

Simulaciones numéricas para la estabilización de sistemas mecánicos subactuados de grado uno, usando el método IDA-PBC: el caso del sistema TORA

Numerical simulations for stabilization of underactuation mechanical systems of degree one, using IDA-PBC method: the TORA system example

Maribel Cecilia Pérez Pirela (maribel.perez@fing.luz.edu.ve)

Centro de Investigación de Matemática Aplicada
Facultad de Ingeniería. Universidad del Zulia
Maracaibo, Venezuela, 4011.

Atilio Morillo Piña (morilloatlio@gmail.com)

Centro de Investigación de Matemática Aplicada
Facultad de Ingeniería. Universidad del Zulia
Maracaibo, Venezuela, 4011.

Resumen

El problema de la estabilización de sistemas no lineales subactuados ha atraído la atención de la comunidad de control en años recientes. Con el método denominado IDA-PBC (Interconexión y Asignación de Amortiguamiento Control basado en Pasividad), desde el punto de vista teórico, se ha logrado describir el comportamiento dinámico de una amplia clase de dichos sistemas, obteniéndose su representación en la denominada forma Hamiltoniana controlada por puertos, mediante la cual se facilita el diseño de un controlador por realimentación que permite estabilizarlos en torno a un punto de equilibrio deseado. El objetivo general de este estudio es analizar la estabilidad de sistemas mecánicos subactuados de grado 1 mediante el método IDA-PBC. Dentro de este enfoque, para lograr el objetivo de control, se interpreta el mecanismo de estabilización en términos del intercambio de la energía del sistema, para lo cual se siguen dos etapas básicas: (1) la etapa del moldeado de la energía, la cual consiste en modificar la función de energía total del sistema para asignar el estado de equilibrio deseado; y (2) la etapa de inyección de amortiguamiento para alcanzar la estabilidad asintótica. El éxito de la aplicación de este método reside en la posibilidad de resolver el conjunto de ecuaciones en derivadas parciales, cuyas soluciones proveen las funciones de energía asignables al sistema en lazo cerrado. El sistema TORA (“translational oscillator with rotational actuator”) es un prototipo de sistema mecánico subactuado que ha merecido gran atención por la comunidad de control no lineal, y en este trabajo, partiendo de la representación Hamiltoniana controlada por puertos basada en la energía total del sistema considerada como energía cinética más energía potencial, se obtiene un controlador que logra estabilizar en forma global y asintótica el punto de equilibrio alcanzando un excelente desempeño. Las simulaciones numéricas mostradas al final del trabajo confirman esta

Recibido 1/2010. Revisado 2/2010. Aceptado 4/2010.

MSC (2010): Primary 93C10; Secondary 93C83.

Autor de correspondencia: Atilio Morillo

apreciación.

Palabras y frases clave: Control no lineal, Sistemas mecánicos subactuados, Método IDA-PBC, sistema Tora.

Abstract

The problem of the stabilization of not linear systems underactuation has attracted the attention of the community of control in the recent years. The so called IDA-PBC method (Interconnection and Damping Assignment Passivity Based Control), from the theoretical point of view, it has been achieved to describe the dynamic behavior of a wide class of the above mentioned systems, obtained a port controlled Hamiltonian form, the controller stabilizes globally and asymptotically the equilibrium point. The general objective of this study is to analyze the stabilization of mechanical systems underactuation degree one using IDA-PBC method. In this method, in order to achieve the control objective, the stabilization mechanism follows two basic stages: (1) energy holding stage, which consists on shaping the total energy function of the system in order to assign the desired equilibrium state, and (2) damping introduction stage, necessary to achieve asymptotic stability. The success of the application of this method resides in the possibility of solving the set of equations in partial derivatives, which solutions provide the assignable functions of energy to the system in closed loop. The TORA system (“translational oscillator with rotational actuator”) is a prototype of a underactuated mechanical system widely studied by the non linear control community. In this paper, a controller is designed taking into account the port controlled Hamiltonian approach based on the total energy of the system, considered as the sum of kinetic and potential energies, the controller stabilizes globally and asymptotically the equilibrium point, showing an excellent performance. The numerical simulations confirm this appreciation.

Key words and phrases: Non linear control, underactuated mechanical systems, IDA-PBC method, Tora system.

1 Introducción

El método IDA-PBC persigue una dinámica en bucle cerrado con función de Hamilton $H_d(q, p)$ y una matriz antisimétrica también llamada de interconexión generalizada de la forma $J_d(q, p) = -J_d(q, p)^T$ que permite aumentar los grados de libertad en el diseño. Las ecuaciones de estado en lazo abierto y cerrado se deben ajustar exactamente. Esto quiere decir que la ley de control u debe calcularse de modo que

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + Gu = \begin{bmatrix} J_d & -R_d \end{bmatrix} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \end{bmatrix} \quad (1)$$

donde $R_d(q) \geq 0$ es la matriz de disipación en bucle cerrado. Las principales dificultades de este método aparecen en el caso de sistemas subactuados, donde el conjunto de funciones de Hamilton H_d alcanzables en bucle cerrado es limitado, y depende de la resolubilidad de un sistema de ecuaciones diferenciales parciales. En efecto, en el caso subactuado existe una matriz de rango $m < n$ siendo n el número de grados de libertad, que representa las direcciones en las que la ley de control no tiene efecto, cumpliéndose que $G^\perp G = 0$, es decir, si G es una matriz constante, las filas de G^\perp forman el núcleo de G . Si se premultiplica (1) por G^\perp se obtiene:

$$G^\perp \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} = G^\perp \begin{bmatrix} J_d & -R_d \end{bmatrix} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \end{bmatrix} \quad (2)$$

Esta ecuación ha de cumplirse para cualquier valor de la ley de control y, por lo tanto, representa una restricción en el conjunto de sistemas hamiltonianos alcanzables en bucle cerrado definidos por las matrices (H_d, J_d, R_d) . Una correcta elección de los parámetros (H_d, J_d, R_d) debe ser compatible con estas ecuaciones de ajuste y al mismo tiempo representar una dinámica en lazo cerrado con las propiedades deseadas en términos de estabilidad. Proporcionar métodos de cálculo de las (H_d, J_d, R_d) adecuadas y de leyes de control para el ajuste lazo abierto-lazo cerrado es la esencia del método IDA-PBC.

En este trabajo se considera el problema de la estabilización asintótica del sistema TORA utilizando una ley de control sintetizada mediante la aplicación de la metodología IDA-PBC, con la cual se obtiene regulación y estabilización por realimentación de la salida del sistema estudiado.

En la sección 2 se describe el modelo matemático del Sistema Tora, y en la sección 3 se presenta el análisis y diseño del controlador que logra estabilizar el sistema, utilizando el Método IDA-PBC aplicado a sistemas mecánicos subactuados desarrollado en [1]. En la sección 4 se presenta la descripción y construcción de los programas de simulación numérica para el sistema estudiado. En la sección 5 se presentan algunas simulaciones numéricas que verifican la eficiencia del controlador diseñado, y por último, en la sección 6 se presentan las conclusiones del trabajo.

2 Método Matemático del Sistema TORA

El denominado sistema TORA (translational oscillator with rotational actuator) fue introducido por primera vez en [2]. La Figura 1 ilustra el sistema TORA consistente de una plataforma oscilante traslacionalmente de masa m_1 , la cual es controlada vía una masa rotacional excéntrica de masa m_2 . El problema es de interés como un caso de estudio en el diseño de controles no lineales, debido a que el modelo exhibe una interacción no lineal entre sus movimientos traslacional y rotacional.

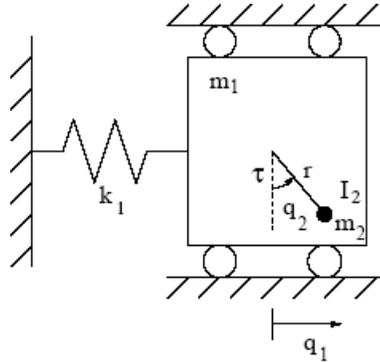


Figura 1: *El Sistema TORA*

La matriz de inercia del sistema posee la forma

$$M = \begin{bmatrix} m_1 + m_2 & m_2 r \cos(q_2) \\ m_2 r \cos(q_2) & m_2 r^2 + I \end{bmatrix} \quad (3)$$

donde q_2 es el ángulo de rotación de la masa m_2 , r es su radio de excentricidad, e I es el momento de inercia. Si denotamos por q_1 la posición generalizada del sistema, por $q = [q_1 \ q_2]^T$, por g la constante de gravedad, y por $V(q_1, q_2)$ la energía potencial de la masa m_2 , el Lagrangiano del sistema viene expresado como:

$$L(q, \dot{q}) = \frac{1}{2} [\dot{q}_1 \ \dot{q}_2] M(q_2) \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} - V(q_1, q_2) \quad (4)$$

con energía potencial dada por

$$V(q_1, q_2) = \frac{1}{2} K q_1^2 + m_2 g r \cos(q_2). \quad (5)$$

siendo K la constante de rigidez del resorte.

Si denotamos por τ la fuerza actuadora sobre el punto de giro de m_2 , las ecuaciones de Euler-Lagrange para el sistema TORA adoptan la forma:

$$\begin{cases} (m_1 + m_2)\ddot{q}_1 + m_2 r \cos(q_2)\ddot{q}_2 - m_2 r \sin(q_2)\dot{q}_2^2 + K q_1 = 0 \\ m_2 r \cos(q_2)\ddot{q}_1 + (m_2 r^2 + I)\ddot{q}_2 + m_2 g r \sin(q_2) = \tau \end{cases} \quad (6)$$

resultando así un sistema con 2 grados de libertad, con grado de subactuación 1, y con q_2 como coordenada actuada. De tal manera que se satisfacen las hipótesis H_1 , H_2 y H_3 , con $G = [0 \ 1]^T$.

