

## Modelado y Resolución del Problema Dinámico Inverso y Directo en Tiempo Real de Robots Industriales

N. Rosillo\*, A. Valera\*\*, F. Benimeli\*\*, V. Mata\*\*\* y F. Valero\*\*\*

\* *Departamento de Ciencias Físicas. Matemáticas y de la Computación  
Universidad Cardenal Herrera-CEU San Pablo, c/ San Bartolomé, 55 46115  
Alfara del Patriarca, Valencia, España (e-mail: nrosillo@uch.ceu.es)*

\*\* *Instituto de Automática e Informática Industrial  
Universidad Politécnica de Valencia, Camino de Vera 14, 46022 Valencia, España  
(e-mail: giuprog@isa.upv.es, frabean@ai2.upv.es)*

\*\*\* *Departamento de Ingeniería Mecánica y de Materiales  
Universidad Politécnica de Valencia, Camino de Vera 14, 46022 Valencia, España  
(e-mail: vmata@mcm.upv.es, fvalero@mcm.upv.es)*

**Resumen:** Este trabajo presenta el desarrollo y la validación de una metodología que permite el modelado y la resolución del problema dinámico en tiempo real de robots manipuladores. La ecuación dinámica del robot está basada en la ecuación de movimiento de Gibbs-Appell, proporcionando un conjunto de ecuaciones bien estructuradas que pueden ser calculadas en tiempo real.

El artículo aborda el cálculo e implementación del problema dinámico directo e inverso de robots, aplicándose al control en tiempo real de un robot industrial PUMA 560 equipado con una arquitectura abierta de control basada en un computador personal industrial. Copyright © 2010 CEA.

**Palabras Clave:** sistemas robotizados, control de robots, ecuaciones de Gibbs-Appell, dinámica inversa, control por computador, aplicaciones digitales de computación.

### 1. INTRODUCCIÓN

En este trabajo se desarrolla el modelo dinámico de robots manipuladores a partir de las ecuaciones de Gibbs-Appell. Se asume que el robot está constituido por elementos rígidos, modelándose de forma independiente el comportamiento dinámico de los actuadores. Además, para modelar el fenómeno de la fricción en las articulaciones del robot se han considerado también distintos modelos de fricción.

Se han reescrito las ecuaciones que constituyen el modelo dinámico en forma lineal con respecto a los parámetros dinámicos del modelo, esto es, masas, momentos de primer orden, componentes de los tensores de inercia de las barras y parámetros de fricción, lo cual facilita en gran medida su identificación.

Estas expresiones lineales se han plasmado en forma matricial, detallándose la obtención de los distintos términos de la matriz resultante, así como su complejidad computacional. La expresión matricial de las ecuaciones del sistema permitirá la aplicación de técnicas numéricas para analizar, reducir y resolver el sistema de ecuaciones que se genera y será utilizada tanto en la fase de identificación como posteriormente en la resolución de los problemas dinámicos directo e inverso.

Mediante un proceso de identificación de parámetros dinámicos se ha obtenido una estimación precisa de los valores de masas, localizaciones de centros de gravedad y componentes de los tensores de inercia de las barras, así como los términos que designan la fricción en los nudos, utilizando para ello las medidas experimentales obtenidas durante el movimiento del robot generado a partir de trayectorias optimizadas capaces de

excitar los distintos términos que componen las ecuaciones que modelizan el comportamiento dinámico del sistema.

Una vez obtenidos los parámetros dinámicos del modelo dinámico del robot, partiendo de la forma matricial de éste, se determinan los distintos términos de la ecuación dinámica del sistema, lo que permitirá resolver los problemas dinámicos directo e inverso.

Tanto el modelado dinámico del robot como los métodos de identificación han sido validados a partir de su aplicación experimental en un robot industrial PUMA 560. Las técnicas propuestas se han validado mediante el desarrollo y la ejecución de varias estrategias de control basadas en la ecuación dinámica cuyos parámetros han sido identificados. Para ello, el robot industrial ha sido equipado con una unidad de control abierta basada en un computador industrial. La arquitectura de control permite, mediante programas C++, el desarrollo de aplicaciones y algoritmos de control en tiempo real.

### 2. MODELADO DINÁMICO DE ROBOTS MANIPULADORES

Las ecuaciones del movimiento que modelizan el comportamiento dinámico de un sistema mecánico se pueden obtener a partir de distintos Principios de la Dinámica. Los primeros trabajos se remontan a 1965 cuando (Uicker, 1965) formuló un método basado en las ecuaciones de Lagrange y matrices de transformación homogéneas 4x4. Ese mismo año, (Hooker y Magulies, 1965) formularon un método basado en las ecuaciones de Newton-Euler. En estos algoritmos, las ecuaciones de movimiento están expresadas de forma cerrada, por lo que la complejidad computacional es elevada, generalmente del orden  $O(n^4)$ .

El primer algoritmo recursivo que resuelve el problema dinámico indirecto basado en el principio de Newton-Euler se debe a (Stepanenko y Vukobratovic, 1976). (Orin et al, 1980) obtuvieron un algoritmo de complejidad  $O(n^2)$  introduciendo sistemas de coordenadas locales y (Luh, Walker y Paul, 1980) obtuvieron un algoritmo recursivo eficiente basado también en Newton-Euler con una complejidad  $O(n)$ .

La aplicación de las ecuaciones de Kane en el campo de la robótica fue iniciada por (Kane y Levinson, 1983), quienes resolvieron los problemas dinámicos directo e inverso con una complejidad similar a los algoritmos más eficientes de la época. Posteriormente, (Angeles, Ma y Rojas, 1989) utilizaron notación tensorial para desarrollar, a partir de las mismas ecuaciones de Kane, un algoritmo eficiente para su resolución.

En este trabajo, se han utilizado las ecuaciones de Gibbs-Appell, ecuaciones que fueron introducidas por Gibbs en 1879 y formalizadas por Appell veinte años después. Estas ecuaciones constituyen, a pesar de su relativa poca utilización, una de las formulaciones dinámicas más simples. Ésta permite expresar las ecuaciones del robot de forma lineal con respecto a los parámetros dinámicos, lo cual es sumamente conveniente para la identificación de los mismos. Además proporcionan otras ventajas muy interesantes como, por ejemplo, permitir la inclusión del efecto de la dinámica de los actuadores de una forma simple y directa (Provenzano, 2001).

Una de las primeras referencias de la utilización de este tipo de ecuaciones aplicadas en el campo de la dinámica de robots se puede encontrar en (Renaud, 1983). (Vukobratovic y Kircanski, 1985) desarrollaron un algoritmo cerrado con un coste computacional  $O(n^3)$ . (Desoyer y Lugner, 1989) desarrollaron un algoritmo recursivo usando la Jacobiana para evitar derivadas algebraicas o numéricas, obteniendo el mismo coste computacional.

Considerando un sistema de referencia inercial, la función de Gibbs para un cuerpo rígido compuesto por  $n$  partículas elementales con masa  $m_i$  y aceleración  $a_i$ , en su forma original, es:

$$G = \frac{1}{2} \sum_{i=1}^n m_i a_i^2 \quad (1)$$

En el caso de un robot manipulador compuesto por  $n$  elementos considerados rígidos, la función de Gibbs del elemento  $i$ -ésimo viene dada por:

$$G_i = \frac{1}{2} \cdot m_i \cdot {}^0\ddot{r}_{G_i}^T \cdot {}^0\ddot{r}_{G_i} + \frac{1}{2} \cdot {}^0\dot{\omega}_i^T \cdot {}^0\mathbf{I}_{G_i} \cdot {}^0\dot{\omega}_i + {}^0\dot{\omega}_i^T \cdot \left[ {}^0\bar{\omega}_i \times ({}^0\mathbf{I}_{G_i} \cdot {}^0\bar{\omega}_i) \right] \quad (2)$$

donde  $m_i$  es la masa,  ${}^0\mathbf{I}_{G_i}$  es el tensor centroidal de inercia,  ${}^0\bar{\omega}_i$  y  ${}^0\dot{\omega}_i$  son respectivamente la velocidad y la aceleración angulares, y  ${}^0\ddot{r}_{G_i}$  es la aceleración del centro de masas del elemento.

