

**UNIVERSIDAD NACIONAL DE INGENIERÍA**  
**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**“DISEÑO Y SIMULACIÓN DE UN SISTEMA DE CONTROL DE  
ARRANQUE Y PARADA DE UN TREN EMPLEANDO CONTROL  
ÓPTIMO, ADAPTIVO Y PREDICTIVO”**

**INFORME DE SUFICIENCIA**

PARA OPTAR EL TÍTULO PROFESIONAL DE:

**INGENIERO ELECTRÓNICO**

**PRESENTADO POR:  
DANTE ROBERTO WATANABE HANGUI**

PROMOCIÓN  
1982-1  
LIMA-PERU  
2003

Dedicado a mis queridos padres:

Pedro Takeshi y Rosa Mitsuko,

con mi eterno agradecimiento

**DISEÑO Y SIMULACIÓN DE UN SISTEMA DE CONTROL DE  
ARRANQUE Y PARADA DE UN TREN EMPLEANDO  
CONTROL ÓPTIMO, ADAPTIVO Y PREDICTIVO**

## **SUMARIO**

El control de vehículos tripulados como es el caso de un tren, requiere de un sistema de control que asegure un arranque y parada suaves, que pueden lograrse con la aplicación de estrategias de control avanzado, tales como el control óptimo, adaptivo y predictivo; optimizando así el funcionamiento del tren y traduciéndose por lo tanto en ventajas competitivas frente a los sistemas convencionales.

Hemos utilizado el programa MATLAB para representar el sistema físico así como el modelado respectivo y para la simulación de los diversos métodos de control mencionados. Incluimos los programas fuente así como los gráficos obtenidos para ilustrar los resultados y explicar las comparaciones entre las diversas técnicas de control.

# ÍNDICE

<b>PRÓLOGO</b>	1
<b>CAPÍTULO I</b>	
<b>CONCEPTOS ACERCA DEL TRANSPORTE FERROVIARIO</b>	
1.1 El ferrocarril	2
1.2 Locomotoras eléctricas	4
1.3 Transporte ferroviario de alta velocidad	6
1.4 Especificaciones de una locomotora y un vagón	8
<b>CAPÍTULO II</b>	
<b>MODELADO DEL PROCESO Y SIMULACIÓN EN TIEMPO CONTINUO Y DISCRETO</b>	
2.1 Descripción del sistema	9
2.2 Modelado del proceso	10
2.3 Representación en el espacio de estado	12
2.4 Controlabilidad	13
2.5 Observabilidad	14
2.6 Simulación en tiempo continuo	15
2.7 Discretización del modelo y simulación en tiempo discreto	16
2.8 Modelamiento y simulación en SIMULINK	18

## **CAPÍTULO III**

### **SISTEMA DE CONTROL ÓPTIMO**

3.1	Diseño del controlador óptimo	25
3.2	Cálculo de la matriz de ganancia óptima	29
3.3	Diseño del observador óptimo cuadrático	30
3.4	Simulación del sistema de control óptimo	34

## **CAPÍTULO IV**

### **SISTEMA DE CONTROL ADAPTIVO**

4.1	Diseño del controlador adaptivo	36
4.2	Estimación de parámetros	37
4.3	Estimación de estados	40
4.4	Ley de control	42
4.5	Ley de control residual	45
4.6	Simulación del sistema de control adaptivo	45

## **CAPÍTULO V**

### **SISTEMA DE CONTROL PREDICTIVO**

5.1	Diseño del controlador predictivo	47
5.2	Determinación del modelo CARIMA del proceso	50
5.3	Cálculo de la respuesta libre del proceso	51
5.4	Cálculo de la respuesta del proceso al escalón	52
5.5	Determinación de la ley de control	53
5.6	Simulación del sistema de control predictivo	54

## **CAPÍTULO VI**

## **COMPARACIÓN DE RESULTADOS**

6.1 Tiempo de ejecución	57
6.2 Tiempo de ascenso	58
6.3 Tiempo de estabilización	58
6.4 Sobrepico	58
6.5 Seguimiento a la señal de referencia	59

<b>CONCLUSIONES</b>	60
---------------------	----

## **ANEXO A**

### **LISTADO DE LOS PROGRAMAS**

A1. Simulación del sistema en lazo abierto	63
A2. Modelo del tren en Simulink	66
A3. Gráfica de respuestas en Simulink	79
A4. Simulación del sistema con control óptimo	80
A5. Simulación del sistema con control adaptivo	83
A6. Simulación del sistema con control predictivo	88

## **ANEXO B**

<b>LISTADO DE FIGURAS</b>	92
---------------------------	----

<b>BIBLIOGRAFÍA</b>	94
---------------------	----

## PRÓLOGO

El presente informe de suficiencia tiene por finalidad mostrar la aplicación de varias técnicas de control automático, para controlar el arranque y parada de un tren y efectuar la comparación de los resultados. Los métodos de Control Avanzado que aplicamos son: control óptimo, control adaptivo y control predictivo.

El capítulo I proporciona una breve introducción a los sistemas de transporte ferroviario actuales. El capítulo II trata sobre el modelado del proceso que deseamos controlar, tanto en tiempo continuo como en tiempo discreto, y la simulación respectiva en SIMULINK. El capítulo III trata sobre el diseño de un controlador óptimo para el modelo del tren. En el capítulo IV se diseña un controlador adaptivo. El capítulo V abarca el diseño de un controlador predictivo. Finalmente, en el capítulo VI se realizan las comparaciones relativas a los resultados de las distintas estrategias de control desarrollados. En el anexo A están todos los programas en MATLAB empleados para simular el modelado y los controladores de los capítulos II al V.

El asesoramiento y orientación del profesor Raúl Benites Saravia han sido invalorable para el desarrollo y culminación satisfactoria del informe de suficiencia. Mi sincero agradecimiento por su apoyo.



# **CAPÍTULO I**

## **CONCEPTOS ACERCA DEL TRANSPORTE FERROVIARIO**

### **1.1 El ferrocarril**

El ferrocarril ha sido un medio de transporte para carga y pasajeros usado por mucho tiempo. Las primeras locomotoras estaban construidas con un caldero de vapor alimentado bien sea por madera o carbón. La presión del vapor era usada para accionar las ruedas. Estas fueron las locomotoras de vapor, las que fueron reemplazadas por las locomotoras diesel, eléctricas o de turbina a gas.

El motor diesel fue patentado por el Dr. Rudolf Diesel (1858-1913) en Alemania en 1892, quien logró hacer trabajar un motor en 1897. La primera locomotora diesel comercial fue construida por General Electric en 1918 para la Jay Street Connecting Railroad en Nueva York. Aunque no fue particularmente exitosa, en 1924 se construyó una maqueta de locomotora diesel .

