



ESTRATEGIAS DE CONTROL PARA SISTEMAS MECÁNICOS

Fabián M. Camusso y Alberto Cardona

Grupo de Tecnología Mecánica de INTEC

Universidad Nacional del Litoral - Conicet

Güemes 3450 - 3000 Santa Fe

RESUMEN

En este trabajo se analiza una estrategia de control basada en geometría diferencial [1], y para verificar la performance se utiliza el programa de simulación *SIMULINK* [2].

Por otra parte, se busca introducir en el programa multipropósito de simulación de mecanismos *MECANO* [3], la capacidad de representar algoritmos y sistemas de control en forma modular -usando "elementos" apropiados, al estilo de los usados para representar el sistema mecánico- para así poder verificar mediante simulación el correcto funcionamiento del sistema completo.

Finalmente se modela un sistema mecánico para simular y verificar la estrategia de control empleada, y se comparan los resultados obtenidos por *SIMULINK* y por *MECANO*.

ABSTRACT

A control strategy based on differential geometry is analyzed [1]. The performance of control algorithms is verified using program *SIMULINK*, which is a well-known software used for simulating general dynamics systems.

Some advancements towards the introduction of the capability of representing control algorithms in the general purpose program for mechanisms simulation *MECANO* are presented. The idea is to represent, in a modular way, control systems using appropriate "elements", in the same style of those used to model the mechanism.

A particular mechanical system is modeled, and a control strategy is developed. The system response is calculated by using *SIMULINK* and *MECANO*, and the computed results of both computations are compared.

1. Introducción

Las estrategias aplicadas en el control de sistemas no lineales usualmente se basan en una linealización local [9], y posterior desarrollo de controladores lineales basados en el modelo linealizado [6]. Para sistemas altamente no lineales, se utilizan con frecuencia controladores *feedback* lineales, que garantizan en gran medida su estabilidad.

En estos últimos años han sido desarrollados métodos basados en geometría diferencial [8], que permiten una linealización exacta del modelo no lineal independientemente del punto de operación.

En este trabajo se analiza una estrategia de control basada en geometría diferencial, y para verificar la performance se utiliza el programa de simulación *SIMULINK*.

Por otra parte, se busca introducir en el programa multipropósito de simulación de mecanismos *MECANO*, la capacidad de representar algoritmos y sistemas de control en forma modular –usando “elementos” apropiados, al estilo de los usados para representar el sistema mecánico– para así poder verificar mediante simulación el correcto funcionamiento del sistema completo.

Finalmente se modela un sistema mecánico para simular y verificar la estrategia de control empleada, y se comparan los resultados obtenidos por *SIMULINK* y por *MECANO*.

2. Teoría de control no lineal

2.1 Cambios no lineales de coordenadas

Un sistema dinámico no lineal puede ser descrito por las siguientes ecuaciones diferenciales:

$$\dot{x}(t) = f(x(t)) + \sum_{j=1}^m g_j(x(t))u_j(t) \quad (1)$$

$$y(t) = h(x(t)) \quad (2)$$

donde

$$u(t) = [u_1(t) \dots u_m(t)] \quad (3)$$

$$x(t) = [x_1(t) \dots x_n(t)] \quad (4)$$

$$y(t) = [y_1(t) \dots y_p(t)] \quad (5)$$

son los vectores de controles, de estados y de salidas, respectivamente, y las funciones $f, g_1, \dots, g_m, h_1, \dots, h_p$ son infinitamente diferenciables (C^∞).