Las ecuaciones de movimiento de Gibbs-Appell se obtienen derivando la función de Gibbs anterior con respecto a las aceleraciones generalizadas  $\ddot{q}_k$ . Las fuerzas de inercia

generalizadas que compensan las fuerzas externas generalizadas  $\tau_k$  se obtienen como:

$$\tau_k = \sum_{i=k}^n \frac{\partial G_i}{\partial \ddot{q}_k}, \quad k = 1, 2 \dots n \quad (3)$$

Desarrollando esta última ecuación:

$$\tau_k = \sum_{i=k}^n \left\{ m_i \cdot {}^i\ddot{r}_{G_i}^T \cdot \frac{\partial {}^i\ddot{r}_{G_i}}{\partial \ddot{q}_k} + {}^i\dot{\omega}_i^T \cdot {}^i\mathbf{I}_{G_i} \cdot \frac{\partial {}^i\dot{\omega}_i}{\partial \ddot{q}_k} + \left( \frac{\partial {}^i\dot{\omega}_i}{\partial \ddot{q}_k} \right)^T \cdot \left[ {}^i\bar{\omega}_i \times ({}^i\mathbf{I}_{G_i} \cdot {}^i\bar{\omega}_i) \right] \right\} \quad (4)$$

Reorganizando la expresión (4) se pueden obtener los dos términos que se indican en la ecuación siguiente:

$$\tau_k = \underbrace{\sum_{i=k}^n \left\{ \left( \frac{\partial {}^i\dot{\omega}_i}{\partial \ddot{q}_k} \right)^T \cdot \left[ {}^i\mathbf{I}_{G_i} \cdot {}^i\dot{\omega}_i + {}^i\bar{\omega}_i \times ({}^i\mathbf{I}_{G_i} \cdot {}^i\bar{\omega}_i) \right] \right\}}_{A_k} + \underbrace{\sum_{i=k}^n \left\{ \left( \frac{\partial {}^i\ddot{r}_{G_i}}{\partial \ddot{q}_k} \right)^T \cdot m_i \cdot {}^i\ddot{r}_{G_i} \right\}}_{B_k} \quad (5)$$

Para poder resolver el problema dinámico directo o inverso a partir de la ecuación del robot (5) se necesitan los parámetros inerciales de dicha ecuación dinámica. Éstos se pueden obtener mediante métodos de identificación, necesitando para ello que se realice una reestructuración de estas ecuaciones de manera que los parámetros inerciales a identificar aparezcan de forma lineal en las expresiones. Empezando por los términos  $A_k$  y  $B_k$ , sustituyendo la expresión de la aceleración lineal del centro de masas del  $i$ -ésimo elemento y considerando el teorema de ejes paralelos, se puede obtener la siguiente ecuación para las fuerzas generalizadas expresadas en forma lineal con respecto a los parámetros inerciales obtenidos:

$$\tau_k = \left( \frac{\partial {}^k\dot{\omega}_k}{\partial \ddot{q}_k} \right)^T \sum_{i=k}^n {}^kR_i \left[ {}^i\hat{\eta}_i \cdot {}^i\bar{I}_i + \left( {}^i\tilde{r}_{O_k O_i} \cdot {}^i\tilde{\eta}_i - {}^i\ddot{r}_{O_i} \right) \cdot m_i \cdot {}^i\tilde{r}_{O_k G_i} + {}^i\tilde{r}_{O_k O_i} \cdot {}^i\ddot{r}_{O_i} \cdot m_i \right] \quad (6)$$

donde

$${}^k\hat{\eta}_i = \left( {}^k\dot{\omega}_i + {}^k\bar{\omega}_i \times {}^k\dot{\omega}_i \right), \quad {}^k\tilde{\eta}_i = \left( {}^k\dot{\omega}_i + {}^k\bar{\omega}_i \times {}^k\dot{\omega}_i \right)$$

Debido a las características de los robots manipuladores, se puede reescribir la ecuación (6) de forma que el vector de pares generalizados es linealmente dependiente de los parámetros dinámicos:

$$\bar{\tau} = W(\bar{q}, \dot{\bar{q}}, \ddot{\bar{q}}) \cdot \Phi \quad (7)$$

donde  $\bar{\tau}$  es el vector de fuerzas generalizadas,  $W$  es la matriz de observación y  $\Phi$  es el vector de parámetros del sistema. Si, por ejemplo, se considera que el robot está compuesto por  $n$  elementos rígidos,  $\Phi$  estará compuesto por los parámetros inerciales del sistema:

$$\Phi = \left[ {}^n\bar{I}_n \dots {}^1\bar{I}_1 \quad m_n \cdot {}^n\tilde{r}_{O_n G_n} \quad m_n \quad \dots \quad m_1 \cdot {}^1\tilde{r}_{O_1 G_1} \quad m_1 \right]^T \quad (8)$$

Para incrementar la precisión del modelo dinámico del robot manipulador se pueden incluir otros comportamientos dinámicos. En (Valera *et al.*, 2007), se muestra cómo se puede considerar en el modelo del robot la fricción en las articulaciones, utilizándose diferentes modelos de fricción: modelos lineales simétricos, el modelo de fricción asimétrico propuesto por (Lu, *et al.*, 1993) y el modelo de (Grotjahn *et al.*, 2001) que incluye un término no lineal con respecto de la velocidad. En este caso, la ecuación (6) está compuesta por las aportaciones:

$$\tau_k = \tau_{l_k} + \tau_{f_k} = \sum_{i=1}^n \frac{\partial G_{l_i}}{\partial \ddot{q}_k} + \underbrace{F_{f_k}(\dot{q}_k)}_{\text{fricción}} \quad (9)$$

*barras rígidas*

de forma que la ecuación (7) se puede reescribir de la forma siguiente:

$$\bar{\tau} = W(\bar{q}, \dot{\bar{q}}, \ddot{\bar{q}}) \cdot \bar{\Phi} = [W_l \quad W_f] \cdot \begin{bmatrix} \bar{\Phi}_l \\ \bar{\Phi}_f \end{bmatrix} \quad (10)$$

donde el vector de parámetros  $\bar{\Phi}_l$  viene dado por la ecuación (8)

y  $\bar{\Phi}_f$  está definido por el vector de los parámetros de fricción:

$$\bar{\Phi}_f = [F_{n_1} \cdots F_{n_m} \cdots F_{l_1} \cdots F_{l_m}]^T \quad (11)$$

En esta última referencia se encuentra también el modelo de robot en el que se han incluido los pares correspondientes al efecto giroscópico de los rotores y las posibles excentricidades de los ejes de rotación.

### 3. OBTENCIÓN DE LA MATRIZ DE OBSERVACIÓN

Como se ha presentado en las ecuaciones (7) o (10) del apartado anterior, para poder obtener la ecuación dinámica del sistema robotizado es necesario obtener la matriz de observación  $W$ . Dicha matriz se puede calcular a partir de la expresión siguiente:

$$W = D \cdot AB \quad (12)$$

donde  $D$  se define como:

$$D = \begin{bmatrix} \left( \frac{\partial^n \ddot{\omega}_n}{\partial \ddot{q}_n} \right)^T & 0_{1 \times 3} & \cdots & 0_{1 \times 3} \\ 0_{1 \times 3} & \left( \frac{\partial^{n-1} \ddot{\omega}_{n-1}}{\partial \ddot{q}_{n-1}} \right)^T & \cdots & 0_{1 \times 3} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{1 \times 3} & 0_{1 \times 3} & \cdots & \left( \frac{\partial^1 \ddot{\omega}_l}{\partial \ddot{q}_l} \right)^T \end{bmatrix} \quad (13)$$

$$\text{con } \left( \frac{\partial^j \ddot{\omega}_j}{\partial \ddot{q}_j} \right)^T = [0 \quad 0 \quad 1] \text{ para todo } j = 1..n.$$

