

**UNIVERSIDAD NACIONAL DE INGENIERÍA**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**CONTROL OPTIMO DE UN MANIPULADOR ROBOTICO  
TRASLACIONAL DE 2 GRADOS DE LIBERTAD**

**INFORME DE SUFICIENCIA**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:**

**INGENIERO ELECTRONICO**

**PRESENTADO POR:**

**JULIO VICTOR OJEDA TUMPAY**

**PROMOCIÓN  
1994-II**

**LIMA – PERÚ  
2006**

CONTROL OPTIMO DE UN MANIPULADOR ROBOTICO  
TRASLACIONAL DE 2 GRADOS DE LIBERTAD

*Dedico este trabajo a:  
Mi familia, inspiración plena de lucha y  
superación.*

## SUMARIO

El presente trabajo pretende presentar una de las muchas aplicaciones del control automático, así como motivar el diseño y la implementación de unidades robóticas comerciales aplicando estrategias de control moderno. En este caso se trata de un manipulador robótico traslacional de dos grados de libertad. Uno de los grados, es la posición angular del brazo, y el otro grado, es la posición de traslación del carro. Por consiguiente, el objetivo es el controlar las posiciones del brazo y del carro, según una referencia deseada, aplicando la técnica de Control Optimo Cuadrático con Observador de estado.

En el Capítulo I se formula el problema a resolver y las consideraciones de diseño.

En el Capítulo II se aborda el modelado de la planta, haciendo uso del método de Lagrange, así como su representación en Espacio de Estado para el Sistema No Lineal y el Sistema Linealizado, la discretización del modelo y finalmente la simulación en lazo abierto utilizando Mat Lab.

En el Capítulo III se diseña el controlador Optimo Cuadrático con Observador de Estado y su correspondiente simulación en Mat Lab.

En el Capítulo IV se hace una propuesta de implementación usando la tarjeta de Adquisición de Datos LAB-PC+ de National Instrument, presentando una descripción general del hardware y Software del sistema.

## ÍNDICE

<b>PROLOGO</b>	1
<b>CAPÍTULO I</b>	
<b>CONSIDERACIONES PREVIAS</b>	
1.1 Formulación del problema	2
1.2 Consideraciones de Diseño	3
1.3 Pautas para el Diseño e Implementación de los Sistemas de Control <sup>1</sup>	4
<b>CAPÍTULO II</b>	
<b>MODELADO DEL MANIPULADOR ROBOTICO DE DOS GRADOS DE LIBERTAD</b>	
2.1 Estructura del manipulador	6
2.2 Ecuaciones Mecánicas del Manipulador	6
2.2.1 Ecuaciones de Energía para el Carro	7
2.2.2 Ecuaciones de Energía para la Varilla	7
2.3 Ecuacion de Lagrange para el Manipulador	8
2.4 Ecuaciones Eléctricas del Manipulador	11
2.4.1 Modelo del Servomotor DC M1 - Polea	11
2.4.2 Modelo del Servomotor DC M2 - Brazo	12
2.4.3 Modelo Lagrangiano del MRTM	14
2.5 Modelo en Espacio de Estado del Manipulador	16
2.5.1 Modelo No Lineal	16
2.5.2 Modelo Linealizado	16
2.5.3 Discretización del Modelo	18
2.5.4 Simulación del Modelo a Lazo Abierto	18
<b>CAPÍTULO III</b>	
<b>CONTROL OPTIMO CUADRATICO DISCRETO DEL MANIPULADOR</b>	

3.1 Control Optimo Cuadrático Estacionario	24
3.1.1 Diseño del Regulador Optimo Cuadrático	25
3.1.2 Diseño del Control Optimo Cuadrático para Sistemas de Seguimiento	27
3.2 Diseño del Control Optimo Cuadrático del Manipulador	28
3.2.1 Diseño del Control Optimo Proporcional Integral SISO	28
3.2.2 Diseño del Control Optimo Proporcional Integral MIMO	30
3.2.3 Diseño del Observador de Estados	35
3.3 Simulación de Sistema de Control Optimo Cuadrático del Manipulador	38

## **CAPÍTULO IV**

### **PROPUESTA DE IMPLEMENTACION**

4.1 Esquema de implementación	40
4.2 Hardware del Sistema	40
4.3 Software del Sistema	42

### **CONCLUSIONES Y RECOMENDACIONES**

<b>ANEXO A:</b> Rutinas de Programación en MATLAB	46
---	----

<b>ANEXO B:</b> Rutinas de Programación en C	59
--	----

<b>BIBLIOGRAFÍA</b>	69
---------------------	----

## **PRÓLOGO**

En la práctica, la mayor parte de los procesos a controlar son no lineales, y si a esto se suma su complejidad, entonces es necesario aplicar estrategias de control no lineal; sin embargo, si las no linealidades inherentes al sistema son suaves, y los parámetros del proceso son prácticamente invariantes en el tiempo, entonces puede obtenerse su representación lineal, lo que posibilita el uso de estrategias de control lineal, naturalmente en un rango de operación.

En tal sentido en este trabajo se modela el proceso no lineal usando la ecuación de Lagrange, luego linealizarlo, discretizarlo, y finalmente abordar el diseño del Observador de Estados y del Control Óptimo Proporcional Integral en Tiempo Discreto, para así cumplir con las especificaciones de diseño.

Este trabajo se basa en trabajos de tesis de Maestría como se puede ver en [8] y [9]. En dichos trabajos se aborda el diseño de sistemas de Control No Lineal Adaptivo y el Control Deslizante, correspondientemente.

# **CAPÍTULO I**

## **CONSIDERACIONES PREVIAS**

### **1.1 Formulación del problema**

El problema que se estudia en este Informe de Suficiencia es el control simultáneo de la posición angular del brazo y la posición de traslación del carro, el problema de control es resuelto por medio de la técnica de Control Optimo Cuadrático. Para esto se requiere que el manipulador robótico realice el seguimiento de trayectorias de referencia arbitrarias empleando la fuerza de control adecuada. Por consiguiente, el proceso a controlar posee dos entradas y dos salidas, es decir, se considera un sistema MIMO (Multiple-Input-Multiple-Output).

El manipulador robótico translacional de dos grados de libertad, es un sistema multivariable que posee un eslabón (el brazo del robot) articulado al eje del motor DC. Este motor está montado sobre un carro que desarrolla su movimiento de traslación por la acción de poleas. Una de las poleas está acoplada al eje de un segundo motor DC. Este último motor va a gobernar el movimiento del carro.

El sistema manipulador translacional en referencia, ha sido construido y diseñado tomando como modelo al sistema de péndulo invertido. La finalidad de construir el sistema es para implementar técnicas de control avanzado, en nuestro caso la implementación de un Controlador Optimo.

Es necesario modelar la dinámica de la planta mediante las ecuaciones de energía de Lagrange. La exactitud del modelo es fundamental para poder aplicar una técnica de control. El modelo requiere de la identificación de los parámetros del servomotor DC cuyos resultados experimentales han sido tomados de un trabajo de investigación realizado en el ambiente de laboratorio 8 (Post-Grado UNI-FIEE).



La propuesta de implementación del sistema de control se hace a la tarjeta de adquisición de datos LABPC+ de National Instrument, que permite manejar los datos de entradas digitales de 16 bits y dos salidas analógicas que son registradas por las tarjetas que generan la señal de Modulación por ancho de pulso (PWM), estas señales se encargan, se encargan de modular la señal que proviene de la salida de la tarjeta LABPC+, para luego ser conectadas a un amplificador de potencia.

## 1.2 Consideraciones de Diseño

Se desea diseñar un Sistema de Control Optimo, que permita obtener un tiempo de asentamiento o establecimiento  $t_s \leq 2 \text{seg.}$  y un porcentaje de sobrepico  $PO \leq 5\%$ , requerimientos necesarios para proveer al sistema alto rendimiento y exactitud, toda vez que se desea orientar a aplicaciones robóticas de alta precisión.

## 1.2 Pautas para el Diseño e Implementación de los Sistemas de Control

Los pasos necesarios para el diseño e implementación de los sistemas de control se pueden resumir en los siguientes pasos:

1. Dada una planta industrial (en la mayoría de los casos de dinámica inalterable), primeramente se deben elegir sensores y actuadores adecuados.
2. Construir modelos matemáticos adecuados de la planta (en la que se incluyan los sensores y actuadores seleccionados).
3. Diseñar un controlador de tal modo que el sistema de lazo cerrado satisfaga las especificaciones dadas.
4. Realizar la simulación del sistema de control diseñado para verificar el comportamiento del sistema resultante en respuesta a diversas señales y perturbaciones.
5. Rediseñar el sistema, y completar el análisis correspondiente en caso que el proceso de simulación no resulta satisfactorio. Repetir el paso de simulación hasta obtener el resultado deseado.
6. Luego de obtener el resultado deseado, se debe construir el prototipo (sistema físico).

7. Editar el programa de control que permita probar el sistema de control físicamente implementado (prototipo). Efectuar ajustes en el prototipo de ser necesario. En esta fase del trabajo, generalmente los problemas que se presentan son relacionados a imperfecciones en el conexionado eléctrico y/o a errores de programación.

## **CAPÍTULO II**

### **MODELADO DEL MANIPULADOR ROBOTICO TRASLACIONAL DE DOS GRADOS DE LIBERTAD**

En el desarrollo de casi todas las estrategias de control, es indispensable la obtención del modelo matemático de la planta a controlarse, aplicándose las leyes físicas del proceso y obteniéndose una ecuación diferencial, sea ésta lineal o no lineal. Este capítulo trata del modelamiento de sistemas en el Espacio de Estado, tanto para procesos lineales y no lineales.

En general, se considera que la mayoría de los enfoques de modelado pertenecen a uno de dos grupos de métodos:

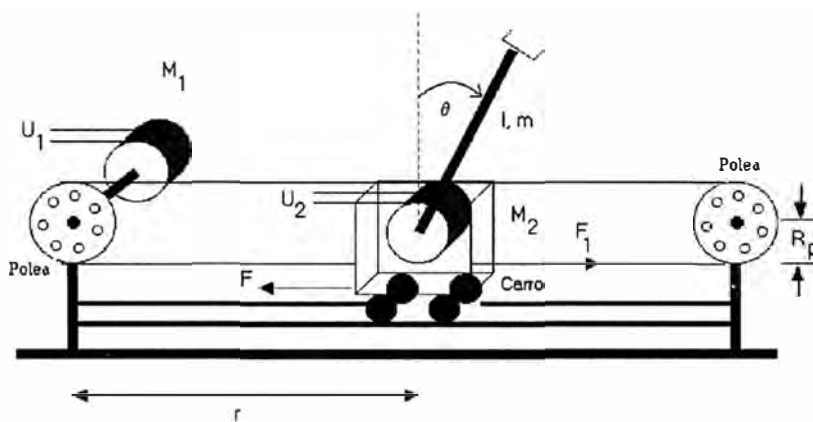
- a) Métodos basados en consideraciones formales de causas y efectos físicos en los componentes de sistemas.
- b) Métodos basados en la ley de conservación de la energía (por ejemplo, el método de Lagrange).

De los grupos de métodos, usaremos el segundo, particularmente interesante porque proporciona la potencialidad en cuanto a exactitud del modelo en referencia.

Por consiguiente, para aplicar la técnica de Control Óptimo, es necesario presentar la dinámica del modelo del Manipulador Robótico translacional [8], [9], usando la ecuación dinámica de Lagrange, que como se indico anteriormente, nos va a proporcionar la precisión del modelo, la misma que debe de responder a las señales de control generadas por el controlador óptimo; y por consiguiente, conseguir que las salidas del sistema sigan a una referencia deseada con un error nulo en estado estacionario.

## 2.1 Estructura del Manipulador

El sistema del Manipulador Robótico Traslacional Multivariable de Dos Grados de Libertad se muestra en la figura 2.1. Donde  $M_1$  es un servomotor de DC con un mecanismo de reducción y un encoder óptico, cuyo eje de salida está articulado a una polea de radio  $R_p$ . Esta polea usa un cable para transmitir la fuerza  $F_1$  que realiza el movimiento corridizo de un carro de masa  $M_c$  montado a un par de rieles a lo largo del eje de  $x$ .  $M_2$  es otro servomotor DC con un encoder óptico, empleado para manejar el movimiento rotatorio del brazo, alrededor de un punto de pivote; asumiendo que dicho punto es el  $CG$  (centro de gravedad) del carro.



**Figura 2.1:** Manipulador Robótico Traslacional Multivariable (MRTM).

En la figura 2.1,  $\theta$  es la posición angular del brazo de longitud  $l$  y masa  $m$ ,  $r$  es la posición del carro,  $F$  es la fuerza friccional opuesta al movimiento del carro. Las entradas al sistema son los voltajes de control  $K_{A1}u_1$  y  $K_{A2}u_2$  aplicados a los terminales de armadura de  $M_1$  y  $M_2$  respectivamente, y las dos salidas del sistema son  $r$  y  $\theta$ . Los parámetros  $K_{A1}$  y  $K_{A2}$  son las ganancias de los amplificadores.

## 2.2 Ecuaciones Mecánicas del Manipulador

El propósito del modelo matemático es desarrollar y probar una ley de control que pueda hacer que la trayectoria del carro y la trayectoria del brazo sigan trayectorias de referencia arbitrarias. Se asume que las dos componentes de salida  $y$  ( $r$ ,  $\theta$ ) son medidas y que las fuerzas de control  $u$  ( $u_1$ ,  $u_2$ ) son las entradas del proceso [9].

En principio, se debe encontrar la relación dinámica entre las salidas y las entradas, para la cual se usa las ecuaciones de Lagrange. Se escriben entonces, las ecuaciones de energía para el carro y el brazo, enseguida se escriben las ecuaciones de los subsistemas

servomotor 1 – polea, servomotor 2 – brazo; para luego encontrar la relación indicada entre las salidas  $y$  ( $r, \theta$ ) y las entradas  $u$  ( $u_1, u_2$ ). Finalmente se determina la ecuación lagrangiana del MRTM.

### 2.2.1 Ecuaciones de Energía para el Carro

Las ecuaciones dinámicas del movimiento del carro comprenden el desarrollo de las energías cinéticas ( $T_1$ ), potencial ( $U_1$ ) y de disipación ( $D_1$ ) dadas por:

$$T_1 = \frac{1}{2} M_c \dot{r}^2 \quad (2.1)$$

$$U_1 = 0 \quad (2.2)$$

$$D = \frac{1}{2} F_c \dot{r}^2 \quad (2.3)$$

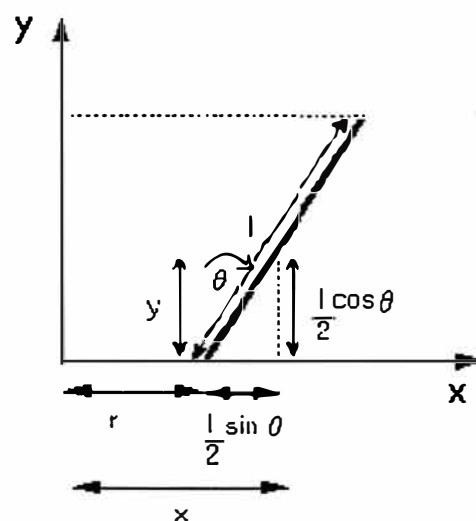
### 2.2.2 Ecuaciones de Energía para la Varilla

Sea la energía cinética ( $T_2$ ) de la Varilla representada por:

$$T_2 = \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} m (\dot{x}^2 + \dot{y}^2) \quad (2.4)$$

donde  $J$  es el momento de inercia referido a la Varilla de masa de  $m$  de la figura 2.1.

La justificación de la ecuación (2.4) se explica con ayuda de la figura 2.2 que muestra las coordenadas ( $x, y$ ) en función de los parámetros del sistema del brazo robótico.



**Figura 2.2:** Subsistema de la Varilla robótica.

Siendo:

$$x = r + \frac{l}{2} \operatorname{sen}\theta \quad (2.5)$$

$$y = \frac{l}{2} \cos\theta \quad (2.6)$$

derivando las ecuaciones (2.5) y (2.6), se obtiene:

$$\dot{x} = \dot{r} + \frac{l}{2} \cos(\theta)\dot{\theta} \quad (2.7)$$

$$\dot{y} = -\frac{l}{2} \sin(\theta)\dot{\theta} \quad (2.8)$$

y elevado al cuadrado las ecuaciones (2.7) y (2.8), se obtiene:

$$\dot{x}^2 = \dot{r}^2 + r\dot{\theta}\cos(\theta) + \frac{l^2}{4} \cos^2(\theta)\dot{\theta}^2 \quad (2.9)$$

$$\dot{y}^2 = \frac{l^2}{4} \sin^2(\theta)\dot{\theta}^2 \quad (2.10)$$

sumando las ecuaciones (2.9) y (2.10) se obtiene:

$$\dot{x}^2 + \dot{y}^2 = \dot{r}^2 + r\dot{\theta}\cos(\theta) + \frac{l^2}{4}\dot{\theta}^2 \quad (2.11)$$

reemplazando (2.11) en (2.4), resulta:

$$T_2 = \frac{1}{2} J\dot{\theta}^2 + \frac{1}{2} m(\dot{r}^2 + r\dot{\theta}\cos(\theta) + \frac{l^2}{4}\dot{\theta}^2) \quad (2.12)$$

La energía potencial y la energía de disipación de la Varilla es tan expresadas por las siguientes ecuaciones:

$$U_2 = \frac{m l g}{2} \cos\theta \quad (2.13)$$

$$D_2 = \frac{1}{2} C\dot{\theta}^2 \quad (2.14)$$

### 2.3 Ecuacion de Lagrange para el Manipulador

La Ecuación de Lagrange generalizada es:

$$\frac{\partial}{\partial t} \left( \frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} + \frac{\partial D}{\partial \dot{q}} = F_q \quad (2.15)$$

donde:

$$L = T - U$$

siendo:

- $L$ : conjunto de energías del sistema.
- $D$ : disipación de energía externa del sistema.
- $q$ : coordenada de variable independiente del sistema.
- $\dot{q}$ : derivada de variable independiente del sistema.
- $F_q$ : fuerza externa que actúa en la dirección de  $q$ .