Denotando $c_1 = m_1 + m_2$, $c_2 = m_2 r$, $c_3 = m_2 r^2 + I$ y definiendo el momento de inercia generalizado mediante $p = M\dot{q}$, la matriz de inercia M se escribe como:

$$M(q_2) = \begin{bmatrix} c_1 & c_2 \cos(q_2) \\ c_2 \cos(q_2) & c_3 \end{bmatrix} \quad (7)$$

donde, a fin de obtener $M(q_2)$ inversible, debe asumirse la siguiente relación entre los parámetros:

$$c_1 c_3 - c_2^2 > 0 \quad (8)$$

3 Estabilización del Sistema TORA

3.1 Realización Hamiltoniana.

El sistema Hamiltoniano Generalizado es:

$$\dot{x} = J\nabla H + G(x)^* u \quad (9)$$

donde la matriz J es antisimétrica, siendo la más sencilla:

$$J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \quad (10)$$

Si se asume que el sistema no posee amortiguamiento natural, las ecuaciones del movimiento pueden escribirse como:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + \begin{bmatrix} 0 \\ G \end{bmatrix} u \quad (11)$$

donde I_n es la matriz identidad $n \times n$, $\nabla_q H$, $\nabla_p H$ son los vectores columna gradiente de HH respecto de q y respecto de p , y $u \in \mathbb{R}^n$ es la función de control. La matriz $G \in \mathbb{R}^{n \times m}$ es determinada por la manera como el control $u \in \mathbb{R}^m$ ingresa en el sistema, y es invertible en el caso que el sistema sea completamente actuado, o sea, $n = m$. En esta investigación se considera el caso más difícil en que el sistema es subactuado, es decir, menos controles que grados de libertad, y asumimos que $\text{rango}(G) = m$.

Para el sistema TORA $n = 2$ y $m = 1 \rightarrow m < n \rightarrow$ Sistema subactuado de grado 1.

La energía total del sistema viene dada por: $H = \text{Energía Cinética} + \text{Energía Potencial}$

$$H = \frac{1}{2} p^T M^{-1} p + V(q) \quad (12)$$

donde $q \in \mathbb{R}^n$, $p \in \mathbb{R}^n$ representan la posición generalizada y el momento generalizado, $M(q) = M^T(q) > 0$ es la matriz de inercia del sistema, y $V(q)$ es la energía potencial.

Para el sistema TORA la matriz M es la matriz de inercia y viene dada por la ecuación (7) como sigue:

$$M = \begin{bmatrix} c_1 & c_2 \cos(q_2) \\ c_2 \cos(q_2) & c_3 \end{bmatrix}$$

y la energía potencial viene dada por la ecuación (5) como sigue:

$$V(q) = \frac{1}{2} K q_1^2 + m_2 g L (1 - \cos(q_2)).$$

Se busca con el método IDA-PBC hacer coincidir el comportamiento del sistema en lazo abierto con la dinámica objetivo.

La dinámica objetivo según éste método viene dada por:

$$\dot{x} = (J_d - R_d) \nabla H_d \quad (13)$$

donde J_d es la matriz de interconexión y R_d la matriz de amortiguamiento, cuyas estructuras son las siguientes:

$$J_d = \begin{bmatrix} 0 & M^{-1} M_d \\ -M_d M^{-1} & J_2(q, p) \end{bmatrix}, \quad R_d = \begin{bmatrix} 0 & 0 \\ 0 & G K_v G^T \end{bmatrix}$$

siendo M_d la matriz de inercia deseada cuya estructura es:

$$M_d = \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix}$$

3.2 Moldeado de la energía.

En la teoría de control, basado en pasividad (PBC), la entrada de control usualmente se descompone en dos términos (véase [3]).

$$u = u_{es}(q, p) + u_{di}(q, p) \quad (14)$$

donde el primer término es designado para alcanzar el moldeado de la energía, mientras que a través del segundo término se introduce amortiguamiento al sistema. Por otro lado, como es bien sabido, el aporte de amortiguamiento en sistemas pasivos se logra vía realimentación negativa de

la nueva salida pasiva, la cual en este caso viene dada por $G^T \nabla_p H_d$. Esta es la razón por la cual se elige para el término u_{di} de la ecuación (14) la expresión:

$$u_{di} = K_v G^T \nabla_p H_d \quad (15)$$

donde se toma $K_v = K_v^T > 0$. Esto justifica el bloque (2,2) en la definición de R_d .

Al igualar el sistema en lazo abierto y la dinámica objetivo, resulta lo siguiente:

$$\begin{aligned} \begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} &= \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + \begin{bmatrix} 0 \\ G \end{bmatrix} u = \\ &= \left\{ \begin{bmatrix} 0 & M^{-1} M_d \\ -M_d M^{-1} & J_2(q, p) \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & G K_v G^T \end{bmatrix} \right\} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \end{bmatrix} \end{aligned} \quad (16)$$

donde H_d es la energía total deseada que mantiene la misma forma de la energía total original del sistema y esta expresada por:

$$H_d = \frac{1}{2} P^T M_d^{-1} P + V(q) \quad (17)$$

Para obtener el término de moldeado de la energía en el controlador, se reemplazan (14) y (15) en (16) y se obtiene:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H \\ \nabla_p H \end{bmatrix} + \begin{bmatrix} 0 \\ G \end{bmatrix} u_{es} = \begin{bmatrix} 0 & M^{-1} M_d \\ -M_d M^{-1} & J_2(q, p) \end{bmatrix} \begin{bmatrix} \nabla_q H_d \\ \nabla_p H_d \end{bmatrix} \quad (18)$$

donde el término R_d de (16) se ha cancelado con el término de (15).

La primera fila de la ecuación (18) produce una identidad para \dot{q} como sigue a continuación:

$$\begin{aligned} \dot{q} &= V_p H = M^{-1} M_d \nabla_p H_d \\ \dot{q} &= M^{-1} p = M^{-1} M_d M_d^{-1} p = M^{-1} p \end{aligned}$$

Con lo que se consigue que la primera ecuación es una igualdad.

De tal manera que \dot{q} equivale a:

$$\dot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = M^{-1} p \quad (19)$$

donde $p = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$ y M es la matriz de inercia dada por la ecuación (7), entonces se procede a calcular M^{-1} como sigue:

$$M^{-1} = \frac{1}{\delta_1} = \begin{bmatrix} c_3 & -c_2 \cos(q_2) \\ -c_2 \cos(q_2) & c_1 \end{bmatrix} \quad (20)$$

donde $\delta_1 = c_1 c_3 - (c_2 \cos(q_2))^2$.

Se sustituye las ecuaciones anteriores en (19) resultando lo siguiente:

$$\dot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = M^{-1} p = \frac{1}{\delta_1} \begin{bmatrix} c_3 & -c_2 \cos(q_2) \\ -c_2 \cos(q_2) & c_1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$$

donde la primera ecuación correspondiente a \dot{q}_1 es la siguiente:

$$\dot{q}_1 = \frac{1}{\delta_1} [c_3 p_1 - c_2 p_2 \cos(q_2)] \quad (21)$$

y la segunda ecuación correspondiente a \dot{q}_2 es la siguiente:

$$\dot{q}_2 = \frac{1}{\delta_1} [-c_2 p_1 \cos(q_2) + c_1 p_2] \quad (22)$$

Ahora se resuelve la segunda fila de la ecuación (18) correspondiente a \dot{p} como sigue a continuación:

$$G u_{es} = \nabla_q H - M_d M^{-1} \nabla_q H_d + J_2 M_d^{-1} p \quad (23)$$

Si el sistema fuese subactuado, o sea si G fuese invertible, para determinar u_{es} bastaría premultiplicar por G^{-1} . En el caso de estudio el sistema es subactuado, luego G ya no es invertible, sino a lo sumo de rango por columnas máximo, y por lo tanto el control u_{es} únicamente ejerce influencia sobre los términos en el espacio imagen del operador G . Esta observación conduce al siguiente conjunto de ecuaciones de restricción, las cuales deben satisfacerse para cualquier escogencia de u_{es} .

$$G^\perp \{ \nabla_q H - M_d M^{-1} \nabla_q H_d + J_2 M_d^{-1} p \} = 0 \quad (24)$$

donde G^\perp es un anulador izquierdo de rango máximo de G (o sea, $GG^\perp = 0$).

