

IDENTIFICACIÓN Y CONTROL DE UN ROBOT MANIPULADOR

F. Dreoni, M.G. Ortega, F.R. Rubio
Depto. Ingeniería de Sistemas y Automática
Escuela Superior de Ingenieros. Universidad de Sevilla
e-mail: {dreoni,ortega,rubio}@cartuja.us.es

J. Gómez-Ortega
Área de Ingeniería de Sistemas y Automática. Depto. Electrónica
Escuela Politécnica Superior. Universidad de Jaén
e-mail: juango@ujaen.es

Resumen

El objetivo de este artículo es exponer el desarrollo e implementación de algunas técnicas avanzadas de control para manipuladores industriales. El equipo de experimentación es un robot industrial de seis grados de libertad controlado por ordenador. La gestión de funciones auxiliares necesarias para el manejo en tiempo real del robot se trata de forma separada a la implementación de controladores en sí, facilitando así las pruebas de nuevas estrategias de control. Con la idea de simplificar el proceso de implementación de controladores, éstos se programan utilizando diagramas de bloques de SIMULINK. Posteriormente se procede a la compilación del programa y se carga la aplicación en una tarjeta en tiempo real de la compañía dSPACE, la cual se encarga de adquirir las medidas de posición y de aplicar las señales de control al manipulador. Mediante software es posible: supervisar el control implementado, modificar sus parámetros en tiempo real y seguir la evolución de variables relevantes del sistema. En este trabajo se ha enfrentado el problema de la identificación del comportamiento dinámico de un manipulador y se han desarrollado dos controladores basados en la teoría del control H_∞ , comparando los resultados con controladores más tradicionales.

Palabras Clave: Robot manipulador, identificación, control movimiento, control robusto.

1 INTRODUCCIÓN

Hoy en día la potencia de cálculo de los ordenadores permite plantear algoritmos de control mucho más complejos y optimizados, lo cual ha contribuido al desarrollo de nuevas metodologías de control. De especial interés es el campo de la robótica, donde es frecuente enfrentarse con comportamientos no lineales y con sistemas afectados por un

acoplamiento entre los ejes no despreciable. Además, el requerimiento de altas prestaciones y desarrollo de manipuladores con acoplamiento directo de los motores en los ejes requieren nuevas estrategias de control capaces de atender a estas especificaciones.

En este trabajo se va a considerar una de dichas estrategias basada en la teoría de control H_∞ no lineal, añadiendo además un aporte para la robustificación del sistema que se oponga a los efectos de la incertidumbre de la dinámica del manipulador. El sistema utilizado se basa en una representación del controlador en un diagrama de bloques que se almacena en una tarjeta de control en tiempo real instalada en un PC, dotado con la herramienta SIMULINK de MATLAB.

El robot empleado para la experimentación es el brazo manipulador RM-10 para entornos industriales [6]. Este robot tiene seis grados de libertad, todos ellos de rotación y con accionamiento indirecto con motores sin escobillas. El robot disponía originalmente de controladores PID para cada motor, de los que se ha prescindido considerando el objetivo del trabajo. En cambio sí se ha conservado la etapa de potencia y los sensores resolver de posición. Además de las no linealidades en las ecuaciones dinámicas del robot (debidas a acoplamiento entre los distintos enlaces), el robot RM-10 presenta fenómenos de fricción en los ejes, lo cual motiva su utilización como equipo de desarrollo de controladores.

La organización del resto del artículo es la siguiente: en la sección 2 se describe la metodología utilizada para la identificación de los efectos gravitatorios dinámicos y de fricción y cómo se ha logrado simplificar el problema dividiéndolo en partes más simples. En la sección 3 se menciona el entorno de trabajo utilizado y cómo ha sido posible simplificar el proceso de implementación de los controladores utilizando los diagramas de bloques de SIMULINK y la herramienta *Real Time Workshop* de MATLAB. En la sección 4 se describe someramente la teoría del

controlado H_∞ y un método para su robustificación. En la sección 5 se presentan los resultados de los ensayos realizados para probar los controladores, y finalmente, en la sección 6 se exponen las conclusiones del artículo.



Figura 1: Robot industrial RM-10.

2 IDENTIFICACIÓN DEL SISTEMA

La estimación de los efectos gravitatorios, dinámicos y de fricción es de fundamental importancia si se quiere desarrollar controladores avanzados para manipuladores. En la mayoría de estos controladores hay presente una componente predictiva que estima el par que necesita el manipulador, una vez conocida la referencia que se proporciona. Hay muchos métodos de plantear la identificación de un robot manipulador, siendo uno de los más conocidos los que utilizan regresiones lineales en los parámetros dinámicos [5]. Generalmente, estos parámetros son los elementos de los tensores de inercia, la posición del centro de masa y la masa de cada eslabón del sistema, referidos en un sistema de coordenadas solidario al enlace del mismo. Además se pueden añadir parámetros de fricción, como se ha realizando en este trabajo.

Normalmente la regresión se hace a partir de la medida de los pares aplicados por los motores, midiendo también posiciones, velocidades y aceleraciones de los brazos en un número N de puntos de trabajo. Con estos valores se soluciona un sistema lineal del tipo [5]:

$$\tau = Y(q, \dot{q}, \ddot{q})\pi \quad (1)$$

donde $\tau \in \mathcal{R}^{N \times 1}$ es el vectores de los pares aplicados, $\pi \in \mathcal{R}^{q \times 1}$ es vector de los parámetros que queremos identificar y $Y \in \mathcal{R}^{N \times q}$ es el regresor, siendo N el número de puntos de trabajo, n el número de grados de libertad del manipulador y q el número de parámetros a identificar.