Para el sistema, los términos de las energías individuales componen en su conjunto la totalidad energética del sistema. Se aprecia cada una de ellas como:

Energía cinética del sistema.

$$T = T_1 + T_2 = \frac{1}{2} M_c \dot{r}^2 + \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} m (\dot{r}^2 + r \dot{\theta} \cos(\theta) + \frac{l^2}{4} \dot{\theta}^2)$$

Energía potencial del sistema

$$U = U_1 + U_2 = \frac{m l g}{2} \cos \theta$$

Energía disipada del sistema

$$D = D_1 + D_2 = \frac{1}{2} F_c \dot{r}^2 + \frac{1}{2} C \dot{\theta}^2$$

Se puede generalizar la ecuación de Lagrange para el vector  $q$  en función de coordenadas generales  $(r, \theta)$  del proceso, formulándose de acuerdo con la ecuación (2.15).

$$\frac{\partial}{\partial t} \left( \frac{\partial T}{\partial \dot{r}} \right) - \frac{\partial T}{\partial r} + \frac{\partial U}{\partial r} + \frac{\partial D}{\partial \dot{r}} = F_1 \quad (2.16)$$

$$\frac{\partial}{\partial t} \left( \frac{\partial T}{\partial \dot{\theta}_1} \right) - \frac{\partial T}{\partial \theta_1} + \frac{\partial U}{\partial \theta_1} + \frac{\partial D}{\partial \dot{\theta}_1} = T_L \quad (2.17)$$

Para poder plantear las ecuaciones de Lagrange, se observa que el término de energía potencial  $U$  no es una función de la derivada de  $r$  respecto al tiempo. Las ecuaciones dinámicas de Lagrange se desarrollan para las variables involucradas en el sistema.

A continuación se desarrollan las derivadas parciales que indica la ecuación (2.16) para determinar la fuerza  $F_1$ :

$$\frac{\partial T}{\partial \dot{r}} = (M_c + m)\dot{r} + \frac{ml}{2}\cos(\theta)\dot{\theta} \quad (2.18)$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{r}}\right) = (M_c + m)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}^2 + \frac{ml}{2}\cos(\theta)\ddot{\theta} \quad (2.19)$$

$$\frac{\partial T}{\partial r} = 0; \quad \frac{\partial U}{\partial r} = 0; \quad \frac{\partial D}{\partial \dot{r}} = F_c \dot{r} \quad (2.20)$$

Se obtiene la primera ecuación parcial del sistema para determinar  $F_1$  Como sigue:

$$F_1 = (M_c + m)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}^2 + \frac{ml}{2}\cos(\theta)\ddot{\theta} + F_c \dot{r} \quad (2.21)$$

Se desarrollan las derivadas parciales que indica la ecuación (2.17) para determinar el torque  $T_L$  Como sigue:

$$\frac{\partial T}{\partial \dot{\theta}} = J\dot{\theta} + \frac{ml}{2}\cos(\theta)\dot{r} + \frac{ml^2}{4}\dot{\theta} \quad (2.22)$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}}\right) = J\ddot{\theta} + \frac{ml}{2}\cos(\theta)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}\dot{r} + \frac{ml^2}{4}\ddot{\theta} \quad (2.23)$$

$$\frac{\partial T}{\partial \theta} = -\frac{ml}{2}\sin(\theta)\dot{r}; \quad \frac{\partial U}{\partial \theta} = -\frac{ml}{2}g\sin(\theta); \quad \frac{\partial D}{\partial \dot{\theta}} = C\dot{\theta} \quad (2.24)$$

Por tanto, se obtiene la segunda ecuación parcial del sistema para  $T_L$ :

$$T_L = \frac{ml}{2}\cos(\theta)\ddot{r} + \left(\frac{ml^2}{4} + J\right)\ddot{\theta} + C\dot{\theta} - \frac{mlg}{2}\sin(\theta) \quad (2.25)$$

El modelo Lagrangiano puede ser obtenido usando las ecuaciones (2.21) y (2.25). Sin embargo, si se desea obtener como variable de control a los voltajes de armadura  $u_1$  y  $u_2$  en lugar de  $F_1$  y  $T_L$ , respectivamente, entonces es necesario modelar los subsistemas eléctricos del MRTM.



## 2.4. Ecuaciones Eléctricas del Manipulador

### 2.4.1. Modelo del Servomotor DC $M_1$ – Polea

La figura 2.3 muestra el servomotor de DC  $M_1$ , que es articulado a una polea. Considerando despreciable la inductancia de armadura  $L_a$ , el voltaje de entrada  $K_A u_1$  aplicado a la armadura está dada por:

$$R_a I_{a1} + V_{b1} = K_A u_1 \quad (2.26)$$

La tensión de fuerza contra-electromotriz es proporcional a la velocidad de motor, y se expresa como:

$$V_{b1} = K_b \dot{\theta}_{m1} \quad (2.27)$$

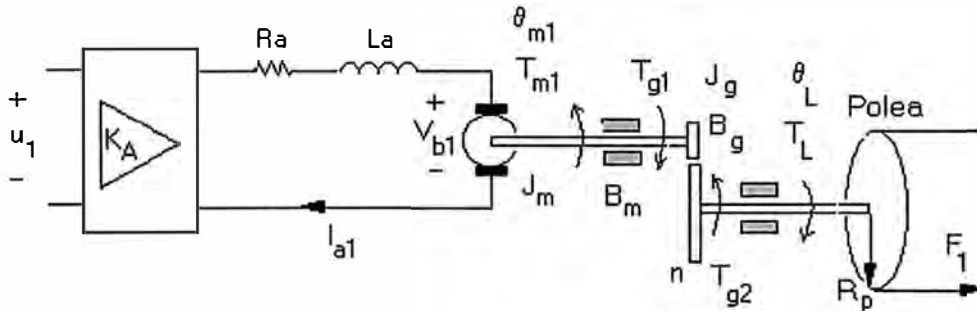
La ecuación del torque  $T_{m1}$  del servomotor se expresa como:

$$T_{m1} = J_m \ddot{\theta}_{m1} + B_m \dot{\theta}_{m1} + T_{g1} \quad (2.28)$$

El torque  $T_{g2}$  requiere la transmisión de la polea, y el torque en la polea es directamente proporcional a la fuerza  $F_1$ , luego formulamos:

$$T_{g2} = n T_{g1} = (J_g + J_p) \ddot{\theta}_L + (B_g + B_p) \dot{\theta}_L + F_1 R_p \quad (2.29)$$

donde  $J_m$ ,  $J_g$  y  $J_p$  son los momentos de inercia de la armadura, mecanismo de reducción de engranaje y polea, respectivamente, mientras  $B_m$ ,  $B_g$  y  $B_p$  son las constantes de fricción de la armadura, mecanismo de reducción de engranaje y polea respectivamente.



**Figura 2.3:** Subsistema motor - polea.

Usando la relación para la conservación de energía de engranajes  $T_{g1} \theta_{m1} = T_{g2} \theta_L$  con  $(n \theta_L = \theta_{m1})$ , de este modo se puede expresar la ecuación (2.28) como:

$$T_{m1} = J_m n \ddot{\theta}_L + B_m n \dot{\theta}_L + T_{g1} \quad (2.30)$$

despejando  $T_{g1}$  de (2.29) y reemplazando en (2.30), luego se agrupa términos adecuadamente se obtiene:

$$n T_{m1} = J_{eq1} \ddot{\theta}_L + B_{eq1} \dot{\theta}_L + F_1 R_p \quad (2.31)$$

donde:

$$J_{eq1} = n^2 J_m + J_g + J_p$$

$$B_{eq1} = n^2 B_m + B_g + B_p$$

el torque del servomotor  $T_{m1}$  es proporcional a la corriente  $I_{a1}$ .

$$T_{m1} = K_m I_{a1} \quad (2.32)$$

reemplazando la ecuación (2.31) en (2.32), y luego despejamos  $I_{a1}$ .

$$I_{a1} = \frac{J_{eq1}}{nK_m} \ddot{\theta}_L + \frac{B_{eq1}}{nK_m} \dot{\theta}_L + \frac{R_p F_1}{nK_m} \quad (2.33)$$

reemplazando las ecuaciones (2.27) y (2.33) en (2.26), y luego despejando  $F_1$ , así:

$$F_1 = \frac{nK_m K_A}{R_a R_p} u_1 - \frac{J_{eq1}}{R_p} \ddot{\theta}_L - \left( \frac{B_{eq1}}{R_p} + \frac{n^2 K_m K_b}{R_a R_p} \right) \dot{\theta}_L \quad (2.34)$$

El movimiento rotacional de la polea puede ser transformado como un movimiento translacional del carro usando la siguiente relación (ver figura 2.3).

$$\theta_L = \frac{r}{R_p} \quad (2.35)$$

derivando la ecuación (2.35) y sustituyendo en la ecuación (2.34), se obtiene la ecuación general del subsistema del motor.

$$F_1 = \frac{nK_m K_A}{R_a R_p} u_1 - \frac{J_{eq1}}{R_p^2} \ddot{r} - \left( \frac{B_{eq1}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2} \right) \dot{r} \quad (2.36)$$

igualando la ecuación (2.36) con la ecuación (2.21), se obtiene la primera ecuación dinámica de Lagrange del sistema como:

$$u_1 = \frac{R_a R_p}{nK_m K_A} \left\{ (M_c + m + \frac{J_{eq1}}{R_p^2}) \ddot{r} + (F_c + \frac{B_{eq1}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2}) \dot{r} + \frac{ml}{2} \cos(\theta) \ddot{\theta} + \dots \right. \\ \left. - \frac{ml}{2} \sin(\theta) \dot{\theta}^2 \right\} \quad (2.37)$$

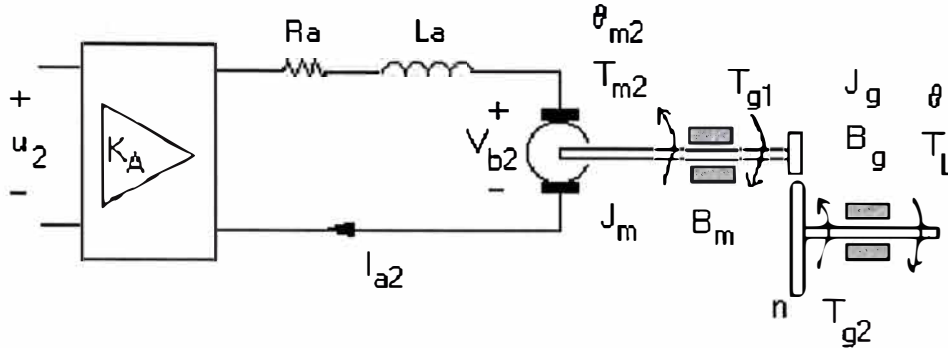
#### 2.4.2. Modelo del Servomotor DC $M_2$ - Brazo

La figura 2.4 muestra el servomotor de DC  $M_2$  que articula a un punto del pivote localizado en el CG del carro. Teniendo en cuenta que  $M_1$  y  $M_2$  tiene los mismos parámetros y despreciando la inductancia de la armadura  $L_a$ , entonces podemos formular lo siguiente:

$$R_a I_{a2} + V_{b2} = K_A u_2 \quad (2.38)$$

donde.

$$V_{b2} = K_b \theta_{m2} \quad (2.39)$$



**Figura 2.4:** Subsistema del torque del brazo.

La relación de desplazamiento angular del eje del motor y del eje de la carga se expresa como:

$$\theta_{m2} = n\theta \quad (2.40)$$

La ecuación del torque  $T_{m2}$  del servomotor se representa como:

$$T_{m2} = J_m n \ddot{\theta} + B_m n \dot{\theta} + T_{g1} \quad (2.41)$$

Mientras que el torque  $T_{g2}$  requerido para manejar el brazo, se expresa por:

$$\bar{T}_{g2} = nT_{g1} = J_g \ddot{\theta} + B_g \dot{\theta} + T_L \quad (2.42)$$

siendo  $T_L$  el torque de la carga. Reemplazando la ecuación (2.42) en la ecuación (2.41) se obtiene:

$$nT_{m2} = J_{eq2} \ddot{\theta} + B_{eq2} \dot{\theta} + T_L \quad (2.43)$$

donde:

$$J_{eq2} = n^2 J_m + J_g$$

$$B_{eq2} = n^2 B_m + B_g$$

El torque  $T_{m2}$  del servomotor es proporcional a la corriente  $I_{a2}$ .

$$T_{m2} = K_m I_{a2} \quad (2.44)$$

sustituyendo las ecuaciones (2.39), (2.40) en la ecuación (2.38) y despejando  $I_{a2}$ , se obtiene:

$$I_{a2} = \frac{k_A}{R_a} u_2 - \frac{nK_b}{R_a} \dot{\theta} \quad (2.45)$$

sustituyendo las ecuaciones (2.44), (2.45) en (2.43) y despejando  $T_L$ , se obtiene:

$$T_L = -J_{eq2}\ddot{\theta} - (B_{eq2} + \frac{n^2 K_b K_m}{R_a})\dot{\theta} + \frac{nK_m K_A}{R_a}u_2 \quad (2.46)$$

igualando la ecuación (2.46) con la ecuación (2.25), se obtiene la segunda ecuación dinámica de Lagrange del sistema, como sigue:

$$u_2 = \frac{R_a}{nK_m K_A} \left\{ \frac{ml}{2} \cos(\theta) \ddot{r} + \left( \frac{ml^2}{4} + J + J_{eq2} \right) \ddot{\theta} + \left( C + B_{eq2} + \frac{n^2 K_m K_b}{R_a} \right) \dot{\theta} - \dots \right. \\ \left. - \frac{mlg}{2} \sin(\theta) \right\} \quad (2.47)$$

### 2.4.3. Modelo Lagrangiano del MRTM

Las ecuaciones (2.21) con (2.37) y (2.25) con (2.47) conforman la formulación Lagrangiana del MRTM que toma la forma siguiente:

$$u_1 = m_{11}\dot{r} + p_{11}\dot{r} + m_{12} \cos\theta \ddot{\theta} + p_{12} \sin\theta \dot{\theta}^2 \quad (2.48)$$

$$u_2 = m_{22}\ddot{\theta} + p_{22}\dot{\theta} + m_{21} \cos\theta \dot{r} + d_{21} \sin\theta \quad (2.49)$$

La tabla 2.1 muestra los coeficientes del sistema, mientras que la tabla 2.2 muestra los parámetros del servomotor DC. El método de identificación de los parámetros del motor y la obtención de las constantes de fricción del carro  $F_e$  y del brazo  $C$  pueden verse en [9]. Habiéndose obtenido que los momentos de inercia y las fricciones equivalentes para los servomotores son bastante similares, se han considerado que:  $J_{eq} = J_{eq1} = J_{eq2}$  y  $B_{eq} = B_{eq1} = B_{eq2}$ .