La ecuación (24) es un conjunto de ecuaciones diferenciales parciales (EDP) no lineales con incógnitas M_d y V_d , y con J_2 siendo un parámetro libre, mientras que p es una coordenada independiente. Si puede obtenerse una solución para esta ecuación, la ley de control resultante u_{es} vendría dada por:

$$u_{es} = (G^T G)^{-1} (\nabla_q H - M_d M^{-1} \nabla_q H_d + J_2 M_d^{-1} p) \quad (25)$$

Las ecuaciones en derivadas parciales (24) pueden de manera natural ser separadas en términos que dependen de p y términos que son independientes de p , o sea, aquéllos que corresponden a la energía cinética, y aquéllos que corresponden a la energía potencial, respectivamente. En tal sentido, la ecuación (24) es equivalente al par de ecuaciones:

$$G^\perp \{ \nabla_q (p^T M^{-1} p) - M_d M^{-1} \nabla_q (p^T M^{-1} p) + 2J_2 M_d^{-1} p \} = 0 \quad (26)$$

$$G^\perp \{ \nabla_q V - M_d M^{-1} \nabla_q V \} = 0 \quad (27)$$

La primera ecuación es una ecuación en derivadas parciales no lineal que debe ser resuelta para los elementos desconocidos de la matriz de inercia en lazo cerrado M_d . Conocida esta M_d , la ecuación (27) es una ecuación en derivadas parciales lineal, y por lo tanto más fácil de resolver, de manera que la mayor dificultad está en resolver (26). Una simplificación, que disminuye considerablemente las dificultades, es asumir la existencia de una matriz de inercia M_d de términos constantes, lo cual permite concentrarse únicamente en hallar la energía potencial deseada V_d .

Afortunadamente, esta simplificación puede llevarse a cabo en el caso de algunos sistemas subactuados, en los cuales la matriz de inercia es función solamente de la coordenada actuada.

En esta investigación, se sigue la orientación de [1] y [4], donde para obtener una reducción de las EDP (26) y (27), se plantean las siguientes hipótesis:

Hipótesis H1: El sistema posee grado de subactuación 1, es decir, $m = n - 1$.

Hipótesis H2: La matriz de inercia depende solamente de la coordenada actuada.

Hipótesis H3: El sistema posee dos grados de libertad, y, sin pérdida de generalidad, la matriz G viene dada por $G = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$.

La hipótesis $H3$ es crucial en el presente desarrollo. Las hipótesis $H1$ y $H2$ garantizan que el término $G^T \nabla_q (p^T M_p^{-1} p)$ en la EDP (26) es nulo. En tal caso (26) puede ser resuelta para una matriz constante M_d , para lo cual, sobre la base de su libertad de escogencia, basta tomar $J_2 = 0$. Esto permite concentrarse en el moldeado de la energía potencial solamente, de manera que la EDP a resolver se reduce a:

$$G^\perp \{M_d M^{-1} \nabla V_d\} = 0 \quad (28)$$

De esta forma la segunda fila de la ecuación (18) queda:

$$\dot{p} = -\nabla_q H + G u_{es} = -M_d M^{-1} \nabla_q H_d \quad (29)$$

y al sustituir quedaría:

$$\begin{aligned} \dot{p} &= \begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \end{bmatrix} = - \begin{bmatrix} \nabla_{q_1} H \\ \nabla_{q_2} H \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{es} = \\ &= - \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix} \frac{1}{\delta_1} \begin{bmatrix} c_3 & -c_2 \cos(q_2) \\ -c_2 \cos(q_2) & c_1 \end{bmatrix} \begin{bmatrix} \nabla_{q_1} H_d \\ \nabla_{q_2} H_d \end{bmatrix}. \end{aligned}$$

Además como la matriz M_d es constante, entonces $\nabla_{q_1} H_d = \nabla_{q_1} V_d$ y $\nabla_{q_2} H_d = \nabla_{q_2} V_d$

$$\begin{aligned} \dot{p} &= \begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \end{bmatrix} = - \begin{bmatrix} \nabla_{q_1} H \\ \nabla_{q_2} H \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_{es} = \\ &= - \frac{1}{\delta_1} \begin{bmatrix} a_1 c_3 - a_2 c_2 \cos(q_2) & -a_1 c_2 \cos(q_2) + a_2 c_1 \\ a_2 c_3 - a_3 c_2 \cos(q_2) & -a_2 c_2 \cos(q_2) + a_3 c_1 \end{bmatrix} \begin{bmatrix} \nabla_{q_1} V_d \\ \nabla_{q_2} V_d \end{bmatrix}. \end{aligned} \quad (30)$$

La primera fila de \dot{p} es \dot{p}_1 :

$$\begin{aligned} \dot{p}_1 &= -\nabla_{q_1} H = \\ &= -\frac{1}{\delta_1} [(a_1 c_3 - a_2 c_2 \cos(q_2)) \nabla_{q_1} V_d + (-a_1 c_2 \cos(q_2) - a_2 c_1) \nabla_{q_2} V_d] \end{aligned} \quad (31)$$

$\nabla_{q_1} H = K q_1$ se sustituye en (31) y se tiene la EDP que permite la búsqueda de la V_d .

$$K q_1 = -\frac{1}{\delta_1} [(a_1 c_3 - a_2 c_2 \cos(q_2)) \nabla_{q_1} V_d + (-a_1 c_2 \cos(q_2) - a_2 c_1) \nabla_{q_2} V_d] \quad (32)$$

Claramente el sistema TORA satisface las hipótesis $H1$, $H2$ y $H3$ comentadas anteriormente, por lo tanto, la atención puede concentrarse en la resolución de la EDP de la ecuación (32). Para ello se define la matriz M_d mediante:

$$M_d = \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix}, \quad \text{con } a_1 > 0 \quad \text{y} \quad a_1 a_3 - a_2^2 > 0 \quad (33)$$

La ecuación (32) puede reescribirse como:

$$\begin{bmatrix} a_1 c_3 - a_2 c_2 \cos(q_2) \\ a_2 c_1 - a_1 c_2 \cos(q_2) \end{bmatrix} \nabla_{q_1} V_d + \nabla_{q_2} V_d = \begin{bmatrix} \delta_1 \\ a_2 c_1 - a_1 c_2 \cos(q_2) \end{bmatrix} K q_1 \quad (34)$$

La ecuación anterior puede intentar resolverse usando el comando *pdsolve* del programa Maple pero se obtiene una solución extremadamente complicada, válida para cualquier conjunto de valores de los parámetros a_1 , a_2 y a_3 . Para resolver esta dificultad, se recurre a un subconjunto de valores posibles para estos parámetros que reducen la complejidad de la ecuación.

Se denota:

$$\frac{\gamma_1}{\gamma_2} = \frac{a_1 c_3 - a_2 c_2 \cos(q_2)}{a_2 c_1 - a_1 c_2 \cos(q_2)} = \frac{b_3 + b_4 \cos(q_2)}{b_1 + b_2 \cos(q_2)} \quad (35)$$

donde $b_1 = a_2 c_1$, $b_2 = -a_1 c_2$, $b_3 = a_1 c_3$ y $b_4 = -a_2 c_2$. Una simple división en el término de la derecha de (35) da lugar a:

$$\frac{\gamma_1}{\gamma_2} = \frac{b_4}{b_2} + \left(\frac{b_3 b_2 - b_1 b_4}{b_2} \right) \frac{1}{b_1 + b_2 \cos(q_2)} \quad (36)$$

de manera que se obtiene:

$$\frac{\gamma_1}{\gamma_2} = \frac{b_4}{b_2} \Leftrightarrow b_3 b_2 - b_1 b_4 = 0, \quad (37)$$

es decir,

$$\frac{\gamma_1}{\gamma_2} = \frac{a_2}{a_1} \Leftrightarrow a_2 = \pm \sqrt{\frac{c_3}{c_1}} a_1 = \alpha a_1. \quad (38)$$

En lo sucesivo, para simplificar, se toma únicamente el valor positivo de la constante α sobre el conjunto:

$$\{(a_1, a_2, a_3) \in \mathbb{R}^3 : a_1, a_2, \text{ y } a_3 \text{ satisfacen (38) y además, } a_2 = \alpha a_1\} \quad (39)$$

Sustituyendo $a_2 = \alpha a_1$ y $\delta_1 = c_1 c_3 - (c_2 \cos(q_2))^2$ en la ecuación (34) y tomando en cuenta que $\alpha = \sqrt{\frac{c_3}{c_1}}$, finalmente se consigue la EDP para hallar la V_d :

$$\frac{\alpha}{q_1} \nabla_{q_1} V_d + \frac{1}{q_1} \nabla_{q_2} V_q = \frac{K}{a_1} [\sqrt{c_1 c_3} + c_2 \cos(q_2)] \quad (40)$$

La aplicación del método de las características a la ecuación (40) conduce a la solución de la EDP. Se toma como condición inicial:

$$\gamma : q_1(s) = s, q_2(s) = 0 \text{ y } V_d(s) = f(s).$$

Para asignar el punto de equilibrio en el origen a la función V_d puede elegirse la función $f(s)$ como $f(s) = \frac{1}{2} R_d (q_1 - \alpha q_2)^2$, donde se ha introducido R_d como un parámetro de diseño y con lo cual se obtiene finalmente para la energía potencial deseada:

$$V_d = \frac{1}{2} R_d (q_1 - \alpha q_2)^2 + \frac{K}{a_1} \left(-\frac{\alpha q_2^2}{2} \sqrt{c_1 c_3} + q_1 q_2 \sqrt{c_1 c_3} + \alpha c_2 \cos(q_2) - \alpha c_2 + c_2 q_1 \sin(q_2) \right) \quad (41)$$

Se sabe por el Principio de Lagrange que:

$$\left\{ \begin{array}{c} \text{Puntos} \\ \text{de equilibrio} \\ \text{del sistema} \end{array} \right\} = \left\{ \begin{array}{c} \text{Puntos} \\ \text{críticos de} \\ V_d \end{array} \right\} \text{ y } \left\{ \begin{array}{c} \text{Puntos} \\ \text{de equilibrio} \\ \text{estables} \end{array} \right\} = \left\{ \begin{array}{c} \text{Puntos} \\ \text{mínimos de} \\ V_d \end{array} \right\}$$

Se procede a calcular los puntos críticos de la función V_d , es decir, $\nabla_q V_d = 0$

$$\begin{aligned}\frac{\partial V_d}{\partial q_1} &= R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_3} q_2 + c_2 \operatorname{sen}(q_2))}{a_1} \\ \frac{\partial V_d}{\partial q_2} &= -R_d(q_1 - \alpha q_2)\alpha + \frac{K(-\alpha\sqrt{c_1 c_3} q_2 + \sqrt{c_1 c_3} q_1 - \alpha c_2 \operatorname{sen}(q_2) + c_2 q_1 \cos(q_2))}{a_1} \\ \frac{\partial V_d}{\partial q_1} &= \frac{\partial V_d}{\partial q_2} = 0 = \nabla_q H_d = \nabla_q V_d \\ R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_3} q_2 + c_2 \operatorname{sen}(q_2))}{a_1} &= 0 \\ -R_d(q_1 - \alpha q_2)\alpha + \frac{K(-\alpha\sqrt{c_1 c_3} q_2 + \sqrt{c_1 c_3} q_1 - \alpha c_2 \operatorname{sen}(q_2) + c_2 q_1 \cos(q_2))}{a_1} &= 0\end{aligned}$$

Resultando ser $q_1 = 0$ y $q_2 = 0$, de manera que son puntos críticos de V_d y por lo tanto, puntos de equilibrio del sistema TORA.