Se propone un cambio de coordenadas con el fin de obtener una forma más “simple” del sistema dinámico anterior, que muestre claramente algunas características del mismo que no están explícitamente manifestadas en las variables originales. Para sistemas no lineales es natural intentar cambios de coordenadas no lineales de la forma siguiente [1]:

$$z = \begin{Bmatrix} z_1 \\ \vdots \\ \cdot \\ \vdots \\ z_n \end{Bmatrix} = \Phi(x) = \begin{Bmatrix} \phi_1(x_1, \dots, x_n) \\ \vdots \\ \vdots \\ \phi_n(x_1, \dots, x_n) \end{Bmatrix} \quad (6)$$

debiendo verificar las siguientes propiedades: Φ es inversible y Φ y Φ^{-1} son infinitamente diferenciables. La primera propiedad permite volver a los estados originales, mientras que la segunda garantiza que la descripción del sistema en las nuevas coordenadas sea tan suave (diferenciable) como la original (propiedad llamada *difeomorfismo*).

2.2 Determinación del grado relativo de un sistema no lineal

Definición 2.2.1: Dadas las funciones $h: \mathbb{R}^n \rightarrow \mathbb{R}$ y $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$, llamaremos derivada de Lie de $h(x)$ en la dirección del campo vectorial $f(x)$ a la función $L_f h: \mathbb{R}^n \rightarrow \mathbb{R}$

$$L_f h(x) = \sum_{i=1}^n \frac{\partial h(x)}{\partial x_i} f_i(x_1, \dots, x_n) = \frac{\partial h(x)}{\partial x} f(x) \quad (7)$$

Se supone que las funciones que intervienen son infinitamente diferenciables, por lo tanto, la aplicación repetida de la operación definida es posible, y podemos escribir:

$$L_g L_f h(x) = \frac{\partial [L_f h(x)]}{\partial x} g(x) \quad (8)$$

$$L_f^k h(x) = \frac{\partial [L_f^{k-1} h(x)]}{\partial x} f(x) \quad (9)$$

Definición 2.2.2. El sistema de una entrada y una salida:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \quad (10)$$

tiene grado relativo r en x_0 si y sólo si:

1. $L_g L_f^k h(x) = 0$, $k = 0, 1, \dots, r-2$, x en un entorno de x_0
2. $L_g L_f^{r-1} h(x_0) \neq 0$.

El grado relativo de un sistema depende de la función de salida $h(x)$, y caracteriza cuan directamente las entradas afectan la salida. Es decir que r es la cantidad de veces que hay que derivar la salida para que aparezca explícitamente el control u .

2.3 Linealización exacta

Consideremos un sistema no lineal con una entrada y una salida, cuyo grado relativo sea n , es decir, igual a la dimensión del espacio de estados. A partir de la forma normal establezcamos condiciones que permitan obtener una versión lineal y controlable del sistema no lineal. Dada la hipótesis hecha sobre el grado relativo, el cambio de coordenadas necesario para la construcción de la forma normal es :

$$z = \begin{Bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{Bmatrix} = \Phi(x) = \begin{Bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n-1} h(x) \end{Bmatrix} \quad (11)$$

En las nuevas coordenadas el sistema se escribe :

$$\begin{Bmatrix} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \dots = \dots \\ \dot{z}_{n-1} = z_n \\ \dot{z}_n = b(z) + a(z)u \\ y = z_1 \end{Bmatrix} \quad (12)$$

con:

$$a(z) = L_g L_f^{n-1} h(\Phi^{-1}(z)) \quad (13)$$

$$b(z) = L_f^n h(\Phi^{-1}(z)) \quad (14)$$

Si definimos $v = b(z) + a(z)u$, resulta:

$$\begin{Bmatrix} \dot{z}_1 = z_2 \\ \dots = \dots \\ \dot{z}_{n-1} = z_n \\ \dot{z}_n = v \\ y = z_1 \end{Bmatrix} \quad (15)$$

que es lineal y controlable.