La matriz  $AB$  está compuesta por dos submatrices correspondientes a cada uno de los términos recursivos  $A_k$  y  $B_k$  de la ecuación (5):

$$AB = \begin{bmatrix} A & B \end{bmatrix} \quad (14)$$

La matriz  $A$  está a su vez compuesta por  $n \times n$  submatrices  $A_{ij}$  definidas como:

$$A = \begin{bmatrix} {}^n \hat{\eta}_n & 0 & \cdots & 0 & 0 \\ {}^{n-1} R_n {}^n \hat{\eta}_n & {}^{n-1} \hat{\eta}_{n-1} & \cdots & 0 & 0 \\ {}^{n-2} R_n {}^n \hat{\eta}_n & {}^{n-2} R_{n-1} {}^{n-1} \hat{\eta}_{n-1} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ {}^2 R_n {}^n \hat{\eta}_n & {}^2 R_{n-1} {}^{n-1} \hat{\eta}_{n-1} & \cdots & {}^2 \hat{\eta}_2 & 0 \\ {}^1 R_n {}^n \hat{\eta}_n & {}^1 R_{n-1} {}^{n-1} \hat{\eta}_{n-1} & \cdots & {}^1 R_2 {}^2 \hat{\eta}_2 & {}^1 \hat{\eta}_1 \end{bmatrix} \quad (15)$$

Así, las submatrices de  $A$  se pueden calcular como:

$$\begin{cases} A_{ij} = 0 & i < j \\ A_{ij} = {}^{n+1-j} \dot{\omega}_{n+1-j} + {}^{n+1-j} \ddot{\omega}_{n+1-j} & i = j \\ A_{ij} = {}^{n+1-i} R_{n+2-i} A_{i-1,j} & i > j \end{cases} \quad (16)$$

La matriz  $B$  se compone también de  $n \times n$  términos, descomponiéndose a su vez en una submatriz  $3 \times 3$   $B1_{ij}$  y una submatriz columna  $3 \times 1$   $B2_{ij}$

$$B_{ij} = \begin{bmatrix} B1_{ij} & B2_{ij} \end{bmatrix} \quad (17)$$

La matriz  $B1$ , constituida a partir de los términos  $B1_{ij}$ , presentará la forma siguiente:

$$B1 = \begin{bmatrix} -{}^n \ddot{r}_{O_n} & 0 & \cdots & 0 & 0 \\ {}^{n-1} \ddot{H}_n & -{}^{n-1} \ddot{r}_{O_{n-1}} & \cdots & 0 & 0 \\ {}^{n-2} \ddot{H}_n & {}^{n-2} \ddot{H}_{n-1} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ {}^2 \ddot{H}_n & {}^2 \ddot{H}_{n-1} & \cdots & -{}^2 \ddot{r}_{O_2} & 0 \\ {}^1 \ddot{H}_n & {}^1 \ddot{H}_{n-1} & \cdots & {}^1 \ddot{H}_2 & 0 \end{bmatrix} \quad (18)$$

Definiéndose los términos auxiliares  $\eta_{ij}$ :

$$\begin{cases} \eta_{ij} = {}^{n+1-j} \ddot{\omega}_{n+1-j} & i = j \\ \eta_{ij} = {}^{n+1-i} R_{n+2-i} \cdot \eta_{i-1,j} & i > j \end{cases} \quad (19)$$

Así, las matrices  $B1_{ij}$  se podrán calcular como:

$$\begin{cases} B1_{ij} = 0 & i < j \\ B1_{ij} = {}^{n+1-j} \ddot{r}_{O_{n+1-j}} & i = j \\ B1_{ij} = {}^{n+1-i} R_{n+2-i} B1_{i-1,j} - {}^{n+1-i} \ddot{r}_{O_{n+1-i} O_{n+2-i}} \eta_{ij} & i > j \end{cases} \quad (20)$$

Para obtener la matriz  $B2$  se utilizará la expresión:

$$B2 = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ {}^{n-1} \vec{h}_n & 0 & \cdots & 0 & 0 \\ {}^{n-2} \vec{h}_n & {}^{n-2} \vec{h}_{n-1} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ {}^2 \vec{h}_n & {}^2 \vec{h}_{n-1} & \cdots & 0 & 0 \\ {}^1 \vec{h}_n & {}^1 \vec{h}_{n-1} & \cdots & {}^1 \vec{h}_2 & 0 \end{bmatrix} \quad (21)$$

Definiendo el término auxiliar  $\delta_{ij}$ :

$$\begin{cases} \delta_{ij} = {}^{n+1-i} \ddot{r}_{O_{n+1-i} O_{n+2-i}} & i = j+1 \\ \delta_{ij} = {}^{n+1-i} \ddot{r}_{O_{n+1-i} O_{n+2-i}} {}^{n+1-i} R_{n+2-i} + {}^{n+1-i} R_{n+2-i} \delta_{i-1,j} & i > j+1 \end{cases} \quad (22)$$

Dichas matrices se calculan como:

$$\begin{cases} B_{2ij} = 0 & i \leq j \\ B_{2ij} = -\delta_{ij}^{n-j} R_{n+1-j}^{n+1-j} \ddot{q}_{n+1-j} & i > j \end{cases} \quad (23)$$

Una vez obtenida la matriz de observación  $W$  únicamente se necesitan conocer los diferentes parámetros dinámicos que aparecen en la ecuación dinámica del robot. En (Valera *et al.*, 2007), se presentan dos métodos para realizar la identificación de dichos parámetros. El primero está basado en el método de identificación directa propuesto en (Gautier, 1991), en el que se han utilizado unas trayectorias optimizadas para realizar la identificación. Estas trayectorias han sido parametrizadas por medio de series finitas de Fourier como las descritas en (Swevers, *et al.*, 1997). En este método se identifica completamente el modelo dinámico del robot con un único experimento.

El segundo método utilizado es una combinación entre los esquemas de identificación directo e indirecto. En este caso, la identificación se realiza en dos pasos. En el primero se realizan experimentos específicos para identificar los coeficientes del modelo de fricción y los parámetros de gravedad. En el segundo paso se identifica el resto del modelo dinámico del robot, esto es, los parámetros inerciales.

Una descripción más detallada de los métodos de identificación propuestos, así como un análisis comparativo de los resultados obtenidos puede encontrarse en (Benimeli, *et al.*, 2005) y (Mata, *et al.*, 2005).

Para verificar estos métodos de identificación se han desarrollado varias identificaciones experimentales. Los experimentos se realizaron tanto como simulaciones como con un robot manipulador industrial: un PUMA 560. Se han generado varias trayectorias de movimiento optimizadas, para las seis articulaciones del robot, basadas en series finitas de Fourier. Utilizando un controlador PID se ha establecido el control del robot, almacenándose tanto la posición como la acción de control (los pares de articulación) reales del robot. A partir de estos datos se ha generado la matriz de observación del sistema y, a partir de dicha matriz y del vector de fuerzas generalizadas se ha obtenido, por último, el vector de los parámetros base del sistema.

Para realizar la validación de la identificación realizada se ha utilizado una trayectoria de movimiento distinta a la empleada en el procedimiento de identificación, obteniéndose la respuesta del robot real y del modelo dinámico descrito mediante los parámetros base identificados. La Figura 1 muestra el esbozo del procedimiento experimental utilizado para validar el modelo dinámico propuesto para el robot industrial.

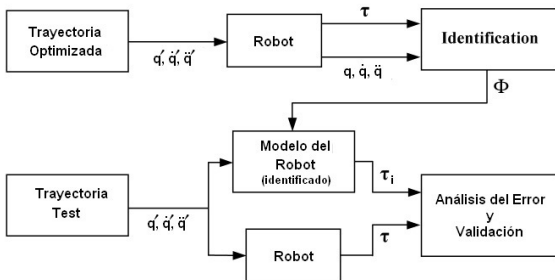


Figura 1. Esquema de validación utilizado.