Se evalúa la función V_d en $(0, 0)$, entonces $V_d(0, 0) = \alpha c_2 - \alpha c_2 = 0$. Para verificar si el punto $(0, 0)$ es un punto mínimo se procede a calcular el Hessiano:

$$\begin{aligned}Hess(V_d) &= \begin{bmatrix} \frac{\partial^2 V_d}{\partial q_1^2} & \frac{\partial^2 V_d}{\partial q_1 \partial q_2} \\ \frac{\partial^2 V_d}{\partial q_1 \partial q_2} & \frac{\partial^2 V_d}{\partial q_2^2} \end{bmatrix}_{(0,0)} \\ Hess(V_d) &= \begin{bmatrix} R_d & -R_d\alpha + \frac{K\sqrt{c_1 c_3} + Kc_2}{a_1} \\ -R_d\alpha + \frac{K\sqrt{c_1 c_3} + Kc_2}{a_1} & R_d\alpha^2 + \frac{K\alpha\sqrt{c_1 c_3} + K\alpha c_2}{a_1} \end{bmatrix}\end{aligned}$$

Para que exista un mínimo en la función V_d los menores deben ser positivos:

- Determinante 1: $R_d > 0$.
- Determinante 2: $-R_d\alpha \left(\frac{-R_d\alpha + K\sqrt{c_1 c_3} - Kc_2}{a_1} \right) - \left(-R_d\alpha + \frac{K\sqrt{c_1 c_3} + Kc_2}{a_1} \right)^2 > 0$. Al resolver resulta lo siguiente

$$R_d < 2 \frac{(K\sqrt{c_1 c_3} + Kc_2)}{a_1 \alpha} \quad (42)$$

Por lo tanto, si se cumple que $R_d < 2 \frac{(K\sqrt{c_1 c_3} + Kc_2)}{a_1 \alpha}$, entonces el $Hess(V_d(0, 0))$ es definido positivo, así $(0, 0)$ es un mínimo y es un punto de equilibrio estable.

Una vez encontrada la V_d correcta se procede a determinar la función de control u_{es} que permita estabilizar el sistema TORA.

Se retoma la ecuación (18) para despejar la u_{es} que logre que el sistema en lazo cerrado tenga el mismo comportamiento de la dinámica objetivo. Para ello se toma la segunda fila expresada por la ecuación (29) como sigue:

$$\dot{p} = -\nabla_q H + G u_{es} = -M_d M^{-1} \nabla_q H_d$$

$G u_{es} = \nabla_q H - M_d M^{-1} \nabla_q H_d$, tomando en cuenta que G no es inversible al resolver se obtiene lo siguiente:

$$u_{es} = -\frac{1}{\delta_1} [\alpha a_1 c_3 - a_3 c_2 \cos(q_2)] \nabla_{q_1} V_d + (-\alpha a_1 c_2 \cos(q_2) + a_3 c_1) \nabla_{q_2} V_d + \nabla_{q_2} H \quad (43)$$

Al sustituir los cálculos correspondientes a $\frac{\partial V_d}{\partial q_1}$, $\frac{\partial V_d}{\partial q_2}$ y $\frac{\partial H}{\partial q_2}$ en la ecuación (43) resulta al simplificar:

$$\begin{aligned} u_{es} = & -\frac{(\alpha a_1 c_3 - a_3 c_2 \cos(q_2))}{\delta_1} \left[R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_2} q_2 + c_2 \sin(q_2))}{a_1} \right] - \\ & -\frac{(-\alpha a_1 c_2 \cos(q_2) + a_3 c_1)}{\delta_1} \left(-R_d(q_1 - \alpha q_2) \alpha + \frac{K}{a_1} (\sqrt{c_1 c_3} (q_1 - \alpha q_2) - \alpha c_2 \sin(q_2) + c_2 q_1 \cos(q_2)) \right) - \\ & -\frac{c_2^2 \sin(q_2) \cos(q_2)}{\delta_1} [c_3 p_1^2 - 2c_2 p_1 p_2 \cos(q_2) + c_1 p_2^2] + \frac{c_2 p_1 p_2 \sin(q_2)}{\delta_1} + m_2 g L \sin(q_2). \end{aligned} \quad (44)$$

3.3 Inyección de Amortiguamiento

Una vez encontrada la u_{es} , se procede a calcular la función de control u que garantiza la estabilización asintótica del sistema TORA. Se sabe que $u = u_{es}(q, p) + u_{di}(q, p)$ y que $u_{di} = -kv G^T \nabla_p H_d$ de tal manera que:

$$\begin{aligned} \nabla_p H_d &= M_d^{-1} p \\ u_{di} &= -kv [0 \quad 1] M_d^{-1} p \end{aligned}$$

Como la matriz de inercia deseada tiene la forma:

$$M_d = \begin{bmatrix} a_1 & \alpha a_1 \\ \alpha a_1 & a_3 \end{bmatrix} \text{ entonces } M_d^{-1} = \frac{1}{\delta_2} \begin{bmatrix} a_3 & -\alpha a_1 \\ -\alpha a_1 & a_1 \end{bmatrix} \text{ donde } \delta_2 = a_1 a_3 - \alpha^2 a_1^2.$$

La inyección de amortiguamiento viene dada por:

$$u_{di} = -\frac{kv}{\delta_2} [0 \quad 1] \begin{bmatrix} a_3 & -\alpha a_1 \\ -\alpha a_1 & a_1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$$

al resolver se obtiene:

$$u_{di} = -\frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]$$

Finalmente la función de control que garantiza que el sistema TORA en lazo cerrado se comporte igual que la dinámica objetivo propuesto por método IDA-PBC y que logra la estabilidad asintótica del sistema.

$$u = u_{es} - \frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]$$

u_{es} es la calculada y corresponde a la ecuación (44). Sustituyendo u_{es} se consigue que la función de control u es la siguiente:

$$\begin{aligned} u = & -\frac{(\alpha a_1 c_3 - a_3 c_2 \cos(q_2))}{\delta_1} \left[R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_2} q_2 + c_2 \sin(q_2))}{a_1} \right] - \\ & -\frac{(-\alpha a_1 c_2 \cos(q_2) + a_3 c_1)}{\delta_1} \left(-R_d(q_1 - \alpha q_2) \alpha + \frac{K}{a_1} (\sqrt{c_1 c_3} (q_1 - \alpha q_2) - \alpha c_2 \sin(q_2) + c_2 q_1 \cos(q_2)) \right) - \end{aligned}$$

$$-\frac{c_2^2 \operatorname{sen}(q_2) \cos(q_2)}{\delta_1^2} [c_3 p_1^2 - 2c_2 p_1 p_2 \cos(q_2) + c_1 p_2^2] \frac{c_2 p_1 p_2 \operatorname{sen}(q_2)}{\delta_1} + m_2 g L \operatorname{sen}(q_2) - \frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]. \quad (45)$$

Una vez encontradas cada una de las funciones necesarias para la aplicación del método IDA-PBC, es posible presentar el sistema TORA en lazo cerrado, sustituyendo $\nabla_{q_2} H$ y u :

$$\begin{aligned} \dot{q}_1 &= \frac{1}{\delta_1} [c_3 p_1 - c_2 p_2 \cos(q_2)] & \dot{q}_2 &= \frac{1}{\delta_1} [-c_2 p_1 \cos(q_2) + c_1 p_2] \\ \dot{p}_1 &= -K q_1 & \dot{p}_2 &= -\nabla_{q_2} H + u \end{aligned}$$