Es decir que un sistema no lineal con grado relativo $r = n$, puede ser transformado en un sistema lineal y controlable [1], a través de (ver figura 2.3.1):

- el cambio (local) de coordenadas $z = \Phi(x)$ dado por ecuación (12)
- la realimentación no lineal :

$$u = \frac{1}{a(z)}[-b(z) + v] \quad (16)$$

Por lo tanto si x_0 es un punto de equilibrio del sistema original, y el sistema tiene grado relativo $r = n$ en x_0 , existe una ley de realimentación no lineal definida en un entorno de x_0 y una transformación de coordenadas – también en un entorno de x_0 – que convierten al sistema en uno lineal y controlable definido en un entorno de $z = 0$. Sobre este sistema lineal puede definirse una ley de control por realimentación para, por ejemplo, obtener un determinado conjunto de autovalores, satisfacer un determinado criterio de optimización, lograr el seguimiento de una trayectoria, o algún otro objetivo.

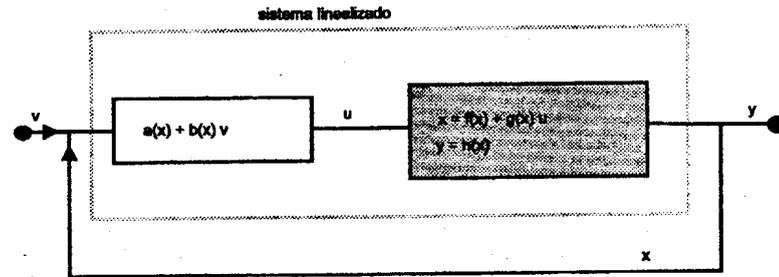


Figura 2.3.1: Sistema linealizado

Si el sistema tiene grado relativo $r < n$, determina parcialmente la transformación dada por la ecuación (11) ya que sólo es posible definir $\phi_1(x), \dots, \phi_r(x)$. Pero es posible agregar a las anteriores otras $n - r$ funciones $\phi_{r+1}(x), \phi_{r+2}(x), \dots, \phi_n(x)$ tales que $\Phi(x)$ tenga jacobiano no singular en x_0 , es decir que sea un difeomorfismo. Por lo tanto resulta :

$$\left\{ \begin{array}{l} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \dots = \dots \\ \dot{z}_r = b(z) + a(z)u \\ \dot{z}_{r+1} = q_{r+1}(z) + p_{r+1}(z)u(t) \\ \dots = \dots \\ \dot{z}_n = q_n(z) + p_n(z)u(t) \end{array} \right\} \quad (17)$$

con:

$$a(z) = L_g L_f^{r-1} h(\Phi^{-1}(z)) \quad (18)$$

$$b(z) = L_f^r h(\Phi^{-1}(z)) \quad (19)$$

$$q_i(z) = L_f \phi_i(\Phi^{-1}(z)), \quad i = r + 1, \dots, n \quad (20)$$

$$p_i(z) = L_g \phi_i(\Phi^{-1}(z)), \quad i = r + 1, \dots, n. \quad (21)$$

Como se observa en la ecuación (17), el control se encuentra en las $n - r$ últimas ecuaciones. Es posible elegir $\phi_{r+1}, \dots, \phi_n$ de tal forma que :

$$L_g \phi_i(x) = 0, \quad i = r + 1, \dots, n \quad (22)$$

para todo x en un entorno de x_0 , el control $u(t)$ desaparece de ellas, y se obtiene la forma normal del sistema :

$$\left\{ \begin{array}{l} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \dots = \dots \\ \dot{z}_{r-1} = z_r \\ \dot{z}_r = b(z) + a(z)u \\ \dot{z}_{r+1} = q_{r+1}(z) \\ \dots = \dots \\ \dot{z}_n = q_n(z)y = z_1 \end{array} \right. \quad (23)$$

Encontrar funciones $\phi_{r+1}, \dots, \phi_n$ que verifiquen $L_g \phi_i(x) = 0$ para $i = r + 1, \dots, n$ significa resolver un sistema de $n - r$ ecuaciones en derivadas parciales :

$$\frac{\partial \phi_{r+1}(x)}{\partial x} g(x) = 0, \dots, \frac{\partial \phi_n(x)}{\partial x} g(x) = 0 \quad (24)$$

3. Aplicación de la teoría de control

Con el objetivo de comprobar la eficiencia de la teoría analizada, se modela y simula el sistema de carro-péndulo mostrado en la figura 3.1.