Una locomotora típica es una planta generadora de energía eléctrica pequeña. Un motor diesel energiza un generador síncrono. La salida del generador es inmediatamente rectificadora a DC mediante diodos rectificadores. Este bus de DC provee de corriente continua a motores de tracción, los que poseen excitación en serie para producir un alto torque a bajas velocidades, ideal para la operación de carga. Cada uno de estos motores es capaz de regenerar cuando el tren está frenando, con la energía disipada por un banco de resistencia de frenos.

Las locomotoras diesel son las más empleadas comúnmente hoy debido a que la eficiencia térmica de las locomotoras diesel es de aproximadamente 30 % comparada con una eficiencia del 6 a 7 % de una locomotora de vapor. Adicionalmente los costos de mantenimiento son mucho menores en una diesel y tienen una disponibilidad del 90% comparada a un 30 a 40 % de una locomotora de vapor promedio.

El motor diesel es un motor de ignición compresión, a diferencia de un motor de petróleo (ó gasolina), el cual es un motor de ignición por explosión. Este motor usa una chispa eléctrica para encender el combustible en los cilindros del motor, mientras que el combustible en los motores diesel es encendido por el calor originado por el aire comprimido en el cilindro. En esta etapa el aire es comprimido a un 25<sup>avo</sup> de su volumen original. Esto se expresa como una proporción de compresión de 25 a 1. Una proporción de compresión de 16 a 1 dará una presión del aire de 35.5 bares e incrementará la temperatura a más de 427°C. El combustible inyectado al aire caliente se enciende y se expande. El gas en expansión fuerza al pistón a moverse hacia abajo, lo cual mueve al eje que se conecta al generador (CD) o al alternador (CA) donde se produce la electricidad. El generador o el alternador proveen de energía a los motores de tracción.

Las locomotoras diesel-eléctricas tienen un motor diesel que a su vez es el principal accionador de un generador síncrono. La mayoría de las locomotoras modernas tienen un generador síncrono entre 3000 a 5000 HP, con una salida rectificada con un diodo. El bus resultante de DC luego provee de energía a los motores de tracción DC. Una locomotora es una pequeña planta de generación sobre ruedas de aproximadamente 4MW.

La mayor parte del tráfico ferroviario en los Estados Unidos, a la fecha, consiste en trenes de carga, los cuales tienden a ser largos, pesados y lentos. Existe también el servicio de pasajeros en la forma de sistemas de trenes livianos cerca de algunas áreas urbanas y algunas líneas de larga trayectoria. Casi todas estas líneas de pasajeros están restringidas por un límite de velocidad, el cual es de 78 millas por hora, lo que equivale a 125.53 km/h. La única excepción es el tren denominado Corredor del Noreste, donde se permite la velocidad de 193.12 km/h en secciones de acceso limitado de la vía.

## **1.2 Locomotoras eléctricas**

Las locomotoras eléctricas reciben su energía de una central eléctrica. La energía es entregada a los trenes ya sea mediante un tercer riel a nivel de la vía o a través de cables elevados conocidos como catenarias o pantógrafos. Con algunas excepciones los sistemas de corriente alterna usan catenarias mientras que los sistemas de corriente directa usan un tercer riel. El rango de es de 500 a 25,000 voltios. Se utiliza un tercer riel hasta los 1,500 voltios. A mayores voltajes, se necesita mas separación por seguridad y se reduce la corriente de tal manera que el conductor sea más liviano, y se prefiera por lo tanto un alambre conductor vía un pantógrafo o catenaria. Por lo general, las locomotoras de bajo voltaje usan corriente directa, mientras que las de alto voltaje usan corriente alterna. Esto se debe a la mayor eficiencia, es decir menores pérdidas de la corriente alterna durante la transmisión a grandes distancias.

Los motores eléctricos tradicionales trabajan de dos maneras: los motores son alimentados directamente con el voltaje completo, o bien se cambia a un voltaje mas bajo por medio de un transformador en la locomotora, luego puede ser transmitida

directamente a los motores de los ejes de tracción o rectificadas si se usan motores de corriente directa.

Las primeras locomotoras eléctricas tenían uno o dos motores grandes en medio del chasis, que movían las ruedas mecánicamente, en forma similar a la transmisión de las locomotoras de vapor. Desde 1940, cada eje tiene su propio motor y cada locomotora tiene dos o tres juegos de ejes con sus respectivos motores.

Las locomotoras de corriente alterna (CA) transforman la corriente a un voltaje mas bajo con un transformador, o en algunos diseños mas modernos, mediante controladores de voltaje. Muchas locomotoras modernas de CA usan motores de tracción de corriente directa o en su defecto motores de inducción asíncronos. Los modelos mas recientes tienen un circuito de corriente directa, el cual es alimentado ya sea por la corriente directa del cable elevado o por la corriente alterna transformada y rectificada. La corriente de este circuito es luego convertida electrónicamente a una corriente trifásica de frecuencia y voltaje variables, con lo cual se alimentan los motores, cuya construcción a la vez es muy simple. Esta nueva tecnología permite que los motores alimenten de energía eléctrica de vuelta a los cables elevados mientras frenan.

Las locomotoras de corriente directa (CD) tradicionalmente controlaban la corriente y el voltaje cambiando las conexiones en los motores de serie a paralelo y añadiendo resistencia para el arranque. Los últimos diseños usan controladores de voltaje de estado sólido para controlar la corriente, algunos usan motores de corriente alterna (inductivos/asíncronos).

La mayoría de los trenes más rápidos del mundo tienen tracción eléctrica. Es oneroso construir transformadores y cables elevados sobre las líneas férreas, pero los

motores son mucho más poderosos y menos pesados que los motores diesel porque no tienen que cargar su propio combustible. Por ejemplo un motor moderno de 80 toneladas y de cuatro ejes pueden tener fácilmente una potencia de más de 6 MW. Las locomotoras eléctricas no necesitan de una transmisión mecánica porque los motores pueden desarrollar una alta potencia a cualquier velocidad. Esto es particularmente cierto para la nueva tecnología de corriente trifásica. Las locomotoras eléctricas son limpias en cuanto a que no contaminan el ambiente, aunque las plantas generadoras pueden depender en como generan la electricidad.

La transmisión eléctrica de la fuente de energía es hecha mejor a un alto voltaje ya que esto reduce las pérdidas debido a la resistencia y puede lograrse con cables mas livianos ( desde que se reduce la corriente). La corriente alterna es el método preferido ya que puede ser convertida fácilmente a corriente directa y puede viajar más lejos a lo largo del cable antes que necesite ser transformada.