**Tabla 2.1:** Coeficientes de las ecuaciones (2.48) y (2.49).

$$\begin{aligned}
m_{11} &= \left( \frac{R_a R_p}{n K_A K_m} \right) \left( M_c + m + \frac{J_{eq}}{R_p^2} \right) & p_{11} &= \left( \frac{R_a R_p}{n K_A K_m} \right) \left( F_c + \frac{B_{eq}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2} \right) \\
m_{12} &= \left( \frac{R_a R_p}{n K_A K_m} \right) \frac{ml}{2} & p_{12} &= - \left( \frac{R_a R_p}{n K_A K_m} \right) \frac{ml}{2} \\
m_{21} &= \left( \frac{R_a}{n K_A K_m} \right) \frac{ml}{2} & p_{21} &= 0 \\
m_{22} &= \left( \frac{R_a}{n K_A K_m} \right) \left( \frac{ml^2}{4} + J + J_{eq} \right) & p_{22} &= \left( \frac{R_a}{n K_A K_m} \right) \left( C + B_{eq} + \frac{n^2 K_m K_b}{R_a} \right) \\
d_{11} &= 0 & d_{21} &= - \left( \frac{R_a}{n K_A K_m} \right) \frac{mlg}{2}
\end{aligned}$$

**Tabla 2.2:** Parámetros estimados del sistema [9].

Simbolo	Descripción	Valor	Unidades
$M_c$	masa del carro + masa de la polea	1.6492	$Kg$
$m$	masa del brazo	0.0519	$Kg$
$l$	longitud del brazo	0.30	$m$
$J$	momento de inercia del brazo robótico	$2.38e^{-3}$	$Kgm^2$
$J_{eq}$	momento de inercia equivalente	$3.0551 \times 10^{-4}$	$Kgm^2$
$B_{eq}$	fricción viscosa equivalente	$5.3848 \times 10^{-4}$	$Nm/rad/s$
$C$	coeficiente de fricción viscosa brazo	$9.5e^{-4}$	$Kgm^2/seg$
$F_c$	coeficiente de fricción del carro	2.81	$Kg/seg$
$g$	aceleración de la gravedad	9.81	$mseg^{-2}$
$K_b$	constante de fuerza contraelectromotriz	0.0565	$V/rad/s$
$K_m$	constante de torque de motor	0.0421	$Nm/A$
$R_p$	radio de polea	0.048	$m$
$R_n$	resistencia de armadura	5.3	$\Omega$
$n$	factor de reducción	12.5	
$K_A$	constante de ganancia de amplificación	5.5	

## 2.5 Modelo en espacio de estado del Manipulador

### 2.5.1 Modelo No Lineal

Eligiendo las siguientes variables de estado:

$$x_1 = r \quad (2.50)$$

$$x_2 = \theta \quad (2.51)$$

$$\dot{x}_3 = \dot{r} \quad (2.52)$$

$$\dot{x}_4 = \dot{\theta} \quad (2.53)$$

y despejando  $\dot{x}_3 = \dot{r}$  y  $\dot{x}_4 = \dot{\theta}$  de las ecuaciones (2.48), (2.49), y derivando las ecuaciones (2.50), (2.51) se obtiene finalmente el modelo no lineal del manipulador:

$$\dot{x}_1 = x_3 \quad \dots\dots\dots f_1(x, u, t) \quad (2.54)$$

$$\dot{x}_2 = x_4 \quad \dots\dots\dots f_2(x, u, t) \quad (2.55)$$

$$\begin{aligned} \dot{x}_3 = & \frac{1}{m_{11} \left(1 - \frac{m_{12} m_{21}}{m_{11} m_{22}} \cos^2(x_2)\right)} \left\{ \frac{m_{12} d_{21}}{m_{22}} \cos(x_2) \sin(x_2) + \frac{P_{22} m_{12}}{m_{22}} \cos(x_2) x_4 - \dots \right. \\ & \left. - p_{12} \sin(x_2) x_4^2 - p_{11} x_3 + \frac{m_{12}}{m_{22}} \cos(x_2) u_2 + u_1 \right\} \quad \dots\dots\dots f_3(x, u, t) \quad (2.56) \end{aligned}$$

$$\begin{aligned} \dot{x}_4 = & \frac{1}{m_{22} \left(1 - \frac{m_{12} m_{21}}{m_{11} m_{22}} \cos^2(x_2)\right)} \left\{ \frac{m_{21} P_{12}}{m_{11}} \cos(x_2) \sin(x_2) x_4^2 - d_{21} \sin(x_2) + \dots \right. \\ & \left. - p_{22} x_4 + \frac{m_{21} P_{11}}{m_{11}} \cos(x_2) x_3 - \frac{m_{21}}{m_{11}} \cos(x_2) u_1 + u_2 \right\} \quad \dots\dots\dots f_4(x, u, t) \quad (2.57) \end{aligned}$$

### 2.5.2 Modelo Linealizado

Un proceso o planta, por lo general tiene un comportamiento no lineal, debido a la existencia de elementos físicos como diodos, transistores, transmisiones, etc, de características no lineales. Sin embargo es muy frecuente el uso de controladores lineales por su facilidad de implementación; esto requiere naturalmente que el proceso no lineal se linealice alrededor de algún punto de equilibrio, permitiendo así, trabajar en una zona aproximadamente no linea. Este rango lineal estará limitado por la magnitud de la señal de control (u-max+ o u-max-).

Considerando las siguientes señales de control para el punto de operación:

$$\bar{u}_1 = \bar{u}_2 = 1 \quad (2.58)$$

e igualando a cero el miembro izquierdo de las ecuaciones (2.54), (2.55), (2.56) y (2.57) se obtiene las variables de estado en el punto de operación:

$$\bar{x}_1 = \bar{x}_2 = \bar{x}_3 = \bar{x}_4 = 0 \quad (2.59)$$

y considerando la siguiente sustitución:

$$d_x = \frac{m_{12}m_{21}}{m_{11}m_{22}} \quad (2.60)$$

$$d_y = 1 - d_x \quad (2.61)$$

Considerando nula la presencia de perturbaciones, y operando alrededor del punto de operación expresadas por las ecuaciones (2.58) y (2.59), se pueden obtener las matrices Jacobianas A, B, Cc, Dc, así:

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m_{12}d_{21}}{m_{11}m_{22}d_y} & \frac{-p_{11}}{m_{22}d_y} & \frac{-p_{22}m_{12}}{m_{11}m_{22}d_y} \\ 0 & \frac{-d_{21}}{m_{22}d_y} & \frac{m_{21}p_{11}}{m_{11}m_{22}d_y} & \frac{-p_{22}}{m_{22}d_y} \end{pmatrix}; \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -m_{12} \\ \frac{m_{11}d_y}{m_{11}m_{22}d_y} & \frac{-m_{12}}{m_{11}m_{22}d_y} \\ \frac{-m_{21}}{m_{11}m_{22}d_y} & \frac{1}{m_{22}d_y} \end{pmatrix}$$

$$C_c = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}; \quad D_c = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

lo que al evaluar queda como:

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.0027 & -13.3994 & -0.0087 \\ 0 & 0.6368 & 1.1852 & -2.0400 \end{pmatrix}; \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 6.2062 & -0.0193 \\ -0.4027 & 4.5534 \end{pmatrix}$$

por lo que, el modelo linealizado queda expresado en la forma:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= C_c x + D_c u \end{aligned} \quad (2.62)$$

### 2.5.3 Discretización del Modelo

Habiendo obtenido el modelo linealizado en tiempo continuo expresado por la ecuación (2.62), se debe encontrar el modelo equivalente discreto, debido que el método de control que se va a implementar, es el Control Óptimo Cuadrático en Tiempo Discreto. Un elemento muy importante en el proceso de discretización es determinar el período de muestreo adecuado, el cual permita adquirir con un mínimo error de conversión la información original (en tiempo continuo).

El sistema discretizado toma la forma:

$$\begin{aligned}x(k+1) &= Gx(k) + Hu(k) \\y(k+1) &= C_d x(k) + D_d u(k)\end{aligned}\quad (2.63)$$

donde las matrices G y H se obtienen mediante las siguientes fórmulas:

$$G = e^{AT}; \quad H = \left[ \int_0^T e^{A\lambda} d\lambda \right] B \quad (2.64)$$

mientras que las matrices  $C_d$  y  $D_d$  son las mismas que en tiempo continuo.

Considerando el tiempo de muestreo  $T = 0.005$  segundos, se obtienen las siguientes matrices discretas:

$$G = \begin{pmatrix} 1.0000 & -0.0000 & 0.0048 & -0.0000 \\ 0 & 1.0000 & 0.0000 & 0.0050 \\ 0 & -0.0000 & 0.9352 & -0.0000 \\ 0 & 0.0032 & 0.0057 & 0.9899 \end{pmatrix}; \quad H = \begin{pmatrix} 0.0001 & -0.0000 \\ -0.0000 & 0.0001 \\ 0.0300 & -0.0001 \\ -0.0019 & 0.0227 \end{pmatrix}$$

$$C_d = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \quad D_d = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

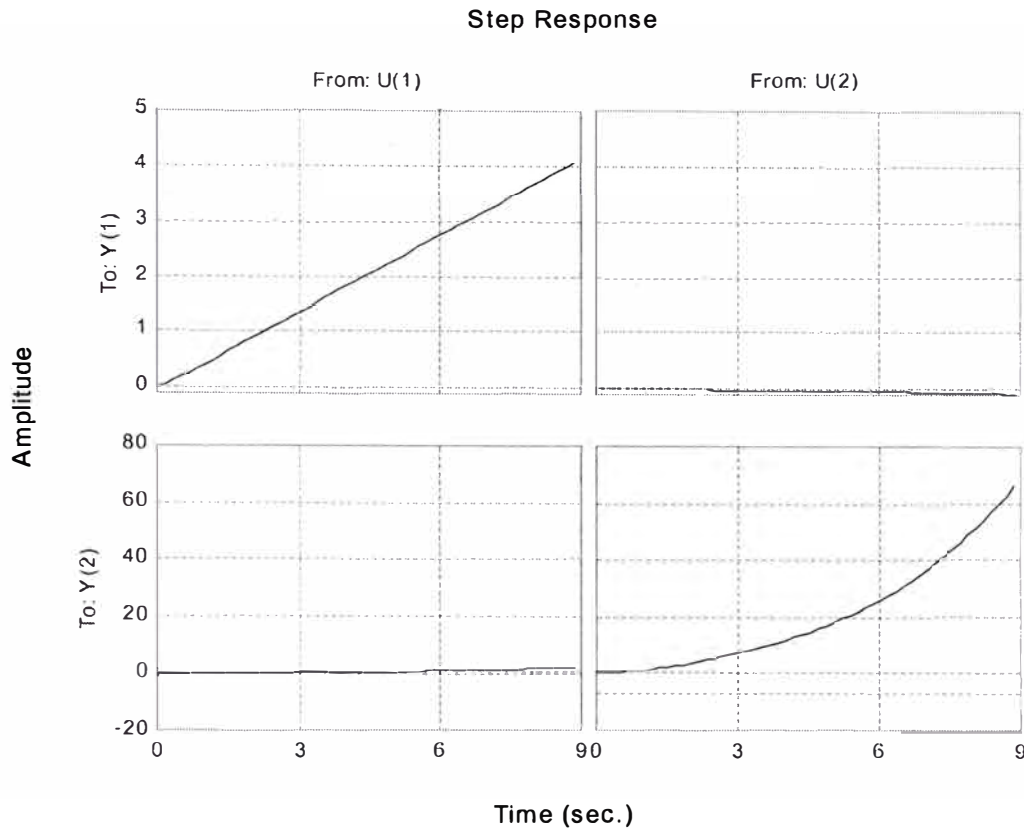
### 2.5.4 Simulación del Modelo a lazo abierto

#### ➤ Simulación en Tiempo Continuo

La respuesta del modelo expresado mediante la ecuación (2.62) en tiempo continuo ante una entrada de un escalón unitario, se muestra en la figura 2.5, que se obtiene mediante el siguiente comando de Matlab:

» step(A,B,Cc,Dc)





**Figura 2.5:** Respuesta del modelo del sistema en tiempo continuo ante una entrada escalón unitario.

De la figura 2.5 se observa el acoplamiento de las señales de control en las variables de salida del sistema. La figura superior izquierda corresponde a la salida  $y_1$  debido a la señal  $u_1$ , y la figura inferior derecha corresponde a la salida  $y_2$  debido a la señal  $u_2$ ; sin embargo la figura superior derecha corresponde al acoplamiento de la señal de control  $u_2$  sobre la salida  $y_1$ , y la figura inferior izquierda corresponden al acoplamiento de la señal de control  $u_1$  sobre la salida  $y_2$ . Se observa también que los acoplamientos tienden a cero, en comparación a las salida  $y_1$  debido a  $u_1$  y la salida  $y_2$  debido a  $u_2$ . Estos acoplamientos no son problema para aplicar Control Optimo al proceso, previa discretización del modelo.

Si se desea obtener las gráficas de respuesta del modelo usando la relación entrada/salida (matriz de transferencia), previamente se tendría que usar dos comandos en Matlab, haciendo referencia a las entradas de control  $u_1$  y  $u_2$ , así:

- **Comando 1:**

$$\gg [\text{num1}, \text{den1}] = \text{ss2tf}(A, B, C_c, D_c, 1)$$

que al ejecutar se obtiene el siguiente resultado:

num1 =

```
0 0.0000 6.2062 12.6641 -3.9509
0 -0.0000 -0.4027 1.9588 0
```

den1 =

```
1.0000 15.4394 26.7082 -8.5293 0
```

lo que significa que se obtiene la siguiente matriz de transferencia:

$$\frac{Y(s)}{U_1(s)} = \frac{\begin{pmatrix} 6.2062s^2 + 12.6641s - 3.9509 \\ -0.4027s^2 + 1.9588s \end{pmatrix}}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.65)$$

La ecuación (2.65) se puede escribir como dos funciones de transferencia, así:

$$\frac{Y_1(s)}{U_1(s)} = \frac{6.2062s^2 + 12.6641s - 3.9509}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.66)$$

$$\frac{Y_2(s)}{U_1(s)} = \frac{-0.4027s^2 + 1.9588s}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.67)$$

Como se podrá apreciar, la ecuación (2.67) es el acoplamiento de  $u_1$  sobre  $y_2$ .

- **Comando 2:**

» [num2,den2]=ss2tf(A,B,Cc,Dc,2)

que al ejecutar se obtiene el siguiente resultado:

num2 =

```
0 -0.0000 -0.0193 -0.0789 -0.0000
0 -0.0000 4.5534 60.9905 0
```

den2 =

```
1.0000 15.4394 26.7082 -8.5293 0
```

lo que significa que se tiene la siguiente matriz de transferencia:

$$\frac{Y(s)}{U_2(s)} = \frac{\begin{pmatrix} -0.0193s^2 - 0.0789s \\ 4.5534s^2 + 60.9905s \end{pmatrix}}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.68)$$

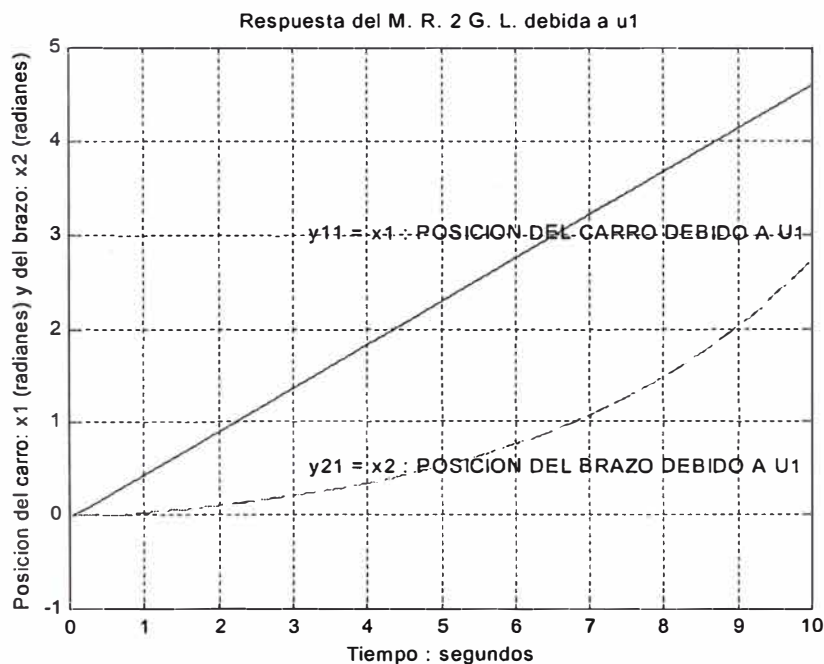
La ecuación (2.68) se puede escribir como dos funciones de transferencia, así:

$$\frac{Y_1(s)}{U_2(s)} = \frac{-0.0193s^2 - 0.0789s}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.69)$$

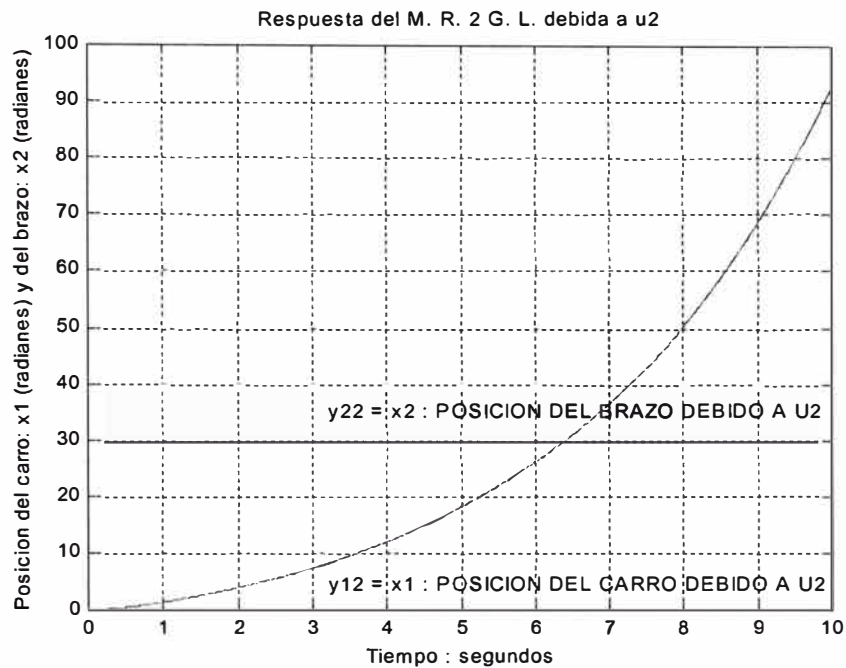
$$\frac{Y_2(s)}{U_2(s)} = \frac{4.5534s^2 + 60.9905s}{s^4 + 15.4394s^3 + 26.7082s^2 - 8.5293s} \quad (2.70)$$

Como se aprecia, la ecuación (2.69) es el acoplamiento de  $u_2$  sobre  $y_1$ .

El programa en Matlab que permite obtener de la respuesta del modelo en tiempo continuo a una entrada escalón unitario se denomina *mr2g\_c*, que se presenta en el Apéndice A, y cuyos resultados se muestran en las figuras 2.6 y 2.7.