El sistema en estudio consta de un carro que se desplaza horizontalmente sobre el cual se encuentra montado un péndulo invertido con desplazamiento angular. El péndulo invertido se encuentra en una situación de equilibrio inestable, pudiendo caer en cualquier momento si no se le aplica una fuerza de control adecuada. Se demuestra que la planta tiene un polo en el eje negativo real ($s \in \mathbb{R}e^-$) y otro en el eje positivo real ($s \in \mathbb{R}e^+$), por lo tanto la planta en lazo abierto es inestable. La fuerza de control u actúa sobre el carro, y es conjugada al desplazamiento horizontal del mismo.

Denotaremos por M la masa del carro, m la masa del péndulo (concentrada en el centro de masas cm), x_1 y x_2 el desplazamiento y velocidad angular del péndulo, l la distancia desde la articulación del péndulo al carro hasta su centro de masas, a la aceleración de la gravedad, y x_{cm} e y_{cm} las coordenadas horizontal y vertical del centro de masas.

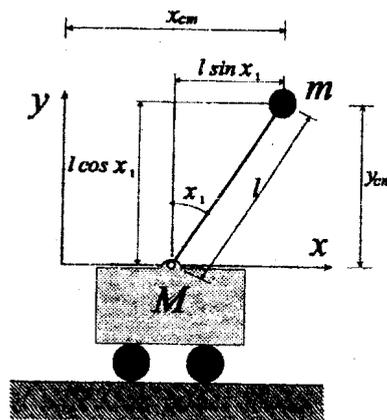


Figura 3.1: Sistema carro-péndulo invertido

3.1 Construcción y control del modelo

Las ecuaciones de movimiento del péndulo fueron desarrolladas a partir del Lagrangiano del sistema en análisis, usando procedimientos generales [4] escritos en el lenguaje propio de *Maple V* [5]. Las ecuaciones diferenciales que rigen al modelo —expresado en la forma de la ecuación (1)— son :

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{-\cos(x_1)m\sin(x_1)x_2^2 - a\sin(x_1)(m+M)}{(-m-M+m\cos(x_1)^2)l} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{\cos(x_1)}{(-m-M+m\cos(x_1)^2)l} \end{pmatrix} u \quad (25)$$

con la salida :

$$y = h(x) = x_1$$

Identificado el sistema se determina el grado relativo. Utilizando el programa *Maple V* se determinó que el grado relativo ($r = 2$), coincide con el espacio de estados. Luego la ley de realimentación que linealiza el sistema está dada por :

$$u = \frac{v - L_f^2 h(x)}{L_g L_f h(x)} \quad (26)$$

En un sistema lineal es posible asignar autovalores o satisfacer un criterio de optimización mediante una ley de control proporcional :

$$v = Kz \quad K = (C_0 \quad \dots \quad C_{n-1}) \quad (27)$$

En sistemas no lineales esto puede realizarse mediante una ley del tipo :

$$v = C_0 h(x) + C_1 L_f h(x) + \dots + C_{n-1} L_f^{n-1} h(x) \quad (28)$$

Por lo tanto, como v puede diseñarse de manera de asignar arbitrariamente los polos a lazo cerrado a los efectos de estabilizar el sistema, tendremos que la ley de control resultante está dada por :

$$u = \frac{w - L_f^2 h(x) - C_1 L_f h(x) - C_2 h(x)}{L_g L_f h(x)} \quad (29)$$

donde C_1 y C_2 son los coeficientes del vector de ganancias K . En este caso :