### **1.3 Transporte ferroviario de alta velocidad**

El diseño tradicional de los motores diesel eléctricos tiene varias desventajas para aplicaciones ferroviarias de alta velocidad. La principal entre ellas está el tamaño y el peso del motor diesel, el cual limita efectivamente las máximas velocidades alcanzables por una locomotora diesel.

El transporte ferroviario de alta velocidad está teniendo mayor atención como una posible alternativa a los viajes aéreos para viajes de negocios entre ciudades que están separadas entre 250 a 1000 km. Los viajes en tren se incrementarán a medida que se desarrollen trenes de alta velocidad, haciendo que el tiempo de viaje en tren desde el centro de una ciudad hasta el centro de otra, sea menor o equivalente al tiempo de viaje por avión. El principal objetivo es el de proveer transporte entre

ciudades que están lo suficientemente alejadas para desalentar el viaje en automóvil y demasiado cercanas para un viaje aéreo eficiente, generalmente en el rango entre 250 a 1000 km. Este servicio de trenes evitaría la congestión y retardos de los vuelos en los aeropuertos así como también disminuiría el tráfico terrestre en los aeropuertos.

Los sistemas ferroviarios de alta velocidad eficientes y rápidos ya existen en Europa y en Japón. La red ferroviaria del Japón posee un tren llamado “tren bala” de alta velocidad que cubre 2,229 km. de recorrido en siete líneas. Este tren recorre 515 km en tres horas y diez minutos, con un promedio de velocidad de 162.5 km/h, pudiendo llegar hasta 350 km/h, alcanzando los 552 km/h en pruebas de laboratorio con trenes que levitan y corren sobre un colchón magnético sin ruedas. En Europa tienen el TGV (Train a grande vitesse) en Francia perteneciente a la SNCF (Societe nationale des chemins de fer francais) y el Transrapid de Alemania donde los trenes exceden los 240 km/h. Esta velocidad aumentará a medida que se usen nuevos sistemas, tales como los de levitación magnética para hacer flotar vehículos sobre una vía de aluminio. La infraestructura europea de rieles está diseñada en locomotoras completamente eléctricas, donde la energía eléctrica para trabajar los motores de tracción está provista a través de un pantógrafo y una catenaria suministrada por una red de distribución de energía eléctrica. Estos tipos de locomotoras se denominan típicamente como locomotoras completamente eléctricas. Sin embargo, la catenaria y la infraestructura de interfase a la empresa de electricidad requerida por este tipo de locomotoras no es económicamente factible en cualquier infraestructura ferroviaria.

Por lo tanto, se están haciendo estudios en locomotoras de alta velocidad con accionadores primarios a bordo que llevan generadores síncronos. Estas locomotoras se denominan locomotoras no-eléctricas aunque usen un sistema de energía eléctrica autónomo a bordo pequeño.

#### **1.4 Especificaciones de una locomotora y un vagón**

La locomotora diesel marca General Motors modelo JT42CWR tiene las siguientes especificaciones técnicas principales:

Potencia:	3000 HP	
Fuerza de tracción:	409 kN	(para el arranque)
	260 kN	(para marcha continua a 25.6km/h)
Velocidad máxima:	121 km/h	
Peso:	126 toneladas	

Un vagón de carga se define por su capacidad volumétrica, capacidad de tracción, capacidad de carga, longitud interna y altura interna. La capacidad de tracción puede ser de 70 toneladas o de 100 toneladas. El primero tiene 100,000 kg de carga bruta, mientras que el segundo tiene 126,000 kg. La capacidad de carga es la diferencia entre la carga bruta y el peso vacío de un vagón. Para nuestro caso consideraremos un vagón de 100 toneladas con carga completa, es decir con una masa total de 126,000 kg.

## CAPÍTULO II

### MODELADO DEL PROCESO Y SIMULACIÓN EN TIEMPO CONTÍNUO Y DISCRETO

#### 2.1 Descripción del sistema

El Sistema que deseamos controlar es un tren compuesto por una locomotora y un vagón. La locomotora y el vagón están unidos por un resorte con coeficiente de rigidez  $k$  y coeficiente de amortiguamiento o coeficiente de fricción viscosa  $B_R$ .  $B_1$  representa el coeficiente de fricción de la locomotora con el aire y de sus ruedas con los rieles y  $B_2$  representa el coeficiente de fricción del vagón con el aire y de sus ruedas con los rieles. El desplazamiento de la locomotora se representa con  $x_1$  y el desplazamiento del vagón se indica con  $x_2$ . La figura 2.1 muestra una representación simplificada del tren.

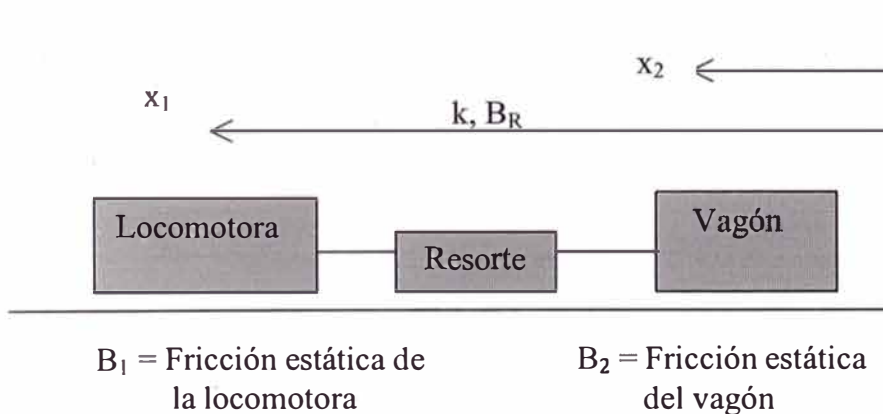


Fig. 2.1 Representación del tren



El problema que deseamos resolver es controlar la velocidad del tren de tal modo que en el arranque alcance rápidamente una velocidad constante con un tiempo de estabilización menor a 30 segundos y un sobreimpulso mínimo de tal manera que presente a los pasajeros un viaje confortable y suave, sin cambios bruscos en la aceleración del tren. Del mismo modo deseamos asegurar una detención rápida y segura. Asumiremos que el tren viaja en una sola dirección y en línea recta. Aplicaremos diversas técnicas de control avanzado al tren de modo que tenga un arranque y una parada suaves.