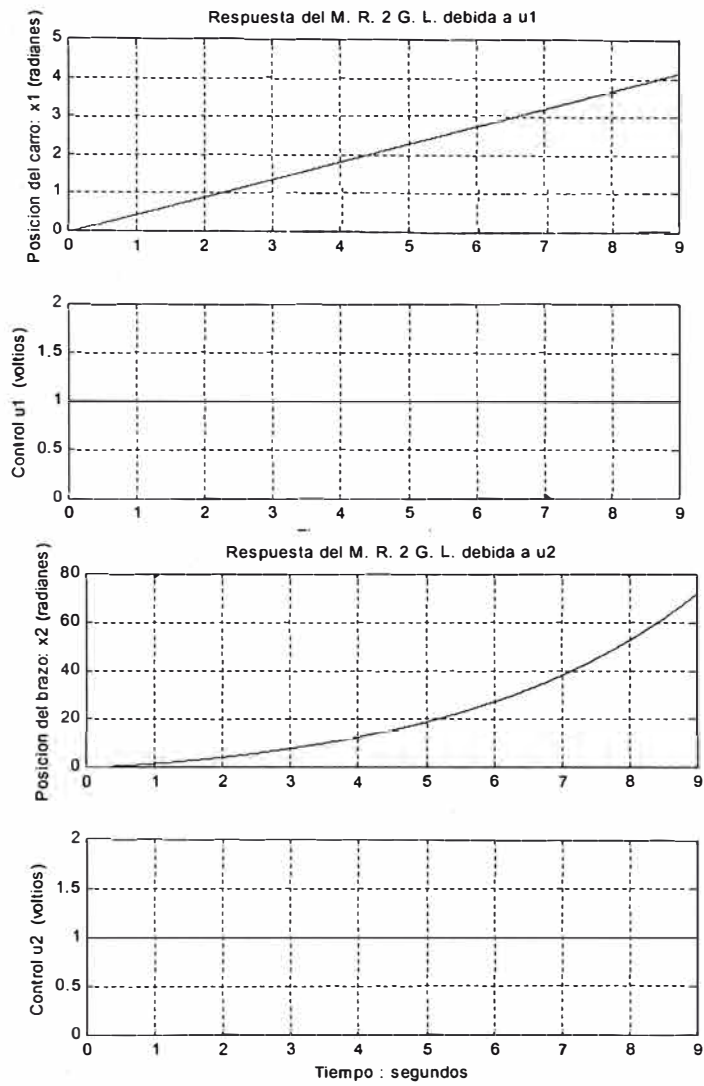
**Figura 2.6:** Posición del carro ( $y_{11}$ ) y del brazo ( $y_{21}$ ) debidos a  $u_1$ . El acoplamiento es  $y_{21}$ .



**Figura 2.7:** Posición del carro ( $y_{12}$ ) y del brazo ( $y_{22}$ ) debidos a  $u_2$ . El acoplamiento es  $y_{12}$ .

### ➤ Simulación en Tiempo Discreto

Considerando un período de muestreo  $T = 0.005$  segundos, el programa en Matlab que permite obtener la respuesta del modelo en tiempo discreto a una entrada escalón unitario se denomina *mr2g\_d*, que se presenta en el Apéndice A, y cuyos resultados se muestran en las figuras 2.8 y 2.9.



**Figura 2.9:** Posición del brazo  $x_2$  y la señal de control  $u_2$ .

## CAPÍTULO III

### CONTROL OPTIMO CUADRATICO DEL MANIPULADOR

#### 3.1 Control Optimo Cuadrático Estacionario

El objetivo del Control Optimo es determinar una ley de control óptima  $u$  que minimice la función de costo  $J$ . En la mayoría de los casos, sin embargo, la búsqueda de la función de costo involucra procedimientos de error y corrección; esto significa que no siempre se podrá estar seguro acerca de la forma exacta que debería tener la función de costo. Es importante indicar que un sistema de control que es óptimo bajo un índice de desempeño es, en general, no óptimo bajo otra función de costo o índice de desempeño.

La solución de un problema de control óptimo consiste en determinar el vector de control óptimo  $u(k)$  que depende de:

- a) La naturaleza de la función de costo y de las restricciones
- b) El estado inicial o la salida inicial
- c) El estado deseado o salida deseada

En general, una solución analítica es muy complicada, por lo que debe usarse el computador. En tal sentido se puede decir que el diseño de sistemas de control óptimo es fundamentalmente un problema computacional.

Para sistemas de control discretos, la función de costo generalmente posee la forma siguiente:

$$J = \sum_{k=0}^N L(x(k), r(k), u(k)) \quad (3.1)$$

donde:

$k$  = tiempo discreto;  $N$  = tiempo discreto final;  $\mathbf{x}(k)$  = vector de estado;  
 $\mathbf{r}(k)$  = vector de referencia;  $\mathbf{u}(k)$  = vector de control óptimo (denominada también fuerza o señal de control). Para aplicaciones prácticas, el vector de control  $\mathbf{u}$  debe estar siempre acotado.

Una condición necesaria para aplicar el control por realimentación de estados es que el proceso sea *completamente controlable*; es *completamente observable* (todos los estados deben estar disponibles o medibles). Si existieran estados que no pudieran medirse directamente, será necesario estimarlos por medio de un estimador u observador de estados.

### 3.1.1 Diseño del Regulador Óptimo Cuadrático

En el sistema de control óptimo cuadrático estacionario, la dinámica de control evoluciona hasta un tiempo  $N$  infinito, por consiguiente, la ganancia  $\mathbf{K}(k)$  se convierte en una matriz constante  $\mathbf{K}$ .

#### Función de costo

La función de costo se representa mediante la siguiente expresión:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} [x^T(k) Q x(k) + u^T(k) R u(k)] \quad (3.2)$$

#### Ecuación de Riccati en estado estacionario

Es necesario encontrar una señal de control  $\mathbf{u}$  óptima que minimice una función de costos  $J$ , se realizan algunas consideraciones de solución, que conllevan a resolver la ya conocida ecuación de Riccati en estado estacionario. dada por:

$$P(k+1) = Q + G^T P(k) G - G^T P(k) H [R + H^T P(k) H]^{-1} H^T P(k) G \quad (3.3)$$

La forma de resolver la ecuación de Riccati en estado estacionario es usar la ecuación de Riccati en estado no estacionario [6], [10]; pero con la inversión en la dirección del tiempo, como se presenta a continuación:

$$P(k+1) = Q + G^T P(k) G - G^T P(k) H [R + H^T P(k) H]^{-1} H^T P(k) G \quad (3.4)$$

Empezando con  $P(0) = 0$ , luego  $P(1)$ , y así sucesivamente hasta obtener una solución en estado estacionario.

### Matriz de ganancia del controlador

Obteniendo la matriz P constante, se usa esta para obtener la matriz ganancia del controlador, según la siguiente ecuación:

$$K = [R + H^T P H]^{-1} H^T P G \quad (3.5)$$

### Ley de control óptima

Habiéndose obtenido la matriz ganancia del controlador, ahora se puede determinar la ley de control óptima para el regulador, mediante la siguiente expresión:

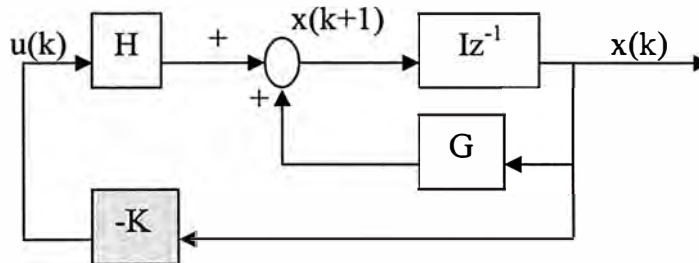
$$u(k) = -K x(k) \quad (3.6)$$

### Función de costo mínima

Finalmente, la función de costo mínima viene expresada por:

$$J_{\min} = \frac{1}{2} x^T(0) P(0) x(0) \quad (3.7)$$

El diagrama de bloques del sistema del control óptimo de estado estacionario se representa en la figura 3.1, en donde se asume que todos los estados se encuentran disponibles.



**Figura 3.1:** Sistema Regulador Óptimo Cuadrático.

Para una operación satisfactoria del sistema de lazo cerrado mostrada en la figura, todas las raíces de la ecuación característica:

$$\det[zI - G + HK] = 0 \quad (3.8)$$

deben ubicarse dentro del círculo unitario del plano Z.

Se puede apreciar que en este caso la ley de control  $u(k)$  sólo depende de la matriz de ganancia del controlador y del estado  $x(k)$ .



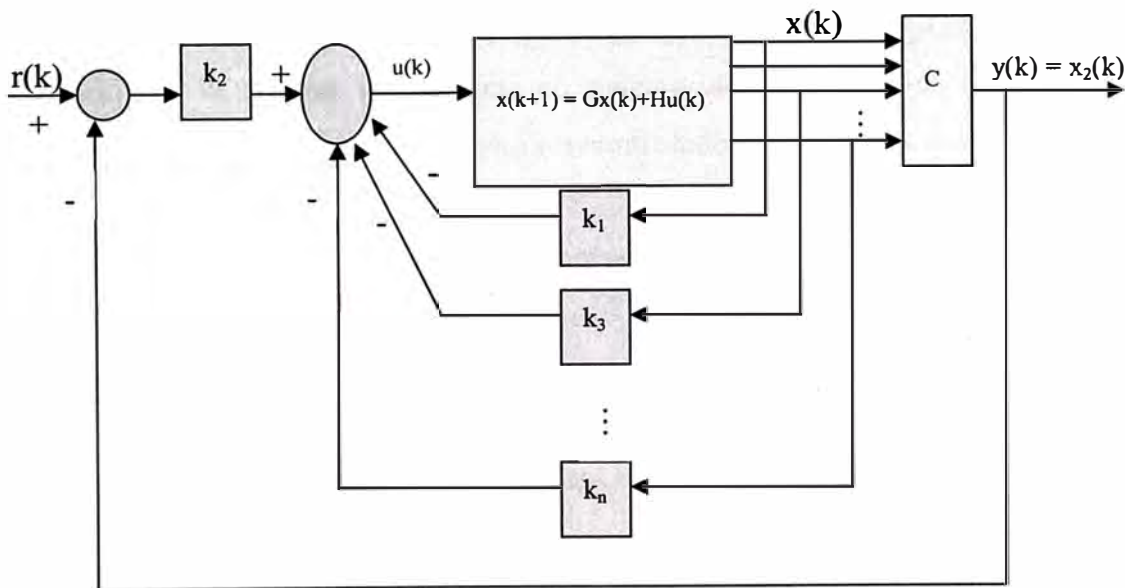
### 3.1.2 Diseño del Control Óptimo Cuadrático para Sistemas de Seguimiento

El controlador óptimo es un sistema de control realimentado, en donde la salida controlada sigue a una señal de referencia  $r(k)$  = (función escalón), es decir estamos considerando un sistema de seguimiento. Esta estructura sólo es aplicable a procesos que poseen un comportamiento integral.

La figura 3.2 muestra el esquema de un controlador donde la variable de estado controlada es  $x_2$ , empleando una ley de control de realimentación de estados que involucra a la matriz ganancia del controlador ( $K$ ), la referencia ( $r$ ) y el vector de estados ( $x$ ).

En consecuencia, la ley de control óptima se expresa mediante la siguiente ecuación:

$$\begin{aligned}
 u(k) &= -k_1 x_1(k) - k_3 x_3(k) - \dots - k_n x_n(k) + k_2 [r(k) - x_2(k)] \\
 &= -\begin{pmatrix} k_1 & k_2 & k_3 & \dots & k_n \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ \vdots \\ x_n(k) \end{pmatrix} + k_2 r(k) \\
 &= -K x(k) + k_2 r(k)
 \end{aligned} \tag{3.9}$$



**Figura 3.2:** Esquema del Controlador óptimo proporcional.

Por otro lado, la ecuación de estado discreto del proceso viene expresada por:

$$x(k+1) = Gx(k) + Hu(k) \tag{3.10}$$

por lo que, reemplazando la ecuación (3.9) en (3.10) se obtiene:

$$\begin{aligned}x(k+1) &= Gx(k) + Hu(k) \\ &= (G - HK)x(k) + H k_2 r(k)\end{aligned}\quad (3.11)$$

que viene a ser la ecuación de estado del sistema de control, es decir del sistema realimentado.

Del modelo del proceso (Capítulo II), se determina que el manipulador es un sistema de tipo 1 debido a la presencia del integrador, lo que significa que si se desea aplicar el Control Optimo Proporcional al modelo del manipulador, se obtendría error nulo en estado estacionario, considerando una entrada escalón; sin embargo si se quisiera que responda de la misma manera para trayectorias variables, no se obtendría los mismos resultados; es decir, no existiría seguimiento perfecto. En tal sentido, en la siguiente sección se aborda el diseño del Control Optimo Proporcional que resuelve el problema indicado.

## 3.2 Diseño del Control Optimo Cuadrático del Manipulador

### 3.2.1 Diseño del Control Optimo Proporcional Integral SISO

Como se había mencionado, para procesos que no poseen propiedades integradoras, la inclusión de la acción integral en el sistema de control permite obtener un error estacionario nulo. La figura 3.3 ilustra un controlador óptimo para sistemas de una entrada y una salida (SISO).

De la figura 3.3 se pueden obtener las siguientes ecuaciones:

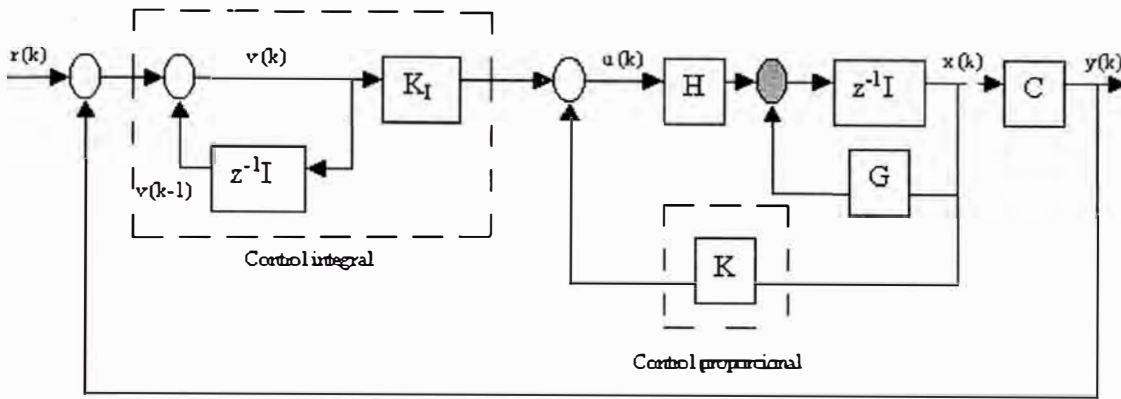
#### Ecuaciones de estado y de salida del proceso:

$$x(k+1) = Gx(k) + Hu(k); \quad (3.12)$$

$$y(k) = Cx(k) \quad (3.13)$$

#### Ley de control:

$$u(k) = -K x(k) + K_1 v(k) \quad (3.14)$$



**Figura 3.3:** Esquema del Controlador Óptimo Proporcional Integral.

**Matriz de ganancia del controlador:**

$$K = [K_1 \quad K_2 \quad \dots \quad K_n] \quad (3.15)$$

**Ecuación para el integrador:**

$$v(k) = v(k-1) + r(k) - y(k) \quad (3.16)$$

$$\begin{aligned} \Rightarrow v(k+1) &= v(k) + r(k+1) - y(k+1) \\ &= v(k) + r(k+1) - C[Gx(k) + Hu(k)] \\ &= (1 - CHK_1)v(k) + (-CG + CHK)x(k) + r(k+1) \end{aligned} \quad (3.17)$$

empleando las ecuaciones (3.12) y (3.14) se obtiene:

$$\begin{aligned} x(k+1) &= Gx(k) + H[-Kx(k) + K_1v(k)] \\ &= (G - HK)x(k) + HK_1v(k) \end{aligned} \quad (3.18)$$

y de las ecuaciones (3.17), (3.18) y (3.13) se deduce:

$$\begin{bmatrix} x(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} G - HK & HK_1 \\ -CG + CHK & 1 - CHK_1 \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k+1) \quad (3.19)$$

$$y(k) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} \quad (3.20)$$

En estado estacionario ( $k \rightarrow \infty$ ), los valores de  $x(k)$ ,  $u(k)$  y  $v(k)$  toman valores  $x(\infty)$ ,  $u(\infty)$  y  $v(\infty)$ . Entonces la ecuación (3.19) se convierte en:

$$\begin{bmatrix} x(\infty) \\ v(\infty) \end{bmatrix} = \begin{bmatrix} G - HK & HK_1 \\ -CG + CHK & 1 - CHK_1 \end{bmatrix} \begin{bmatrix} x(\infty) \\ v(\infty) \end{bmatrix} + \begin{bmatrix} 0 \\ r \end{bmatrix} \quad (3.21)$$

Si se efectúa la siguiente asignación:

$$x(k) - x(\infty) = x_e(k) \quad (3.22)$$

$$v(k) - v(\infty) = v_e(k) \quad (3.23)$$

y se resta la ecuación (3.21) de (3.19) y se usa las relaciones (3.22) y (3.23) se obtiene:

$$\begin{aligned} \begin{bmatrix} x_e(k+1) \\ v_e(k+1) \end{bmatrix} &= \begin{bmatrix} G - HK & HK_I \\ -CG + CHK & I - CHK_I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix} \\ &= \begin{bmatrix} G & 0 \\ -CG & I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix} + \begin{bmatrix} H \\ -CH \end{bmatrix} \begin{bmatrix} -K & K_I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix} \end{aligned} \quad (3.24)$$

que finalmente se reescribe como:

$$\xi(k+1) = \tilde{G}\xi(k) + \tilde{H}\omega(k) \quad (3.25)$$

La ecuación de Riccati y la ecuación de ganancia del controlador  $\tilde{K}$  son:

$$\tilde{P} = \tilde{Q} + \tilde{G}^T \tilde{P} \tilde{G} - \tilde{G}^T \tilde{P} \tilde{H} [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.26)$$

$$\tilde{K} = [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.27)$$

donde:

$$\begin{aligned} \omega(k) &= -\tilde{K}\xi(k); \quad \xi(k) = \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix}; \quad \tilde{G}(k) = \begin{bmatrix} G & 0 \\ -CG & I \end{bmatrix} \\ \tilde{H}(k) &= \begin{bmatrix} H \\ -CH \end{bmatrix}; \quad \tilde{K}(k) = [K \quad -K_I] \end{aligned}$$

Finalmente, con el calculo de la ecuación (3.27) se logra obtener el diseño del controlador para el sistema SISO.