$$L_f h(x) = x_2 \quad (30)$$

$$L_f^2 h(x) = \frac{-\cos(x_1)m\sin(x_1)x_2^2 - a\sin(x_1)(m+M)}{(-m-M+m\cos(x_1)^2)l} \quad (31)$$

$$L_g L_f h(x) = \frac{\cos(x_1)}{(-m-M+m\cos(x_1)^2)l} \quad (32)$$

La función de transferencia del sistema linealizado y estabilizado resultante será :

$$H^*(s) = \frac{Y(s)}{w(s)} = \frac{1}{s^2 + C_1 s + C_2} \quad (33)$$

4. Simulación y resultados

4.1 Empleo del programa SIMULINK

Se modela y simula el sistema carro-péndulo, utilizando el programa *SIMULINK*, con el propósito de evaluar la teoría de control propuesta y comparar los resultados con los que se obtendrán en el programa *MECANO*.

SIMULINK es un programa para simular sistemas dinámicos. Funciona como extensión al programa *MATLAB* y agrega numerosas funcionalidades específicas de sistemas dinámicos al mismo tiempo que conserva toda la capacidad de *MATLAB*. La figura 4.1.1 muestra el esquema de control con realimentación modelado en *SIMULINK*.

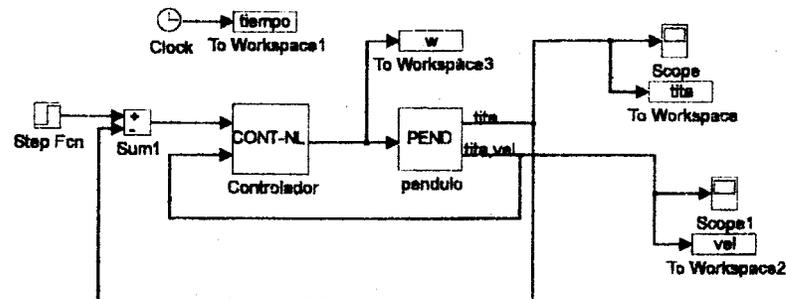


Figura 4.1.1: Esquema de control con realimentación

La función "Pendulo" contiene las ecuaciones de movimiento del sistema modelado -- ecuación (25) --, mientras que "Controlador" contiene las ecuaciones que definen al controlador -- ecuaciones (29), (30), (31) y (32).

Los parámetros utilizados en la simulación son :

- m : masa del péndulo = 2.0 kg
- M : masa del carro = 8.0 kg
- l : longitud del péndulo = 1.0 m
- a : aceleración de la gravedad = 9.8 m/s^2

El sistema es perturbado de su posición de equilibrio inestable mediante la aplicación de un torque T en la base del péndulo. ($T = 10[Nm] \forall t < 1$ y $T = 0.[Nm] \forall t > 1$). La figura 4.1.2 muestra la evolución de la posición y la velocidad angular a lo largo del tiempo. En la figura 4.1.3 se presenta la fuerza ejercida al carro para estabilizar el péndulo.

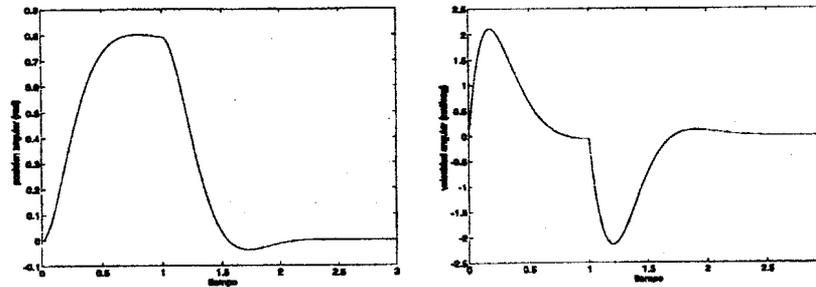


Figura 4.1.2: evolución de la posición y velocidad angular

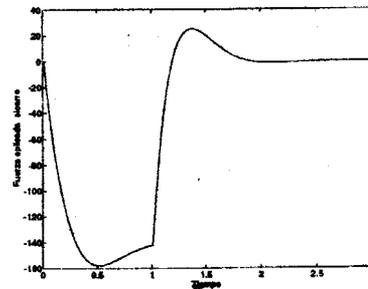


Figura 4.1.3: fuerza de control

La integración numérica en todos los casos es realizada utilizando el esquema de *Runge-Kutta* de 5° orden.