## 2.2 Modelado del proceso

El modelo matemático del movimiento de traslación del tren se efectuará considerando los elementos mecánicos siguientes: la masa de la locomotora, la masa del vagón y la constante del resorte. Las variables que se utilizan para describir este movimiento de traslación son: la aceleración, la velocidad y el desplazamiento. La ecuación que gobierna el movimiento de los sistemas mecánicos y que aplicaremos es la segunda ley de Newton. La masa de la locomotora y del vagón están representadas por  $M_1$  y  $M_2$ , respectivamente.  $F$  representa la fuerza aplicada por la locomotora. La figura 2.2 muestra la representación del sistema en un diagrama de cuerpo libre.

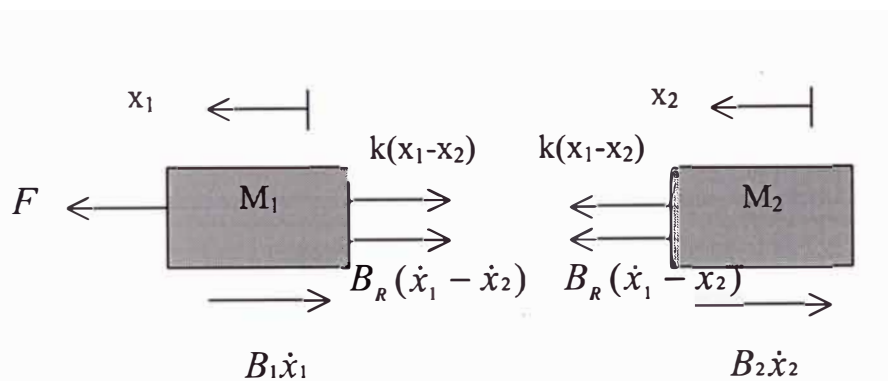


Fig. 2.2 Diagrama de cuerpo libre del tren

La segunda ley de Newton establece que la suma algebraica de las fuerzas que actúan sobre un cuerpo rígido en una dirección dada es igual al producto de la masa del cuerpo por su aceleración en la misma dirección. En este caso, las fuerzas que actúan horizontalmente sobre la locomotora son el resorte y la fricción. En la dirección vertical, la fuerza gravitacional es cancelada por la fuerza normal aplicada por la tierra de tal modo que no habrá aceleración en esta dirección.

Las ecuaciones diferenciales para el movimiento en la dirección horizontal de los dos cuerpos son las siguientes:

$$M_1\ddot{x}_1 = F - k(x_1 - x_2) - B_R(\dot{x}_1 - \dot{x}_2) - B_1\dot{x}_1 \quad (2.1)$$

$$M_2\ddot{x}_2 = k(x_1 - x_2) + B_R(\dot{x}_1 - \dot{x}_2) - B_2\dot{x}_2 \quad (2.2)$$

Este conjunto de ecuaciones diferenciales se pueden replantear de la siguiente manera:

$$\ddot{x}_1 = \frac{F}{M_1} - \frac{k}{M_1}x_1 + \frac{k}{M_1}x_2 - \frac{B_1 + B_R}{M_1}\dot{x}_1 + \frac{B_R}{M_1}\dot{x}_2 \quad (2.3)$$

$$\ddot{x}_2 = \frac{k}{M_2}x_1 - \frac{k}{M_2}x_2 + \frac{B_R}{M_2}\dot{x}_1 - \frac{B_2 + B_R}{M_2}\dot{x}_2 \quad (2.4)$$

Las variables de estado elegidas son:

$x_1$  : Posición de la locomotora

$v_1$ : Velocidad de la locomotora

$x_2$  : Posición del vagón

$v_2$ : Velocidad del vagón

Considerando las variables de estado elegidas, se obtienen las ecuaciones de estado derivando cada una de las variables de estado y utilizando las ecuaciones (2.3) y (2.4), así:

$$\dot{x}_1 = v_1 \quad (2.5)$$

$$\dot{v}_1 = -\frac{k}{M_1}x_1 - \frac{B_1 + B_R}{M_1}v_1 + \frac{k}{M_1}x_2 + \frac{B_R}{M_1}v_2 + \frac{F}{M_1} \quad (2.6)$$

$$\dot{x}_2 = v_2 \quad (2.7)$$

$$\dot{v}_2 = \frac{k}{M_2}x_1 - \frac{B_R}{M_2}v_1 - \frac{k}{M_2}x_2 - \frac{B_2 + B_R}{M_2}v_2 \quad (2.8)$$

La variable de salida del sistema es la velocidad de la locomotora. Por lo tanto la ecuación de salida será:

$$y = v_1 \quad (2.9)$$

### 2.3 Representación en el espacio de estado

Escribiendo estas ecuaciones en el espacio de estado en forma matricial obtenemos:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k}{M_1} & -\frac{B_1 + B_R}{M_1} & \frac{k}{M_1} & \frac{B_R}{M_1} \\ 0 & 0 & 0 & 1 \\ \frac{k}{M_2} & \frac{B_R}{M_2} & -\frac{k}{M_2} & -\frac{B_2 + B_R}{M_2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} [F] \quad (2.10)$$

$$y = [0 \quad 1 \quad 0 \quad 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + [0][F] \quad (2.11)$$

Siendo que las ecuaciones de estado en tiempo continuo están dadas por:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.12)$$

$$y(t) = Cx(t) + Du(t) \quad (2.13)$$

Las matrices A, B, C, D serán las siguientes:

$$A = \begin{bmatrix} \frac{0}{k} & \frac{1}{B_1 + B_R} & \frac{0}{k} & \frac{0}{B_R} \\ \frac{M_1}{M_1} & \frac{M_1}{M_1} & \frac{M_1}{M_1} & \frac{M_1}{M_1} \\ \frac{0}{k} & \frac{0}{B_R} & \frac{0}{k} & \frac{1}{B_2 + B_R} \\ \frac{M_2}{M_2} & \frac{M_2}{M_2} & \frac{M_2}{M_2} & \frac{M_2}{M_2} \end{bmatrix} \quad (2.14)$$

$$B = \begin{bmatrix} 0 \\ F \\ M_1 \\ 0 \\ 0 \end{bmatrix} \quad (2.15)$$

$$C = [0 \quad 1 \quad 0 \quad 0] \quad (2.16)$$

$$D = [0] \quad (2.17)$$

## 2.4 Controlabilidad

Decimos que nuestro sistema es controlable si para cada estado inicial  $x(t_0)$  y cada estado final  $x(t_f)$ , existe una entrada  $u(t)$  que transfiere el sistema desde  $x(t_0)$  en  $t=t_0$  a  $x(t_f)$  en  $t=t_f$ . Si un sistema es controlable, podemos mover el sistema desde cualquier estado inicial a cualquier estado final en un tiempo finito  $t_f - t_0$ . Definimos la matriz de controlabilidad M para un proceso de orden 4 como el nuestro, del modo siguiente:

$$M = [B \quad AB \quad A^2B \quad A^3B] \quad (2.18)$$

Donde el determinante de M debe ser diferente de cero, para lo cual la matriz de controlabilidad debe poseer filas o columnas linealmente independientes. Por consiguiente, para que un proceso sea completamente controlable se debe cumplir que el rango de la matriz de controlabilidad M sea igual al orden del proceso. El valor de la matriz de controlabilidad M, obtenido de la ejecución del programa que se muestra posteriormente, es:

$$M = \begin{bmatrix} 0 & 2.0635 & -0.1801 & -16.3611 \\ 2.0635 & -0.1801 & -16.3611 & 3.1310 \\ 0 & 0 & 0.0172 & 17.1927 \\ 0 & 0.0172 & 17.1927 & -3.3569 \end{bmatrix}$$

En nuestro caso, la matriz de controlabilidad es de rango 4 e igual al orden del proceso, por lo tanto es completamente controlable.