La Figura 2 muestra los pares para los 6 ejes del PUMA de uno de los experimentos realizados. En línea continua se muestran los pares medidos en el robot real y, en línea discontinua, se muestran los pares estimados por el modelo del robot obtenido.

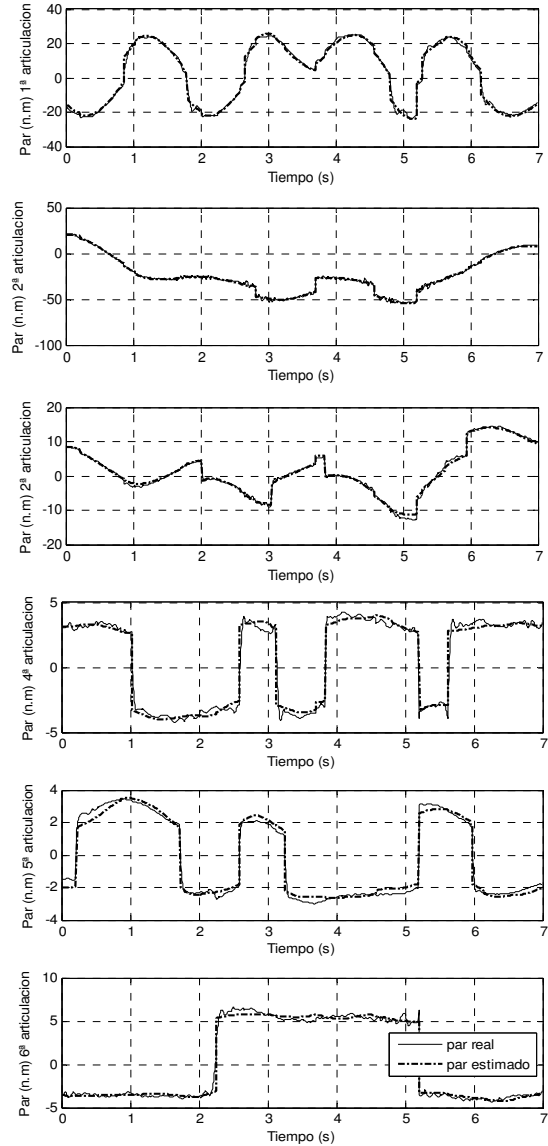


Figura 2. Pares medidos y calculados del robot PUMA.

#### 4. CÁLCULO E IMPLEMENTACIÓN DEL PROBLEMA DINÁMICO DEL ROBOT EN TIEMPO REAL

Como se ha visto en el apartado anterior, a partir de la matriz de observación del sistema y de la identificación del vector base de los parámetros inerciales se puede obtener el vector de fuerzas generalizadas del sistema:

$$W(\ddot{q}, \dot{q}, q, t) \cdot \ddot{\Phi} = \ddot{\tau}(\ddot{q}, \dot{q}, f, \ddot{\Phi}, t) \quad (24)$$

A partir de esta expresión se puede plantear el cálculo de la ecuación dinámica del sistema robotizado:

$$M(\ddot{q}) \cdot \ddot{q} + C(\ddot{q}, \dot{q}) \cdot \dot{q} + G(\ddot{q}) = \ddot{\tau} \quad (25)$$

donde  $M(\ddot{q})$  es la matriz de masas del sistema,  $C(\ddot{q}, \dot{q})$  es el vector de fuerzas centrífugas y de Coriolis y  $G(\ddot{q})$  son los términos gravitacionales.

Para poder resolver el problema dinámico inverso del sistema robotizado se ha desarrollado un algoritmo secuencial inspirado en los algoritmos propuestos por Walter y Orin (Walker y Orin, 1982). Se puede comprobar que los términos gravitacionales  $G(\vec{q})$  dependen únicamente de las coordenadas generalizadas  $\vec{q}$ . Por ello, se pueden obtener estos términos inicializando a cero las velocidades y aceleraciones generalizadas ( $\dot{\vec{q}}=0$  y  $\ddot{\vec{q}}=0$ ) en la matriz de observación  $\mathbf{W}$  de la ecuación (24):

$$\mathbf{W}(\vec{q}, 0, 0) \cdot \vec{\Phi} = \vec{G}(\vec{q}) \quad (26)$$

Para hallar los términos centrífugos y de Coriolis  $C(\vec{q}, \dot{\vec{q}})$  se puede observar que éstos dependen tanto de las coordenadas como de las velocidades generalizadas. Por ello, si se inicializan a cero las aceleraciones generalizadas en la ecuación (24):

$$\mathbf{W}(\vec{q}, \dot{\vec{q}}, 0) \cdot \vec{\Phi} = \vec{C}(\vec{q}, \dot{\vec{q}}) + \vec{G}(\vec{q}) \quad (27)$$

De esta forma, se puede obtener  $C(\vec{q}, \dot{\vec{q}})$  restando directamente las ecuaciones (26) y (27):

$$(\mathbf{W}(\vec{q}, \dot{\vec{q}}, 0) \cdot \vec{\Phi} - \vec{G}(\vec{q})) = \vec{C}(\vec{q}, \dot{\vec{q}}) \quad (28)$$

El último término de la ecuación dinámica del robot que falta es la matriz de masas del sistema. Éste depende, además del vector que agrupa los parámetros inerciales  $\vec{\Phi}$ , de la geometría de las coordenadas generalizadas  $\vec{q}$  empleadas para modelizar el mecanismo. Para obtener dicha matriz se pueden utilizar los términos gravitacionales y de Coriolis calculados previamente:

$$\mathbf{M}(\vec{q}) \cdot \ddot{\vec{q}} = \vec{\tau} - \vec{G}(\vec{q}) - \vec{C}(\vec{q}, \dot{\vec{q}}) \quad (29)$$

Sin embargo, hay que tener en cuenta que  $\mathbf{M}(\vec{q})$  es una matriz  $n \times n$  (6x6 en el caso del robot PUMA) y no un vector. Por ello, a diferencia de lo que ocurría con los términos gravitacionales y de Coriolis, no se puede calcular directamente. Para poder solucionar el problema se pueden obtener las columnas de forma independiente mediante la expresión:

$$\begin{bmatrix} m_{11} & m_{12} & \cdots & m_{16} \\ \vdots & & \ddots & \vdots \\ m_{61} & m_{62} & \cdots & m_{66} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} m_{11} \\ m_{21} \\ \vdots \\ m_{61} \end{bmatrix} \quad (30)$$

Así, para poder obtener la matriz completa lo que se hace es calcular la columna  $i$ -ésima de dicha matriz, uniéndolas posteriormente:

$$(\mathbf{W}(\vec{q}, 0, \vec{e}_i) \cdot \vec{\Phi} - \vec{G}(\vec{q})) = \vec{M}_i(\vec{q}) \quad (31)$$

siendo  $\vec{M}_i(\vec{q})$  la columna  $i$  de la matriz de inercias y  $\vec{e}_i = [0 \dots 1 \dots 0]^T$  un vector columna donde la posición  $i$  será 1 e indicará la columna a hallar.

A fin de desarrollar el problema dinámico inverso y poder validar el algoritmo propuesto, se ha utilizado la aplicación *Real-Time Workshop* de *Matlab*, que permite la implementación automática de programas ejecutables en tiempo real a partir de esquemas *Simulink*.

De esta forma, se han programado las funciones para obtener los términos gravitacionales de la ecuación (26), los términos de Coriolis de la ecuación (28) y los términos inerciales de la ecuación (31). A partir de estas funciones se puede obtener la ecuación dinámica del sistema robotizado (25). La programación de los términos dinámicos se realiza a partir de la matriz de observación  $\mathbf{W}$  de los parámetros dinámicos identificados del sistema robotizado  $\vec{\Phi}$ .