El sistema (1) se obtiene a partir de las ecuaciones de Euler Lagrange para manipuladores [1,5]:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) \quad (2)$$

donde τ es el par aplicado, M es la matriz de inercias, C es la matriz de los efectos centripetos y de Coriolis, G es el vector de lo pares debidos a la gravedad y F es el vector de los efectos de fricción. De la ecuación (2) se han obtenido los parámetros que caracterizan la estructura (introduciéndolos en el vector π) y las funciones de las posiciones, velocidades y aceleraciones de la articulaciones (recogidas en la matriz Y).

Dado que el sistema (1) tiene más ecuaciones que incógnitas ($Nn \gg q$), el valor de los parámetros se obtiene minimizando la norma 2 del vector de error $\|\tau - Y(q, \dot{q}, \ddot{q})\pi\|_2$, lo cual se puede realizar utilizando la pseudo-invertida de Moore Penrose [5]:

$$\pi = Y^+ \tau \quad (3)$$

donde

$$Y^+ = (Y^T Y)^{-1} Y^T \quad (4)$$

La solución que se obtiene de esta forma es la que mejor aproxima a los valores de par medidos en el sentido de los mínimos cuadrados.

En el entorno de este trabajo, el sistema real no permite medir el valor de la corriente de los motores. Por eso la medida del par aplicado no puede ser muy precisa. Si además se tiene en cuenta que en este caso se tiene que deducir los valores de velocidad y de aceleración de las coordenadas articulares a partir de las posiciones, queda claro que el ruido (debido a la incertidumbre sobre el valor del par realmente aplicado y a los errores numéricos en la derivación de la señal de posición) no es irrelevante en las medidas. De este hecho se deduce que no se pueden obtener los parámetros dinámicos solamente resolviendo el sistema (1), sino que el problema de la identificación se ha de dividir en problemas más sencillos que sean más fáciles de resolver [4,8].

El primer paso que se ha realizado ha sido la identificación de los efectos gravitatorios, es decir,

las ecuaciones (funciones de las posiciones de las articulaciones) que describen el par que tenemos que aplicar a la estructura para que esté parada en una posición cualquiera. Una vez identificados estos efectos, se ha afrontado el problema del comportamiento dinámico. De nuevo en este caso no se resuelve un sólo sistema, sino que hay que plantear un sistema para cada eje del robot. El método adoptado para identificar el comportamiento dinámico ha constado de los siguientes pasos:

- Realización de ensayos de movimiento del manipulador.
- Cálculo del par aplicado por los motores en el punto de trabajo.
- Cálculo de las velocidades y aceleraciones de las articulaciones en los puntos de trabajo.
- Formulación de las ecuaciones completas de la dinámica de cada enlace.
- Elección de un grupo de parámetros dinámicos (los más representativos) para cada eje.
- Extracción del efecto de los parámetros elegidos en las ecuaciones completas.
- Formulación de las ecuaciones de cada articulación en la forma de la ecuación (1).
- Sustracción del efecto de la gravedad, dado por los parámetros de gravedad ya encontrados, de la medida del par aplicado.
- Resolución de los sistemas de cada eje como en (3).

Esta metodología no lleva a la identificación de los valores reales de masas o de inercia de los enlaces, sino a combinaciones lineales de estos parámetros. En el proceso de identificación se elige una función para cada eje en la que aparecen los parámetros a identificar. Las funciones que describen los efectos dinámicos del sistema derivan de las ecuaciones de movimiento de Euler Lagrange (2) para que se adapten lo mejor posible al comportamiento real. De todas formas, lo que se requiere no es identificar los valores reales que caracterizan a cada enlace, sino encontrar funciones para cada eje que puedan describir apropiadamente el comportamiento dinámico.

2.1 FORMULACIÓN DE LAS ECUACIONES

Dado que el robot con el que se trabaja tiene seis grados de libertad, las ecuaciones dinámicas son muy complejas, y resulta muy tedioso escribirlas sin el auxilio de un software adecuado. El software utilizado en este trabajo para desarrollar los cálculos es el programa de cálculo simbólico MAPLE, que permite de manipular formalmente ecuaciones complejas.

Antes de escribir las ecuaciones dinámicas del robot se ha de elegir los sistemas de coordenadas para cada eslabón del manipulador. Para ello se ha utilizado el convenio de Denavit Hartenberg, en el que el *i-ésimo* sistema de referencia se sitúa en el *i-ésimo* eslabón [1,5].

Se han escrito seis códigos en lenguaje MAPLE que permiten obtener las ecuaciones dinámicas de Euler Lagrange (2) del manipulador. Nótese que se puede utilizar estos códigos para cualquier sistema dinámico con cadena cinemática abierta, con ejes de revolución o de traslación, y con cualquier número de grados de libertad. En estos códigos, a partir de los parámetros de Denavit Hartenberg relativos a los enlaces del sistema, se obtienen las matrices de transferencia que permiten describir posición y orientación de cualquier brazo del manipulador respecto al sistema base, los jacobianos, la matriz de masa del sistema y la de los efectos centrífugos y de Coriolis, la expresión de los pares debidos a la gravedad y la matriz *Y* del sistema (1).

2.2 LA MEDIDA DEL PAR

Dado que no se tiene acceso a la medida de la intensidad de los motores, se tiene que inferir el par aplicado de la magnitud de la señal de control. Esto se puede lograr, una vez conocidos los coeficientes de reducción de los ejes, las constantes de par y la corriente máxima de los motores utilizando la fórmula [8]:

$$\tau = \frac{\sqrt{3}}{2} K_r I_{max} K_r u \quad (5)$$

donde K_r e I_{max} son las constantes de par y la corriente máxima de los motores respectivamente, K_r es el coeficiente de reducción y u es la señal de control.

