4

Modelo dinámico

Introducción

Para tratar la dinámica de los robots manipuladores es posible optar por emplear MATLAB directamente o bien Simulink. Si se pretende obtener resultados simbólicos y trabajar con matrices, resultará más conveniente el uso de MATLAB, pero en el resto de casos ambos entornos resultan equiparables en lo que se refiere a la facilidad de manejo. En Ollero [3], se proporcionan ejemplos que permiten al lector familiarizarse con ambos modos de trabajar.

Para resolver la dinámica de un robot son necesarios un conjunto de parámetros, que habrán de ser introducidos en una matriz siguiendo un formato que se explica en la sección dyn de este capítulo. Por tanto lo primero que deberá hacer el lector será acudir a dicha sección para aprender el modo en que deberá introducir los parámetros cinemáticos y dinámicos del robot para el que quiera simular su dinámica. Dichos parámetros se deberán corresponder con la notación de Craig [2].

También se ha contemplado la posibilidad de que la base del robot sea móvil, por lo que existen entradas para las velocidades angulares y lineales de la base.

Este capítulo ha sido dividido en tres partes: una con la descripción de las funciones MATLAB disponibles, otra con los bloques Simulink correspondientes al caso de base fija, y finalmente una sección con los bloques Simulink necesarios para simular la dinámica del brazo sobre una plataforma móvil.

Instrucciones MATLAB relacionadas con la dinámica

En la siguiente tabla se resumen las instrucciones que se pueden emplear en relación con la dinámica de los brazos manipuladores:

Tabla 4.1: Instrucciones relacionadas con la dinámica.

	Propósito
coriolis	Calcular el par correspondiente a los términos centrífugos y de Coriolis
dyn	Contener los parámetros cinemáticos y dinámicos del manipulador
fdyn	Permite integrar la dinámica directa para obtener las trayectorias articulares
friction	Calcular el par correspondiente a las fricciones según un cierto modelo
gravity	Calcular el par correspondiente al término gravitatorio

inertia	Calcular la matriz de masas del manipulador
rne	Calcular el modelo dinámico completo del manipulador mediante el método recursivo de Newton-Euler

coriolis

Propósito

Calcular el par correspondiente a los términos centrífugos y de Coriolis.

Sintaxis

```
tau_c=coriolis(dyn,q,qd)
```

Descripción

Devuelve el par (vector fila `tau_c`) que corresponde a los términos centrífugos y de Coriolis para el estado definido por los vectores fila `q` y `qd`, que son respectivamente las posiciones y velocidades articulares. La matriz `dyn` debe contener los parámetros cinemáticos y dinámicos del manipulador.

Para el caso en que `q` y `qd` sean matrices, cada fila es interpretada como un vector de variables/velocidades articulares y `tau_c` es una matriz que en cada fila contiene el par correspondiente.

Algoritmo

Se calcula a partir de las ecuaciones de Newton-Euler (función `rne`) haciendo nulas las aceleraciones articulares y la aceleración de la gravedad, obteniéndose por tanto $V(q, q')$.

Ejemplos

Ver Ejemplo 5.1 de Ollero [3].

Ver también

`dyn`, `gravity`, `inertia`, `rne`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

dyn

Propósito

Contener los parámetros cinemáticos y dinámicos del manipulador.

Descripción

Cada vez que se quiera utilizar una función de HEMERO relacionada con la dinámica será necesario introducir en una matriz los parámetros de Denavit-Hartenberg del manipulador, junto con ciertos parámetros dinámicos. El modo de introducir esta información en dicha matriz (a la que se denominará genéricamente `dyn`) es el siguiente:

- Habrá una fila por cada enlace que tenga el manipulador.
- Cada fila tendrá el siguiente formato:

Columna	Símbolo	Descripción
1	α_{i-1}	Parámetros de Denavit-Hartenberg
2	a_{i-1}	
3	θ_i	
4	d_i	
5	σ_i	Tipo de articulación; 0 si es de rotación y 1 si es prismática
6	masa	Masa del enlace i
7	rx	Centro de masas del enlace respecto al cuadro de referencia de dicho enlace
8	ry	
9	rz	
10	Ixx	Elementos del tensor de inercia referido al centro de masas del enlace
11	Iyy	
12	Izz	
13	Ixy	
14	Iyz	
15	Ixz	
16	Jm	Inercia de la armadura
17	G	Velocidad de la articulación / velocidad del enlace
18	B	Fricción viscosa, referida al motor
19	Tc+	Fricción de Coulomb (rotación positiva), referida al motor
20	Tc-	Fricción de Coulomb (rotación negativa), referida al motor

Así pues, para un robot con *n* enlaces, la matriz `dyn` tendría dimensiones *n*x20.