## 2.5 Observabilidad

Un sistema es observable si y si solo si existe un tiempo finito T tal que el estado inicial  $x(t_0)$  puede ser determinado de la observación de  $y(t)$ , dada la señal de control  $u(t)$ . Como en el caso de la controlabilidad, determinamos la observabilidad de un sistema mediante una condición matricial simple. Definimos la matriz de observabilidad N para un proceso de orden 4 como:

$$N = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} \quad (2.19)$$

Donde el determinante de N debe ser diferente de cero, para lo cual la matriz de observabilidad debe poseer filas o columnas independientes. Por consiguiente para

que un proceso sea completamente observable se debe cumplir que el rango de la matriz de observabilidad  $N$  sea igual al orden del proceso. A continuación mostramos el resultado del cálculo de la matriz de observabilidad  $N$ :

$$N = \begin{bmatrix} 0 & 1.0000 & 0 & 0 \\ -7.9365 & -0.0873 & 7.9365 & 0.0079 \\ 0.7590 & -7.9288 & -0.7590 & 7.9351 \\ 129.0529 & 1.5173 & -129.0529 & -1.5493 \end{bmatrix}$$

En este proceso, la matriz de observabilidad es de rango 3, por lo tanto el proceso no es completamente observable.

## 2.6 Simulación en tiempo continuo

Las ecuaciones de estado del modelo propuesto constituyen un sistema lineal invariante en el tiempo (SLIT) de una entrada y una salida (SISO).

Los valores numéricos de los parámetros que se usan en este proceso son:

- Masa de la locomotora  $M_1 = 126000 \text{ kg}$
- Masa del vagón  $M_2 = 120000 \text{ kg}$
- Coeficiente de rigidez del resorte  $k = 1000000 \text{ N/m}$
- Coef. de fricción de la locomotora  $B_1 = 10000 \text{ N-s/m}$
- Coef. de fricción estrozamiento  $B_2 = 10000 \text{ N-s/m}$
- Coef. de fricción viscosa del resorte  $B_R = 1000 \text{ N-s/m}$
- Fuerza de la locomotora  $F = 260000 \text{ N}$

Se ha usado MATLAB para simular el proceso. MATLAB es un programa interactivo para computación numérica y visualización de datos, es usado extensamente por ingenieros de control para el diseño y análisis. Para simular el sistema se ha utilizado el programa MATLAB versión 5.3.0.10183 Release 11 en un

computador personal tipo notebook marca Toshiba con procesador Pentium II de 266 Mhz.

El programa de simulación del sistema en lazo abierto modelo.m se lista en el anexo A1. El resultado de evaluar las matrices A y B, reemplazando los valores numéricos de los parámetros se muestra a continuación:

$$A = \begin{bmatrix} 0 & 1.0000 & 0 & 0 \\ -7.9365 & -0.0873 & 7.9365 & 0.0079 \\ 0 & 0 & 0 & 1.0000 \\ 8.3333 & 0.0083 & -8.3333 & -0.0917 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 2.0635 \\ 0 \\ 0 \end{bmatrix}$$

La gráfica superior de la figura 2.3 ilustra el resultado del programa en tiempo continuo, donde el sistema del tren opera en lazo abierto sin ningún tipo de control, se estabiliza y alcanza una velocidad final de 13.0000 m/s

## 2.7 Discretización del modelo y simulación en tiempo discreto

Las ecuaciones de estado en tiempo discreto se describen con las ecuaciones siguientes:

$$x(k+1) = Gx(k) + Hu(k) \quad y(k) = Cx(k) + Du(k) \quad (2.20)$$

donde:

$$G = \Phi(T) = e^{AT} \quad H = \left[ \int_0^T \phi(\lambda) d\lambda \right] B \quad (2.21)$$

Para la discretización del proceso se usa un período de muestreo de 1 segundo. El programa en MATLAB para la simulación del modelo del proceso en tiempo

discreto está en el Anexo A1. Los resultados principales de la simulación son los siguientes:

El resultado obtenido de calcular las matrices G y H es como sigue:

$$G = \begin{matrix} 0.2149 & 0.4025 & 0.7851 & 0.5581 \\ 1.4567 & 0.1844 & -1.4567 & 0.7371 \\ 0.8220 & 0.5860 & 0.1780 & 0.3742 \\ -1.5326 & 0.7740 & 1.5326 & 0.1483 \end{matrix}$$

$$H = \begin{matrix} 0.6141 \\ 0.8305 \\ 0.4100 \\ 1.2093 \end{matrix}$$

Nuevamente se calculan la matriz M de controlabilidad y la matriz N observabilidad, esta vez en el tiempo discreto, con el resultado siguiente:

$$M_d = \begin{matrix} 0.6141 & 1.4631 & 2.3298 & 3.3110 \\ 0.8305 & 1.3419 & 0.5443 & 0.8080 \\ 0.4100 & 1.5170 & 2.4497 & 3.1180 \\ 1.2093 & 0.5093 & 1.1969 & 0.7825 \end{matrix}$$

rangoMD = 4 => Es completamente controlable

$$N_d = \begin{matrix} 0 & 1.0000 & 0 & 0 \\ 1.4567 & 0.1844 & -1.4567 & 0.7371 \\ -1.7455 & 0.3372 & 1.7455 & 0.5132 \\ 0.7644 & 0.7798 & -0.7644 & 0.0036 \end{matrix}$$

rangoNd = 3 => No es completamente Observable

El tiempo de ascenso 10%-90%  $T_{r1}$  es de 28 segundos, considerando que alcanza los 2.1724 m/s (10%) a los 3 segundos y 11.7296m/s (90%) a los 30 segundos.

El tiempo de estabilización  $T_s$  es de 50 segundos, pues alcanza la velocidad de 13.00 m/s en ese tiempo, considerando un margen de error  $\delta = \pm 2\%$ .