2.3 IDENTIFICACIÓN DE LOS PARES GRAVITATORIOS

Analizando las expresiones de los pares debidos a la gravedad, se han obtenido siete parámetros, combinaciones lineales de momentos estáticos de la estructura, que permiten describir las funciones [4]. Se han calculado además los valores de los pares proporcionados para mantener parado el robot en 130 posiciones diferentes, obteniendo un sistema como en (1), con la diferencia de que *Y* es solamente función de las posiciones, y solucionándolo como en (3).

2.4 IDENTIFICACIÓN DE LOS EFECTOS DINÁMICOS

Se han realizado unos ensayos moviendo el manipulador según referencias adecuadas (trayectorias trapezoidales suavizadas mediante splines cúbicas y trayectorias variadas realizadas utilizando splines de quinto orden) y eligiendo unos

puntos de trabajo en los que se calculan el par aplicado, la posición, la velocidad y la aceleración para cada articulación.

Para inferir los valores de velocidad y aceleración de los ejes se han interpolado previamente las señales de posición utilizando el comando *csaps* de MATLAB, que permite interpolar con splines cúbicas de las que se puede controlar la magnitud de la derivada segunda.

Para cada enlace han sido añadidos tres parámetros adicionales de fricción. La función elegida para describir el fenómeno de la fricción ha sido [8]:

$$F(\dot{q}) = F_1 \text{sign}(\dot{q}) + F_2 \dot{q} + F_3 \text{sign}(\dot{q}) |\dot{q}|^{1/3} \quad (6)$$

donde \dot{q} es la velocidad del eje y F_1 , F_2 y F_3 los tres parámetros a identificar.

Para resolver el problema de la identificación dinámica se ha sustraído, como se comento anteriormente, la estimación de los pares gravitatorios de los pares medidos, eligiendo para cada enlace los valores más representativos que se ha querido identificar, poniendo el sistema en la forma (1) y resolviéndolo según el método de los mínimos cuadrados (3).

3 EL ENTORNO DE TRABAJO

En esta sección se explica como ha sido utilizada la tarjeta de la marca comercial *dSPACE* para probar los controladores diseñados.

La tarjeta de control utilizada ha sido la *DS1103* de la compañía *dSPACE*, que consta de un procesador *PowerPC*, un *DSP* esclavo *TMS320F240* y una memoria *RAM* de 32 Mb. En la tarjeta *dSPACE* se ejecuta el controlador diseñado y se gestionan las posibles alarmas y señales de los circuitos electrónicos. Así mismo lleva a cabo la aplicación que provee las señales de control a los motores del robot manipulador.

En primer lugar, hay que distinguir entre aplicaciones utilizadas para implementar el controlador y aplicaciones de monitorización y sintonización de variables del controlador implementado.

Para descargar en la tarjeta las aplicaciones del primer tipo, se ha desarrollado un modelo de SIMULINK con distintos bloques que simplifica la introducción de nuevos controladores [8]. El controlador se construye con bloques de SIMULINK, lo cual evita la utilización de lenguajes de programación de bajo nivel e incide directamente en un menor tiempo de implementación de

controladores. Una vez introducido el controlador en el modelo de SIMULINK, se utiliza la herramienta *Real Time Workshop* de MATLAB para traducir el modelo a código C. El código se descarga de forma automática en la tarjeta *dSPACE* [2].

Finalmente se utilizan las aplicaciones *Cockpit* y *Trace* de *dSPACE*, las cuales pertenecen al segundo tipo de aplicaciones. El programa *Cockpit* permite crear pantallas con controles y gráficos para modificar en tiempo real cualquier parámetro del modelo de SIMULINK y mostrar la evolución en el tiempo de variables de interés, mientras que el programa *Trace* se utiliza para supervisar los ensayos, representando gráficamente la evolución de cualquier variable definida previamente en el modelo de SIMULINK.

4 LOS CONTROLADORES IMPLEMENTADOS

Se han implementado y comparado cuatro tipos de controladores diferentes: un *PID* clásico, un *PID* con dinámica invertida, un controlador H_∞ y el mismo controlador H_∞ robustificado para tener en cuenta de forma expresa las incertidumbres de los parámetros dinámicos. Los tres últimos controladores se pueden representar con el siguiente diagrama de bloques, cambiando simplemente las expresiones del controlador externo:

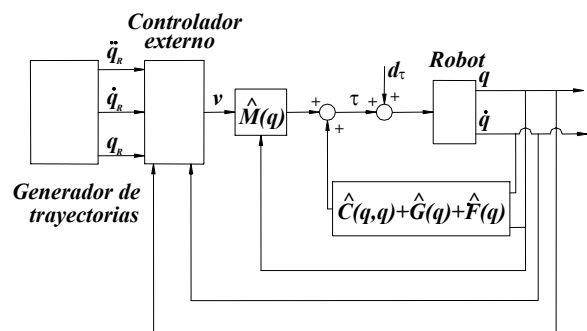


Figura 2: Esquema de bloques de los controladores *PID* con dinámica invertida, H_∞ y H_∞ robustificado.

4.1 TEORÍA DE LOS CONTROLADORES

Tanto el controlador *PID* como el controlador *PID* con dinámica invertida no tienen ninguna dificultad de nivel teórico, y su descripción puede encontrarse en cualquier libro básico de robótica. Las leyes de control del *PID* y del *PID* e dinámica invertida son respectivamente las siguientes:

$$v = K_D \dot{e} + K_P e + K_I \int e dt \quad (7)$$

$$\tau = \hat{M}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q} + \hat{G}(q) + \hat{F}(\dot{q}) - \hat{M}(K_D \dot{e} + K_P e + K_I \int e dt) \quad (8)$$

donde v y τ son los pares aplicados por el controlador PID y PID a dinámica invertida respectivamente, K_P , K_D y K_I son las matrices de las ganancias para el error de posición $e(t)$, su derivada y su integral, y \hat{M} , \hat{C} , \hat{G} , \hat{F} son las estimaciones de M , C , G y F de la ecuación (2). El esquema de bloques del controlador PID con dinámica invertida es el representado en la Figura 2, donde el controlador externo es un PID con ganancias constantes K_P , K_D y K_I de la ecuación (8).