Todos los ángulos deberán ser introducidos en radianes. El resto de parámetros de la matriz podrán tener las unidades que se deseen, siempre que se sea coherente en el uso de dichas unidades. Es decir que si se introducen las masas en Kg y los centros de masas en metros, al escribir el tensor de inercia se deberá expresar en Kg·m^2.

Ejemplos

Ver Ejemplos 5.1, 5.2, 5.3 y 5.4 de Ollero [3].

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

fdyn

Propósito

Permite integrar la dinámica directa para obtener las trayectorias articulares.

Sintaxis

```
[t q qd]=fdyn(dyn,t0,t1)
[t q qd]=fdyn(dyn,t0,t1,grav,torqfun)
[t q qd]=fdyn(dyn,t0,t1,grav,torqfun,q0,qd0)
```

Descripción

La función `fdyn` se encarga de integrar las ecuaciones del movimiento del manipulador en el intervalo de tiempo que va desde `t0` a `t1`, usando la función de integración `ode45` de MATLAB. Devuelve como resultado un vector de tiempo `t`, y dos matrices `q` y `qd`, que se corresponden respectivamente con las posiciones y velocidades articulares. Cada una de estas matrices tiene una fila por cada instante contenido en el vector `t`, y tantas columnas como variables articulares haya.

El par con el que se actúa sobre el manipulador se debe especificar mediante una función:

```
tau=torqfun(t,x)
```

donde `t` es el instante actual y `x=[q;qd]` es un vector columna con $2n$ elementos que contiene las posiciones y velocidades articulares. Dicha función debe devolver un vector fila. Si no se especifica el parámetro `torqfun`, entonces se asume que se está aplicando un par nulo al manipulador.

También es posible pasar como parámetros dos vectores `q0` y `qd0` con los valores iniciales de las posiciones y velocidades articulares.

Asimismo, en el parámetro `grav`, se puede introducir el vector aceleración de la gravedad que sufre el manipulador. Si no se indica ninguno, por defecto se asume una aceleración de 9.81 m/s^2 en la dirección y sentido del vector $-\hat{Z}$ (es decir, `grav=[0 0 -9.81]`).

Algoritmo

Esta función se encarga de resolver el sistema de ecuaciones diferenciales:

$$q'' = M^{-1}(q)[\tau - V(q, q') - G(q) - F(q')] \qquad (4.1)$$

El modelo de fricciones que se emplea es el que se presenta en la descripción de la función `friction`.

El vector τ que aparece en la ecuación es el que se introduce en el parámetro `torqfun`.

Ejemplos

Ver Ejemplo 5.2 de Ollero [3].

Ver también

`dyn`, `ode45`, `rne`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

friction

Propósito

Calcular el par correspondiente a las fricciones según un cierto modelo.

Sintaxis

```
tau_f=friction(dyn,qd)
```

Descripción

Devuelve el par (vector fila `tau_f`) que corresponde a las fricciones que sufren las articulaciones para las velocidades articulares contenidas en el vector `qd`. En la matriz `dyn` están los parámetros necesarios:

Columna	Símbolo	Descripción
17	G	Velocidad de la articulación / velocidad del enlace
18	B	Fricción viscosa, referida al motor
19	Tc+	Fricción de Coulomb (rotación positiva), referida al motor
20	Tc-	Fricción de Coulomb (rotación negativa), referida al motor

para calcular las fricciones según el modelo que se indica a continuación.

Algoritmo

El modelo de fricciones incluye términos correspondientes a las fricciones viscosa y de Coulomb. Los parámetros de fricción (B, Tc+ y Tc-) que se introducen en la matriz `dyn` están referidos al motor, por lo que para referir los valores de fricción a los enlaces es necesario también el parámetro de reducción G.

El modelo que se emplea es:

$$F_i(q'_i) = \begin{cases} G_i^{\,2} B_i q'_i + G_i \tau_i^- & , q'_i < 0 \\ G_i^{\,2} B_i q'_i + G_i \tau_i^+ & , q'_i > 0 \end{cases} \tag{4.2}$$

donde el subíndice i se refiere a la articulación i-ésima.

Ver también

`coriolis`, `dyn`, `gravity`, `rne`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

gravity

Propósito

Calcular el par correspondiente al término gravitatorio.

Sintaxis

```
tau_g=gravity(dyn,q)
tau_g=gravity(dyn,q,grav)
```

Descripción

Devuelve el par (vector fila `tau_g`) que corresponde al término gravitatorio para el estado definido por el vector fila `q` de variables articulares. La matriz `dyn` debe contener los parámetros cinemáticos y dinámicos del manipulador.

Para el caso en que `q` sea una matriz, cada fila es interpretada como un vector de variables articulares y el resultado (`tau_g`) es una matriz que en cada fila contiene el par correspondiente.