4.2 Empleo del programa MECANO

El programa *MECANO*, basado en el método de elementos finitos, ha sido concebido para el análisis de mecanismos y estructuras no lineales, es decir el estudio del sistema a lazo abierto. Para analizar problemas de control es necesario un trabajo adicional: el lazo puede cerrarse empleando el "elemento utilizador" a través del comando "SOL".

El objetivo de este punto es el de programar un algoritmo de control para el sistema analizado en la sección 3, empleando el programa multipropósito de simulación de mecanismos *MECANO* con "elementos" apropiados, al estilo de los usados para representar el sistema mecánico para poder verificar mediante simulación el correcto funcionamiento del sistema completo.

La función "user", dentro del programa *MECANO*, permite la creación de elementos con varios números de nodos y es posible determinar una ley en función del tiempo y dependiente de desplazamientos nodales, velocidades, y/o aceleraciones del elemento.

4.2.1 Modelo de elementos finitos y simulación del sistema

El carro está definido por un elemento *masa rígido*, se encuentra libre en el eje x , restringido en y y en las rotaciones ϕ_x y ϕ_y . El vínculo entre el carro y el péndulo está definido por un elemento *Hinge* (elemento propio del módulo *MECANO*, para definir vínculos). Estos están construidos en base a los nodos que definen el carro y el extremo del péndulo y utilizando un tercer nodo adicional sobre el cual se asigna la rotación relativa interna. El péndulo consta de un elemento *masa rígido* en su extremo –donde se concentra la masa –, libre en los ejes x, y y fijo en la rotación ϕ_z .

Sobre el carro se aplica una fuerza implementada por un elemento “fuerza” que adopta la consigna de mover a éste para que el péndulo se mantenga en posición vertical. Este elemento permite que la carga pueda ser expresada como función explícita del tiempo y de las posiciones de los grados de libertad seleccionados junto con las primeras derivadas (posición y velocidad).

El sistema de ecuaciones, en el elemento utilizador, adopta la forma de un vector de ecuaciones de segundo orden :

$$M\ddot{q} + g^{int} = g^{ext} \quad (34)$$

La salida de la ecuación empleada son las fuerzas internas f_{int} , fuerzas externas f_{ext} y la matriz de iteración S , cuyas expresiones están definidas por :

$$f_{int} = M\ddot{q} \quad (35)$$

$$f_{ext} = g^{int} \quad (36)$$

$$S = \alpha_1 K_T + \alpha_2 M + \alpha_3 C \quad (37)$$

Donde los valores de α_i son constantes definidas por de *MECANO*, función del algoritmo de integración y del paso de tiempo empleado. K_T y C son la matriz de rigidez y amortiguamiento respectivamente, definidas como :

$$K_T = \frac{\partial f_{int}(q)}{\partial q} \quad (38)$$

$$C = \frac{\partial f_{int}(q)}{\partial \dot{q}} \quad (39)$$

Se utiliza para el análisis de repuesta dinámica un esquema de integración predictor corrector implícito de *Hilber, Hughes y Taylor* con selección automática del paso del tiempo.

El sistema y la fuerza excitatriz son definidos de la misma manera que en la sección 4.1. La figura 4.2.1.1 muestra las posiciones y velocidades angulares que adopta el péndulo a lo largo del tiempo. En la figura 4.2.1.2 se presenta la fuerza ejercida al carro para estabilizar el péndulo.