4.1.1 El controlador H_∞

El controlador H_∞ desarrollado está basado en la teoría del control H_∞ no lineal [7]. Su formulación general es la siguiente:

Dado un nivel de atenuación γ , hallar una ley de control por realimentación del vector de estado tal que:

$$\int_0^T \|z(t)\|_2^2 dt \leq \gamma^2 \int_0^T \|\omega(t)\|_2^2 dt, \quad \forall T \quad (9)$$

donde $\omega(t)$ representa las perturbaciones del sistema y $z(t)$ representa el vector objetivo cuya norma se quiere hacer pequeña.

En el caso que el sistema sea un manipulador [3], se puede elegir el vector $\omega(t)$ de la ecuación (9) como una medida escalada del vector de perturbación de par d_τ (véase la Figura 2) y el vector $z(t)$ como sigue:

$$z(t) = W \begin{bmatrix} \dot{e}(t) & e(t) & \int e dt & u(t) \end{bmatrix}^T \quad (10)$$

siendo $e(t)$ el error de seguimiento de la trayectoria de referencia, $u(t)$ la señal de control aplicada y W una matriz de ponderación. En la implementación del controlador se ha elegido la matriz W de la ecuación (10) de forma diagonal:

$$W = \begin{bmatrix} \text{diag}(w_{D_i}) & 0_{n \times n} & 0_{n \times n} & 0_{n \times n} \\ 0_{n \times n} & \text{diag}(w_{P_i}) & 0_{n \times n} & 0_{n \times n} \\ 0_{n \times n} & 0_{n \times n} & \text{diag}(w_{I_i}) & 0_{n \times n} \\ 0_{n \times n} & 0_{n \times n} & 0_{n \times n} & \text{diag}(w_{U_i}) \end{bmatrix} \quad (11)$$

donde n es el número de grados de libertad del manipulador.

Desarrollando las expresiones anteriores se llega a encontrar la formulación del par óptimo, que resulta tener la siguiente expresión [3]:

$$\tau = \hat{M}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q} + \hat{G}(q) + \hat{F}(\dot{q}) - (K_D \dot{e} + K_P e + K_I \int e dt) \quad (12)$$

donde \hat{M} , \hat{C} , \hat{G} , \hat{F} vuelven a ser las estimaciones de M , C , G y F de la ecuación (2), y K_D , K_P y K_I son matrices de ganancias variables con el tiempo, las cuales tienen las siguientes expresiones:

$$K_{P_{i,j}} = \frac{w_{I_j}}{w_{D_i}} \hat{M}_{i,j} + \frac{\sqrt{w_{P_j}^2 + 2w_{D_j} w_{I_j}}}{w_{D_i}} \left(\hat{C}_{i,j} + \frac{1}{w_{U_i}^2} \delta_{ij} \right)$$

$$K_{I_{i,j}} = \frac{w_{I_j}}{w_{D_i}} \left(\hat{C}_{i,j} + \frac{1}{w_{U_i}^2} \delta_{ij} \right) \quad (13)$$

$$K_{D_{i,j}} = \frac{\sqrt{w_{P_j}^2 + 2w_{D_j} w_{I_j}}}{w_{D_i}} \hat{M}_{i,j} + \frac{1}{w_{U_i}^2} \delta_{ij}$$

donde $\delta_{ij} = 1$ en caso de que $i = j$, y 0 en caso contrario.

Si se escribe la ecuación (12) de la siguiente forma:

$$\tau = \hat{M}\ddot{q}_r + \hat{C}\dot{q}_r + \hat{G} + \hat{F} - \hat{M}(\hat{M}^{-1}K_D \dot{e} + \hat{M}^{-1}K_P e + \hat{M}^{-1}K_I \int e dt) \quad (14)$$

la ecuación del par calculado (14) es algebraicamente parecida a la ecuación de un controlador PID con dinámica invertida (8). La diferencia fundamental reside en que el controlador PID externo no es el mismo. En el caso del PID con dinámica invertida, los valores de K_D , K_P , K_I son constantes, mientras que en el controlador H_∞ , estos valores varían en el tiempo, ya que dichas matrices dependen de $M(q)$ y $C(q, \dot{q})$. Por ello se puede afirmar que el controlador H_∞ desarrollado equivale a un PID con dinámica invertida, con la diferencia que las matrices de las ganancias del PID son funciones no lineales de la posición y de la velocidad de las articulaciones. Por lo tanto, utilizando la ecuación del par calculado (14), el esquema de bloques del controlador H_∞ es el mismo de lo representado en la Figura 2, teniendo en cuenta que el controlador externo es un PID no lineal.

4.1.2 Robustificación del controlador H_∞

En el desarrollo del controlador H_∞ se ha supuesto que tanto las matrices M y C como los vectores G y F de la ecuación (2) son conocidos. Sin embargo, en la realidad siempre se cometen errores en la estimación de los parámetros dinámicos. Por eso conviene añadir

al par calculado (14) un aporte que tenga en cuenta la incertidumbre sobre los parámetros dinámicos [3,5].