El parámetro `grav` debe contener el vector con la aceleración de la gravedad que sufre el manipulador. Si no se le da este parámetro, se asume que la aceleración de la gravedad tiene un valor de 9.81 m/s^2, y actúa en la dirección y sentido del vector $-\hat{Z}$ (es decir `grav=[0 0 -9.81]`).

Algoritmo

Se calcula a partir de las ecuaciones de Newton-Euler (función `rne`) haciendo nulas las velocidades y aceleraciones articulares, obteniéndose por tanto $G(q)$.

Ejemplos

Ver Ejemplo 5.1 de Ollero [3].

Ver también

`coriolis`, `dyn`, `inertia`, `rne`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

inertia

Propósito

Calcular la matriz de masas del manipulador.

Sintaxis

```
M=inertia(dyn,q)
```

Descripción

Devuelve la matriz de masas $M(q)$ correspondiente al vector fila `q` que contiene las posiciones articulares. En la matriz `dyn` están contenidos los parámetros cinemáticos y dinámicos.

Para un manipulador con n articulaciones la matriz de masas tiene dimensiones nxn y es simétrica.

Algoritmo

Se calcula a partir de las ecuaciones de Newton-Euler (función `rne`) haciendo nulas las velocidades articulares y la aceleración de la gravedad, al mismo tiempo que se hacen igual a uno las aceleraciones articulares.

Precauciones

Hay que recordar que si en la matriz `dyn` aparecen parámetros relativos a la inercia del motor, entonces dicha inercia, referida al cuadro de referencia del enlace, aparecerá en la diagonal de la matriz M.

Ejemplos

Ver Ejemplo 5.1 de Ollero [3].

Ver también

`coriolis`, `dyn`, `gravity`, `rne`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

rne

Propósito

Calcular el modelo dinámico completo del manipulador mediante el método recursivo de Newton-Euler.

Sintaxis

```
tau=rne(dyn,q,qd,qdd)
tau=rne(dyn,[q qd qdd])

tau=rne(dyn,q,qd,qdd,grav)
tau=rne(dyn,[q qd qdd],grav)

tau=rne(dyn,q,qd,qdd,grav,fext)
tau=rne(dyn,[q qd qdd],grav,fext)
```

Descripción

La función `rne` se encarga de calcular las ecuaciones del movimiento para proporcionar el par total en función de la posición, velocidad y aceleración articulares.

Si `q`, `qd` y `qdd` son vectores fila, el par `tau` es un vector fila. En el caso de que `q`, `qd` y `qdd` sean matrices, cada fila se interpretará como un vector de posiciones/velocidades/aceleraciones articulares y el resultado será una matriz `tau` en la que cada fila tendrá el par correspondiente.

El vector aceleración de la gravedad deseado se le puede pasar en el parámetro `grav`. Si no se le da este parámetro, se toma por defecto una aceleración de 9.81 m/s^2 en la dirección y sentido del vector $-\hat{Z}$ (es decir, se asume que `grav=[0 0 -9.81]`).

También es posible especificar una fuerza/momento externo actuando al final del manipulador mediante un vector de 6 elementos `fext=[Fx Fy Fz Mx My Mz]` expresado en el sistema de referencia del efector final.

El par total que devuelve la función también contiene términos debidos a la inercia de la armadura y a las fricciones. Dichos términos se calculan a partir de ciertos parámetros de la matriz `dyn` mediante el modelo que se presenta en la descripción de la función `friction`.

Algoritmo

Calcula el par:

$$\tau = M(q)q'' + G(q) + F(q') + V(q, q') + F_{ext} \tag{4.3}$$

donde:

- $M(q)$ es la matriz de masas.
- $G(q)$ es el término gravitatorio.
- $F(q')$ es el término que incluye las fricciones viscosa y de Coriolis, según el modelo que se muestra en la descripción de la función `friction`.
- $V(q, q')$ se corresponde con los términos centrífugos y de Coriolis.
- F_{ext} incluye aquellos términos que aparecen si hay un vector de fuerza/momento externo distinto del vector nulo aplicado sobre el final del manipulador.

Ejemplos

Ver Ejemplos 5.1 y 5.3 de Ollero [3].

Ver también

`coriolis`, `dyn`, `gravity`, `inertia`

Referencias

Corke, P.I., *A robotics toolbox for MATLAB*, IEEE Robotics and Automation Magazine, Vol.3, núm. 1, pp. 24-32, 1996.