### 3.2.2 Diseño del Control Óptimo Proporcional Integral MIMO

El esquema del sistema de control óptimo Proporcional Intergral MIMO para el manipulador robótico traslacional es el mismo que para el sistema SISO, con la diferencia que al tratarse de un sistema MIMO, deben considerarse el orden de los vectores y matrices presentes en el sistema, así:

- $r(k)$  es un vector de referencia de orden  $r$
- $u(k)$  es un vector de orden  $m$
- $x(k)$  es un vector de estados de orden  $n$
- $y(k)$  es un vector de salida de orden  $r$
- $G$  es una matriz de estados de dimensión  $n \times n$
- $H$  es una matriz de control de dimensión  $n \times m$
- $C$  es una matriz de salida de dimensión  $r \times n$
- $D$  es una matriz de transmisión directa de dimensión  $r \times m$  donde:

$n = 4$  (número de variables de estado),  $r = 2$  (número de salidas del proceso a ser controladas),  $m = 2$  (número de entradas al proceso).

Reescribiendo las ecuaciones de estado y de salida del proceso linealizado y discretizado como:

$$x(k+1) = Gx(k) + Hu(k) \quad (3.28)$$

$$y(k) = Cx(k) + Du(k) \quad (3.29)$$

y considerando un período de muestreo de  $T = 0.02$ , las matrices  $G$ ,  $H$ ,  $Cd$  y  $Dd$  vienen a ser:

$$G = \begin{pmatrix} 1 & 0 & 0.0175 & 0 \\ 0 & 1.0001 & 0.0002 & 0.0196 \\ 0 & 0 & 0.7649 & -0.0001 \\ 0 & 0.0125 & 0.0204 & 0.9601 \end{pmatrix}; \quad H = \begin{pmatrix} 0.0011 & 0 \\ -0.0001 & 0.0009 \\ 0.1089 & -0.0003 \\ -0.0066 & 0.0892 \end{pmatrix}$$

$$Cd = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}; \quad D = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (3.30)$$

La dinámica del sistema de control multivariable se determina por las ecuaciones de estado, de salida y de control, como sigue:

$$x(k+1) = Gx(k) + Hu(k) \quad (3.31)$$

$$y(k) = Cx(k) + Du(k) \quad (3.32)$$

$$u(k) = -Kx(k) + K_I v(k) \quad (3.33)$$

siendo  $v(k)$  el vector de salida del integrador de orden  $m$  (2),  $K$  es la matriz ganancia del controlador de dimensión  $m \times n$  (2 x 4) y la matriz ganancia del integrador de dimensión  $m \times r$  que adquieren la siguiente forma:

$$K = \begin{pmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \end{pmatrix}; \quad K_I = \begin{pmatrix} ki_{11} & ki_{12} \\ ki_{21} & ki_{22} \end{pmatrix} \quad (3.34)$$

$$v(k) = v(k-1) + r(k) - y(k) \quad (3.35)$$

Finalmente, la ecuación de estado del controlador óptimo proporcional integral MIMO se expresa por:

$$\zeta(k+1) = \tilde{G}\zeta(k) + \tilde{H}\omega(k) \quad (3.36)$$

donde:

$$\omega(k) = -\tilde{K}\zeta(k); \quad \zeta(k) = \begin{pmatrix} x_e(k) \\ v_e(k) \end{pmatrix}; \quad \tilde{G}(k) = \begin{pmatrix} G & 0 \\ -CG & I \end{pmatrix}$$

$$\tilde{H}(k) = \begin{pmatrix} H \\ -CH \end{pmatrix}; \quad \tilde{K} = (K \quad -K_I)$$

La solución al problema de diseño de control óptimo proporcional integral multivariable es, calcular la matriz ganancia ampliada del controlador  $\tilde{K}$  que minimice la función de costo. El calculo de la matriz ganancia  $\tilde{K}$  está condicionada a la solución de la ecuación asociada de Riccati, por lo que se reescribe las ecuaciones siguientes:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} [\zeta^T(k) \tilde{Q} \zeta(k) + \omega^T(k) \tilde{R} \omega(k)] \quad (3.37)$$

$$\tilde{P} = \tilde{Q} + \tilde{G}^T \tilde{P} \tilde{G} - \tilde{G}^T \tilde{P} \tilde{H} [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.38)$$

$$\tilde{K} = [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.39)$$

donde:

- $\tilde{P}$  es una matriz simétrica definida positiva de dimensión  $(n+r) \times (n+r)$
- $\tilde{Q}$  es una matriz simétrica definida positiva o semidefinida positiva de dimensión  $(n+r) \times (n+r)$
- $\tilde{R}$  es una matriz simétrica definida positiva de dimensión  $m \times m$

El diseño del sistema de control óptimo proporcional integral multivariable contempla los siguientes pasos:

1. Elegir las matrices de ponderación  $\tilde{Q}$  y  $\tilde{R}$ , de la forma:

$$\tilde{Q} = \begin{pmatrix} q_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & q_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & q_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & q_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & q_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & q_{66} \end{pmatrix}; \quad \tilde{R} = \begin{pmatrix} r_{11} & 0 \\ 0 & r_{22} \end{pmatrix}$$

El criterio de elección de los valores  $q$  y  $r$  de las matrices  $\tilde{Q}$  y  $\tilde{R}$  son:

- a) El valor de  $q_{11}$  pondera a la variable de estado  $x_1$ ,  $q_{22}$  pondera a  $x_2$ ,  $q_{33}$  pondera a  $x_3$ ,  $q_{44}$  pondera a  $x_4$ ,  $q_{55}$  pondera al primer integrador y  $q_{66}$  pondera al segundo integrador. Por ejemplo dar un mayor peso a  $q_{11}$  significa incrementar la velocidad de respuesta de la variable  $x_1$  para llegar a tiempo estable; dar un mayor peso a  $q_{55}$  permite incrementar la velocidad del integrador 1 para obtener error estacionario nulo.

- b) El valor de  $r_{11}$  pondera a la señal de control  $u_1$  y el valor  $r_{22}$  pondera a la señal de control  $u_2$ . Un mayor peso en  $r_{11}$  significa disminuir la magnitud de la señal de control en el transitorio; pero el tiempo de establecimiento aumentará, haciéndose más lento.

En definitiva, se tiene que escoger una solución de compromiso entre los valores de  $\tilde{Q}$  y  $\tilde{R}$ .

2. Resolver la ecuación de Riccati en forma recursiva, que a continuación se escribe:

$$\tilde{P}(k+1) = \tilde{Q} + \tilde{G}^T \tilde{P}(k) \tilde{G} - \tilde{G}^T \tilde{P}(k) \tilde{H} [\tilde{R} + \tilde{H}^T \tilde{P}(k) \tilde{H}]^{-1} \tilde{H}^T \tilde{P}(k) \tilde{G}$$

3. Luego de determinar la matriz  $\tilde{P}$  (en tiempo estacionario), se puede obtener la matriz ganancia del controlador  $\tilde{K}$ , usando la siguiente ecuación:

$$\tilde{K} = [K \quad -K_r] = [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G}$$

A continuación las matrices asumidas para Matlab:

$$\tilde{Q} = \begin{pmatrix} 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 \end{pmatrix}; \quad \tilde{R} = \begin{pmatrix} 200 & 0 \\ 0 & 200 \end{pmatrix}$$

y las matrices obtenidas siguientes son :

$$\tilde{P} = 1.0e + 005 * \begin{pmatrix} 1.8434 & 0.0121 & 0.1231 & 0.0006 & -0.0583 & -0.0007 \\ 0.0121 & 0.8187 & 0.0099 & 0.1382 & -0.0002 & -0.0378 \\ 0.1231 & 0.0099 & 0.0087 & 0.0021 & -0.0036 & -0.0003 \\ 0.0006 & 0.1382 & 0.0021 & 0.0286 & 0.0000 & -0.0049 \\ -0.0583 & -0.0002 & -0.0036 & 0.0000 & 0.0035 & 0.0000 \\ -0.0007 & -0.0378 & -0.0003 & -0.0049 & 0.0000 & 0.0028 \end{pmatrix}$$

$$\tilde{K} = \begin{pmatrix} 7.5049 & 0.0546 & 0.5166 & 0.0120 & -0.2164 & -0.0008 \\ -0.0019 & 6.0428 & 0.0896 & 1.2628 & 0.0010 & -0.2107 \end{pmatrix}$$

El programa en Matlab que permite diseñar el sistema de Control Optimo Proporcional Integral MIMO se puede encontrar en el Apéndice con el nombre *mr2d\_opt.m*, y que a continuación se presenta un extracto del mismo:

```
% DISEÑO DEL SISTEMA DE CONTROL OPTIMO PROPORCIONAL INTEGRAL

% MATRICES AMPLIADAS DEBIDO A LA INSERCIÓN DEL INTEGRADOR EN EL
CONTROLADOR
Ga = [G zeros(4,2)
      -Cd*G eye(2,2)];
Ha = [H;-Cd*H];

% CÁLCULO DE LA CONTROLABILIDAD EN TIEMPO DISCRETO DEL SISTEMA
AMPLIADO
Md=[Ha Ga*Ha Ga^2*Ha Ga^3*Ha Ga^4*Ha Ga^5*Ha];
rangoMd=rank(Md); %RANGO DEL SISTEMA AMPLIADO= 6
% CÁLCULO DE LA OBSERVABILIDAD EN TIEMPO DISCRETO DEL SISTEMA
AMPLIADO
Nd=[Cd' G'*Cd' (G')^2*Cd' (G')^3*Cd'];
rangoNd=rank(Nd); %RANGO DEL SISTEMA AMPLIADO= 4
% MATRICES DE PONDERACIÓN
R=200*eye(2);
Qa=10*eye(6);
P=zeros(6,6);
for i=1:1000
P=Qa+Ga'*P*Ga-Ga'*P*Ha*inv(R+Ha'*P*Ha)*Ha'*P*Ga;
end
K1=inv(R+Ha'*P*Ha)*Ha'*P*Ga;
[KK,PP]=dlqr(Ga,Ha,Qa,R); % K1=KK, P=PP
K=[KK(1,1) KK(1,2) KK(1,3) KK(1,4)
   KK(2,1) KK(2,2) KK(2,3) KK(2,4)];
KI = - [KK(1,5) KK(1,6)
        KK(2,5) KK(2,6)];
```



$$K_{c1} = [K(1,1) \ K(1,2) \ K(1,3) \ K(1,4)];$$

$$K_{c2} = [K(2,1) \ K(2,2) \ K(2,3) \ K(2,4)];$$

$$K_{I1} = - [K(1,5)];$$

$$K_{I2} = - [K(2,6)];$$

### 3.2.3 Diseño del Observador de Estados

#### ➤ El Observador Óptimo Cuadrático SISO

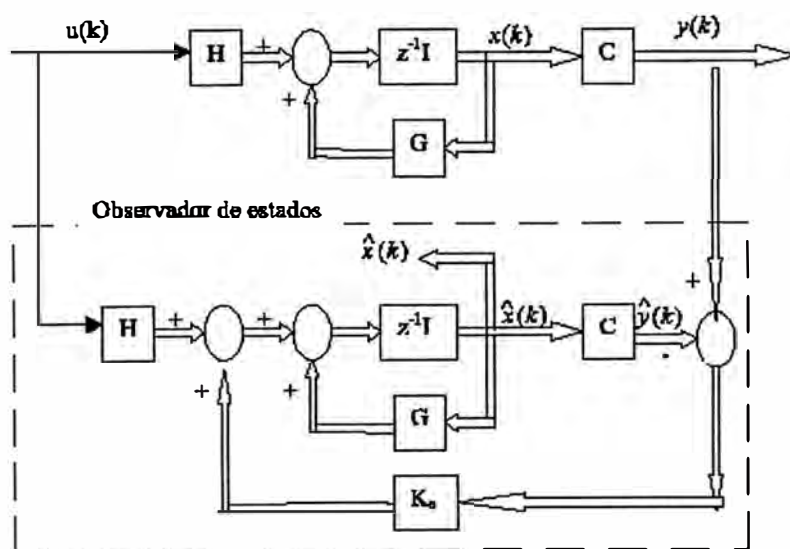
Cuando no es posible medir todas las variables de estado, ya sea por razones de encarecimiento de los sensores o porque algunas de las variables no son físicamente medibles, entonces es necesario estimar al vector de estado  $x(k)$ , usando para ello el algoritmo del observador de estados [2], [6].

En la figura 3.4 se muestra el diagrama de bloques del observador de estados discreto, donde:

- $K_e$  es la matriz ganancia del observador con dimensión  $n \times 1$ .
- $C$  es la matriz de salida de dimensión  $1 \times n$ .
- $\hat{x}(k)$  es el vector de estado estimado de dimensión  $n$ .
- $\hat{y}(k)$  es el vector de salida estimado de dimensión  $1$ .

La ecuación de estado del observador se obtienen de la figura 3.4, así:

$$\begin{aligned} \hat{x}(k+1) &= G\hat{x}(k) + Hu(k) + K_e[y(k) - \hat{y}(k)] \\ &= G\hat{x}(k) + Hu(k) + K_e C[x(k) - \hat{x}(k)] \end{aligned} \quad (3.40)$$



**Figura 3.4:** Diagrama de bloques del Observador de Estados del sistema.

Restando la ecuación (3.40) de la ecuación (3.31) se obtiene la ecuación del error del observador, dada por:

$$e(k+1) = [G - K_e C]e(k) \quad (3.41)$$

donde:

$$e(k) = x(k) - \hat{x}(k)$$

La ecuación característica del observador viene dada por:

$$|zI - G + K_e C| = 0 \quad (3.42)$$

y permite verificar que las raíces de la ecuación característica deben posicionarse dentro del círculo unitario para una operación satisfactoria del observador. Asimismo, la ganancia del estimador debe escogerse adecuadamente para que el error tienda asintóticamente a una velocidad suficiente.

Para obtener la matriz ganancia del observador en tiempo estacionario, previamente se resuelve la ecuación de Riccati para el observador, así como sigue:

$$P_e = Q_e + GP_e G^T - GP_e C^T [R_e + CP_e C^T]^{-1} CP_e G^T \quad (3.43)$$

La forma de obtener  $P_e$  es usar la ecuación de Riccati para calcular  $P$ , pero con la inversión del tiempo y realizando las siguientes modificaciones:

$$G \rightarrow G^T; \quad H \rightarrow C^T; \quad K \rightarrow K_e^T$$

por lo que la ecuación de Riccati viene a ser:

$$P_e(k+1) = Q_e + GP_e(k)G^T - GP_e(k)C^T [R_e + CP_e(k)C^T]^{-1} CP_e(k)G^T \quad (3.44)$$

Finalmente, podemos calcular la matriz ganancia del observador, dada por la siguiente ecuación:

$$K_e = [R_e + CP_e C^T]^{-1} CP_e G^T \quad (3.45)$$

### ➤ El Observador Optimo Cuadrático MIMO

Las ecuaciones del Observador Optimo Cuadrático MIMO son las mismas que para el caso SISO, con las siguientes consideraciones :

- $K_e$  es una matriz de ganancia del observador de dimensión  $n \times r$  (en este caso es  $4 \times 2$ ).
- $P_e$  es una matriz simétrica definida positiva de dimensión  $n \times n$  (es este caso es  $4 \times 4$ ).