Sustituyendo $\nabla_{q_2} H$ y u en el sistema de EDO anterior se tiene que:

$$\begin{aligned} \dot{q}_1 &= \frac{1}{\delta_1} [c_3 p_1 - c_2 p_2 \cos(q_2)] & \dot{q}_2 &= \frac{1}{\delta_1} [-c_2 p_1 \cos(q_2) + c_1 p_2] & \dot{p}_1 &= -K q_1 \\ \dot{p}_2 &= -\frac{(\alpha a_1 c_3 - a_3 c_2 \cos(q_2))}{\delta_1} \left[R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_3} q_2 + c_2 \operatorname{sen}(q_2))}{a_1} \right] - \\ &-\frac{(-\alpha a_1 c_2 \cos(q_2) + a_3 c_1)}{\delta_1} \left(-R_d(q_1 - \alpha q_2) \alpha + \frac{K}{a_1} (\sqrt{c_1 c_3} (q_1 - \alpha q_2) - \alpha c_2 \operatorname{sen}(q_2) + c_2 q_1 \cos(q_2)) \right) - \\ &-\frac{c_2^2 \operatorname{sen}(q_2) \cos(q_2)}{\delta_1^2} [c_3 p_1^2 - 2c_2 p_1 p_2 \cos(q_2) + c_1 p_2^2] \frac{c_2 p_1 p_2 \operatorname{sen}(q_2)}{\delta_1} + m_2 g L \operatorname{sen}(q_2) - \frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]. \quad (46) \end{aligned}$$

Por otro lado, se determina la dinámica objetivo del sistema TORA como se muestra a continuación. De la ecuación (1) se sabe que la dinámica objetivo tiene la forma:

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = [J_d - R_d] \nabla H_d.$$

Una vez realizadas las sustituciones y cálculos se obtiene el sistema para la dinámica objetivo:

$$\begin{aligned} \dot{q}_1 &= \frac{1}{\delta_1} [c_3 p_1 - c_2 p_2 \cos(q_2)] & \dot{q}_2 &= \frac{1}{\delta_1} [-c_2 p_1 \cos(q_2) + c_1 p_2] \\ \dot{p}_1 &= -\frac{(-a_1 c_3 - \alpha a_1 c_2 \cos(q_2))}{\delta_1} \left[R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_3} q_2 + c_2 \operatorname{sen}(q_2))}{a_1} \right] - \\ &-\frac{(-a_1 c_2 \cos(q_2) + \alpha a_1 c_1)}{\delta_1} \left(-R_d(q_1 - \alpha q_2) \alpha + \frac{K}{a_1} (\sqrt{c_1 c_3} (q_1 - \alpha q_2) - \alpha c_2 \operatorname{sen}(q_2) + c_2 q_1 \cos(q_2)) \right) - \\ &-\frac{c_2^2 \operatorname{sen}(q_2) \cos(q_2)}{\delta_1^2} [c_3 p_1^2 - 2c_2 p_1 p_2 \cos(q_2) + c_1 p_2^2] \frac{c_2 p_1 p_2 \operatorname{sen}(q_2)}{\delta_1} + m_2 g L \operatorname{sen}(q_2) - \frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]. \\ \dot{p}_2 &= -\frac{(\alpha a_1 c_3 - a_3 c_2 \cos(q_2))}{\delta_1} \left[R_d(q_1 - \alpha q_2) + \frac{K(\sqrt{c_1 c_3} q_2 + c_2 \operatorname{sen}(q_2))}{a_1} \right] - \\ &-\frac{(-\alpha a_1 c_2 \cos(q_2) + a_3 c_1)}{\delta_1} \left(-R_d(q_1 - \alpha q_2) \alpha + \frac{K}{a_1} (\sqrt{c_1 c_3} (q_1 - \alpha q_2) - \alpha c_2 \operatorname{sen}(q_2) + c_2 q_1 \cos(q_2)) \right) - \\ &-\frac{c_2^2 \operatorname{sen}(q_2) \cos(q_2)}{\delta_1^2} [c_3 p_1^2 - 2c_2 p_1 p_2 \cos(q_2) + c_1 p_2^2] \frac{c_2 p_1 p_2 \operatorname{sen}(q_2)}{\delta_1} + m_2 g L \operatorname{sen}(q_2) - \frac{kv}{\delta_2} [-\alpha a_1 p_1 + a_1 p_2]. \quad (47) \end{aligned}$$

4 Programas de Simulación Numérica.

Para corroborar el comportamiento esperado de los sistemas controlados se diseñaron programas utilizando la herramienta computacional *Matlab 7.1*. Para simular un sistema no lineal controlado utilizando Matlab, se requieren dos programas llamados: Programa de simulación y Programa del sistema o modelo a simular. El programa de simulación permite definir los lineamientos básicos de la simulación: tiempo de simulación (inicial y final), condiciones iniciales y tipo de algoritmo de simulación (*ode23*, *ode45*,...); inclusive se puede definir los parámetros del sistema controlado y hasta graficar los resultados de la simulación. Este programa consiste en un conjunto lógico de instrucciones de ejecución secuencial denominado *script* en el ambiente Matlab. El corazón principal de este programa es el algoritmo de simulación. En esta investigación se utilizó el algoritmo *ode45*, el cual es un método de resolución de ecuaciones diferenciales ordinarias mediante fórmulas de Runge-Kutta de cuarto y quinto orden.

El modelo a simular se presenta por medio de un programa o función, o *function* en Matlab, en el cual se plantean explícitamente las ecuaciones diferenciales asociadas al sistema de control. Esencialmente, posee dos parámetros de entrada, el tiempo t de simulación y la variable de estado x , representadas por $(q_1, q_2, p_1$ y $p_2)$, debido a que éstas son las variables utilizadas directamente por los algoritmos de simulación. Junto con las ecuaciones diferenciales que representan el modelo del sistema, aparece la ley de control diseñada, la cual puede estar definida a través de variables auxiliares locales. A diferencia del *script*, el modelo a simular requiere un encabezado con la palabra *function* $\langle salida \rangle = \langle nomarch \rangle (t, x)$, donde $\langle salida \rangle$ corresponde al vector \dot{x} que refleja la dinámica del sistema, $\langle nomarch \rangle$ es el nombre original del programa, que tiene la extensión *.m* y el par (t, x) representan las variables de tiempo y de estado correspondientes de la simulación y del sistema a simular.

El diseño de estos programas de simulación numérica, permiten proporcionar información numérica y gráfica, sobre la función correcta de H_d , J_d , R_d y la ley de control para el ajuste lazo abierto-lazo cerrado, que permitan que el sistema en lazo cerrado tenga el mismo comportamiento de la dinámica objetivo.

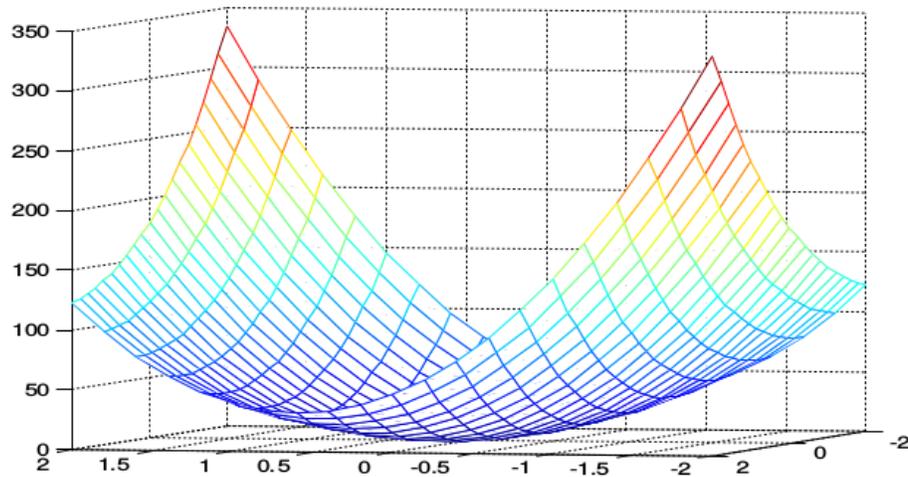
Para el sistema TORA, se diseñaron 6 programas descritos como: 3 programas de simulación y 3 programas del sistema o modelo a simular. Los dos primeros denominados VDESTORA Y SIMVDESTORA como se muestra en el anexo A, están dirigidos a verificar si la V_d hallada en la resolución de la EDP, cumple con ser una función de Lyapunov. Los dos siguientes programas denominados TORALAZ y SIMTORALAZ como se muestra en el anexo B, están dirigidos a corroborar el comportamiento de la dinámica en lazo cerrado. Asimismo, los programas denominados TORADIN y SIMTORADIN como se muestra en el anexo C, están dirigidos a corroborar el comportamiento de la dinámica objetivo.

5 Simulaciones Numéricas.

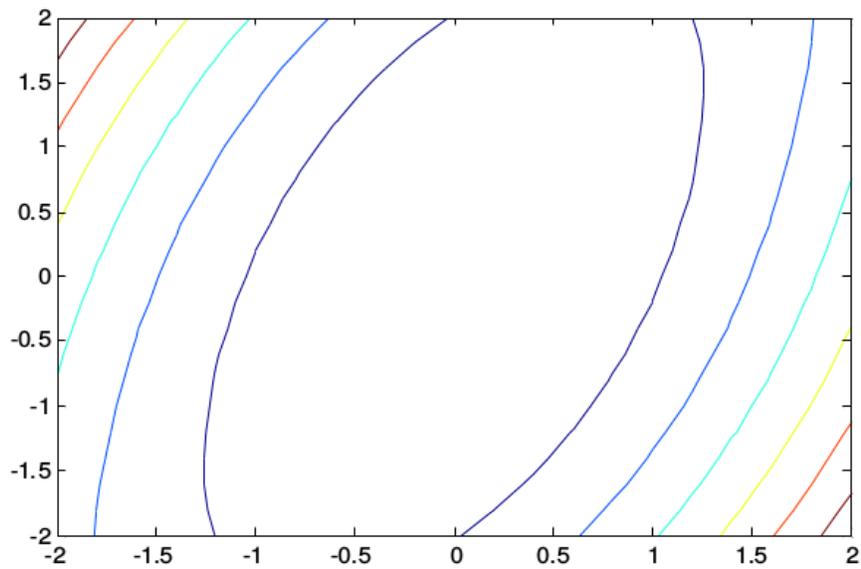
A continuación se presenta los resultados obtenidos utilizando los programas computacionales VDESTOR Y SIMVDESTOR, los cuales fueron diseñados para comprobar que la V_d hallada al resolver la EDP, es la función Lyapunov que se necesita para calcular la función de control que permita estabilizar el sistema TORA.