La figura 2.3 muestra la velocidad en tiempo continuo y en tiempo discreto:



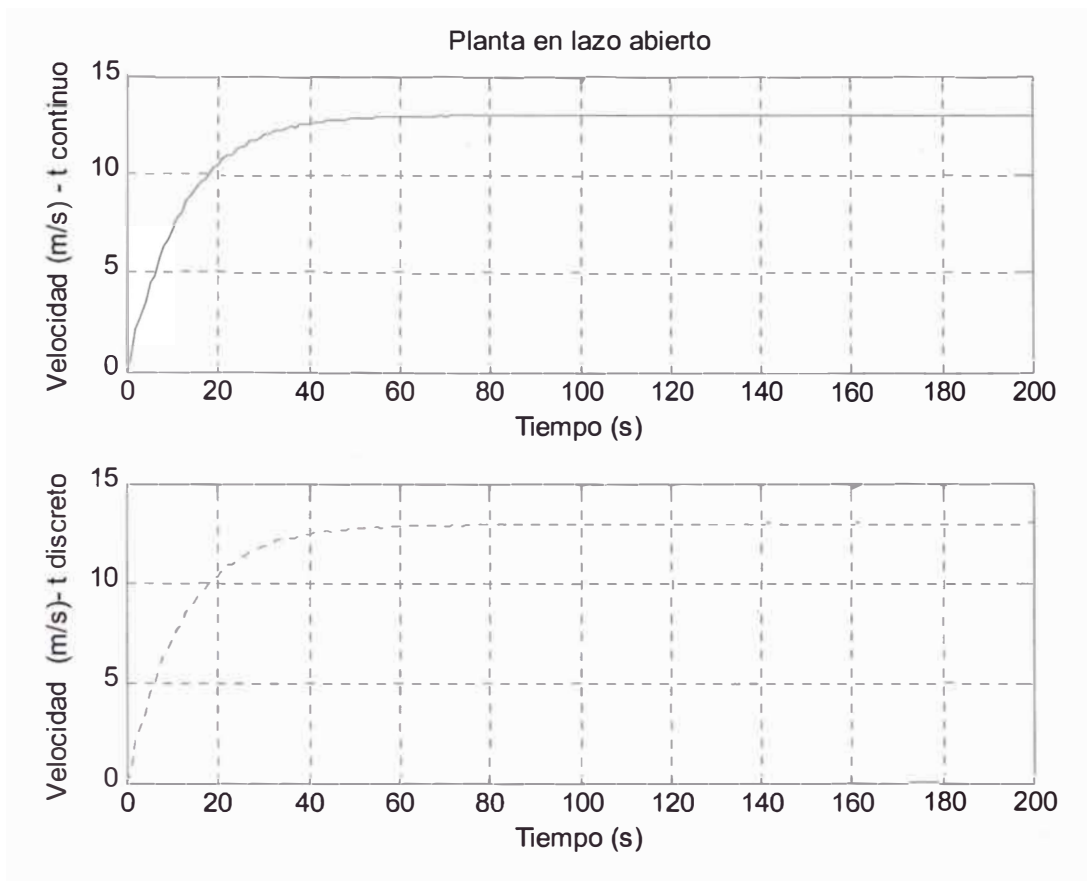


Fig. 2.3 Respuesta del proceso al escalón, en tiempo continuo y tiempo discreto.

## 2.8 Modelamiento y simulación en SIMULINK

SIMULINK permite simular sistemas de control con realimentación; constituyendo otra herramienta de MATLAB para diseñar y analizar sistemas de control. Los sistemas físicos y modelos pueden representarse directamente con SIMULINK, tal como se muestra a continuación.

La figura 2.4 muestra el modelo del tren en SIMULINK. La entrada del sistema es la fuerza  $F$ , proporcionada por la locomotora; la cual se ha representado con la función escalón. La salida del sistema es la velocidad de la locomotora. Se han colocado cinco osciloscopios para monitorear la entrada y las cuatro salidas posibles: Vista escalón, Vista \_v1, Vista \_v2, Vista x1, y Vista x2.

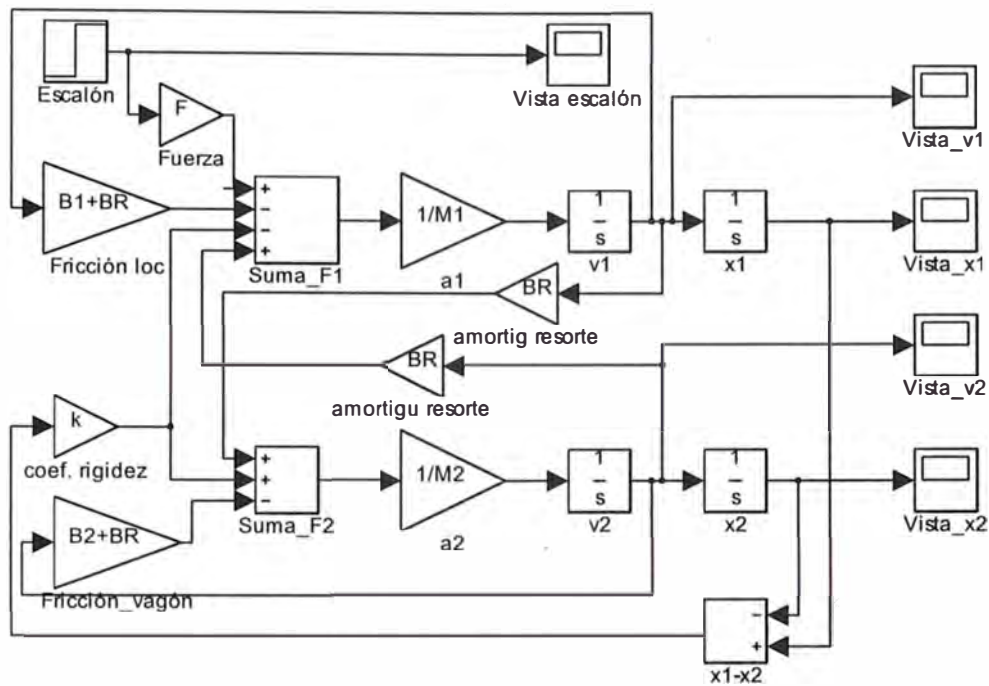


Fig. 2.4 Modelo del tren en SIMULINK.

#### Monitoreo de las señales con osciloscopios

El programa modelotren.mdl que corresponde al modelo del tren en SIMULINK está listado en el anexo A2.

