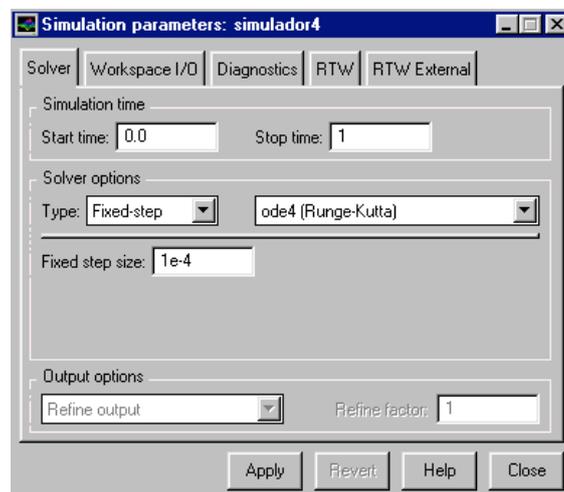


Figura 6.10. Parámetros para realizar la simulación del robot de 4 GDL.

Una vez que se han establecido los parámetros de simulación de Simulink®, ya puede realizarse la simulación del modelo. Una vez finalizada la simulación, podremos, desde el entorno de Matlab®, realizar una animación para comprobar visualmente que el robot realiza la trayectoria deseada. Para efectuar la animación se utilizará la función desarrollada en Matlab® *animacion4.m*. El código de esta función se muestra a continuación:

```
% ANIMACION4 Animación de la trayectoria de un robot de 4 GDL
% ANIMACION(MAT_Q) realiza la animación de la trayectoria, expresada
% en la matriz MAT_Q, de un brazo robot de 4 GDL. MAT_Q contiene 4 filas
% y una columna para cada disposición del robot.
%
% Ver también DRAWROBOT3D4.

function animacion4(mat_q)

% Parámetros Denavit-Hartenberg del robot. Los parámetros correspondientes
% a variables articulares aparecen con valor 0
teta = [0 0 0 0];
```

```

d = [0.4 0 0 0.2];
a = [0 -0.1 0 0 ];
alfa = [0 -pi/2 0 0 ];

% Vector de posicion (x, y, z) del sistema de coordenadas de referencia
x0 = 0; y0 = 0; z0 = 0;

% Se dibuja el sistema de coordenadas de referencia. Se asigna el modo XOR
para borrar
% sólo el robot dibujado anteriormente. Se utiliza un grosor de línea de 2
unidades
p = plot3(x0,y0,z0,'EraseMode','xor','LineWidth',2);
% Se asigna una rejilla a los ejes
grid;
% Se establecen los límites de los ejes
axis([-1.5 1.5 -1.5 1.5 0 1.5]);
% Mantiene el gráfico actual
hold on;

% Número de columnas de la matriz
n = size(mat_q,2);

% Se dibuja la disposición del robot correspondiente a cada columna
for i=1:n

    % Variables articulares del brazo robot
    q1 = mat_q(1,i);
    q2 = mat_q(2,i);
    q3 = mat_q(3,i);
    q4 = mat_q(4,i);

    % Matrices de transformación homogénea entre sistemas de coordenadas
    consecutivos
    A01 = denavit(q1, d(1), a(1), alfa(1));
    A12 = denavit(teta(2), q2, a(2), alfa(2));
    A23 = denavit(teta(3), q3, a(3), alfa(3));
    A34 = denavit(q4, d(4), a(4), alfa(4));

    % Matrices de transformación del primer sistema al correspondiente
    A02 = A01 * A12;
    A03 = A02 * A23;
    A04 = A03 * A34;

    % Vector de posicion (x, y, z) de cada sistema de coordenadas
    x1 = A01(1,4); y1 = A01(2,4); z1 = A01(3,4);
    xi = x1; yi = y1; zi = z1 + q2;
    x2 = A02(1,4); y2 = A02(2,4); z2 = A02(3,4);
    x3 = A03(1,4); y3 = A03(2,4); z3 = A03(3,4);
    x4 = A04(1,4); y4 = A04(2,4); z4 = A04(3,4);

    % Se dibuja el robot
    x = [x0 x1 xi x2 x3 x4];
    y = [y0 y1 yi y2 y3 y4];
    z = [z0 z1 zi z2 z3 z4];
    set(p,'XData',x,'YData',y,'ZData',z);
    % Se fuerza a MATLAB a actualizar la pantalla
    drawnow;
end

```

Para realizar la animación simplemente se ejecutará desde Matlab:

```
» animacion4(mat_q')
```

En la figura 6.11 se muestra el instante final de la animación del robot cuyo extremo realiza una trayectoria en línea recta entre la posición $(0,-0.6,0.5)$ y $(0,-1,1.2)$.

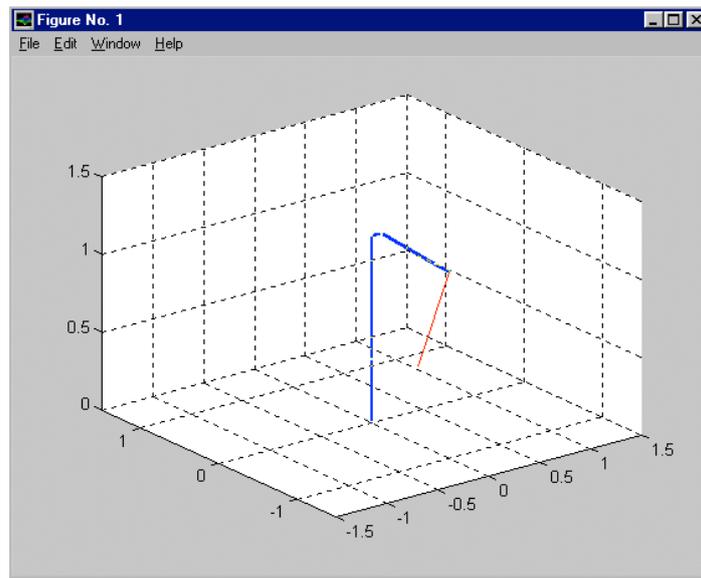


Figura 6.11. Instante final de la animación del robot de 4 GDL. Se ha añadido la trayectoria seguida por el extremo del robot.

EJERCICIOS PROPUESTOS

Implementar las funciones del ejemplo 6.1 para el caso del robot rotacional de 6 gdl mostrado en las prácticas anteriores.

Implementar las funciones del ejemplo 6.2 para el caso del robot rotacional de 6 gdl mostrado en las prácticas anteriores.