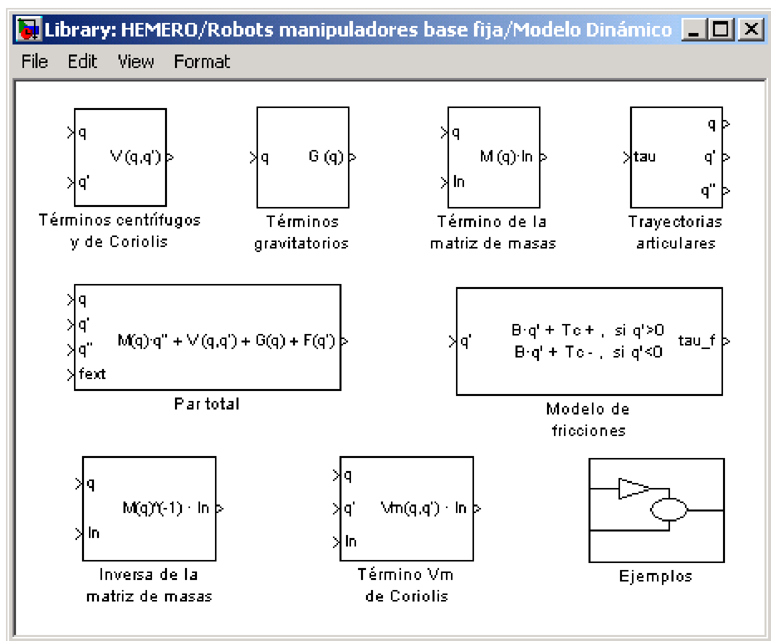
Craig, J.J., *Introduction to robotics*. Addison Wesley, segunda ed., 1989.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Dinámica en Simulink de manipuladores con base fija

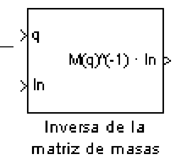
En las siguientes páginas se describen los bloques de Simulink que corresponden al modelo dinámico de un robot manipulador con base fija. Al pulsar sobre esos bloques se solicita la introducción de una serie de parámetros. En casi todos los bloques se solicita la matriz de parámetros cinemáticos y dinámicos, que es aquella cuyo formato se describió en el apartado `dyn` de la sección anterior.

El conjunto de bloques disponibles es el siguiente:



y se describen a continuación.

Inversa de la matriz de masas



Propósito

Calcular el producto de la inversa de la matriz de masas del manipulador por un vector columna (In) que se le pasa como entrada.

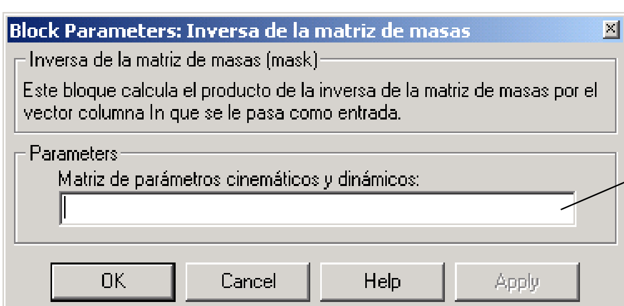
Descripción

Este bloque se encarga de obtener el resultado del producto:

$$M^{-1}(q) \cdot In \tag{4.4}$$

donde In es un vector columna cualquiera que se le da como entrada. La otra entrada del bloque es evidentemente un vector columna con las posiciones articulares. A la salida, el resultado es un vector columna también.

Parámetros



Esta matriz debe tener el formato que se indica en el apartado dyn.

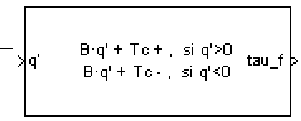
Ejemplos

Ver Ejemplos 5.2 y 5.4 de Ollero [3].

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Modelo de fricciones



Modelo de fricciones

Propósito

Calcular el par correspondiente a las fricciones viscosa y de Coulomb empleando cierto modelo para las mismas.

Descripción

Devuelve el par (vector columna tau_f) que contiene las fricciones a partir del vector columna de velocidades articulares (q') que se le pase.

El modelo que se emplea es:

$$F_i(q'_i) = \begin{cases} G_i^2 B_i q'_i + G_i \tau_i^- & , q'_i < 0 \\ G_i^2 B_i q'_i + G_i \tau_i^+ & , q'_i > 0 \end{cases} \qquad (4.5)$$

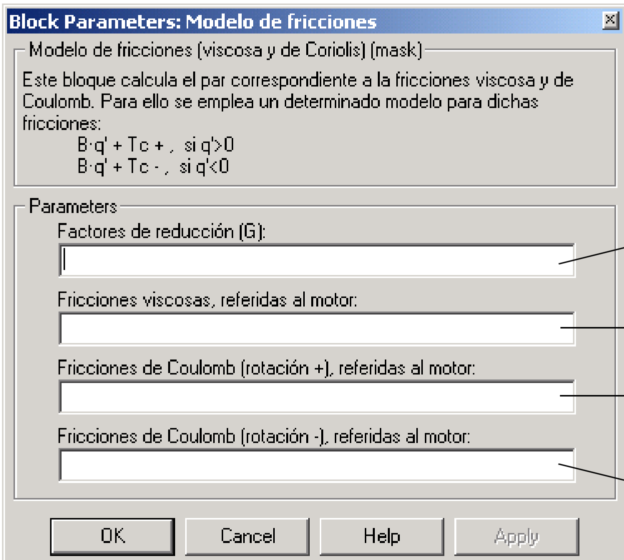
donde el subíndice i se refiere a la articulación i-ésima. Los parámetros que aparecen en este modelo se detallan a continuación.