La figura 2.5 muestra la entrada al sistema, la cual permite simular el arranque y la parada del tren. Es un escalón de amplitud 1 y de 100 segundos de duración, de tal manera que el flanco de subida simula el arranque del tren y el flanco de bajada simula la parada del tren.

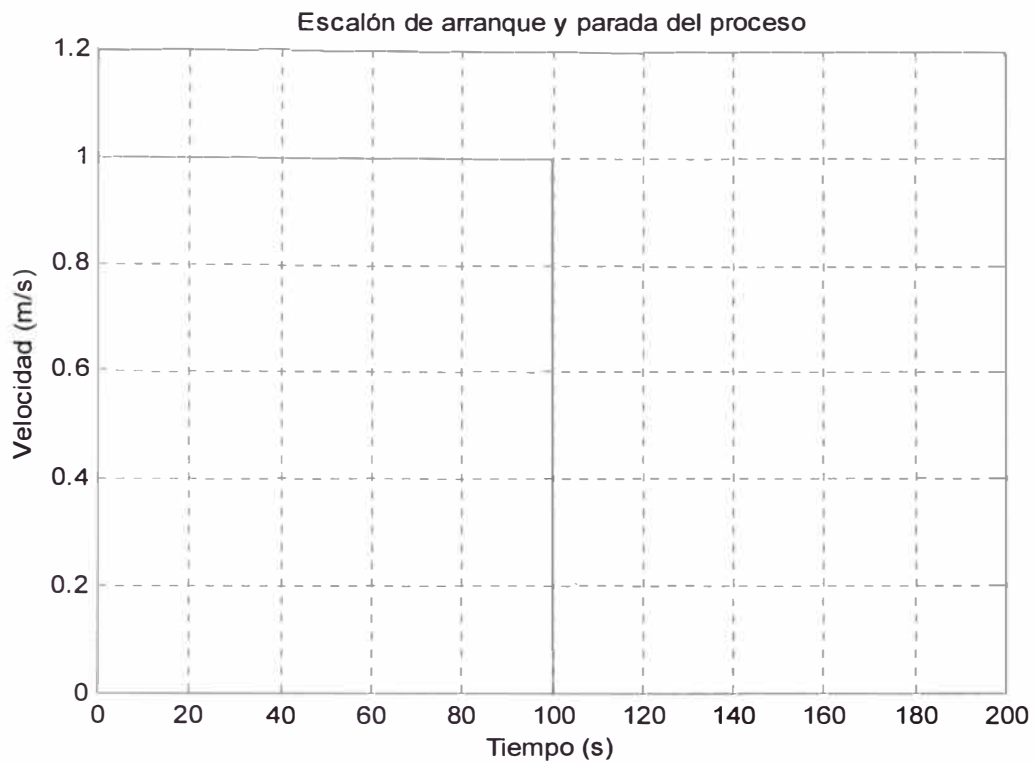


Fig. 2.5 Escalón de entrada en lazo abierto

La velocidad de la locomotora se muestra en la figura 2.6. Observamos que llega a su velocidad final  $v_1$  de 13.00 m/s en aproximadamente 50 segundos con un error menor al 2 %. Por otro lado observamos en la gráfica que el sistema inicia su detención a los 100 segundos, alcanzando su detención total después de 90 segundos.

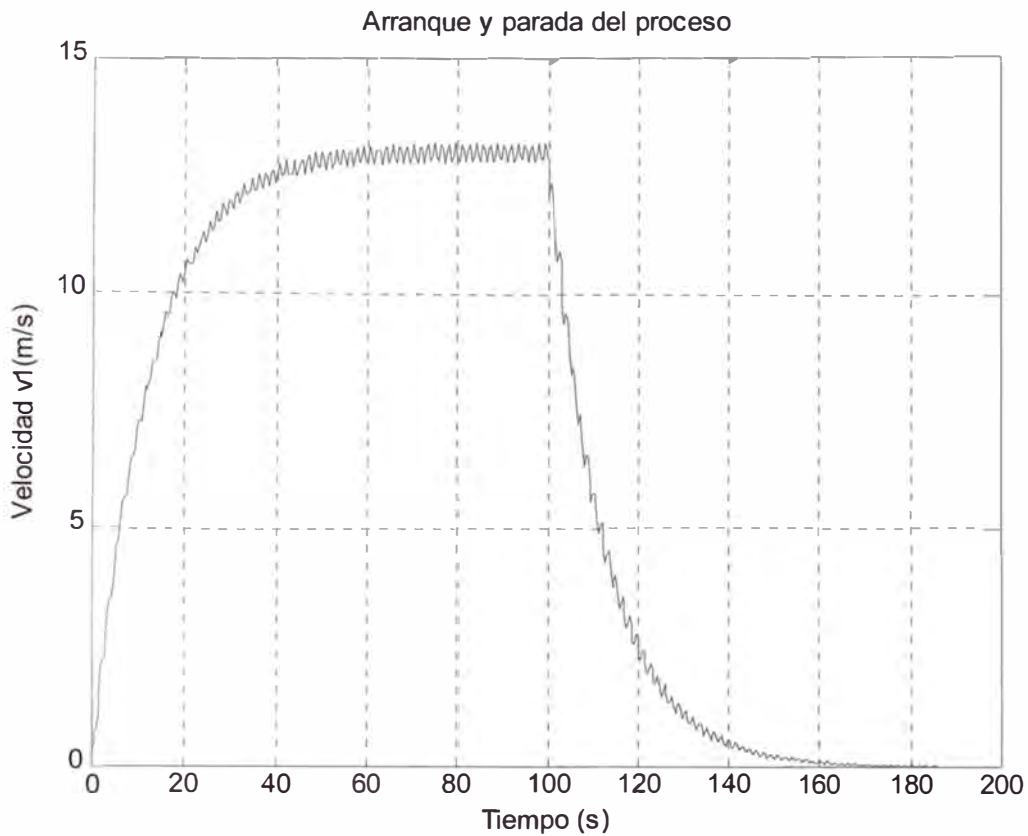


Fig. 2.6 Respuesta del proceso al escalón.  
Velocidad de la locomotora

La velocidad del tren para efectos de nuestro proceso se descompone en la velocidad de la locomotora ( $v_1$ ) y la velocidad del vagón ( $v_2$ ); los cuales son iguales en la práctica. Sin embargo, con fines ilustrativos se muestran en la figura 2.7 el gráfico de la velocidad del vagón ( $v_2$ ). La diferencia fundamental estriba en que la velocidad del vagón muestra mayores variaciones que la de la locomotora, lo que se debe al resorte, que amortigua las variaciones de velocidad de la locomotora.