Se puede plantear este problema con la misma metodología con la que se resuelve el problema de la robustificación de un controlador *PID* con dinámica invertida (véase la Figura 2). Para ello, si se diseña una ley de control externa v (véase la Figura 2) como sigue [3]:

$$v(t) = \ddot{q}_r - (\hat{M}^{-1}K_D \dot{e} + \hat{M}^{-1}K_P e + \hat{M}^{-1}K_I \int edt) + \Delta v(t) \quad (15)$$

donde K_D , K_P , K_I son las matrices de ganancias dadas en la ecuaciones (13). Se puede observar como la expresión del controlador es la misma de la del controlador H_∞ , siendo añadido el término de robustificación $\Delta v(t)$.

La diferencia fundamental de este problema respecto a la robustificación del controlador *PID* con dinámica invertida es que, dado que el *PID* externo no es lineal, la ecuación a la que se llega es una ecuación diferencial a las derivadas parciales no lineal del tipo de Hamilton-Jacobi-Bellman-Isaac [3], una tipología para la que todavía no existe una solución genérica. Para resolver ese problema se puede linearizar dicha ecuación diferencial en cada punto de trabajo, encontrando una ecuación matricial de Lyapunov [3]:

$$A^T P + PA + Q = 0 \quad (16)$$

Para implementar el controlador se requeriría la solución de dicha ecuación en tiempo real en cada instante, lo cual significaría un costo computacional demasiado grande para el procesador de la tarjeta *dSPACE* en la que se implementan los controladores. Por eso, aprovechando la forma particular que posee la ecuación (16) en el caso con el que se ha trabajado, se ha encontrado una solución algébrica de dicha ecuación utilizando el programa MAPLE. Para encontrar la solución se requiere de fijar los valores de la matriz Q , que ha sido elegida de forma diagonal. Una vez resuelta en P la ecuación (16), el término $\Delta v(t)$ resulta ser [3,5]:

$$\Delta v(t) = \begin{cases} -\rho(x,t) \frac{B^T P x}{\|B^T P x\|} & \text{si } \|B^T P x\| \geq \varepsilon \\ -\frac{\rho(x,t)}{\varepsilon} B^T P x & \text{si } \|B^T P x\| < \varepsilon \end{cases} \quad (17)$$

donde

$$\rho(x,t) = \frac{1}{1-\alpha} (\alpha Q_1 + \alpha \|Kx\| + \bar{M} \phi(x,t))$$

$$x = \begin{bmatrix} \dot{e} & e & \int edt \end{bmatrix}^T$$

$$B = \begin{bmatrix} I_{n \times n} & 0_{n \times n} & 0_{n \times n} \end{bmatrix}^T$$

$$\alpha = \sup_q \|M(q)^{-1} \hat{M}(q) - I\| \quad (18)$$

$$K = \hat{M}^{-1} K_D \dot{e} + \hat{M}^{-1} K_P e + \hat{M}^{-1} K_I \int edt$$

$$Q_1 = \sup_{t \geq 0} \|\ddot{q}_r\|$$

$$\bar{M} = \sup_q \|M^{-1}(q)\|$$

$$\phi(x,t) = \sup_{q,\dot{q}} \|(C\dot{q} - \hat{C}\dot{q}) + (F - \hat{F}) + (G - \hat{G})\|$$

Nótese como se ha linealizado esta ley de control para valores de $\|B^T P x\|$ inferiores a ε para evitar el efecto de las vibraciones [3,5].

El esquema de bloques del controlador H_∞ robustificado de nuevo se ajusta al esquema representado en la Figura 2, donde la ley de control v proporcionada por el controlador externo queda definida por la ecuación (15).

5 ENSAYOS REALIZADOS

5.1 TRAYECTORIAS DE REFERENCIA

Todos los controladores han sido probados en el seguimiento de las mismas trayectorias de referencias. Los ensayos realizados han sido de tres tipos:

- Seguimiento de trayectoria trapezoidal realizada con splines cúbicas y movimiento de un sólo eje cada vez.
- Seguimiento de trayectorias trapezoidales (splines cúbicas) con movimiento simultáneo de todos los ejes.
- Movimiento simultáneo de todos los ejes según trayectorias variadas realizadas por spline del quinto orden.

Un ejemplo de trayectoria trapezoidal puede verse en la siguiente figura:

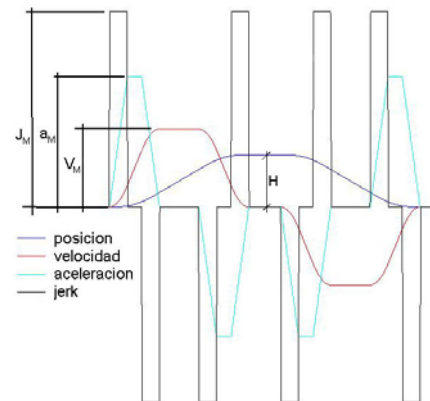


Figura 3: Ejemplo de trayectoria trapezoidal

El generador de trayectorias implementado junto a los controladores permite además de controlar en tiempo real los valores indicados en la Figura 3: H (magnitud del escalón), V_M , a_M y J_M (valores de velocidad, aceleración y derivada de la aceleración máximos alcanzados en la trayectoria).

La figura que sigue muestra un ejemplo de trayectoria spline del quinto orden, con evolución de posición, velocidad, aceleración y derivada de la aceleración (jerk).