Parámetros

Todos los parámetros que se solicitan en el cuadro de diálogo que se muestra a continuación son vectores con un número de elementos igual al número de articulaciones del manipulador.

Cada elemento de dichos vectores hace referencia a la correspondiente articulación del manipulador. Así por ejemplo, el segundo elemento del vector de fricciones viscosas, es la fricción viscosa de la segunda articulación del manipulador.

Los parámetros de fricción han de estar referidos al motor, por lo que el cuadro de diálogo solicita también el parámetro de reducción G con la finalidad de poder referir los valores de fricción a los enlaces.



G es velocidad de la articulación/velocidad del enlace

Son los elementos B_i del modelo

Son los elementos τ_i^- del modelo

Son los elementos τ_i^+ del modelo

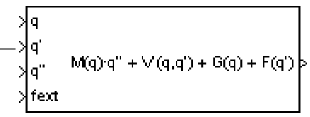
Ejemplos

Ver Ejemplo 5.4 de Ollero [3].

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Par total



Par total

Propósito

Calcular el modelo dinámico completo del manipulador mediante el método recursivo de Newton-Euler.

Descripción

El bloque se encarga de calcular las ecuaciones del movimiento para proporcionar el par total (vector columna) en función de las posiciones, velocidades y aceleraciones articulares que se introducen mediante los vectores columna q, q' y q''. Asimismo se puede introducir un vector columna (fext), que permite especificar una fuerza/momento externo actuando sobre el final del manipulador. Dicho vector tendrá seis elementos y será de la forma [Fx Fy Fz Mx My Mz]' (expresado en el sistema de referencia del efector final).

El par viene dado por:

$$\tau = M(q)q'' + G(q) + V(q, q') + F(q') + F_{ext} \tag{4.6}$$

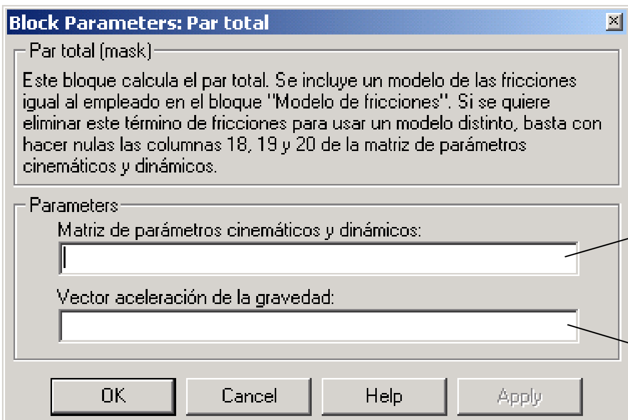
donde:

- $M(q)$ es la matriz de masas.
- $G(q)$ es el término gravitatorio.
- $F(q')$ es el término que incluye las fricciones viscosa y de Coriolis, según el siguiente modelo:

$$F_i(q'_i) = \begin{cases} G_i^2 B_i q'_i + G_i \tau_i^- & , q'_i < 0 \\ G_i^2 B_i q'_i + G_i \tau_i^+ & , q'_i > 0 \end{cases} \tag{4.7}$$

- $V(q, q')$ se corresponde con los términos centrífugos y de Coriolis.
- F_{ext} incluye aquellos términos que aparecen si hay un vector de fuerza/momento externo distinto del vector nulo aplicado sobre el final del manipulador.

Parámetros



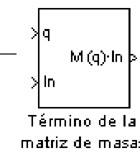
Esta matriz debe tener el formato que se indica en el apartado `dyn`.

Se introduciría [0 0 -9.81] para el caso en que la aceleración fuera en la dirección y sentido del vector $-\hat{Z}$.

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Término de la matriz de masas



Propósito

Calcular el par asociado a la matriz de masas.

Descripción

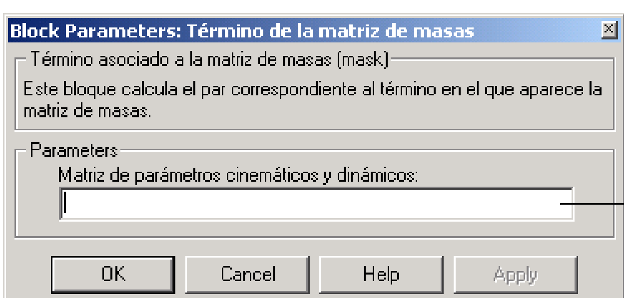
Calcula el producto de la matriz de masas por el vector de entrada `In`. En la otra entrada hay que introducir el vector de posiciones articulares `q`. Si el vector `In` es el vector de aceleraciones articulares, entonces a la salida del bloque se tiene el par asociado a la matriz de masas.:

$$M(q) \cdot q'' \tag{4.8}$$

Para un manipulador con n articulaciones la matriz de masas tiene dimensiones nxn y es simétrica.