El cálculo de  $\mathbf{W}$  se realiza tal y como se ha mostrado en el apartado 3 y los parámetros dinámicos utilizados se obtuvieron previamente con un procedimiento fuera de línea. El cálculo de  $\mathbf{W}$  se realiza a partir de la ecuación (12) mediante el producto de dos matrices:  $\mathbf{D}$  y  $\mathbf{AB}$ . El cálculo de  $\mathbf{D}$  no tiene coste computacional asociado porque es simplemente una matriz cuyos elementos son 0 ó 1. La matriz  $\mathbf{AB}$  se calcula a partir de las matrices  $\mathbf{A}$  y  $\mathbf{B}$ . El cálculo de las submatrices que forman  $\mathbf{A}$  se realiza a partir de la ecuación (16), y son necesarios:

$$\begin{aligned} \text{Sumas} & \begin{cases} 9n & i=j \\ 12 \frac{n(n-1)}{2} & i>j \end{cases} \\ \text{Productos} & \begin{cases} 6n & i=j \\ 27 \frac{n(n-1)}{2} & i>j \end{cases} \end{aligned}$$

Por lo tanto, para la construcción de la matriz  $\mathbf{A}$  se necesitan  $12 \frac{n(n-1)}{2} + 9n$  sumas y  $27 \frac{n(n-1)}{2} + 6n$  productos.

La matriz  $\mathbf{B}$  se compone de las matrices  $\mathbf{B1}_{ij}$  y  $\mathbf{B2}_{ij}$ . Para calcular la primera matriz se necesitan los términos auxiliares  $\eta_{ij}$  de la ecuación (19). La complejidad de estos términos es:

$$\begin{aligned} \text{Sumas} & \begin{cases} 6n & i=j \\ 18 \frac{n(n-1)}{2} & i>j \end{cases} \\ \text{Productos} & \begin{cases} 6n & i=j \\ 18 \frac{n(n-1)}{2} & i>j \end{cases} \end{aligned}$$

A partir de los  $\eta_{ij}$  se pueden obtener las matrices  $\mathbf{B1}_{ij}$  según la ecuación (20). El coste computacional de este cálculo (para  $i>j$ ) es  $18n(n-1)$  sumas y  $21n(n-1)$  productos. De esta forma, el coste total del cálculo de las matrices  $\mathbf{B1}$  son  $27n(n-1) + 6n$  sumas y  $30n(n-1) + 6n$  productos.

Las matrices columna  $\mathbf{B2}_{ij}$  necesita los términos auxiliares  $\delta_{ij}$ . El coste de este cálculo es el siguiente:

$$\begin{aligned} \text{Sumas} & \begin{cases} 0 & i=j \\ 26 \frac{(n-1)(n-2)}{2} & i>j \end{cases} \\ \text{Productos} & \begin{cases} 0 & i=j \\ 30 \frac{(n-1)(n-2)}{2} & i>j \end{cases} \end{aligned}$$

El cálculo de las matrices  $\mathbf{B2}_{ij}$  se define en la ecuación (23) y tiene el coste siguiente:

$$\text{Sumas} \begin{cases} 0 & i = j \\ 24 \frac{n(n-1)}{2} & i > j \end{cases}$$

$$\text{Productos} \begin{cases} 0 & i = j \\ 36 \frac{n(n-1)}{2} & i > j \end{cases}$$

La obtención de la matriz  $B$  requiere, por lo tanto  $25(n-1)(n-2)$  sumas y  $33(n-1)(n-2)$  productos.

De esta forma, el coste computacional total de la matriz  $W$  es:

$$\begin{cases} 58n^2 - 93n + 50 & \text{Sumas} \\ \frac{153}{2}n^2 - \frac{261}{2}n + 66 & \text{Productos} \end{cases}$$

En el caso del robot PUMA 560 con el que se ha trabajado, como tiene 6 grados de libertad, el coste computacional del cálculo de la matriz  $W$  es 1.580 sumas y 2.037 productos, un coste completamente asumible para su implementación en tiempo real con cualquiera de los procesadores actuales.

La Figura 3 muestra el esquema de *Simulink* con la aplicación *Real-Time Workshop* con el que se obtiene el problema dinámico inverso del sistema robotizado. El esquema toma como entradas la posición, velocidad y aceleración del robot de uno de los experimentos reales realizados con el robot PUMA 560, calculando el par de las 6 articulaciones a partir de la ecuación dinámica y los parámetros base identificados. El periodo de muestreo utilizado para la resolución del problema dinámico inverso es de 5ms, que es el mismo que se utilizó en la etapa de identificación del apartado 3.

La implementación de las funciones de los términos dinámicos de gravedad ( $G(q)$ ), fricción ( $C(q, \dot{q})$ ) e inerciales ( $M(q)$ ) se ha realizado mediante los bloques “*Embedded MATLAB Function*” de *Simulink*. Se ha elegido esta opción porque éstos permiten obtener, gracias a la *Real-Time Workshop*, programas ejecutables en tiempo real a partir de rutinas estándar “.m” de *Matlab*, lo que permite reutilizar directamente el trabajo previo que se tenía para realizar las simulaciones.

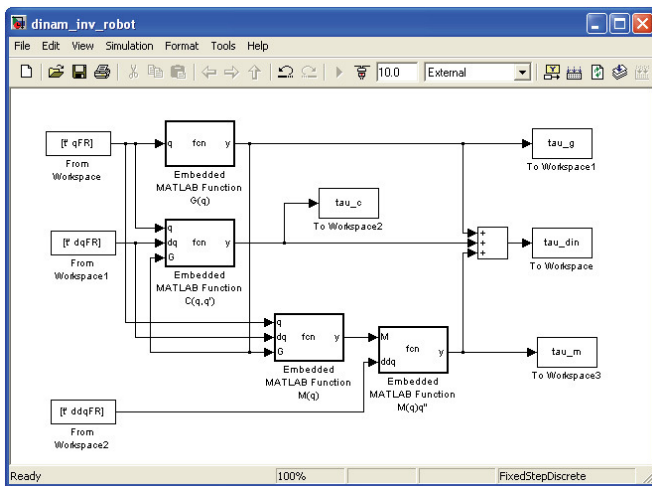


Figura 3. Esquema Real-Time Workshop/Simulink para la resolución del problema dinámico inverso del sistema robotizado.

La Figura 4 muestra el par real leído del robot PUMA 560 y el calculado a partir de la ecuación dinámica del robot de la 1ª articulación del robot. La Figura 5 muestra el error entre el par real y el calculado. Se puede observar que, a pesar de que en algunos instantes de tiempo dicho error alcanza valores cercanos a  $\pm 10$  N.m, su valor medio es -0.088, el cual puede considerarse aceptable y validar de este modo el procedimiento de identificación y resolución del problema dinámico del robot. Los picos de error son debidos al modelo de fricción lineal asimétrico utilizado y, en concreto, a su discontinuidad ante un cambio en signo de la velocidad. Mediante un modelo de fricción más ajustado, se podría mejorar la respuesta. La Figura 6 muestra la aportación que tienen los términos gravitacionales, centrífugos y de Coriolis e inerciales en el par calculado mediante la resolución del problema dinámico inverso. Las Figuras 7, 8 y 9 muestran los valores correspondientes a la 2ª articulación.

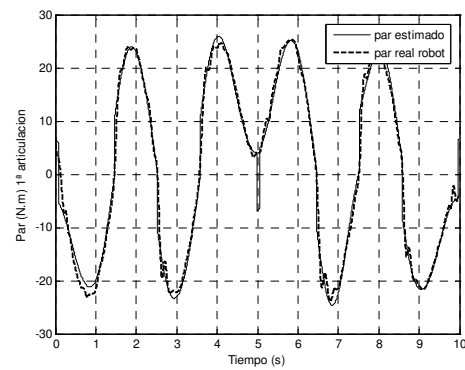


Figura 4. Par real y calculado de la 1ª articulación del PUMA.