Por consiguiente, rescribiendo las ecuaciones que gobiernan la dinámica del observador MIMO se obtiene:

$$\begin{aligned}\hat{x}(k+1) &= G\hat{x}(k) + Hu(k) + K_e[y(k) - \hat{y}(k)] \\ &= G\hat{x}(k) + Hu(k) + K_e C[x(k) - \hat{x}(k)]\end{aligned}\quad (3.46)$$

$$e(k+1) = [G - K_e C]e(k) \quad (3.47)$$

$$P_e(k+1) = Q_e + GP_e(k)G^T - GP_e(k)C^T [R_e + CP_e(k)C^T]^{-1} CP_e(k)G^T \quad (3.48)$$

$$K_e = [R_e + CP_e C^T]^{-1} CP_e G^T \quad (3.49)$$

A continuación las matrices asumidas para Matlab:

$$Q_e = \begin{pmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{pmatrix}; \quad R_e = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix}$$

y las matrices obtenidas siguientes son :

$$P_e = 1.0e+003 * \begin{pmatrix} 0.1092 & 0.0000 & 0.0035 & 0.0003 \\ 0.0000 & 0.1097 & 0.0002 & 0.0251 \\ 0.0035 & 0.0002 & 0.2409 & 0.0133 \\ 0.0003 & 0.0251 & 0.0133 & 1.2264 \end{pmatrix}$$

$$K_e = \begin{pmatrix} 0.9166 & 0.0001 & 0.0221 & 0.0033 \\ 0.0000 & 0.9207 & 0.0015 & 0.2129 \end{pmatrix}$$

A continuación se presenta nuevamente un extracto del programa *mr2d\_opt.m*, que permite obtener la solución de la ecuación de Riccati y la matriz ganancia del observador.

```
% CALCULO DE LA GANANCIA DEL OBSERVADOR DE ESTADOS
```

```
% MATRICES DE PONDERACION PARA EL OBSERVADOR
```

```
Qe=100*eye(4); Re=10*eye(2);
```

```
Pe = zeros(4,4);
```

```
for i=1:200
```

```
Pe = Qe + G*Pe*G' -G*Pe*Cd'*inv(Re+Cd*Pe*Cd')*Cd*Pe*G';
```

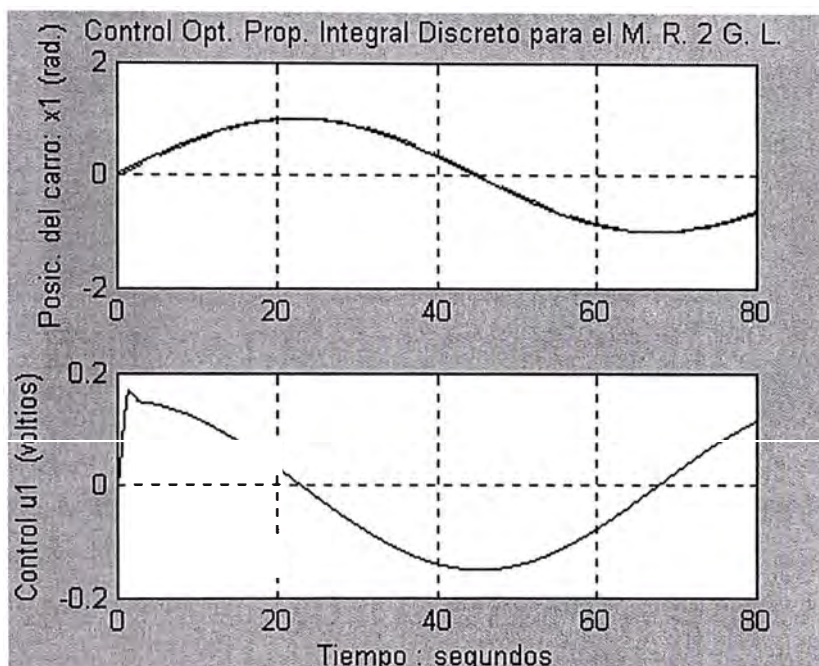
```
end
```

$$KeT = \text{inv}(\text{Re} + Cd * Pe * Cd') * Cd * Pe * G'; Ke = KeT';$$

$$[KKeT, PPe] = \text{dlqr}(G', Cd', Qe, Re); KKe = KKeT'; \% Pe = PPe, Ke = KKe$$

### 3.3 Simulación del Sistema de Control Optimo Cuadrático del Manipulador

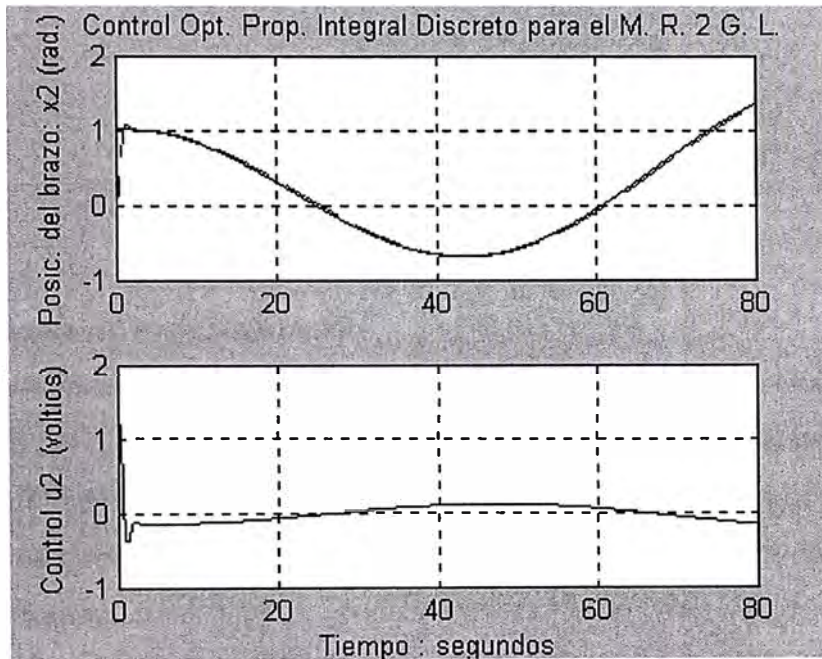
Los resultados de la simulación del Sistema de Control Optimo Proporcional Integral con Observador Optimo, se puede ver en las figuras 3.5 y 3.6. El tiempo de muestreo utilizado es de  $T = 0.02$  segundos, que corresponde a una frecuencia de muestreo de 50 Hz.



**Figura 3.5:** Posición del carro  $x_1$  y la señal de control correspondiente ( $u_1$ ).

Del resultado gráfico de la figura 3.5 se puede observar que la señal de control  $u_1(t)$  necesario para que la posición angular  $x_1(t)$  del carro siga a la referencia  $r_1(t)$ , alcanza un valor pico transitorio de aproximadamente 0.17 voltios, y manteniéndose alrededor de  $\pm 0.15$  voltios (de forma sinusoidal), debido justamente a una referencia sinusoidal. En forma idéntica, de la figura 3.6 se puede observar que la tensión de control  $u_2(t)$  necesario para llevar la posición angular del brazo del manipulador robótico traslacional a la posición deseada  $r_2(t)$ , alcanza un valor pico transitorio de aproximadamente 1.05 voltios, para luego tender alrededor de cero en tiempo estacionario.

De la figura 3.5 se puede observar, que el carro sigue con exactitud la trayectoria de referencia  $r_1(t)$  ; y en la figura 3.6 se observa que hay un pequeño sobrepico en el transitorio del orden 1 mv, luego sigue exactamente a la trayectoria de referencia  $r_2(t)$ .



**Figura 3.6:** Posición del brazo  $x_2$  y la señal de control correspondiente ( $u_2$ ).

## **CAPÍTULO IV**

### **PROPUESTA DE IMPLEMENTACION**

#### **4.1 Esquema de implementación**

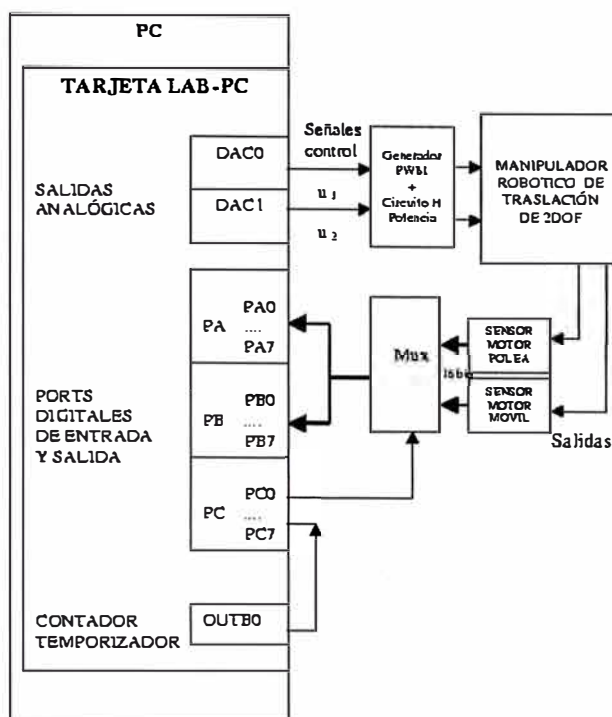
La implementación del sistema manipulador robótico traslacional de dos grados de libertad (MRT2GL) se puede realizar usando DSP, microcontroladores y tarjetas de adquisición de datos y PC. En la figura 4.1 se muestra un esquema de implementación del sistema del manipulador referido, usando una tarjeta de adquisición de datos LAB-PC+ de National Instrument.

En trabajos anteriores realizados en el laboratorio de Post-Grado FIEE, se llegó a implementar el sistema, usando estrategias de control avanzado distintas al óptimo [8] [9].

#### **4.2 Hardware del Sistema**

Los componentes físicos del sistema de control está compuesto por:

➤ **El actuador:** Conformado por dos servomotores DC de campo magnético permanente y escobillas conmutadas (con encoder óptico incremental que puede ser de 1250 cuentas por vuelta), dos generadores PWM y dos amplificadores de potencia que generen las tensiones necesarias para el control de giro de los servomotores DC.



**Figura 4.1:** Diagrama de bloques del sistema.

➤ **El Sensor:** Conformado por un codificador óptico rotario incremental por cada servomotor (incorporado en la parte posterior interna de cada motor), un decodificador en desfase de cuadratura por cada servomotor y un contador UP/DOWN de 16 bits doble (dos canales de sensado). El contador UP/DOWN efectúa la cuenta ascendente o descendente, dependiendo de la información que se obtenga del decodificador de cuadratura. Si el decodificador entrega cuatro pulsos, entonces el contador cuenta hasta cuatro, indicando que el motor se ha movido 4 ranuras de las 1250 con que cuenta, por lo que convertido a su valor en ángulo se tiene la información correspondiente al movimiento de cada servomotor. Esta información binaria de 16 bits por canal se selecciona previamente por un circuito de multiplexación, que permite en un instante dado pasar el dato digital de 16 bits de un canal hacia la tarjeta de adquisición de datos LABPC+ de National Instrument, y en otro instante pasar la información digital del otro canal, repitiéndose este proceso en todo el tiempo de funcionamiento del sistema de control. En la figura 4.2 se aprecia con mejor detalle el diagrama de bloques del sistema descrito anteriormente.

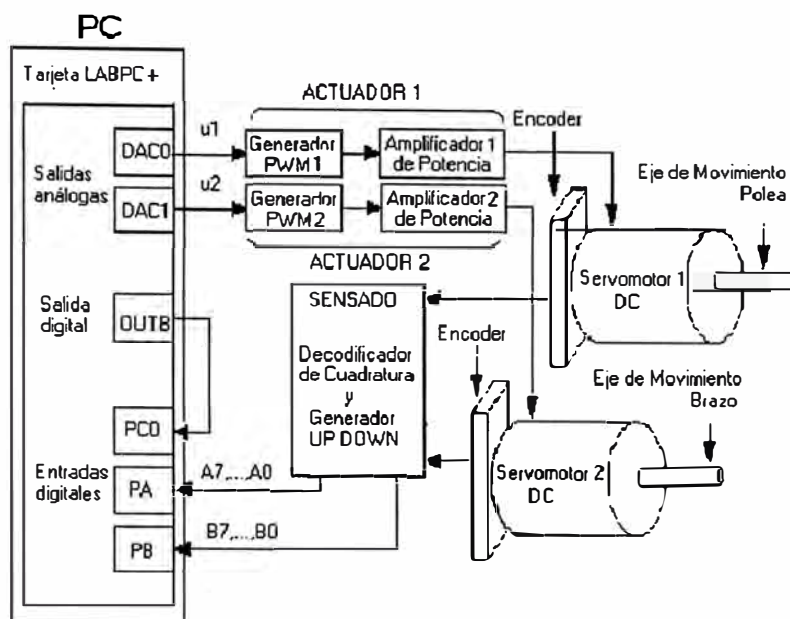


Figura 4.2: Diagrama de bloques detallado del sistema.

➤ **Interface LABPC+:** La interface entre la señal de posición sensada por cada servomotor y el computador, que permita generar la señal de control correspondiente es una tarjeta de adquisición de datos ABPC+ de National Instrument, cuyas entradas digitales de 8 bits cada una (puertos digitales *PA* y *PB*) que conforman un grupo de entradas de 16 bits por canal, reciben la información digital que luego de convertida en ángulo en el programa ejecutado en el computador, permite comparar con las referencias correspondientes al carro y al brazo, para activar las señales de control digitales  $u_1$  y  $u_2$ , las que ingresarán a los convertidores D/A correspondientes (*DAC0* y *DAC1* de la tarjeta LABPC+) para producir en sus salidas las señales analógicas que viene a ser las entradas de los generadores PWM, y así cerrar el ciclo de control del sistema.

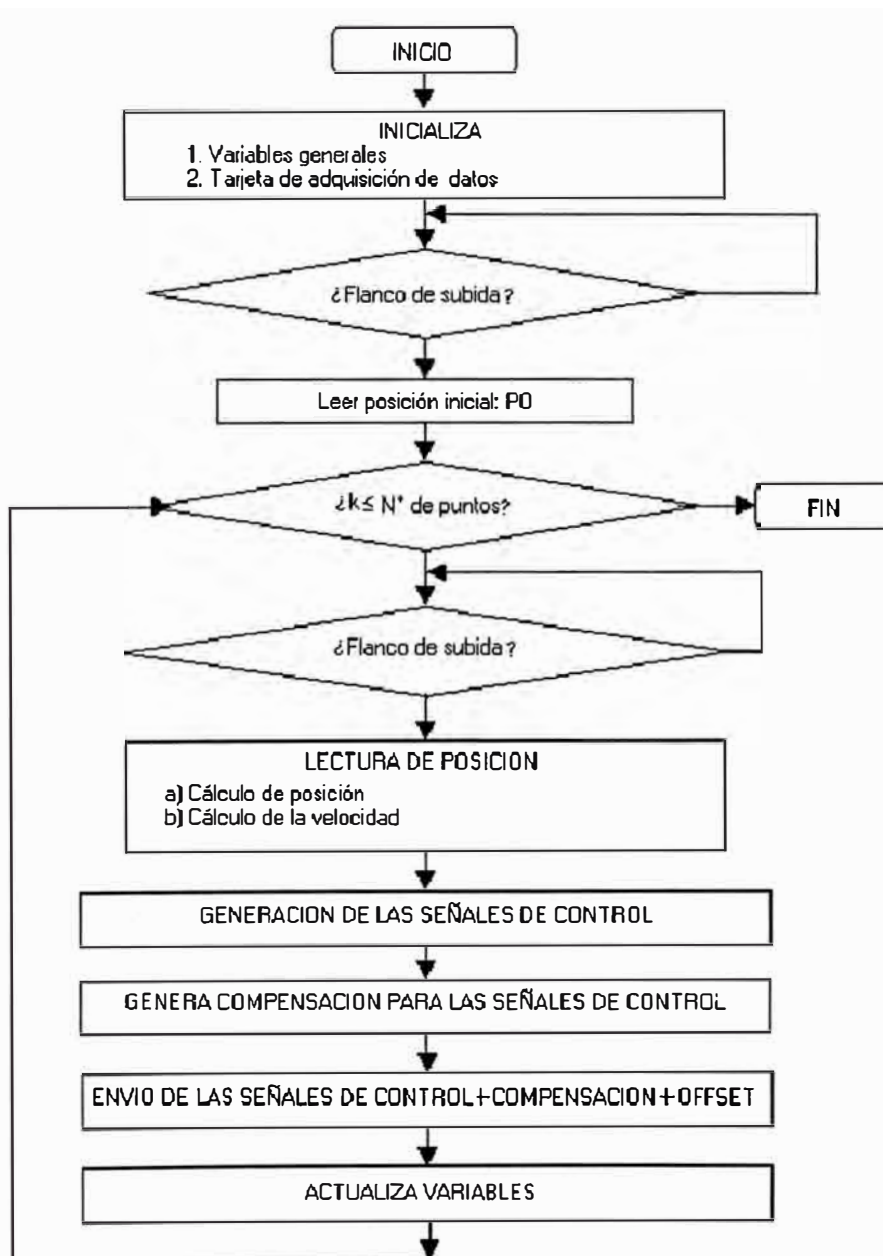
### 4.3 Software del Sistema

Similarmente como en [8], [9] y [10], la propuesta de implementación del programa es en lenguaje C. El diagrama de flujo del sistema de control que permita elaborar el programa de Control Optimo Proporcional Integral con Observador de Estados, se muestra en la figura 4.3. Como se podrá apreciar el diagrama de flujo describe los siguientes pasos:

1. Inicialización de la tarjeta de adquisición de datos LABPC+ y de las variables utilizadas en el programa.
2. Detección del flanco de subida del reloj (clock) y lectura del primer dato de la posición inicial del carro y del brazo, y se inicializa el algoritmo.



3. Comprobación que la variable  $k$  sea menor que el número de puntos  $n$ . El tiempo de duración de funcionamiento  $t$  está dada por  $t = kT$ .
4. Detección del flanco de subida del reloj, iniciándose un período de adquisición de datos y control. Se lee la posición  $y$  y luego se estima la velocidad.
5. Se calcula las señales de control  $u_1$  y  $u_2$ , se compensa las tensiones de las fricciones estáticas y de Coulomb  $v$ , se aplica restricciones para las señales de control y luego se envían las señales de control  $u_1$   $u_2$  por el DAC0 y DAC1 de la tarjeta de adquisición hacia los generadores PWM.
6. Se actualizan las variables y se repite el lazo de control hasta que  $k$  alcance el valor de  $n$ .



**Figura 4.3:** Diagrama de flujo del algoritmo de control del MRT2GL.

## CONCLUSIONES Y RECOMENDACIONES

1. De los resultados gráficos obtenidos en la simulación del sistema de Control Optimo con Observador de Estados, se puede concluir que la aplicación de esta estrategia de Control es de mejor resultado que comparada al Control Adaptivo por Modelo de Referencia Multivariable [8] y muy parecido a la estrategia de Control por Modo Deslizante [9] , ya que los tiempos de establecimiento o asentamiento son menores a los dos segundos, el sobrepico o sobreimpulso para el movimiento del carro es nulo, así como el sobrepico del movimiento del brazo es muy pequeño (prácticamente despreciable), cumpliéndose con las especificaciones de funcionamiento requeridas.
2. Por otro lado la señal de control  $u_1$  alcanza un valor pico transitorio de aproximadamente 0.17 voltios, y manteniéndose alrededor de  $\pm 0.15$  voltios en estado estacionario. La señal de control  $u_2$  alcanza un valor pico transitorio de aproximadamente 1.05 voltios, para luego tender alrededor de cero en tiempo estacionario. Si estos valores de tensión están saliendo hacia los amplificadores de potencia para cada servomotor, suponiendo una ganancia de 18, entonces ¿estoy? hablando de una tensión máxima de aproximadamente 19 voltios en los bornes del servomotor.
3. En el programa de Control Optimo Proporcional Integral Discreto con Observador de Estados, se ha previsto incluir una limitación de las señales de control  $u_1$  y  $u_2$  que salen de la posible tarjeta de adquisición de datos, de tal forma de no sobredimensionar ni sobrepasar los límites requeridos de salida analógica de las salidas DAC0 y DAC1 (que pueden soportar niveles TTL).