Al ejecutar dichos programas se obtiene los siguientes resultados para la función V_d del sistema TORA:

- (1) Grafica la función.

Figura 2: Gráfica de la función V_d

(2) Grafica las curvas de nivel de la función.

Figura 3: Curvas de nivel de la función V_d

(3) Determina si la función es cóncava o convexa en $(0,0)$.

Ingrese x: 0
 Ingrese y: 0
 valores_propios=
 22.7129
 90.0000

Existe un mínimo y la función es convexa.

(4) Puntos críticos de la función

x=
 $1/18*33^{(1/2)}*atan(i*131^{(1/2)},-2*33^{(1/2)})-1/18*i*131^{(1/2)}$
 $1/18*33^{(1/2)}*atan(-i*131^{(1/2)},-2*33^{(1/2)})+1/18*i*131^{(1/2)}$
 0.
 0.
 y=
 $atan(i*131^{(1/2)},-2*33^{(1/2)})$
 $atan(-i*131^{(1/2)},-2*33^{(1/2)})$
 0.
 0.

(5) Hessiano y gradiente en (0,0).

grad=
 0 0
 hess=
 90.0000 0
 0 22.7129

Para el modelo del sistema TORA, se realizaron simulaciones computacionales con los programas TORALAZ, SIMTORALAZ, TORADIN y SIMTORADIN, usando la plataforma de Matlab 7.1. Con estas simulaciones se muestra el desempeño del controlador propuesto. Se emplearon para el sistema TORA, los parámetros contenidos en el Cuadro 1 que fueron tomados de [4].

Se tomaron para la matriz M_d , los valores $a_1 = 1$, $a_2 = \alpha a_1$, $a_3 = 4$, con $\alpha = \sqrt{\frac{c_3}{c_1}} = \sqrt{\frac{11}{12}}$.

Se debe tomar en cuenta que el parámetro de diseño R_d debe cumplir:

$$R_d < 2 \frac{(K\sqrt{c_1 c_3} + Kc_2)}{a_1 \alpha},$$

Parámetro	c_1	c_2	c_3	k
Valor	12	1	11	5

Cuadro 1: Parámetros para el sistema TORA

lo que conlleva a que el $Hess(V_d(0,0))$ sea definido positivo, lo que implica que $(0,0)$ es un mínimo y es un punto de equilibrio es estable. Sustituyendo:

$$R_d < 2 \frac{(5\sqrt{(12)(11)} + 5(1))}{(1)\sqrt{\frac{11}{12}}} < 130,5$$

Se realizó la simulación, con las condiciones iniciales ubicadas en $q_1 = 1$, $q_2 = \frac{\pi}{2}$, $p_1 = 0$ y $p_2 = 0$. Después de algunos ensayos para obtener la mejor respuesta transitoria, la cual corresponde a la etapa de rediseño de las funciones, se seleccionaron $R_d = 90$ y $K_v = 70$.

Es importante señalar, que para obtener una rápida convergencia a la estabilización del sistema, se añade a la función de control propuesto el término $-50p_1$.

La respuesta del sistema con la dinámica objetivo propuesto por el método IDA-PBC, se muestra en las Figuras 4 y 7. Como puede verse, el sistema exhibe su respuesta en forma de oscilaciones amortiguadas, y la estabilización se produce a los 100 segundos.

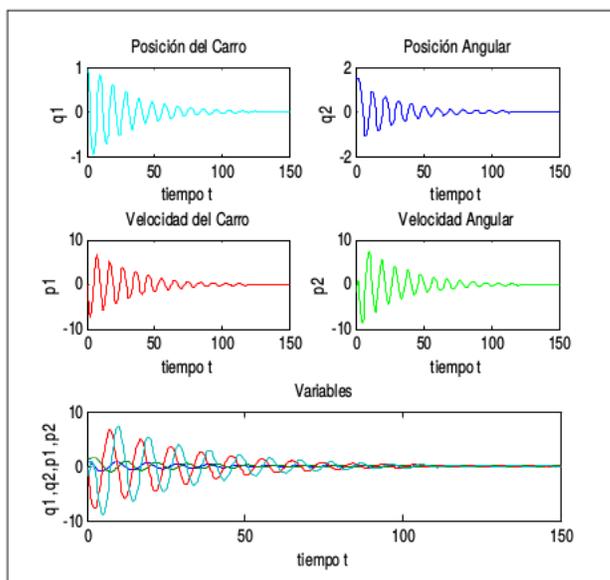


Figura 4: Simulación del Sistema TORA con la dinámica objetivo: Comportamiento de las variables.

Una típica respuesta del sistema en lazo cerrado con el controlador, se muestra en las Figuras ?? y ??. Como puede verse, el sistema exhibe su respuesta en forma de oscilaciones amortiguadas, y la estabilización se produce a los 100 segundos.

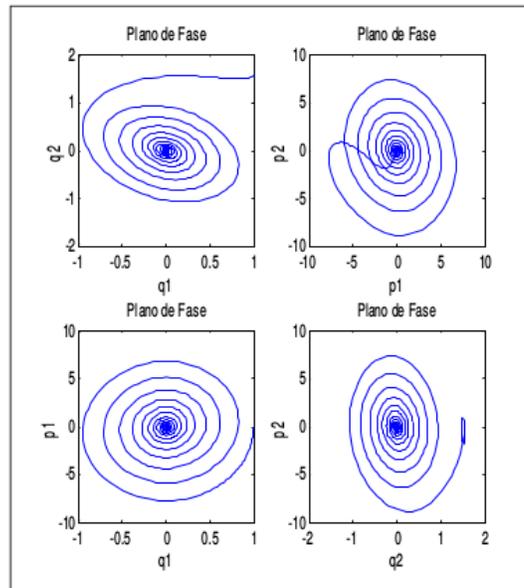


Figura 5: Simulación del Sistema TORA con la dinámica objetivo: Plano de Fase.

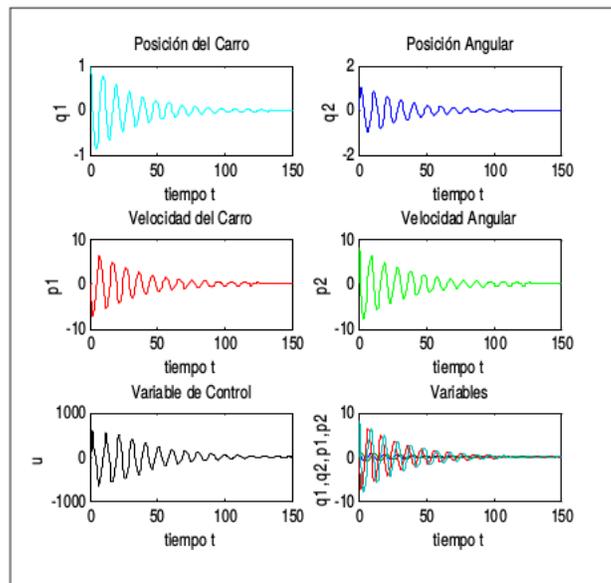


Figura 6: Simulación del Sistema TORA en lazo cerrado: Comportamiento de las variables y función de Control.

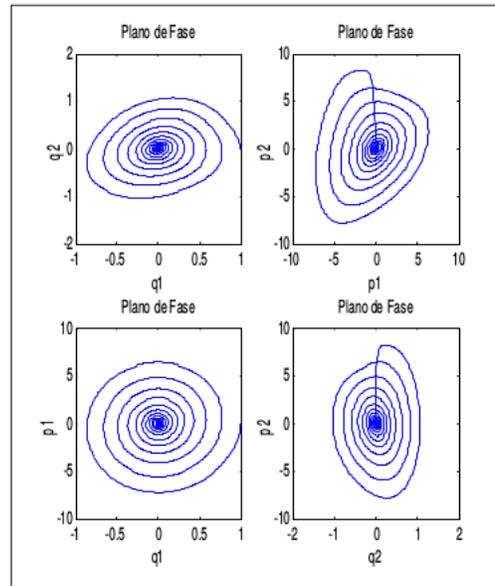


Figura 7: Simulación del Sistema TORA en lazo cerrado: Plano de Fase.

Esto demuestra que una vez aplicada la función de control propuesta, el sistema en lazo cerrado tiene el mismo comportamiento de la dinámica objetivo; por lo tanto, el método IDA-PBC, resulta ser efectivo para estabilizar el sistema TORA.

Para ilustrar la naturaleza global de la ley de control obtenida se realizan simulaciones cambiando las condiciones iniciales del sistema. De esta manera se prueba la naturaleza global de la ley de control obtenida, ya que para diferentes condiciones iniciales, se observa que la convergencia es preservada tal como lo predice la teoría. También es importante destacar que dicha ley de control logra estabilizar el sistema en un tiempo de simulación o tiempo de asentamiento perfectamente aceptable.