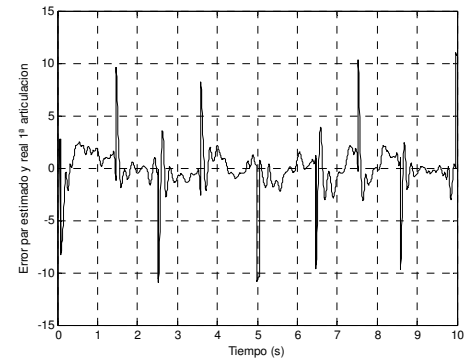


Figura 5. Error entre el par real y calculado de la 1ª articulación del PUMA.

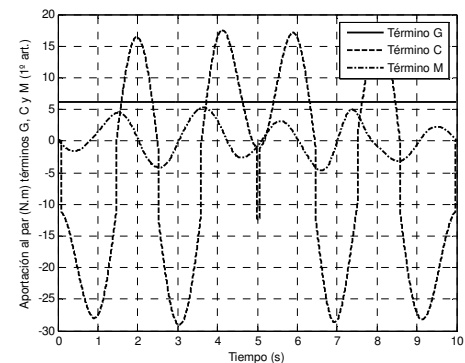


Figura 6. Aportación de los términos dinámicos de gravedad, Coriolis e inerciales al par de la 1ª articulación del PUMA.

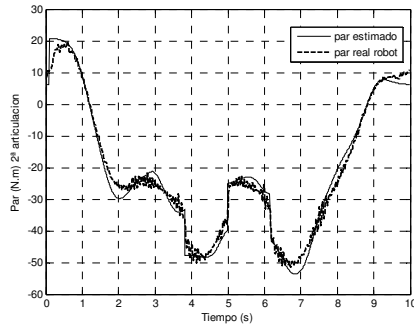


Figura 7. Par real y calculado de la 2ª articulación del PUMA.

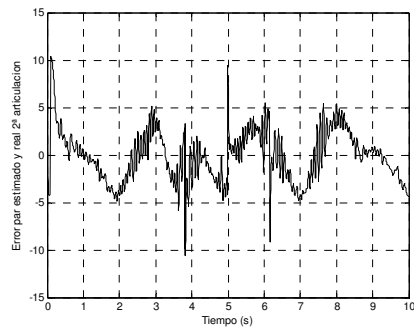


Figura 8. Error entre el par real y calculado de la 2ª articulación del PUMA.

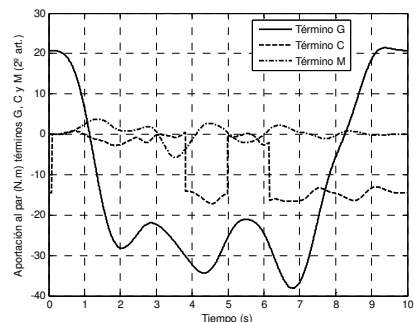


Figura 9. Aportación términos dinámicos de gravedad, Coriolis e inerciales al para de la 2ª articulación del PUMA.

Por último, en este trabajo también se ha planteado y resuelto el problema dinámico directo de robots. Éste se puede obtener de una forma análoga a la planteada para la resolución del problema dinámico inverso. En este caso se parte de la ecuación (25) y se despeja el término de la aceleración generalizada. De esta forma, la ecuación dinámica que se obtiene es la siguiente:

$$\ddot{\bar{q}} = M(\bar{q})^{-1} (\bar{\tau} - C(\bar{q}, \dot{\bar{q}}) \cdot \dot{\bar{q}} - G(\bar{q})) \quad (32)$$

##### 5. CONTROL DINÁMICO DEL ROBOT PUMA 560 EN TIEMPO REAL

Una vez resueltos el problema dinámico directo e inverso y obtenido y validado los parámetros dinámicos del robot, se puede abordar el control en tiempo real del sistema robotizado. En este trabajo, se han implementado varias estrategias de control basadas en el método de dinámica inversa (o par calculado). Este tipo de control se analiza con mayor detalle en diversos textos como, por ejemplo, (Craig, 1989), (Spong, Vidyasagar, 1989), (Yoshikawa, 1990) o (Canudas, *et al.*, 1996).

Las estrategias de control implementadas vienen dadas por la expresión general siguiente:

$$\tau_c = M(q)a + C(q, \dot{q})\dot{q} + G(q) \quad (33)$$

donde

$\tau_c$ : par generalizado de control

$M(q)$ ,  $C(q, \dot{q})\dot{q}$ ,  $G(q)$ : matriz de inercias, vector de Coriolis y fuerzas gravitacionales del sistema robotizado

$a$ : acción de control lineal

En la ecuación (33) se puede observar como el controlador cancela los términos no lineales de la ecuación dinámica del robot (los términos de Coriolis y gravitacionales) sumando dichas fuerzas a la acción de control. Una vez canceladas el sistema se convierte en un sistema lineal simple de entrada-salida.

$$\ddot{q} = a \quad (34)$$

Tal como se muestra en la Tabla 1, dependiendo de la expresión que se utilice para el término  $a$ , se pueden obtener diferentes estrategias de control.

**Tabla 1. Acción de control lineal  $a$  de los controladores por dinámica inversa**

Controlador	Acción de control lineal $a$
Punto a punto	$-K_d \dot{q} - K_p e$
Control trayectoria	$\ddot{q}_d - K_d \dot{e} - K_p e$
Control Trayectoria con acción integral	$\ddot{q}_d - K_d \dot{e} - K_p e - K_i \int_0^t e(u) du$

donde  $e = q - q_d$  y  $K_d$ ,  $K_p$  y  $K_i$  son las matrices (diagonales y definidas positivas) de ganancias derivativas, proporcionales e integradoras.

El primero es un controlador punto a punto exponencialmente estable con una selección adecuada de las matrices  $K_d$  y  $K_p$ . El segundo controlador es similar al primero, pero en este caso se establece un control de trayectoria en el que la respuesta del robot debe seguir una trayectoria variable  $q_d(t)$  y sus sucesivas derivadas que describen la velocidad  $\dot{q}_d(t)$  y la aceleración  $\ddot{q}_d(t)$  deseada. El tercer controlador es también un controlador de trayectoria en el que se ha añadido un término integral del error.

Estas estrategias se han implementado para establecer el control en tiempo real del robot industrial PUMA 560, robot que se muestra en la Figura 10.



Figura 10. Robot industrial PUMA.560 utilizado.

Para poder acceder a las variables de estado del sistema se ha sustituido la unidad de control original por una unidad nueva basada en un computador personal industrial equipado con un procesador Intel Pentium IV a 2.66GHz. La Figura 11 muestra la arquitectura de control utilizada.

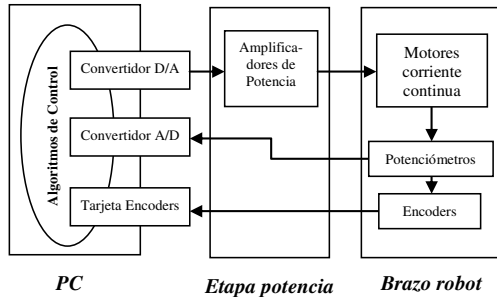


Figura 11. Arquitectura abierta de control del PUMA.560.

En la nueva unidad de control desarrollada las lecturas de las posiciones angulares de los nudos del brazo robot se obtienen a través de las tarjetas de los encoders incrementales PCL-833 de Advantech. Cabe destacar que, puesto que se trata de encoders incrementales, es necesario programar el valor inicial de partida del robot en el momento de la inicialización del sistema. Esto se realiza de forma automática gracias a los convertidores analógico/digital PCI-1711 de Advantech, que leen el valor de los potenciómetros de cada una de las articulaciones del robot del PUMA 560.

Con la información del estado del sistema, las referencias de movimiento y el algoritmo de control programados, el PC calcula las acciones de control necesarias para las distintas articulaciones del robot. Estas acciones de control se suministran a los amplificadores de potencia de los motores mediante convertidores Digital/Analógico, utilizándose en este trabajo las tarjetas PCL-726 de Advantech. Una información más detallada de la arquitectura desarrollada para el control de este robot se puede encontrar en (Valera, *et al.*,2003).