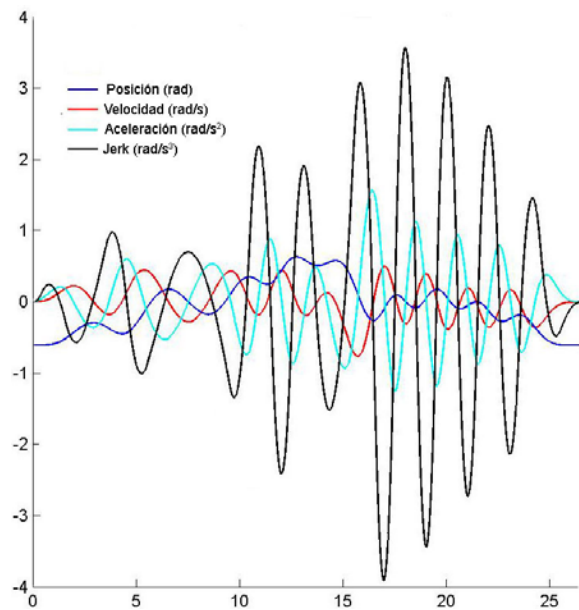


Figura 4: Ejemplo de spline del quinto orden

5.2 RESULTADOS EXPERIMENTALES

En esta sección se presentan los resultados de los ensayos de los controladores implementados en el seguimiento de las trayectorias de referencia.

5.2.1 La medida de la velocidad

Antes de presentar los resultados es necesario reflexionar sobre la instrumentación utilizada, y en particular sobre la medida de la velocidad con la que se ha tenido que trabajar.

Dado que el manipulador utilizado no tiene sensores de velocidad, es necesario calcular la velocidad derivando la señal de posición. Los motores están dotados de sensores de posición con 4048 impulsos por vuelta, y la resolución de posición de las articulaciones viene dada por la resolución de los ejes de los motores multiplicada por la constante de reducción. Cuando se deriva la posición para inferir la velocidad, la resolución de la velocidad crece, con respecto a la resolución de posición, según el tiempo

de muestreo. Teniendo en cuenta que los controladores han sido implementados con un tiempo de muestreo de un milisegundo, la resolución aumenta mil veces. El manipulador utilizado tiene una resolución de velocidad entre $0.6^\circ/s$ y $1.8^\circ/s$ según el eje. Por eso, como se observa en la siguiente figura, la señal de velocidad lleva asociado un ruido significativo.

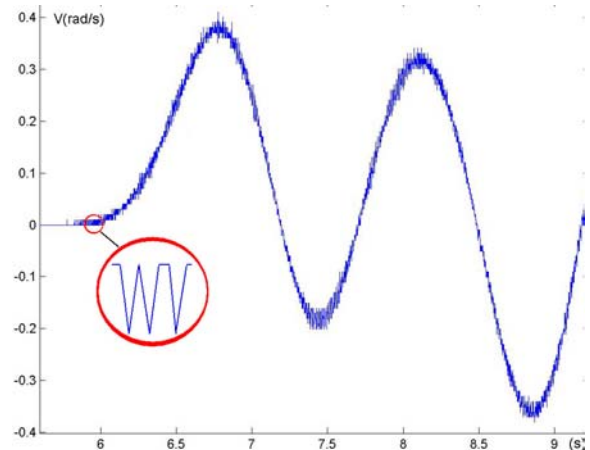


Figura 5: Señal de velocidad

El efecto de la discretización de la señal de posición (y de la velocidad) se nota especialmente al controlar el manipulador para que siga una referencia de velocidad constante. En este caso la velocidad oscila alrededor del valor de referencia por efecto de su discretización, y si la ganancia del error de velocidad es demasiado grande, el sistema puede empezar a vibrar. Las consecuencias de este comportamiento pueden observarse en la Figura 5, en la que se puede notar cómo alrededor de los puntos en los que la derivada de la velocidad se anula (puntos de máximo o de mínimo), el error de velocidad tiene vibraciones de mayor magnitud.

Han sido probados filtros de primer y segundo orden para obtener una señal de velocidad mejor, pero en estos casos se introducen retrasos en la realimentación de velocidad que pueden volver inestable el sistema en bucle cerrado. Por eso se ha comprobado que es mejor no filtrar la velocidad y tener bajas las ganancias del error de velocidad para evitar vibraciones.

5.2.2 La importancia de la referencia

En la sintonización de controladores tiene mucha importancia el tipo de trayectoria de referencia, debiéndose proporcionar referencias que el manipulador sea capaz de seguir. Como se expuso en la sección 5.2.1, las ganancias del error de velocidad tienen que ser bajas, si no queremos que se produzcan vibraciones en el movimiento del brazo manipulador. De hecho, un controlador que no tenga un buen control de velocidad puede presentar un

comportamiento oscilante frente a una referencia a escalón. Eso se puede evitar teniendo bajas las ganancias del error de posición y de su integral, aunque como contraprestación, de esa manera los errores serán más grandes.

Por esto, en la implementación de leyes de control se ha tenido que elegir entre controladores con ganancias bajas que pudiera soportar referencias discontinuas y controladores con ganancias más altas, pero necesitando las referencias oportunas. En este proyecto se ha elegido de desarrollar controladores para el seguimiento de trayectorias suaves y sin discontinuidades.

5.2.3 Comparación de los resultados

Comparando los resultados obtenidos con los controladores *PID* y *PID* con dinámica invertida, se ha observado que aplicando éste último se ha obtenido una reducción de los errores máximos entre un 2% y un 7%. Eso es debido sobre todo a la parte predictiva del controlador *PID* con dinámica invertida, que proporciona una estimación del par necesario para el seguimiento de la referencia. Desde luego, la presencia de la estimación de los efectos gravitatorios permite eliminar el error que comete el controlador *PID* desde el momento en que se desbloqueen los frenos. En efecto, el *PID* con dinámica invertida proporciona el par necesario para mantener parado el manipulador sin necesidad que se genere un error. Por el contrario, el controlador *PID* no puede aplicar ningún par, a menos que no se genere, o se haya generado, un error previo.