El sistema TORA es un prototipo de sistema mecánico subactuado que ha merecido gran atención por la comunidad de control no lineal, y en este trabajo, partiendo de la representación Hamiltoniana controlada por puertos, basada en la energía total del sistema considerada como energía cinética más energía potencial, se obtiene un controlador que logra estabilizar en forma global y asintótica el punto de equilibrio.

6 Conclusiones.

De acuerdo a los objetivos propuestos en la investigación y en atención a los resultados obtenidos luego de aplicada la metodología propuesta, se emiten las siguientes conclusiones.

- En el presente trabajo se ha desarrollado un esquema de control basado en el método IDA-PBC para el sistema prototipo TORA. La principal característica de este método es que explota la estructura física del sistema, en este caso, se aprovecha que toda la información del

comportamiento dinámico del sistema mecánico está contenida en las funciones de energía y de disipación, en consecuencia el diseño del controlador se ha concentrado en el manejo y modificación de estas variables.

- Para el moldeado de la energía total se tomó ventaja de la posibilidad de obtener una matriz deseada con términos constantes, y para la síntesis de la energía potencial deseada se obtuvo una reducción del conjunto de parámetros posibles para la matriz deseada, que facilita enormemente la resolución de la ecuación en derivadas parciales característica del método IDA-PBC.
- Al final se realizaron simulaciones numéricas que muestran el excelente comportamiento del controlador diseñado, reduciendo significativamente las oscilaciones de la plataforma, y con un tiempo de asentamiento perfectamente aceptable.

Referencias

- [1] Ortega R., Spong M., Gómez F., y Blankenstein G. *Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment*. IEEE Transactions on Automatic Control, Vol. C, AC-47, **8**(2002), 1218-1233.
- [2] Escobar G., Ortega R. and Sira-Ramírez H. *Output feedback global stabilization of a nonlinear benchmark system using a saturated passivity-based controller*. IEEE Transactions on Control Systems Technology, Vol. 7, **2**(1999), 289-293.
- [3] Ortega R., Loria A., Nicklasson J. and Sira-Ramírez H. *Passivity-Based Control of euler-Lagrange Systems*. Berlin, Germany, Springer-Verlag, Communications and Control Engineering, 1998.
- [4] Jankovic M., Fontaine D., and Kokotovic P. *TORA Example: -cascade and and passivity-based control designs*. IEEE Transactions on Control Systems Technology, Vol. 4, **3**(1996), 292-297.
- [5] Mahindrakar A., Astolfi A., Ortega R. y Viola G. *Further constructive results on interconnection and damping assignment control of underactuated mechanical systems: The acrobot example*. International Journal of Robust and Nonlinear Control, Vol. 16, **14**(2006), 671-685.
- [6] Morillo A., Ríos M., y Acosta V. *Control no lineal de sistemas mecánicos sub-actuados basado en el enfoque IDA-PBC: el caso del sistema TORA*. XIII Congreso Latinoamericano de Control y Automatización, Noviembre 25-28. Mérida, Venezuela, 2008.
- [7] Morillo A., Ríos M., y Arteaga F. *Una estrategia de control no lineal para el sistema Acrobot basada en el enfoque IDA-PBC*. XIII Congreso Latinoamericano de Control y Automatización, Noviembre 25-28. Mérida, Venezuela, 2008.
- [8] Grajales T. *Tipos de investigación*. Segunda Edición. Editorial Mc Graw Hill. México, 1998.
- [9] Hernández, R., Fernández, C. y Baptista, P. *Metodología de la Investigación*. Segunda Edición. Editorial Mc Graw Hill. México, 1998.
- [10] Khalil H. *Nonlinear Systems*. Prentice Hall, 2002.

- [11] Sira H., Márquez R., Rivas F. y Llanes O. *Control de sistemas no lineales: Linealización aproximada, extendida, exacta*. Editorial Pearson-Prentice Hall. España, 2005.
- [12] Acosta J., Ortega R., and Astolfi A. *Interconnection and damping assignment passivity-based control of mechanical systems with actuation degree one*. IEEE Trans. Automat. Contr., Vol. 50, 12(2005), 1936-1955.
- [13] Ortega R., Van der Schaft A., Masche B. and Escobar G. *Stabilization of port-controlled Hamiltonian systems: Energy balancing and passivation*. Automática, Vol. 38, 4(2002), April. Preprint.

7 Anexos.

ANEXO A

%%Programa VDESTORA:

```
function F=VDESTORA(x,y)
c1=12;
c2=1;
c3=11;
alpha=sqrt(c3/c1);
a1=1;
%a2=alpha*a1;
K=4;
Rd=90;
F=(1/2)*Rd*(x-alpha*y).^2+K/a1*((-alpha*sqrt(c1*c3)*y.^2)/2+sqrt(c1*c3)*x.*y
+alpha*c2*cos(y)-alpha*c2+c2*x.*sin(y));
```

%%Programa SIMVDESTORA

```
clear all
%Vd=(1/2)*Rd*(x-alpha*y)^2+K/a1*((alpha*sqrt(c1*c3)*y^2)/2+sqrt(c1*c3)*x*y
-alpha*sqrt(c1*c3)*y^2+alpha*c2*cos(y)+alpha*c2*y*sin(y)-alpha*c2+c2*x*sin(y)
-alpha*c2*y*sin(y))
while menu ~= 10
menu=0;
[x,y]=meshgrid(-2:0.2:2,-2:0.2:2);
z=VDESTORA(x,y);
disp(' (1) grafica la función')
disp(' (2) grafica las curvas de nivel de la función')
disp(' (3) dibuja los gradientes en el gráfico de curvas de nivel de la función')
disp(' (4) determina si la función es cóncava o convexa')
disp(' (5) Puntos críticos de la función')
disp(' (6) Hessiano y Gradiente')
disp(' (10) Salir del Programa')

menu=input('elija la opcion a estudiar: ');
```

```

switch menu
    case 1
mesh(x,y,z)
    case 2
contour(x,y,z)
    case 3
[x,y]=meshgrid(-2:.2:2,-2:.2:2);
z=VDESTORA(x,y);
contour(x,y,z)
hold on
[px,py]=gradient(z);
quiver(x,y,px,py)
    case 4
clear all
syms x y
z=VDESTORA(x,y);
a=input('Ingrese x: ');
b=input('Ingrese y: ');
d1=diff(z,'x',2);
d2=diff(z,'x','y');
d3=diff(z,'x','y');
d4=diff(z,'y',2);
a11=subs(d1,{x,y},{a,b});
a12=subs(d2,{x,y},{a,b});
a21=subs(d3,{x,y},{a,b});
a22=subs(d4,{x,y},{a,b});
hess=[a11,a12;a21,a22];
valores_propios=eig(hess)
c1=0;
c2=0;
for i = 1:length(valores_propios)
    if (valores_propios(i)>=0)
        c1=c1+1;
    else
        c2=c2+1;
    end
end
if (c1==1)&(c2==1)
fprintf ('La función es indefinida');
end
if (c1==2)&(c2==0)
fprintf ('Existe un mínimo y la función es convexa ');
end
if (c1==0)&(c2==2)
fprintf ('Existe un máximo y la función es cóncava');
end
pause

```

```

case 5
clear all
syms x y
z=VDESTORA(x,y);
disp('Puntos Críticos de la Función:')
[x,y] = solve(diff(z,'x'),diff(z,'y'))
case 6
clear all
syms x y
z=VDESTORA(x,y);
m=input('Ingrese x: ');
n=input('Ingrese y: ');
disp('Gradiente y Hessiano la Función:')
d1=diff(z,'x',2);
d2=diff(z,'x','y');
d3=diff(z,'x','y');
d4=diff(z,'y',2);
d5=diff(z,'x');
d6=diff(z,'y');
a1=subs(d5,{x,y},{m,n});
a2=subs(d6,{x,y},{m,n});
a11=subs(d1,{x,y},{m,n});
a12=subs(d2,{x,y},{m,n});
a21=subs(d3,{x,y},{m,n});
a22=subs(d4,{x,y},{m,n});
grad=[a1,a2]
hess=[a11,a12;a21,a22]
pause
end
end
end

```

ANEXO B

%%Programa TORALAZ:

```

function xdot=TORALAZ(t,x)
%%TORALAZ.m

```

```

%%Control del Sistema TORA Lazo Cerrado
%%

```

```

%%Parámetros del sistema
c1=12;
c2=1;
c3=11;
alpha=sqrt(c3/c1);
a1=1;
a2=alpha*a1;