La Figura 12 muestra uno de los esquemas de la *Real-Time Workshop* utilizados para la generación automática del controlador en tiempo real. Se trata de un controlador punto a punto en el que se pueden apreciar los bloques para el acceso a la posición de las 6 articulaciones del robot (*Tarjeta A* y *Tarjeta B* de los encoders) y el bloque de los convertidores D/A (*Analog Output*). Como se ha comentado anteriormente, los bloques *Embedded Matlab Function G*, *Embedded Matlab Function C* y *Embedded Matlab Function M* se encargan de obtener los términos gravitacionales, de Coriolis e inerciales de la ecuación dinámica del robot a partir de la resolución de la matriz  $W$ .

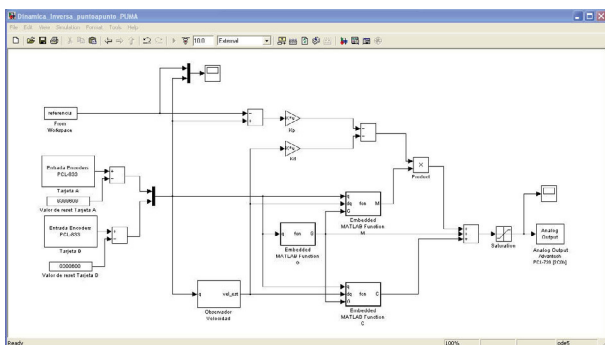


Figura 12. Controlador por dinámica inversa punto a punto del robot industrial PUMA.560.

Como en las fases de identificación del apartado 3 y de la resolución del problema dinámico del apartado 4, el periodo de muestreo utilizado para establecer el control del robot industrial es de 5ms. En las pruebas y las ejecuciones realizadas se ha comprobado que este periodo es suficientemente grande puesto que el tiempo medio de ejecución del controlador (que incluye las lecturas de los encoders, el cálculo del controlador y las conversiones Digital/Analógico) que se obtuvo fue 3.55ms, con un tiempo de ejecución máximo de 3.57ms.

Las figuras siguientes muestran una ejecución real del robot industrial PUMA 560. En la Figura 13 se muestran la referencia (línea continua) y la posición real (línea discontinua) para las 6 articulaciones del robot. La Figura 14 muestra los errores cometidos.

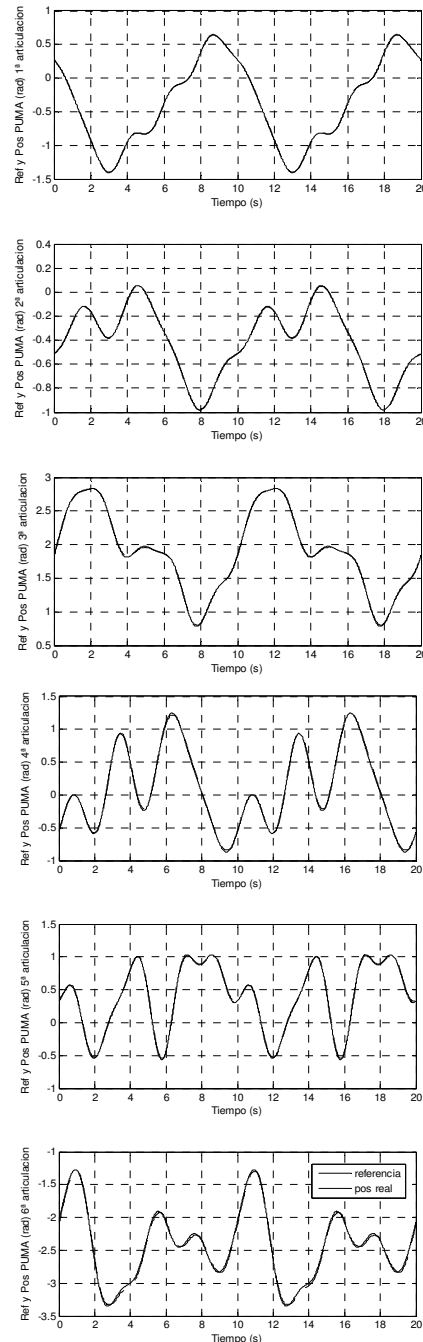


Figura 13. Referencias y posiciones reales de las seis articulaciones del robot industrial PUMA.560.

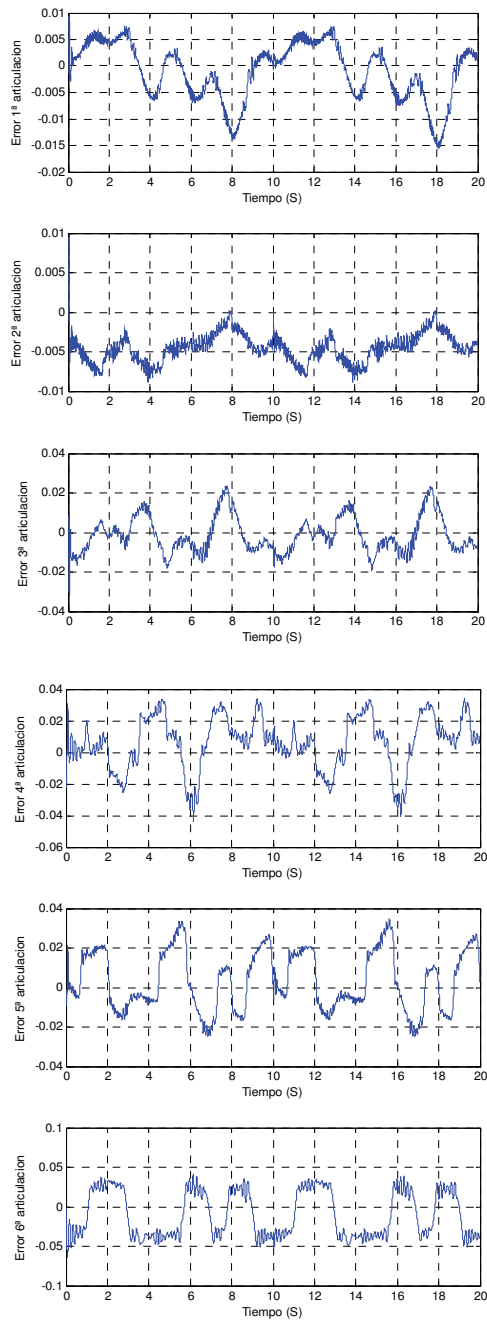


Figura 14. Error referencias y posiciones reales del robot.

La Tabla 2 muestra el valor medio y la media cuadrática (RMS) del error entre la referencia y la posición real de cada articulación.

**Tabla 2. Error (medio y RMS) de posición de las articulaciones del robot industrial PUMA 560**

Articulación	$\frac{\sum e_i}{n}$	$\sqrt{\frac{\sum e_i^2}{n}}$
1	-0.00095	0.0054
2	-0.00452	0.0048
3	-0.00141	0.0092
4	0.00625	0.0177
5	0.00363	0.0154
6	-0.00979	0.0310

Por último, la Figura 15 muestra las acciones de control aplicadas sobre las articulaciones del robot.