Hay que recordar que si se introducen parámetros dinámicos relativos a la inercia del motor, entonces dicha inercia, referida al cuadro de referencia del enlace, aparecerá en la diagonal de la matriz M.

Parámetros

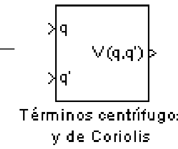


Esta matriz debe tener el formato que se indica en el apartado `dyn`.

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Términos centrífugos y de Coriolis



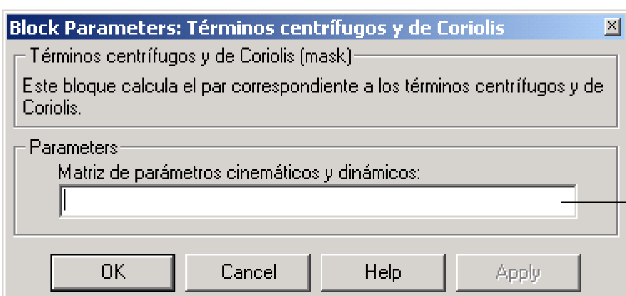
Propósito

Calcular el par correspondiente a los términos centrífugos y de Coriolis.

Descripción

Devuelve un vector columna con el par que corresponde a los términos centrífugos y de Coriolis para el estado definido por los vectores columna q y q', que son respectivamente las posiciones y velocidades articulares.

Parámetros



Esta matriz debe tener el formato que se indica en el apartado dyn.

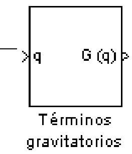
Ejemplos

Ver Ejemplos 5.2 y 5.4 de Ollero [3].

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Términos gravitatorios



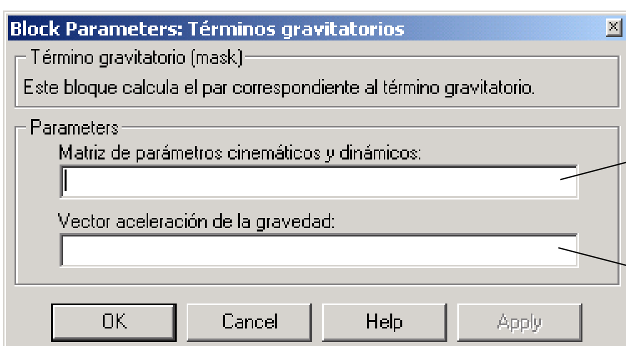
Propósito

Calcular el par debido a los términos gravitatorios.

Descripción

Devuelve un vector columna con el par que corresponde al término gravitatorio para el estado definido por el vector columna `q` de variables articulares.

Parámetros



Esta matriz debe tener el formato que se indica en el apartado `dyn`.

Se introduciría [0 0 -9.81] para el caso en que la aceleración fuera en la dirección y sentido del vector $-\hat{Z}$.

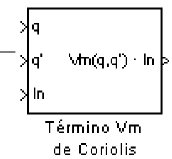
Ejemplos

Ver Ejemplos 5.2 y 5.4 de Ollero [3].

Referencias

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Término Vm de Coriolis



Propósito

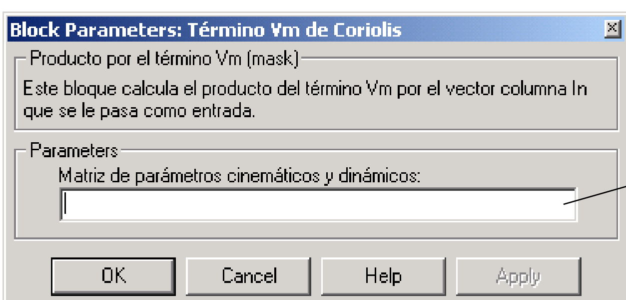
Calcular el producto de un vector por el término V_m de Coriolis.

Descripción

Calcula el producto del término V_m de Coriolis por el vector de entrada In. En la otras entradas hay que introducir los vectores con las posiciones y las velocidades articulares (q y q' respectivamente). Si el vector In es el vector de velocidades articulares, entonces a la salida del bloque se tiene el par asociado al término centrífugo y de Coriolis:

$$V(q, q') = V_m(q, q')q' \tag{4.9}$$

Parámetros



Esta matriz debe tener el formato que se indica en el apartado dyn.