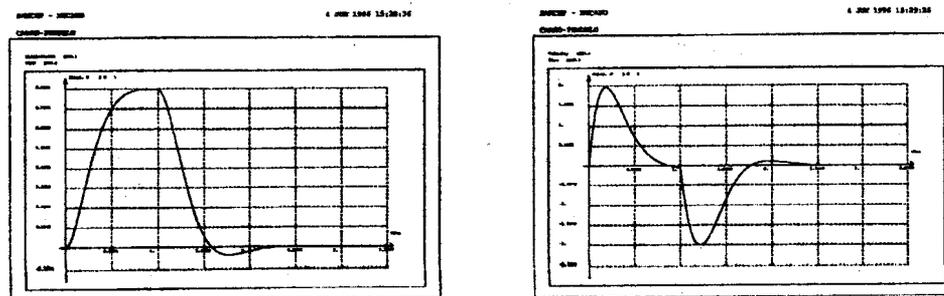


Figura 4.2.1.1: evolución de la posición y velocidad angular

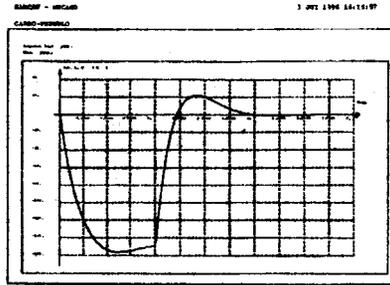


Figura 4.2.1.2: fuerza de control

5. Conclusión

Se estudió una teoría de control, aplicable a sistemas no lineales, que fue implementada en el programa de simulación de mecanismos *MECANO* utilizando un elemento “fuerza” para estabilizar y controlar un sistema carro-péndulo. La solución fue comparada con la obtenida en el programa *SIMULINK* obteniendo curvas idénticas.

En un paso posterior, el estudio proseguirá con la incorporación de nuevos elementos al programa *MECANO* con el objetivo de representar sistemas de control en forma modular -semejantes a los del programa *SIMULINK*- empleando elementos apropiados, al estilo de los usados para representar el sistema mecánico.

6. Referencias

- [1]. A. Isidori, *Nonlinear Control Systems*, Spinger-Verlag, segunda ed., 1989.
- [2]. *Simulink: A Program for Simulating Dynamic Systems*, The MathWorks Inc., 1992.
- [3]. *Samcef: Module d'Analyse de Mecanismes MECANO (Manuel d'Utilisation)*, A. Cardona y M. Gérardin, Facultad de Ciencias Aplicadas, Universidad de Lieja, 1987.
- [4]. F. Camusso y A. Cardona, *Determinación de las Ecuaciones de Movimiento y Análisis de la Estabilidad de un Sistema de Péndulo Invertido*, *Mecánica Computacional Vol XV*, p. 459-468, 1995.
- [5]. B. Char, K. Geddes, G. Gonnet, M. Monagan and S. Watt, *First Leaves for the Macintosh, a Tutorial Introduction to Maple*, Brooks/Cole Publishing Co., 1989.
- [6]. C. Kravaris, *Input/output linearization: a nonlinear analog of placing poles at the process zeros*, *AIChE J.*, vol. 34, p. 1803, 1988.
- [7]. C. Kravaris and C. Chung, *Nonlinear state feedback synthesis by global input/output linearization*, *AIChE J.*, vol. 33, p. 592, 1987.
- [8]. M. A. Henson and D. E. Seborg, *A critique of differential geometric control strategies for nonlinear proces control*, *IFAC World Congress, Tallinn, Estonia*, vol. , p. , 1990.
- [9]. M. A. Henson and D. E. Seborg, *Input-output linerization of general nonlinear processes*, *AIChE J.*, vol. 36, p. 1753, 1990.
- [10]. W. Baumann and W. Rugh, *Feedback control of nonlinear systems by extended linearization*, *IEEE Transactions on Automatic Control*, vol. AC-31, pp. 40-45, 1986.