El controlador H_∞ se ha comportado de manera muy parecida al controlador *PID* con dinámica invertida. De hecho se puede afirmar que, para este caso particular, los dos controladores son casi equivalentes. La única excepción se ha tenido en el primer eje, consiguiendo una reducción del error máximo, con respecto al *PID* con dinámica invertida, hasta el 20% y una reducción de la norma dos del error de posición hasta el 10%.

La robustificación del controlador H_∞ ha mejorado sobre todo el comportamiento de los dos primeros ejes, bajando los errores máximos de posición hasta el 8% y la norma dos del error hasta el 5%. Para las otras articulaciones el comportamiento ha sido muy parecido al comportamiento del controlador H_∞ no robustificado.

Se puede notar cómo, en líneas generales, la implementación del controlador H_∞ no ha mejorado de manera sensible el comportamiento del controlador *PID* con dinámica invertida. Este hecho puede ser explicado si se tiene en cuenta el tipo de manipulador utilizado. El robot con el que se ha

trabajado presenta constantes de reducción elevadas en todos los ejes [6] y en particular, dichas constantes son especialmente elevadas para los ejes en los que se pueden esperar un mayor acoplamiento de los efectos dinámicos. Por eso, el comportamiento dinámico del sistema es casi completamente desacoplado y lineal. Nótese además que, en la implementación del controlador H_∞ , la no linealidad del controlador externo (véase la Figura 2) depende de la no linealidad de las matrices M y C de la ecuación (2). Si en estas matrices los efectos de las no linealidades son pequeños, el *PID* externo del controlador H_∞ es muy parecido a un *PID* clásico, por lo que el controlador H_∞ se convierte en este caso particular en un *PID* con dinámica invertida. Tras estas consideraciones, se puede afirmar que el controlador H_∞ es más apropiado para sistema con dinámica fuertemente no lineal, como en el caso de manipuladores con acoplamiento directo de los motores en los ejes, donde los requerimientos de velocidades sean más elevadas.

Hay que hacer notar también la importancia del modelado del manipulador. Los controladores *PID* con dinámica invertida, H_∞ y H_∞ robustificado basan sus leyes de control en el modelo del sistema, y aunque sea posible añadir una señal que se oponga a la incertidumbre sobre los parámetros dinámicos (controlador H_∞ robustificado), es evidente que es mejor estimar la dinámica del sistema de forma más precisa antes que intentar disminuir los efectos de los errores en el modelado. Por eso, tras la implementación de los cuatro controladores descritos, se ha vuelto a hacer una identificación del sistema intentando obtener un modelo más preciso. El nuevo modelo ha sido utilizado en la implementación de un nuevo *PID* con dinámica invertida, obteniendo una reducción de los errores máximos, con respecto al *PID* con dinámica invertida implementado con el modelo anterior, desde el 2% hasta el 17% y una reducción de la norma dos del error hasta el 20%.

En las Figuras 6 a 9 se muestran los gráficos de los errores de posición para el eje 1 en el seguimiento de la trayectoria trapezoidal con movimiento de todos los ejes. Se puede notar como el comportamiento del *PID* con dinámica invertida, del controlador H_∞ y del controlador H_∞ robustificado es muy parecido.

En las Figuras 10 y 11 se comparan los errores cometidos en el eje 3 en el seguimiento de la trayectoria trapezoidal con movimiento de todos los ejes. La comparación se realiza entre los dos controladores *PID* con dinámica invertida implementados, observándose como el nuevo modelo del sistema lleva a una reducción sensible de los errores.

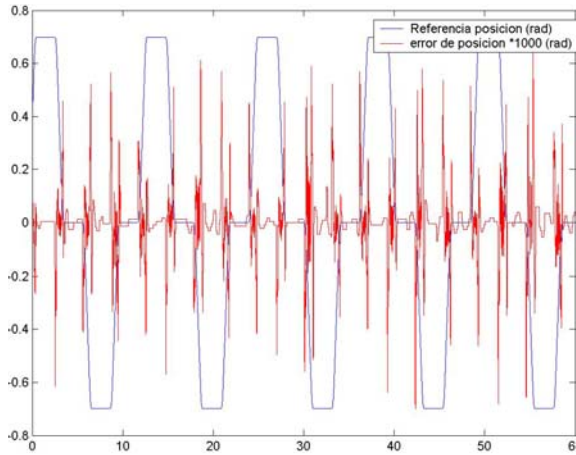


Figura 6: Errores en el eje 1 obtenidos con el controlador PID

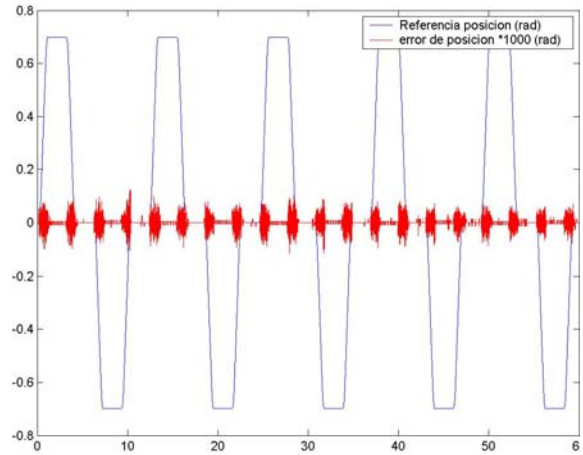


Figura 9: Errores en el eje 1 obtenidos con el controlador H_∞ robustificado

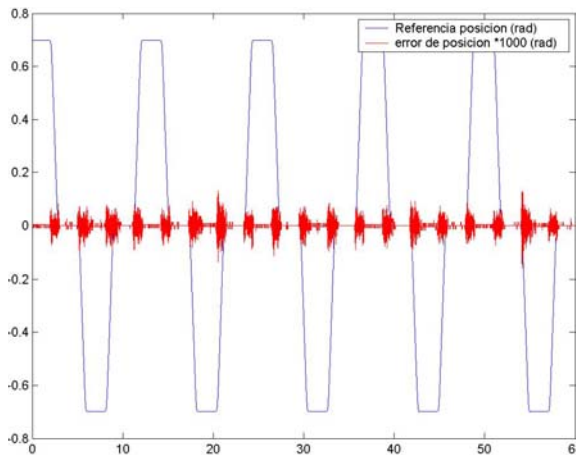


Figura 7: Errores en el eje 1 obtenidos con el controlador PID con dinámica invertida