```

```

K=5;
%a3 PUEDE SER MAYOR QUE 2.
a3=4;
%%Parámetros de Diseño
kv=70;%PUEDE VARIAR.
Rd=90;%Puede ser mayor de 65.

%%Otros Parámetros que no influyen
%g=32.2;%pie/seg^2.
g=981;%cm/seg^2.
%g=9.81; %m/seg^2.
m2=1;
L=1;
delta1 =c1*c3-c2*c2*cos(x(2)).*cos(x(2));
delta2= a1*a3-(c3/c1)*a1*a1;

%%Ley de control
u=-(alpha*a1*c3-a3*c2*cos(x(2)))/(delta1).*(Rd*(x(1)-alpha*x(2))
+(K*sqrt(c1*c3)*x(2)/a1)-(K*c2*sin(x(2))/a1))
-(-alpha*a1*c2*cos(x(2))+a3*c1)/delta1).*(-Rd*(x(1)-alpha*x(2))*alpha
+(K*sqrt(c1*c3)*(x(1)-alpha*x(2))/a1)-(K*alpha*c2*sin(x(2))/a1)+
(K*c2*x(1).*cos(x(2))/a1)-(c2*c2*sin(x(2)).*cos(x(2)))/(delta1)*(delta1))
.*(c3*x(3).*x(3)-2*c2*x(3).*x(4).*cos(x(2))+c1*x(4).*x(4))+c2*(x(4).*x(3)
.*sin(x(2))/delta1)+m2*g*L*sin(x(2))-(kv/delta2)*(-alpha*a1*x(3)+a1*x(4))
-50*x(3));

%%Ecuaciones de estado
xdot=[(1/delta1)*(c3*x(3)-c2*x(4).*cos(x(2)));
(1/delta1)*(-c2*x(3).*cos(x(2))+c1*x(4));
(-K*x(1));
((c2*c2*sin(x(2)).*cos(x(2)))/delta1*delta1)*(c3*x(3).*x(3)-2*c2*x(3)
.*x(4).*cos(x(2))+c1*x(4).*x(4))-((c2*x(3).*x(4).*sin(x(2)))/delta1)
-m2*g*L*sin(x(2))+u];

%%Fin de TORALAZ.m
%%Programa SIMTORALAZ:
SIMTORALAZ.m
Programa de generación de los gráficos TORALAZ.m

%Tiempo de simulación
ti=0; tf=150;

%%Condiciones Iniciales:
x0=[1 0 0 0]';

%%Simulación:

```

```
[t,x]=ode45('TORALAZ',[ti tf],x0);

%%Posición del Carro (x(1))
subplot(3,2,1),plot(t,x(:,1),'c')
title('Posición del Carro')
xlabel('tiempo t')
ylabel('q1')

%%Posición Angular (x(2))
subplot(3,2,2),plot(t,x(:,2),'b')
title('Posición Angular')
xlabel('tiempo t')
ylabel('q2')

%%Velocidad del Carro (x(3))
subplot(3,2,3),plot(t,x(:,3),'r')
title('Velocidad del Carro')
xlabel('tiempo t')
ylabel('p1')

%%Velocidad Angular (x(4))
subplot(3,2,4),plot(t,x(:,4),'g')
title('Velocidad Angular')
xlabel('tiempo t')
ylabel('p2')

%%Parámetros del sistema
c1=12;
c2=1;
c3=11;
alpha=sqrt(c3/c1);
a1=1;
a2=alpha*a1;
K=5;
%a3 PUEDE SER MAYOR QUE 2.
a3=4;

%%Parámetros de Diseño
kv=70;%PUEDE VARIAR.
Rd=90;%Puede ser mayor de 65.

%%Otros Parámetros que no influyen
%g=32.2;%pie/seg^2.
g=981;%cm/seg^2.
%g=9.81; %m/seg^2.
m2=1;
L=1;
```

```

delta1 =c1*c3-c2*c2*cos(x(2)).*cos(x(2));
delta2= a1*a3-(c3/c1)*a1*a1;

%%Ley de Control
u=-(alpha*a1*c3-a3*c2*cos(x(:,2)))/(delta1).*(Rd*(x(:,1))-alpha*x(:,2))+
(K*sqrt(c1*c3)*x(:,2)/a1)-(K*c2*sin(x(:,2))/a1)-
(-alpha*a1*c2*cos(x(:,2))+a3*c1)/delta1).*(-Rd*(x(:,1))-alpha*x(:,2))*alpha+
(K*sqrt(c1*c3)*(x(:,1))-alpha*x(:,2))/a1)-(K*alpha*c2*sin(x(:,2))/a1)+
(K*c2*x(:,1).*cos(x(:,2))/a1)-(c2*c2*sin(x(:,2)).*cos(x(:,2)))/(delta1)
*(delta1)).*(c3*x(:,3).*x(:,3)-2*c2*x(:,3).*x(:,4).*cos(x(:,2)))
+c1*x(:,4).*x(:,4))+c2*(x(:,4).*x(:,3)).*sin(x(:,2))/delta1
+m2*g*L*sin(x(:,2))-(kv/delta2)*(-alpha*a1*x(:,3)+a1*x(:,4))-50*x(:,3);

%%Control (u)
subplot(3,2,5),plot(t,u,'k')
title('Variable de Control')
xlabel('tiempo t')
ylabel('u')

%%Todas las Variables
subplot(3,2,6),plot(t,x)
title('Variables')
xlabel('tiempo t')
ylabel('q1,q2,p1,p2')

pause
figure
%%Plano de Fase
subplot(2,2,1),plot(x(:,1),x(:,2))
title('Plano de Fase')
xlabel('q1')
ylabel('q2')

%%Plano de Fase
subplot(2,2,2),plot(x(:,3),x(:,4))
title('Plano de Fase')
xlabel('p1')
ylabel('p2')

%%Plano de Fase
subplot(2,2,3),plot(x(:,1),x(:,3))
title('Plano de Fase')
xlabel('q1')
ylabel('p1')

%%Plano de Fase
subplot(2,2,4),plot(x(:,2),x(:,4))

```

```
title('Plano de Fase')
xlabel('q2')
ylabel('p2')
```

```
%Fin de SIMTORALAZ.m
```

ANEXO C

```
%% Programa TORADIN:
```

```
function xdot=TORADIN(t,x)
%%TORADIN.m
```

```
%%Control del Sistema TORA Dinámica Objetivo
%%%
```

```
%%Parámetros del sistema
```

```
c1=12;
c2=1;
c3=11;
alpha=sqrt(c3/c1);
a1=1;
a2=alpha*a1;
K=5;
%a3 PUEDE SER MAYOR QUE 2.
a3=15;
```

```
%%Parámetros de Diseño
```

```
kv=70;%PUEDE VARIAR.
Rd=90;%Puede ser mayor de 65.
```

```
%%Otros Parámetros que no influyen
```

```
%g=32.2;%pie/seg^2.
g=981;%cm/seg^2.
m2=1;
L=1;
delta1 =c1*c3-c2*c2*cos(x(2)).*cos(x(2));
delta2= a1*a3-(c3/c1)*a1*a1;
```

```
%%Ecuaciones de estado
```

```
xdot=[(1/delta1)*(c3*x(3)-c2*x(4).*cos(x(2)));
      (1/delta1)*(-c2*x(3).*cos(x(2))+c1*x(4));
      -(((a1*c3-alpha*a1*c2*cos(x(2)))/delta1).*(Rd*(x(1)-alpha*x(2))
      +(K*sqrt(c1*c3)*x(2)/a1)+(K*c2*sin(x(2))/a1)))-(((a1*c2*cos(x(2))
      +alpha*a1*c1)/delta1).*(-Rd*(x(1)-alpha*x(2))*alpha
      +(K*sqrt(c1*c3)*(x(1)-alpha*x(2))/a1)-(K*alpha*c2*sin(x(2))/a1
      +(K*c2*x(1).*cos(x(2))/a1)));-(((alpha*a1*c3-a3*c2*cos(x(2)))/delta1
      .*(Rd*(x(1)-alpha*x(2)))+(K*sqrt(c1*c3)*x(2)/a1)+(K*c2*sin(x(2))/a1)))]
```

```

-((( -alpha*a1*c2*cos(x(2))+a3*c1)/delta1).*(-Rd*(x(1)-alpha*x(2))*alpha
+(K*sqrt(c1*c3)*(x(1)-alpha*x(2))/a1)-(K*alpha*c2*sin(x(2))/a1)
+(K*c2*x(1).*cos(x(2))/a1)))-((-kv*alpha*a1*x(3)+kv*a1*x(4))/delta2)
-10*x(3)];

%%Fin de TORADIN.m

%%Programa SIMTORADIN:

SIMTORADIN.m
Programa de generación de los gráficos toradin.m

Tiempo de simulación
ti=0; tf=150;

%%Condiciones Iniciales:
x0=[1 pi/2 0 0]';

%%Simulación:
[t,x]=ode45('TORADIN',[ti tf],x0);

%%Posición del Carro (x(1))
subplot(3,2,1),plot(t,x(:,1),'c')
title('Posición del Carro')
xlabel('tiempo t')
ylabel('q1')

%%Posición Angular (x(2))
subplot(3,2,2),plot(t,x(:,2),'b')
title('Posición Angular')
xlabel('tiempo t')
ylabel('q2')

%%Velocidad del Carro (x(3))
subplot(3,2,3),plot(t,x(:,3),'r')
title('Velocidad del Carro')
xlabel('tiempo t')
ylabel('p1')

%%Velocidad Angular (x(4))
subplot(3,2,4),plot(t,x(:,4),'g')
title('Velocidad Angular')
xlabel('tiempo t')
ylabel('p2')

%%Todas las Variables
subplot(3,2,[5 6]),plot(t,x)
title('Variables')

```

```
xlabel('tiempo t')
ylabel('q1,q2,p1,p2')

pause
figure
%%Plano de Fase
subplot(2,2,1),plot(x(:,1),x(:,2))
title('Plano de Fase')
xlabel('q1')
ylabel('q2')

%%Plano de Fase
subplot(2,2,2),plot(x(:,3),x(:,4))
title('Plano de Fase')
xlabel('p1')
ylabel('p2')

%%Plano de Fase
subplot(2,2,3),plot(x(:,1),x(:,3))
title('Plano de Fase')
xlabel('q1')
ylabel('p1')

%%Plano de Fase
subplot(2,2,4),plot(x(:,2),x(:,4))
title('Plano de Fase')
xlabel('q2')
ylabel('p2')
%Fin de SIMTORADIN.m
```