4. Finalmente, se sugiere utilizar la estrategia de Control Adaptivo Autosintonizado, el cual debería proveer en línea la estimación de los parámetros y de los estados y hacer el ajuste automático de los parámetros del controlador a fin de obtener resultados de alto rendimiento en tiempo real. Esta técnica es bastante útil su uso en procesos o plantas en las que el modelado del sistema es complicado y/o cuando hay incertidumbres en el proceso no contempladas.

## ANEXO A

### RUTINAS DE PROGRAMACION EN MATLAB

A continuación se presentan los programas en código Matlab correspondientes a la respuesta del MRT2GL en lazo abierto, tanto en tiempo continuo como en discreto, así como el programa de Control Optimo Proporcional Integral Discreto con Observador Optimo.

#### **1. Programa de Respuesta en Tiempo Continuo del MRT2GL a una Entrada Escalón**

```
% ARCHIVO : mr2g_c.m
% PROCESO : MANIPULADOR ROBOTICO TRASLACIONAL DE DOS
GRADOS DE LIBERTAD
% SIMULACION DEL MODELO DEL PROCESO EN TIEMPO CONTINUO
% INFORME DE SUFICIENCIA - 5TO. PROGRAMA DE TITULACION
% VICTOR OJEDA
% UNI, 28/02/2005

clear all; % close all clc

%PARAMETROS RESULTANTES DEL MODELO DE LAGRANGE

m11=0.161172355; m12=0.00068426; m21=0.014255409; m22=0.219675182;
p11=2.942715601; p22=0.448010511; p12=-0.00068426;
d21=-0.139845564;
dx=(m12*m21)/(m11*m22);
dy=1-dx;
```

```
%MATRICES EN CONTINUO
```

```
%*****
```

```
A=[0 0 1 0;
```

```
0 0 0 1;
```

```
0 (m12*d21)/(m22*m11*dy) -p11/(m22*dy) -p22*m12/(m22*m11*dy);
```

```
0 -d21/(m22*dy) (m21*p11)/(m11*m22*dy) -p22/(m22*dy)];
```

```
B=[0 0;0 0;1/(m11*dy) -m12/(m11*m22*dy);-m21/(m11*m22*dy) 1/(m22*dy)];
```

```
Cc=[1 0 0 0;0 1 0 0];
```

```
Dc=[0 0;0 0];
```

```
% AL EJECUTARSE, SE OBTIENE LA SIGUIENTE MATRIZ B:
```

```
%B =
```

```
%
```

```
% 0 0
```

```
% 0 0
```

```
% 6.2062 -0.0193
```

```
% -0.4027 4.5534
```

```
% LO QUE INDICA ACOPLAMIENTO.
```

```
% CALCULO DE LA CONTROLABILIDAD EN TIEMPO CONTINUO
```

```
M=[B A*B A^2*B A^3*B];
```

```
rangoM=rank(M); %RANGO DEL SISTEMA = 4
```

```
% CALCULO DE LA OBSERVABILIDAD EN TIEMPO CONTINUO
```

```
N=[Cc' A'*Cc' A'^2*Cc' A'^3*Cc'];
```

```
rangoN=rank(N);
```

```
% RANGO DEL SISTEMA = 4
```

```
% DETERMINACION DE LA MATRIZ DE TRANSFERENCIA DEL PROCESO:
```

```
[num1,den1]=ss2tf(A,B,Cc,Dc,1);
```

```
[num2,den2]=ss2tf(A,B,Cc,Dc,2);
```

```
%PLOTEO DEL RESPUESTA DEL SISTEMA ACOPLADO
```

```
num11=[0 0.0000 6.2062 12.6641 -3.9509];
num21=[0 -0.0000 -0.4027 1.9588 0];
num12=[0 -0.0000 -0.0193 -0.0789 -0.0000];
num22=[0 -0.0000 4.5534 60.9905 0];
den=[1.0000 15.4394 26.7082 -8.5293 0];

t=0:0.0025:10;
figure(1)
[y11,x3,t]=step(num11,den,t);
[y21,x3,t]=step(num21,den,t);
plot(t,y11,'-',t,y21,'--'); grid;
title('Respuesta del M. R. 2 G. L. debida a u1')
ylabel('Posicion del carro: x1 (radianes) y del brazo: x2 (radianes)')
xlabel('Tiempo : segundos')
text(3.2,3,'y11 = x1 : POSICION DEL CARRO DEBIDO A U1')
text(3.2,0.5,'y21 = x2 : POSICION DEL BRAZO DEBIDO A U1')
figure(2)
[y12,x4,t]=step(num12,den,t);
[y22,x4,t]=step(num22,den,t);
plot(t,y12,'-',t,y22,'-'); grid;
title('Respuesta del M. R. 2 G. L. debida a u2')
ylabel('Posicion del carro: x1 (radianes) y del brazo: x2 (radianes)')
xlabel('Tiempo : segundos')
text(3.2,4.2,'y12 = x1 : POSICION DEL CARRO DEBIDO A U2')
text(3.2,35,'y22 = x2 : POSICION DEL BRAZO DEBIDO A U2')
```

## 2. Programa de Respuesta en Tiempo Discreto del MRT2GL a una Entrada Escalón

```
% ARCHIVO : mr2g_d.m
```

```
% PROCESO : MANIPULADOR ROBOTICO TRASLACIONAL DE DOS
GRADOS DE LIBERTAD
```

```
% SIMULACION DEL MODELO DEL PROCESO EN TIEMPO DISCRETO
```

```
% INFORME DE SUFICIENCIA - 5TO. PROGRAMA DE TITULACION
```

```
% VICTOR OJEDA
```

```
% UNI, 28/02/2005
```

```
clear all; % close all clc
```

```
%PARAMETROS RESULTANTES DEL MODELO DE LAGRANGE
```

```
m11=0.161172355; m12=0.00068426; m21=0.014255409; m22=0.219675182;
```

```
p11=2.942715601; p22=0.448010511; p12=-0.00068426;
```

```
d21=-0.139845564;
```

```
dx=(m12*m21)/(m11*m22);
```

```
dy=1-dx;
```

```
%MATRICES EN CONTINUO
```

```
%*****
```

```
A=[0 0 1 0;
```

```
0 0 0 1;
```

```
0 (m12*d21)/(m22*m11*dy) -p11/(m22*dy) -p22*m12/(m22*m11*dy);
```

```
0 -d21/(m22*dy) (m21*p11)/(m11*m22*dy) -p22/(m22*dy)];
```

```
B=[0 0;0 0;1/(m11*dy) -m12/(m11*m22*dy);-m21/(m11*m22*dy) 1/(m22*dy)];
```

```
Cc=[1 0 0 0;0 1 0 0];
```

```
Dc=[0 0;0 0];
```

```
% AL EJECUTARSE, SE OBTIENE LA SIGUIENTE MATRIZ B:
```

```
%B =
```

```
%
```

```
% 0 0
```

```
% 0 0
```

```
% 6.2062 -0.0193
```

% -0.4027 4.5534

% LO QUE INDICA ACOPLAMIENTO.

%\*\*\*\*\*

%Discretizacion

%\*\*\*\*\*

T=0.005;

%T=0.02;

[G,H,Cd,Dd]=c2dm(A,B,Cc,Dc,T,'zoh');

Cd1=[Cd(1,1) Cd(1,2) Cd(1,3) Cd(1,4)];

Cd2=[Cd(2,1) Cd(2,2) Cd(2,3) Cd(2,4)];

H1=[H(1,1); H(2,1); H(3,1); H(4,1)];

H2=[H(1,2); H(2,2); H(3,2); H(4,2)];

% CONDICIONES INICIALES

x1=0; x2=0; x3=0; x4=0;

x = [x1;x2;x3;x4];

% LAZO DE CONTROL

N = 2000; % TIEMPO EN SEGUNDOS: T\*N

for k=1:N

% LEY DE CONTROL

u1=1;

u2=1;

u=[u1;u2]; % REPRESENTACION VECTORIAL

% PROCESO LINEAL DISCRETIZADO USANDO T=0.01 SEG.

%x=G\*x+H\*u;



```

x1=G(1,1)*x1+G(1,2)*x2+G(1,3)*x3+G(1,4)*x4+H(1,1)*u1+H(1,2)*u2;
x2=G(2,1)*x1+G(2,2)*x2+G(2,3)*x3+G(2,4)*x4+H(2,1)*u1+H(2,2)*u2;
x3=G(3,1)*x1+G(3,2)*x2+G(3,3)*x3+G(3,4)*x4+H(3,1)*u1+H(3,2)*u2;
x4=G(4,1)*x1+G(4,2)*x2+G(4,3)*x3+G(4,4)*x4+H(4,1)*u1+H(4,2)*u2;

```

```
% ASIGNACION DE VARIABLES PARA PLOTEO
```

```

y1(k)=x1; y2(k)=x2;
U1(k) =u1; U2(k)=u2;

```

```
end
```

```
% PLOTEO
```

```

t = linspace(0,N*T,N);
figure(1)
subplot(2,1,1)
plot(t,y1,'-'); grid
title('Respuesta del M. R. 2 G. L. debida a u1')
ylabel('Posicion del carro: x1 (radianes)')
xlabel('Tiempo : segundos')
subplot(2,1,2)
plot(t,U1), grid
ylabel('Control u1 (voltios)')
xlabel('Tiempo : segundos')

```

```

figure(2)
subplot(2,1,1)
plot(t,y2,'-'); grid
title('Respuesta del M. R. 2 G. L. debida a u2')
ylabel('Posicion del brazo: x2 (radianes)')
xlabel('Tiempo : segundos')
subplot(2,1,2)
plot(t,U2), grid

```

```
ylabel('Control u2 (voltios)')
```

```
xlabel('Tiempo : segundos')
```

### 3. Programa de Control Optimo Proporcional Integral Discreto

```
% ARCHIVO : mr2d_opt.m
```

```
% PROCESO : MANIPULADOR ROBOTICO TRASLACIONAL DE DOS  
GRADOS DE LIBERTAD
```

```
% SIMULACION CON CONTROL OPTIMO PROPORCIONAL INTEGRAL  
DISCRETO
```

```
% INFORME DE SUFICIENCIA - 5TO. PROGRAMA DE TITULACION
```

```
% VICTOR OJEDA
```

```
% UNI, 02/03/2005
```

```
clear all; % close all clc
```

```
%PARAMETROS RESULTANTES DEL MODELO DE LAGRANGE
```

```
m11=0.161172355; m12=0.00068426; m21=0.014255409; m22=0.219675182;
```

```
p11=2.942715601; p22=0.448010511; p12=-0.00068426;
```

```
d21=-0.139845564;
```

```
dx=(m12*m21)/(m11*m22);
```

```
dy=1-dx;
```

```
%MATRICES EN TIEMPO CONTINUO
```

```
%*****
```

```
A=[0 0 1 0;
```

```
0 0 0 1;
```

```
0 (m12*d21)/(m22*m11*dy) -p11/(m22*dy) -p22*m12/(m22*m11*dy);
```

```
0 -d21/(m22*dy) (m21*p11)/(m11*m22*dy) -p22/(m22*dy)];
```

```
B=[0 0;0 0;1/(m11*dy) -m12/(m11*m22*dy);-m21/(m11*m22*dy) 1/(m22*dy)];
```

```
Cc=[1 0 0 0;0 1 0 0];
```

```
Dc=[0 0;0 0];
```

% AL EJECUTARSE, SE OBTIENE LA SIGUIENTE MATRIZ B:

%B =

%

% 0 0

% 0 0

% 6.2062 -0.0193

% -0.4027 4.5534

% LO QUE INDICA ACOPLAMIENTO.

% DETERMINACION DE LA MATRIZ DE TRANSFERENCIA DEL PROCESO:

[num1,den1]=ss2tf(A,B,Cc,Dc,1);

[num2,den2]=ss2tf(A,B,Cc,Dc,2);

% CALCULO DE LA CONTROLABILIDAD EN TIEMPO CONTINUO

M=[B A\*B A^2\*B A^3\*B];

rangoM=rank(M); %RANGO DEL SISTEMA = 4

% CALCULO DE LA OBSERVABILIDAD EN TIEMPO CONTINUO

N=[Cc' A'\*Cc' A'^2\*Cc' A'^3\*Cc'];

rangoN=rank(N);

% RANGO DEL SISTEMA = 4

%\*\*\*\*\*

%Discretizacion

%\*\*\*\*\*

%T=0.01;

%T=0.005;

T=0.02; % ADECUADO PARA TRAYECTORIAS Y REFERENCIAS

CONSTANTES

[G,H,Cd,Dd]=c2dm(A,B,Cc,Dc,T,'zoh');

% AL CORRER EL PROGRAMA PARA T=0.005 SEG. SE OBTIENEN LAS SIGUIENTES MATRICES:

%G =

%

% 1.0000 -0.0000 0.0048 -0.0000

% 0 1.0000 0.0000 0.0050

% 0 -0.0000 0.9352 -0.0000

% 0 0.0032 0.0057 0.9899

%

%

%H =

%

% 0.0001 -0.0000

% -0.0000 0.0001

% 0.0300 -0.0001

% -0.0019 0.0227

%

%Cd =

%

% 1 0 0 0

% 0 1 0 0

%

%Dd =

%

% 0 0

% 0 0

% DISEÑO DEL SISTEMA DE CONTROL OPTIMO PROPORCIONAL INTEGRAL

% MATRICES AMPLIADAS DEBIDO A LA INSERCIÓN DEL INTEGRADOR EN EL CONTROLADOR

```
Ga = [G zeros(4,2)
      -Cd*G eye(2,2)];
Ha = [H;-Cd*H];
```

```
% CALCULO DE LA CONTROLABILIDAD EN TIEMPO DISCRETO DEL
SISTEMA AMPLIADO
```

```
Md=[Ha Ga*Ha Ga^2*Ha Ga^3*Ha Ga^4*Ha Ga^5*Ha];
rangoMd=rank(Md); %RANGO DEL SISTEMA AMPLIADO= 6
```

```
% CALCULO DE LA OBSERVABILIDAD EN TIEMPO DISCRETO DEL
SISTEMA AMPLIADO
```

```
Nd=[Cd' G'*Cd' (G')^2*Cd' (G')^3*Cd'];
rangoNd=rank(Nd); %RANGO DEL SISTEMA AMPLIADO= 4
```

```
% MATRICES DE PONDERACION
```

```
R=200*eye(2);
Qa=10*eye(6);
P=zeros(6,6);
for i=1:1000
P=Qa+Ga'*P*Ga-Ga'*P*Ha*inv(R+Ha'*P*Ha)*Ha'*P*Ga;
end
```

```
K1=inv(R+Ha'*P*Ha)*Ha'*P*Ga;
[KK,PP]=dlqr(Ga,Ha,Qa,R); % K1=KK, P=PP
K=[KK(1,1) KK(1,2) KK(1,3) KK(1,4)
   KK(2,1) KK(2,2) KK(2,3) KK(2,4)];
KI = - [KK(1,5) KK(1,6)
        KK(2,5) KK(2,6)];
```

```
Kc1=[KK(1,1) KK(1,2) KK(1,3) KK(1,4)];
Kc2=[KK(2,1) KK(2,2) KK(2,3) KK(2,4)];
KI1 = - [KK(1,5)];
KI2 = - [KK(2,6)];
```

```
% CALCULO DE LA GANANCIA DEL OBSERVADOR DE ESTADOS
```

```
% MATRICES DE PONDERACION PARA EL OBSERVADOR
```

```
Qe=100*eye(4); Re=10*eye(2);
```

```
Pe = zeros(4,4);
```

```
for i=1:200
```

```
Pe = Qe + G*Pe*G' -G*Pe*Cd'*inv(Re+Cd*Pe*Cd')*Cd*Pe*G';
```

```
end
```

```
KeT = inv(Re+Cd*Pe*Cd')*Cd*Pe*G'; Ke=KeT';
```

```
[KKeT,PPe] = dlqr(G',Cd',Qe,Re); KKe=KKeT';% Pe=PPe, Ke=KKe
```

```
% SIMULACION DEL SISTEMA DE CONTROL OPTIMO MULTIVARIABLE
```

```
% CONDICIONES INICIALES
```

```
x1=0; x2=0; x3=0; x4=0;
```

```
xe1=0; xe2=0; xe3=0; xe4=0;
```

```
x = [x1;x2;x3;x4];
```

```
xe = [xe1;xe2;xe3;xe4];
```

```
v1=0; v2=0;
```

```
v=[v1;v2];
```

```
% LAZO DE CONTROL
```

```
N = 4000; % TIEMPO EN SEGUNDOS: T*N
```

```
for k=1:N
```

```
% LEY DE CONTROL
```

```
u1=-Kc1*xe+KI1*v1;
```

```
u2=-Kc2*xe+KI2*v2;
```

```
u=[u1;u2]; % REPRESENTACION VECTORIAL
```

```
% RESTRICCIONES PARA NO SATURAR A LOS AMPLIFICADORES
```

```
if(u1 > 1.4), u1 = 1.4;
```

```
elseif(u1 < -1.4), u1 = -1.4;
```

```

end

if(u2 > 1.4), u2 = 1.4;
elseif(u2 < -1.4), u2 = -1.4;
end

% ESTADOS ESTIMADOS
xe = G*xe + H*u + Ke*([x1;x2] - Cd*xe);

% PROCESO LINEAL DISCRETIZADO
x1=G(1,1)*x1+G(1,2)*x2+G(1,3)*x3+G(1,4)*x4+H(1,1)*u1+H(1,2)*u2;
x2=G(2,1)*x1+G(2,2)*x2+G(2,3)*x3+G(2,4)*x4+H(2,1)*u1+H(2,2)*u2;
x3=G(3,1)*x1+G(3,2)*x2+G(3,3)*x3+G(3,4)*x4+H(3,1)*u1+H(3,2)*u2;
x4=G(4,1)*x1+G(4,2)*x2+G(4,3)*x3+G(4,4)*x4+H(4,1)*u1+H(4,2)*u2;

% SETPOINT
r1(k)=sin(0.0014*k);           % TRAYECTORIA 1
r2(k)=cos(0.0014*k)+0.00014*k; % TRAYECTORIA 2
%r1(k)=1; r2(k)=1.5;         % REFERENCIAS 1 Y 2 CONSTANTES

% ERROR DE SEGUIMIENTO
e1=r1(k)-x1; e2=r2(k)-x2;

% SALIDA DEL INTEGRADOR
v1 = v1 + r1(k) - x1;
v2 = v2 + r2(k) - x2;
v=[v1;v2]; % REPRESENTACION VECTORIAL

% ASIGNACION DE VARIABLES PARA PLOTEO
y1(k)=x1; y2(k)=x2;
U1(k) =u1; U2(k)=u2;

end