Algoritmo

El término V_m se calcula mediante la fórmula:

$$V_m(q, q') = \frac{1}{2}(M' + U^T - U) \tag{4.10}$$

donde:

$$M'(q) = (q' \otimes I_n)\frac{\partial M}{\partial q}; \quad U(q, q') = (I_n \otimes q'^T)\frac{\partial M}{\partial q} \tag{4.11}$$

siendo $\otimes$ el producto de Kronecker [1] y $\frac{\partial M}{\partial q} = \begin{bmatrix} \frac{\partial M}{\partial q_1} \\ \dots \\ \frac{\partial M}{\partial q_n} \end{bmatrix}$.

Precauciones

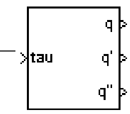
Se debe tener en cuenta que este bloque emplea una variable global denominada VM.

Referencias

Lewis, F.L., C.T. Abdallah, D.M. Dawson, *Control of robot manipulators*, Macmillan Publishing Company, 1993.

Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Trayectorias articulares



Trayectorias articulares

Propósito

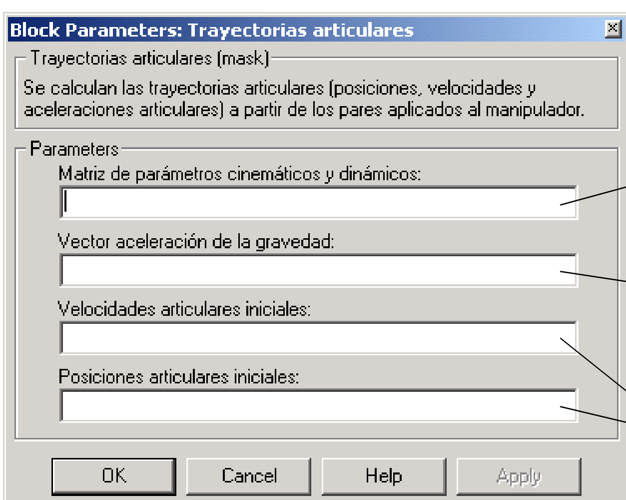
Calcular las variables articulares de un determinado manipulador a partir del vector que contiene los pares aplicados en cada una de las articulaciones.

Descripción

Devuelve los vectores columna correspondientes a las posiciones, velocidades y aceleraciones angulares (q, q' y q''). Como entrada hay que introducir un vector columna (tau) con los pares aplicados a cada una de las articulaciones.

Por tanto mediante este bloque es posible simular el comportamiento dinámico del manipulador ante determinados pares.

Parámetros



Esta matriz debe tener el formato que se indica en el apartado dyn.

Se introduciría [0 0 -9.81] para el caso en que la aceleración fuera en la dirección y sentido del vector $-\hat{Z}$.

Valores iniciales de posiciones y velocidades articulares, necesarios para resolver la ecuación diferencial que permite obtener las trayectorias articulares.

Algoritmo

Este bloque resuelve la siguiente ecuación diferencial:

$$\theta'' = M^{-1}(\theta)[\tau - V(\theta, \theta') - G(\theta) - F(\theta')] \quad (4.12)$$

mediante métodos de integración numérica. En concreto se ha empleado el esquema de la Figura 4.1, en el que se puede observar que las fricciones han sido modeladas como:

$$F_i(q'_i) = \begin{cases} G_i^2 B_i q'_i + G_i \tau_i^- & , q'_i < 0 \\ G_i^2 B_i q'_i + G_i \tau_i^+ & , q'_i > 0 \end{cases} \tag{4.13}$$

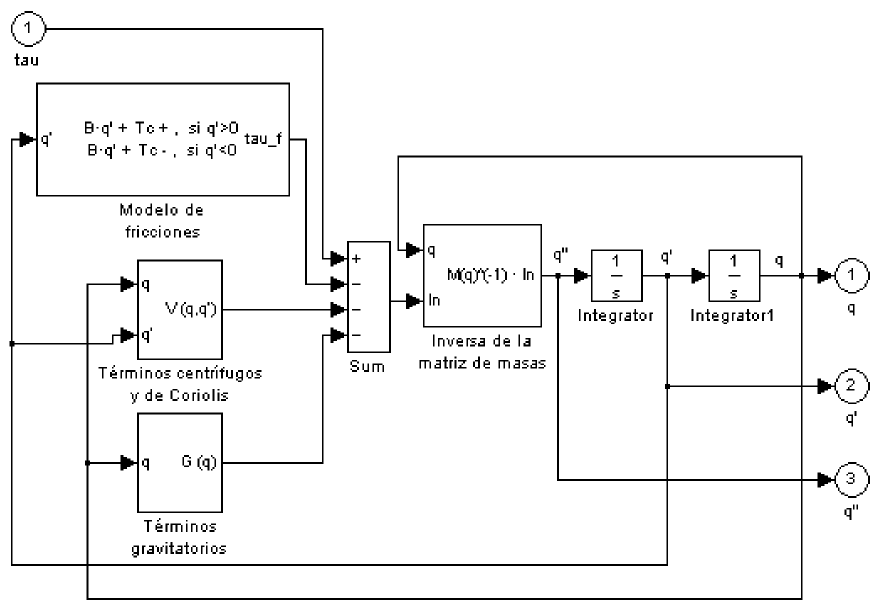


Figura 4.1: Diagrama que permite determinar el comportamiento dinámico del manipulador ante la aplicación de determinados pares.