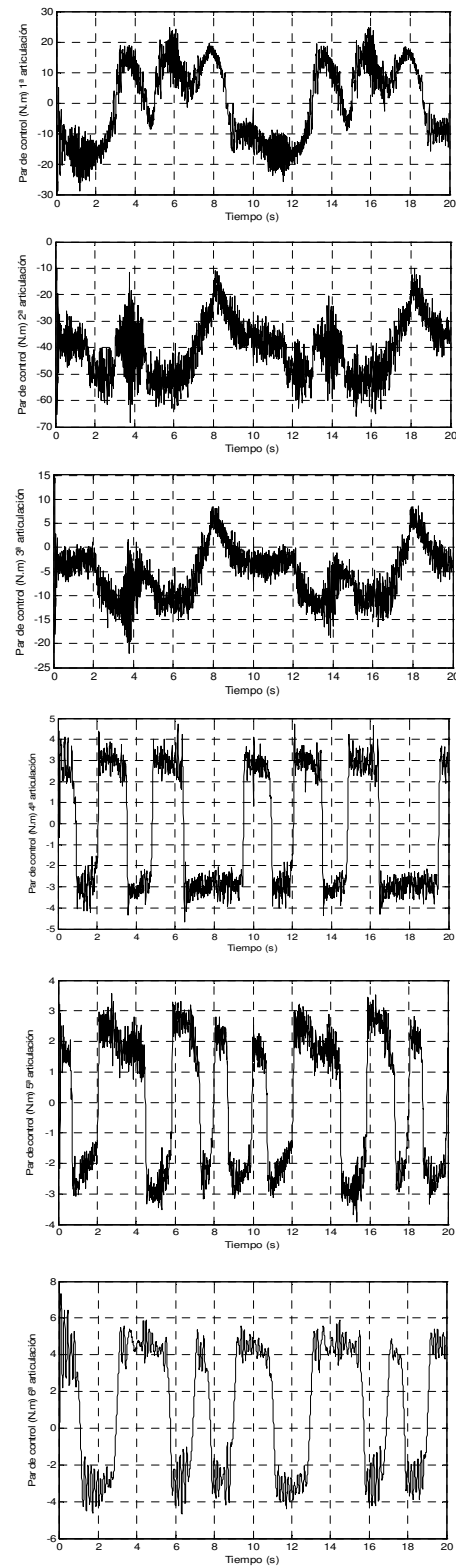


Figura 15. Acciones de control aplicadas en el robot.

## 6. CONCLUSIÓN

Este artículo ha abordado la resolución del problema dinámico directo e inverso de sistemas robotizados y su aplicación para el establecimiento del control de éstos. El modelo dinámico del robot se ha desarrollado a partir de las ecuaciones dinámicas de Gibbs-Appell, que proporcionan características muy interesantes como la simplicidad de las expresiones que se obtienen y la fácil ampliación del modelo dinámico para poder incorporar efectos como la dinámica de los actuadores o el fenómeno de la fricción. Las ecuaciones del modelo dinámico se pueden expresar en forma lineal con respecto a los parámetros dinámicos del sistema, utilizando una formulación matricial.

El artículo ha mostrado cómo se puede resolver el problema dinámico de robots. Partiendo de la matriz de observación del sistema y de los parámetros dinámicos del robot identificados fuera de línea, se pueden obtener los términos que componen su ecuación dinámica. La implementación del problema dinámico se realiza mediante la aplicación *Real-Time Workshop*, utilizada en este trabajo para realizar la simulación y establecer el control de un robot real: el robot industrial PUMA 560.

Como trabajos futuros, se plantea incorporar la dinámica de los rotores en el modelo del robot y analizar qué efecto tiene en la mejora de la precisión del sistema. Además, también se está planteando la utilización de este modelo para otras configuraciones de robots, como el caso de los robots paralelos.

## AGRADECIMIENTOS

Los autores desean expresar su agradecimiento al Plan Nacional de I+D, Comisión Interministerial de Ciencia y Tecnología (FEDER-CICYT) por la financiación parcial de este trabajo bajo los proyectos DPI2008-06737-C02-01y DPI2009-13830-C02-01.

## REFERENCIAS

- J. Angeles, O. Ma, A. Rojas (1989). An Algorithm for Inverse Dynamics of n-Axis General Manipulators Using Kane's Equations. *Computer Math. Applic.*, **vol. 17**, pp. 1545-15561.
- F. Benimeli, V. Mata, F. Valero (2005). *A Comparison between Direct and Mixed Dynamic Parameter Identification Methods in Mechanical Systems*, presented at MULTIBODY DYNAMICS 2005, ECCOMAS Thematic Conf., Madrid, Spain.
- C. Canudas, B. Siciliano, G. Basting (1996). *Theory of Robot Control*. Ed. Springer
- J. Craig (1989). *Introduction to robotics: Mechanics and control*. Ed. Addison-Wesley
- K. Desoyer, P. Lugner (1989). Recursive Formulation for the Analytical or Numerical Application of the Gibbs-Appell Method to the Dynamics of Robots", *Robotica*, **vol. 7**, pp. 343-347.
- M. Gautier (1991). Numerical-Calculation of the Base Inertial Parameters of Robots, *Journal of Robotic Systems*, **vol. 8**, pp. 485-506, 1991.
- M. Grotjahn, M. Daemi, B. Heimann (2001). Friction and rigid body identification of robot dynamics, *International Journal of Solids and Structures*, **vol. 38**, pp. 1889-1902.
- W. Hooker, G. Magulies (1965), The Dynamical Attitude Equations for an n-Body Satellite, *J. of Astronautical Science*, **vol. 12**, pp. 123-128.
- T. R. Kane, D. A. Levinson (1983). The use of Kane's Dynamical Equations in Robotics, *Int. J. of Robotics Research*, **vol. 2**, pp. 3-21.
- Z. R. Lu, K. B. Shimoga, A. A. Goldenberg (1993). Experimental-Determination of Dynamic Parameters of Robotic Arms, *Journal of Robotic Systems*, **vol. 10**, pp. 1009-1029.
- J. Y. S. Luh, M. W. Walker, R. P. Paul (1980). On-line Computational Scheme for Mechanical Manipulators. *J. of Dynamical Systems, Measurement, and Control*, **vol. 102**, pp. 69-79.
- V. Mata, F. Benimeli, N. Farhat, A. Valera (2005). Dynamic Parameter Identification in Industrial Robots Considering Physical Feasibility, *Advanced Robotics*, **vol. 19**, pp. 101-119.
- D. E. Orin, R. B. McGhee, M. Vukobratovic, G. Hartoch (1980). Kinematic and Kinetic Analysis of Open-Chain Linkages utilising Newton-Euler Methods. *J. of Dynamic Systems, Measurements, and Control*, **vol. 102**, pp. 69-79.
- S. Provenzano (2001). *Aplicación de las Ecuaciones de Gibbs-Appell a la Dinámica de Robots*, Ed. Universitat Politecnica de Valencia.
- M. Renaud (1983). An Efficient Iterative Analitical Procedure for obtaining a Robot Manipulator Dynamic Model, *Proc. 1st Int. Symp. On Robotic Research*, New Hampshire, pp. 749-762.
- M. Spong, M. Vidyasagar (1989). *Robot dynamics and control*. Ed. John Wiley & Sons.
- Y. Stepanenko, M. Vukobratovic (1976). Dynamics of Articulated Open-Chain Active Mechanisms, *Matematical Biosciences*, **vol. 28**, pp. 137-170.
- J. Swevers, C. Ganseman, D. B. Tukel, J. DeSchutter, H. VanBrussel (1997). Optimal robot excitation and identification, *IEEE Transactions on Robotics and Automation*, **vol. 13**, pp. 730-740.
- J. J. Uicker (1965). *On Dynamic Analysis of Spatial Linkages using 4x4 Matrices*. Ph.D. Dissertation, Northwestern University.
- A. Valera, V. Mata, M. Valles, F. Valero, N. Rosillo, F. Benimeli (2003). Solving the Inverse Dynamics Control for Low Cost Real-Time Industrial Robot Control Applications, *Robotica*, **vol. 21**, pp. 261-269, 2003.
- A. Valera, F. Benimelli, V. Mata, F. Valero, N. Rosillo (2007). Modelling and Dynamic Parameter Identification in Robots. Applications to the Control of an Industrial Robot. *Int. J. of Factory Automation, Robotics and Soft Computing*, pp 44-51.
- M. Vukobratovic, N. Kircanski (1985). *Real Time Dynamics of Manipulation Robots*, Springer-Verlag, Berlin, 1985.
- M. W. Walker, D.E. Orin (1982). Efficient Dynamic Computer Simulation of Robotic Manipulators. *ASME Journal of Dynamic Systems, Measurement, and Control*, **vol. 104**, pp. 205-211.
- T. Yoshikawa (1990). *Foundations of Robotics: Analysis and control*. Ed. The MIT Press