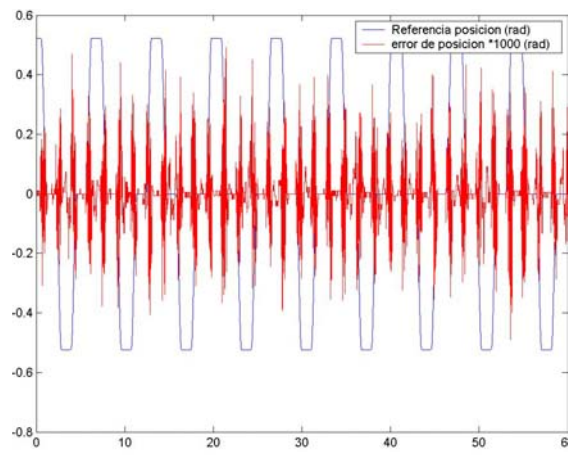


Figura 10: Errores en el eje 3 obtenidos con el controlador PID con dinámica invertida con el modelo inicial

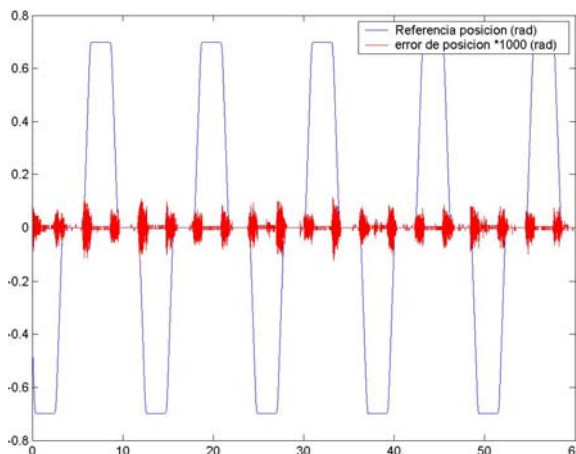


Figura 8: Errores en el eje 1 obtenidos con el controlador H_∞

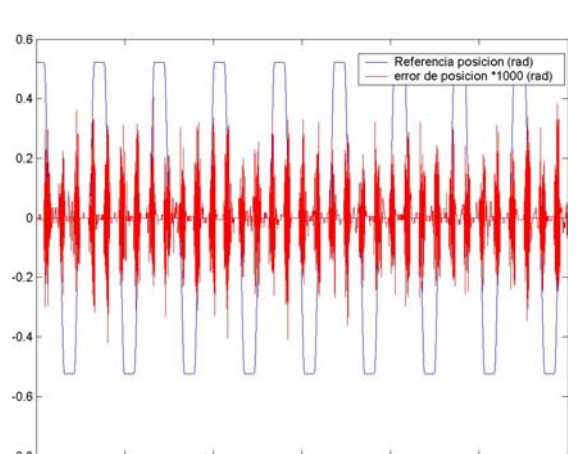


Figura 11: Errores en el eje 1 obtenidos con el controlador PID con dinámica invertida con el nuevo modelo

6 CONCLUSIONES

En este artículo se ha abordado la identificación de un brazo manipulador de carácter industrial y la

implementación de diferentes estrategias de control. Respecto a la identificación, se ha utilizado regresiones lineales, aprovechando la estructura lineal de la dinámica de los manipuladores respecto a los parámetros dinámicos del mismo. Se han implementado controladores clásicos, de tipo *PID* y control con dinámica invertida, y controladores avanzados basados en la teoría del control H_∞ no lineal. Finalmente, se ha mostrado resultados experimentales, obteniendo como principal conclusión que las técnicas de control avanzado no mejoran excesivamente el comportamiento del sistema debido a las restricciones de comportamiento (por las reductoras en las articulaciones) del propio sistema.

Agradecimientos

Los autores quieren agradecer a la CICYT la financiación de este trabajo con cargo a los proyectos de investigación DPI2001-2424-C02 y DPI2000-1218-C0401.

Referencias

- [1] Craig, J.J., (1989) Introduction to robotics: Mechanics and control, Addison Wesley.
- [2] dSPACE, (1999) Matlab-dspace interface libraries, Manua.
- [3] Ortega, M.G., (2001) Aportaciones al Control H_∞ de Sistemas Multivariables, Escuela Superior de Ingenieros, Universidad de Sevilla.
- [4] Pérez, C., (1999) Simulación y control de un robot manipulador, Escuela Superior de Ingenieros, Universidad de Sevilla.
- [5] Sciavicco, L., Siciliano, B., (1996) Modeling and Control of Robot Manipulators, The McGraw-Hill Companies, Inc.
- [6] System Robot, (1991) Manuale d'uso rm10.
- [7] van der Schaft, A.,(2000) L_2 -Gain and Passivity Techniques in Nonlinear Control. Springer-Verlag. New York.
- [8] Vicente, R. R., (2001) Plataforma de Experimentación de Controladores para Robots Industriales, Escuela superior de Ingenieros, Univerlidad de Sevilla.