Ejemplos

Ver Ejemplo 5.2 de Ollero [3].

Referencias

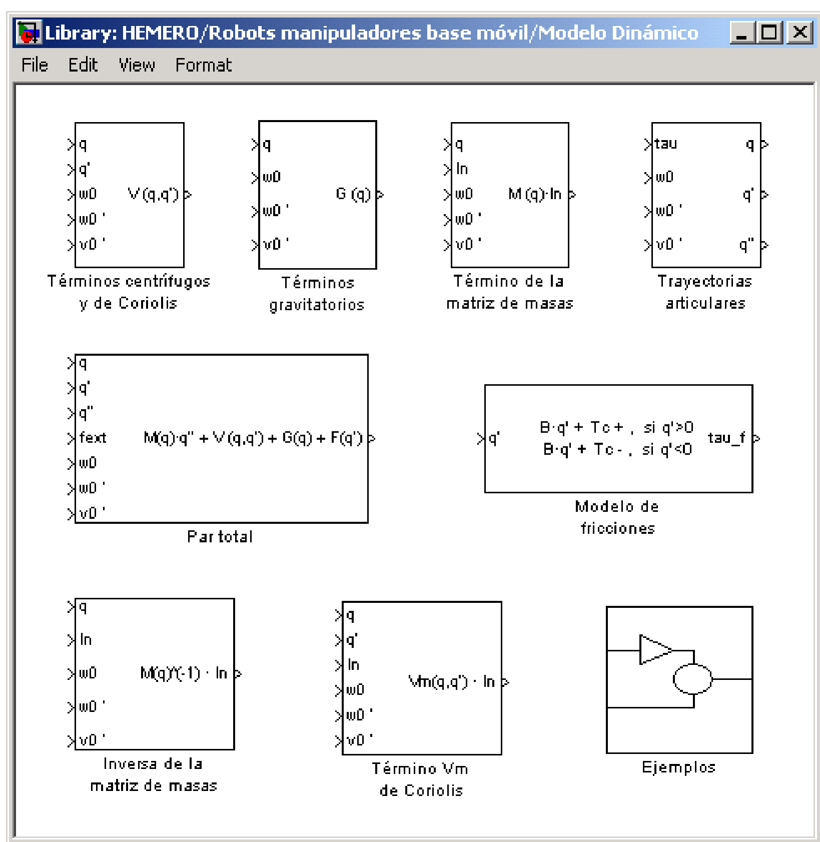
Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.

Dinámica en Simulink de manipuladores con base móvil

Los bloques de Simulink que corresponden al modelo dinámico de un robot manipulador con base móvil son básicamente los mismos que se han presentado para el caso de manipuladores con base fija. La única diferencia es que para el caso de base móvil, la mayoria de los bloques disponen de tres entradas adicionales. Dichas entradas son:

Símbolo	Descripción
w0	Velocidad angular de la base
w0'	Aceleración angular de la base
v0'	Aceleración lineal de la base

El conjunto de bloques disponibles es el siguiente:



Ejemplos

Ejemplo H.4.1 (archivo `ejh41.m`): Ver Ejemplo 5.1 en Ollero [3].

Ejemplo H.4.2 (archivos `ejh42.m` y `ejh42a.mdl`): Ver Ejemplo 5.2 en Ollero [3].

Ejemplo H.4.3 (archivo `ejh43.m`): Ver Ejemplo 5.3 en Ollero [3].

Ejemplo H.4.4 (archivo `ejh44a.mdl`): Ver Ejemplo 5.4a en Ollero [3].

Ejemplo H.4.5 (archivo `ejh44b.mdl`): Ver Ejemplo 5.4b en Ollero [3].

Ejemplo H.4.6 (archivo `ejh44c.mdl`): Ver Ejemplo 5.4c en Ollero [3].

Referencias

[1] Brewer, J. W., *Kronecker products and matrix calculus in matrix theory*, IEEE Trans. Circuits Syst., vol. CAS-25, nº 9, pp. 772-781, Sept. 1978.

[2] Craig, J.J., *Introduction to robotics*. Addison Wesley, segunda ed., 1989.

[3] Ollero, A., *Robótica: Manipuladores y robots móviles*, Marcombo-Boixareu editores, 2001.