```

```

% PLOTEO

t = linspace(0,N*T,N);
figure(1)
subplot(2,1,1)
plot(t,r1,'-',t,y1,'--'); grid
title('Control Optimo Proporcional Integral Discreto para el M. R. 2 G. L.')
ylabel('Posicion del carro: x1 (radianes)')
%xlabel('Tiempo : segundos')
%text(1.2,0.4,'- : r1(t)')
%text(1.2,0.1,'- - : x1(t)')
subplot(2,1,2)
plot(t,U1), grid
ylabel('Control u1 (voltios)')
xlabel('Tiempo : segundos')
%print -f -deps mr2d_opt1

figure(2)
subplot(2,1,1)
plot(t,r2,'-',t,y2,'--'); grid
title('Control Optimo Proporcional Integral Discreto para el M. R. 2 G. L.')
ylabel('Posicion del brazo: x2 (radianes)')
%xlabel('Tiempo : segundos')
%text(2.2,0.6,'- : r2(t)')
%text(2.2,0.2,'- - : x2(t)')
subplot(2,1,2)
plot(t,U2), grid
ylabel('Control u2 (voltios)')
xlabel('Tiempo : segundos')
%print -f -deps mr2d_opt2

```



## ANEXO B

### RUTINAS DE PROGRAMACION EN C

A continuación se presenta el programa de Control Optimo Discreto Proporcional Integral con Observador Optimo.

```
/*Archivo : mr2d_opt.m.cpp
PROGRAMA DEL SISTEMA DE CONTROL OPTIMO PROPORCIONAL INTEGRAL
DISCRETO DEL MRT2GL
INFORME DE SUFICIENCIA - 5TO. PROGRAMA DE TITULACION
VICTOR OJEDA
UNI, 23 DE MARZO DEL 2005 */

#include "r2_llib.h"
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <assert.h>
#include <alloc.h>
#include <conio.h>
#include <time.h>

#define IZQ 0
#define DER 1
#define MOTOR1 IZQ
#define MOTOR2 DER
#define A_RADIANES1 2.*M_PI/100.
#define A_RADIANES2 2.*M_PI/100.
#define Nm1 1/18.5 //servosystem serie: 5911 (base)
#define Nm2 1/18.5 //servosystem serie: 5807 (brazo)
```

```

/* Declaracion de las funciones propias del programa */

void Inicializa Sistema();
void Mide_Salidas(float *y1, float *vell, float *y2, float *vel2);
float Aplica_Control1(float u1, float yvel1);
float Aplica_Control2(float u2, float yvel2);

/* Declaración de las variables globales */

float y1, ylant, y2, y2ant;
float T, t, tsimul, Fs, N;
float VcFC, VcFE, Wth, Ofst1, Ofst2;
float Ke[4][2]; Ka[2][6]; G[4][4]; H[4][2]; Cd[2][4]; // ganancia
del observ. y del controlador

/*****
** FUNCIONES
*****/

void Inicializa_Sistema()

VcFE= .2; VcFC= .15;
Wth = .1;
Ofst1= 2.16;
Ofst2= 2.16;
//T=0.005; Fs=200 Hz
Fs= 50.; T= 1./Fs;
tsimul= 20.;

/* Condiciones iniciales */
//ulp=0.; u2p=0.;
vlp=0.; v2p=0.; x1=0.; x2=0.;
xelp=0.; xe2p=0.; xe3p=0.; xe4p=0.;

/* Matriz Ganancia del Observador de Estados */
Ke[0][0]=0.9164; Ke[0][1]=0.;
Ke[1][0]=0.; Ke[1][1]=0.9164;

```

```

Ke[2][0]=0.0562; Ke[2][1]=0.;
Ke[3][0]=0.; Ke[3][1]=0.0500;

/* Matriz Ganancia del Controlador Proporcional Integral */
Ka[0][0]=15.0906; Ka[0][1]=0.; Ka[0][2]=0.7924; Ka[0][3]=0.;
Ka[0][4]=-0.6995; Ka[0][5]=0.;
Ka[1][0]=0.; Ka[1][1]=14.4815; Ka[1][2]=0.; Ka[1][3]=0.7530;
Ka[1][4]=0.; Ka[1][5]=-0.6756;

KI1=-Ka[0][4]; KI2=-Ka[1][5];

/* Matrices del proceso */
G[0][0]=1.; G[0][1]=0.; G[0][2]=0.0048; G[0][3]=0.;
G[1][0]=0.; G[1][1]=1.; G[1][2]=0.; G[1][3]=0.0050;
G[2][0]=0.; G[2][1]=0.; G[2][2]=0.9352; G[2][3]=0.;
G[3][1]=0.; G[3][2]=0.0032.; G[3][3]=0.0057.; G[3][4]=0.9899;

H[0][0]=0.0001; H[0][1]=0.;
H[1][0]=0.; H[1][1]=0.0001;
H[2][0]=0.03; H[2][1]=-0.0001;
H[3][1]=-0.0019; H[3][2]=0.0227;

Cd[0][0]=1.; Cd[0][1]=0.; Cd[0][2]=0.; Cd[0][3]=0.;
Cd[1][0]=0.; Cd[1][1]=1.; Cd[1][2]=0.; Cd[1][3]=0.;

ConfigurarHardware(Fs);
EnviarVoltaje1(Ofst1);
EnviarVoltaje2(Ofst2);
}

void Mide_Salidas(float *y1, float *vell, float *y2, float *vel2)
{
static float y1ant=0;
static float y2ant=0;
static unsigned contC1_ant=0;
static unsigned contC2_ant=0;
unsigned contC1_act, contC2_act;

```

```

//Lee Posicion del MOTOR1

contC1_act=LeerSensor(MOTOR1);
if(contC1_act<contC1_ant)
{
    if((contC1_ant-contC1_act)>32768)
        *y1=y1ant+(65536-contC1_ant+contC1_act)*A_RADIANES1*Nm1;
    else
        *y1=y1ant-(contC1_ant-contC1_act)*A_RADIANES1*Nm1;
}
else
{
    if((contC1_act-contC1_ant)>32768)
        *y1=y1ant-(65536-contC1_act+contC1_ant)*A_RADIANES1*Nm1;
    else
        *y1=y1ant+(contC1_act-contC1_ant)*A_RADIANES1*Nm1;
}
*vel1=(5.*(*y1-y1ant)+*vel1)/(1.+5.*T);
contC1_ant=contC1_act;
y1ant=*y1;

//Lee Posicion del MOTOR2

contC2_act=LeerSensor(MOTOR2);
if(contC2_act<contC2_ant)
{
    if((contC2_ant-contC2_act)>32768)
        *y2=y2ant+(65536-contC2_ant+contC2_act)*A_RADIANES2*Nm2;
    else
        *y2=y2ant-(contC2_ant-contC2_act)*A_RADIANES2*Nm2;
}
else
{
    if((contC2_act-contC2_ant)>32768)
        *y2=y2ant-(65536-contC2_act+contC2_ant)*A_RADIANES2*Nm2;
    else
        *y2=y2ant+(contC2_act-contC2_ant)*A_RADIANES2*Nm2;
}
*vel2=(5.*(*y2-y2ant)+*vel2)/(1.+5.*T);
contC2_ant=contC2_act;

```

```
y2ant=*y2;
}

float Aplica_Control1(float u1, float yvel1)
{
    float v1;

    if(fabs(yvel1)<=Wth)
    {
        if(u1<0.) v1 = -VcFE;
        if(u1>0.) v1 = VcFE;
    }
    else
    {
        if(u1<0.) v1 = -VcFC;
        if(u1>0.) v1 = VcFC;
    }

    if(u1+v1<-1.4) u1= -1.4;
    else if(u1+v1>1.4) u1= 1.4;

    EnviarVoltaje1(u1+v1+Ofst1);

    return u1;
}

float Aplica_Control2(float u2, float yvel2)
{
    float v2;

    if(fabs(yvel2)<=Wth)
    {
        if(u2<0.) v2 = -VcFE;
        if(u2>0.) v2 = VcFE;
    }
    else
    {
        if(u2<0.) v2 = -VcFC;
        if(u2>0.) v2 = VcFC;
    }
}
```

```

    if(u2+v2<-1.4)      u2= -1.4;
    else if(u2+v2>1.4) u2=  1.4;

    EnviarVoltaje2(u2+v2+Ofst2);

    return u2;
}

void main()
{
    int ant=0, act=0, k, n;
    FILE *out;
    float *yb1, *ub1, *velb1, vell1, y1, u1, ylant;
    float *yb2, *ub2, *velb2, vel2, y2, u2, y2ant;
    float r1, r2; x1, x2, x3, x4; xe1, xelp, xe2, xe2p, xe3, xe3p, xe4,
xe4p;
    float e1, e2; //ulp, u2p;
    float p33, pa, pb, q42, q43, q44, g3, g4;
    float f, w;

    Inicializa_Sistema();
    clrscr();
    assert(tsimul<100);
    N= tsimul/T;
    n= (int)N;
    t=0.;
    vell1=0.;
    vel2=0.;

    assert(NULL!=(ub1= (float *)calloc(n/5,sizeof(float))));
    assert(NULL!=(yb1= (float *)calloc(n/5,sizeof(float))));
    assert(NULL!=(velb1= (float *)calloc(n/5,sizeof(float))));
    assert(NULL!=(ub2= (float *)calloc(n/5,sizeof(float))));
    assert(NULL!=(yb2= (float *)calloc(n/5,sizeof(float))));
    assert(NULL!=(velb2= (float *)calloc(n/5,sizeof(float))));

    for(k=0; k<n;)
    {
        act=NivelClock();

```

```

if ((ant==0)&&(act==1))

Mide_Salidas(&y1, &vel1, &y2, &vel2);

        /* Definiendo las referencias constantes*/
r1=M_PI/2;
r2=M_PI/4;

        /* Definiendo trayectorias */
//r1=sin(t)+0.1*t;
//r2=cos(t);

        /* Error de Posici'on */
e1=r1-x1; e2=r2-x2;

        /* Observador: xe(k+1)=G*xe(k)+H*u(k)+Ke*([x1(k);
x2(k)]-Cd*xe(k)) */

        // Asignando las siguientes variables para comodidad
del cálculo

GX1=G[0][0]*xe1p+G[0][1]*xe2p+G[0][2]*xe3p+G[0][3]*xe4p;

GX2=G[1][0]*xe1p+G[1][1]*xe2p+G[1][2]*xe3p+G[1][3]*xe4p;

GX3=G[2][0]*xe1p+G[2][1]*xe2p+G[2][2]*xe3p+G[2][3]*xe4p;

GX4=G[3][0]*xe1p+G[3][1]*xe2p+G[3][2]*xe3p+G[3][3]*xe4p;

HU1=H[0][0]*u1+H[0][1]*u2;
HU2=H[1][0]*u1+H[1][1]*u2;
HU3=H[2][0]*u1+H[2][1]*u2;
HU4=H[3][0]*u1+H[3][1]*u2;

XC1=x1-
(Cd[0][0]*xe1p+Cd[0][1]*xe2p+Cd[0][2]*xe3p+Cd[0][3]*xe4p);
XC2=x2-
(Cd[1][0]*xe1p+Cd[1][1]*xe2p+Cd[1][2]*xe3p+Cd[1][3]*xe4p);

```

```

KCX1=Ke[0][0]*XC1+Ke[0][1]*XC2;
KCX2=Ke[1][0]*XC1+Ke[1][1]*XC2;
KCX3=Ke[2][0]*XC1+Ke[2][1]*XC2;
KCX4=Ke[3][0]*XC1+Ke[3][1]*XC2;

xe1=GX1+HU1+KCX1;
xe2=GX2+HU2+KCX2;
xe3=GX3+HU3+KCX3;
xe4=GX4+HU4+KCX4;

/* Leyes de Control u1 y u2 */
u1=-(Ka[0][0]*xe1+Ka[0][1]*xe2+Ka[0][2]*xe3-
Da[0][3]*xe4)+KI1*v1p;
u2=-(Ka[1][0]*xe1+Ka[1][1]*xe2+Ka[1][2]*xe3-
Da[1][3]*xe4)+KI2*v2p;

/* Ecuación de los integradores */
v1=v1p+r1-x1;
v2=v2p+r2-x2;

u1= Aplica_Control1(u1, vel1);
if(!(k%5))
{
    yb1[k/5]=y1*180./M_PI; ub1[k/5]=u1;
    velb1[k/5]=vel1;
}

u2=Aplica_Control2(u2, vel2);
if(!(k%5))
{
    yb2[k/5]=y2*180./M_PI; ub2[k/5]=u2;
    velb2[k/5]=vel2;
}

/* Actualizaci'on de Variables */
%u1p=u1; u2p=u2;
xelp=xel; xe2p=xe2; xe3p=xe3; xe4p=xe4;
v1p=v1; v2p=v2;

k++;

```



```

        t+=T;
        gotoxy(15,5);
printf("*****");
        gotoxy(18,7); printf("RESPUESTA DEL SISTEMA DE CONTROL
OPTIMO ROBUSTO");
        gotoxy(18,9); printf("DEL MANIPULADOR ROBOTICO TRASLACIONAL
DE DOS GRADOS DE LIBERTAD");
        gotoxy(15,11);
printf("*****");
        gotoxy(21,13);printf("tiempo           : t=%+5.3f",t);
        gotoxy(21,15);printf("se~nal de control 1 : u1=%+5.3f",u1);
        gotoxy(21,17);printf("posic.    angular    1
y1=%+5.3f",y1*180./M_PI);
        gotoxy(21,19);printf("referencia 1           : r1= %+5.3f",
r1*180./M_PI);
        gotoxy(21,21);printf("error 1           : e1= %+5.3f",
e1*180./M_PI);
        gotoxy(21,23);printf("veloc.    angular    1
vell=%+5.3f",vell);
        gotoxy(21,25);printf("se~nal de control 2 : u2=%+5.3f",u2);
        gotoxy(21,27);printf("posic.    angular    2
y2=%+5.3f",y2*180./M_PI);
        gotoxy(21,29);printf("referencia 2           : r2= %+5.3f",
r2*180./M_PI);
        gotoxy(21,31);printf("error 2           : e2= %+5.3f",
e2*180./M_PI);
        gotoxy(21,33);printf("veloc.    angular    2
vel2=%+5.3f",vel2);

gotoxy(15,35);printf("***** *****
");

        if(kbhit())    if(getch()==27) break;
    }
    ant=act;
}

EnviarVoltaje1(Ofst1);
EnviarVoltaje2(Ofst2);

```

```
out=fopen("mr2d_opt.m.dat","wt");
for (int i=0; i<n/5; i++)
    fprintf(out,"%10f %10f %10f %10f %10f %10f %10f\n",
        i*5*T,ub1[i],yb1[i],velb1[i],ub2[i],yb2[i], velb2[i]);
delete ub1, yb1, velb1, ub2, yb1, velb2;
fclose(out);
```

```
}
```

## BIBLIOGRAFIA

- [1] Ogata, Katushito, “Ingeniería de Control Moderna”, Segunda Edición, Prentice Hall Hispanoamericana, 1993.
- [2] Ogata, Katushito, “Sistemas de Control en Tiempo Discreto”, Segunda Edición, Prentice Hall Hispanoamericana, 1996.
- [3] E. Slotine and W. Li, “Applied Nonlinear Control”, Prentice Hall Inc, 1991.
- [4] Umez-Eronini, Eronini, “Dinámica de Sistemas y Control”,. International Thomson Editores, 2001.
- [5] Ogata, Katushito, “Problemas de Ingeniería de Control utilizando Matlab”, Prentice Hall Iberia, 1999.
- [6] Rojas-Moreno, Arturo, “Control Avanzado - Diseño y Aplicaciones en Tiempo Real”, Publicación Independiente, 2001.
- [7] N. Rao Sripada, D. Grant Fisher, “Improved Least Squares Identification”, Int. J. Control, Vol. 46, no. 6 , 1889{1913 (Dic 1987).
- [8] Castillo Casanova, Percy, “Diseño e Implementación de un Sistema de Control No Lineal Adaptivo Multivariable Aplicado a un Manipulador Robótico Traslacional de 2do.F”, Tesis de Maestría, Universidad Nacional de Ingeniería, Lima, Perú, 2004.
- [9] Rodríguez Bustinza, Ricardo, “Diseño e Implementación de un Sistema de Control No Lineal Multivariable por Modo Deslizante Aplicado a un Manipulador Robótico Traslacional de 2do.F”, Tesis de Maestría, Universidad Nacional de Ingeniería, Lima, Perú, 2003.
- [10] Benites Saravia, N. Raúl, “Diseño de un Sistema de Control de Posición Optimo Multivariable para un Manipulador Robótico Esférico”, Tesis de Maestría, Universidad Nacional de Ingeniería, Lima, Perú, 2002.