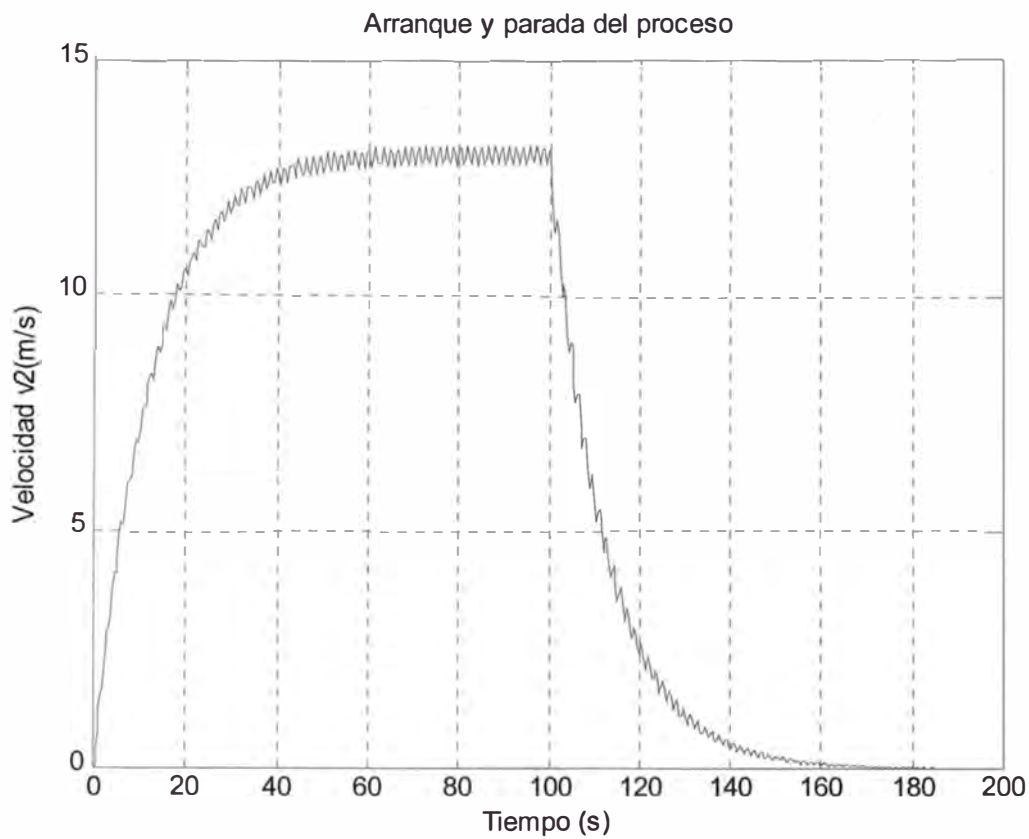


Fig. 2.7 Respuesta del proceso al escalón.  
Velocidad del vagón

Tenemos a continuación la gráfica de la posición  $x_1$  la figura 2.8. Esta corresponde al monitoreo obtenido de Vista\_x1 de la figura 2.4. Observamos que la distancia recorrida se detiene a los 140 segundos de iniciado el proceso, es decir 40 segundos después que deja de actuar la fuerza  $F$  o lo que es lo mismo cuando se inicia el flanco de bajada del escalón unitario usado para simular la parada del tren.

En la gráfica se verifica que la pendiente de la curva se mantiene constante durante el tiempo de aplicación de la fuerza  $F$  en la simulación del recorrido del tren.

La obtención de las figuras 2.5 a 2.8 se logró reemplazando los osciloscopios por dispositivos de almacenaje de datos a archivos. Luego la información recopilada en dichos archivos fue graficada con el programa captura.m listado en el anexo A3.

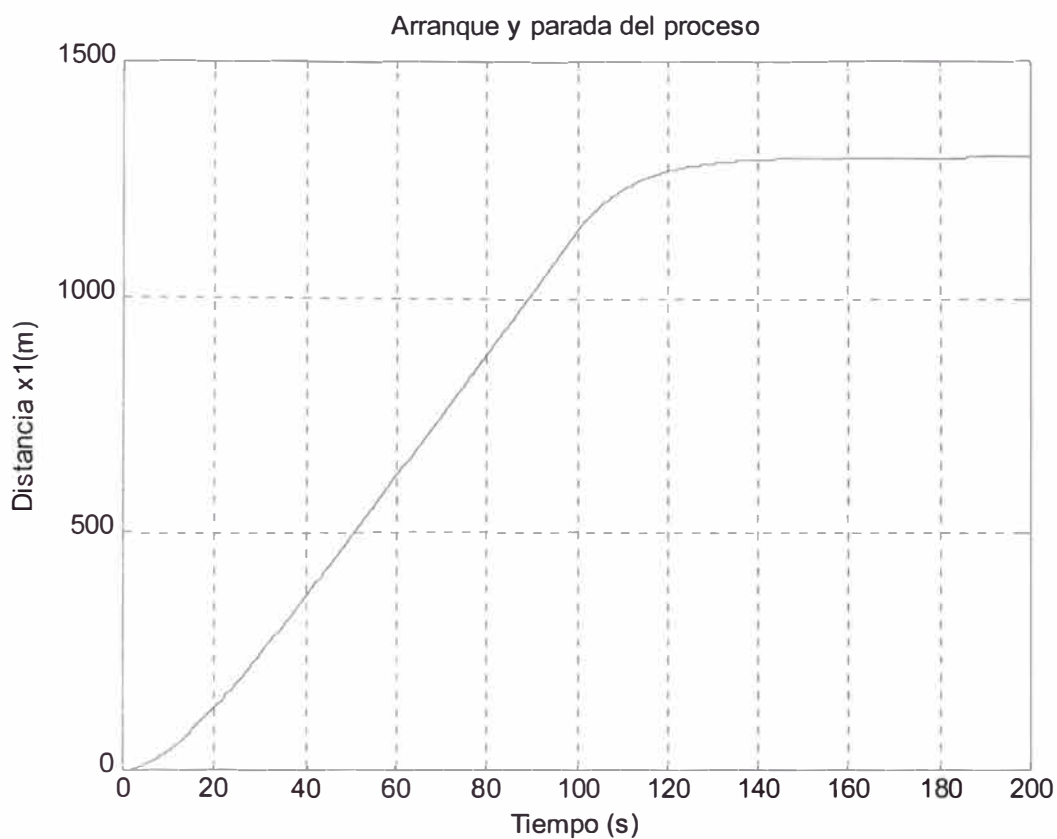


Fig. 2.8 Distancia recorrida por el tren.

La figura 2.9 muestra el modelo del tren en SIMULINK reemplazando los osciloscopios por dispositivos de almacenaje de datos a archivos.

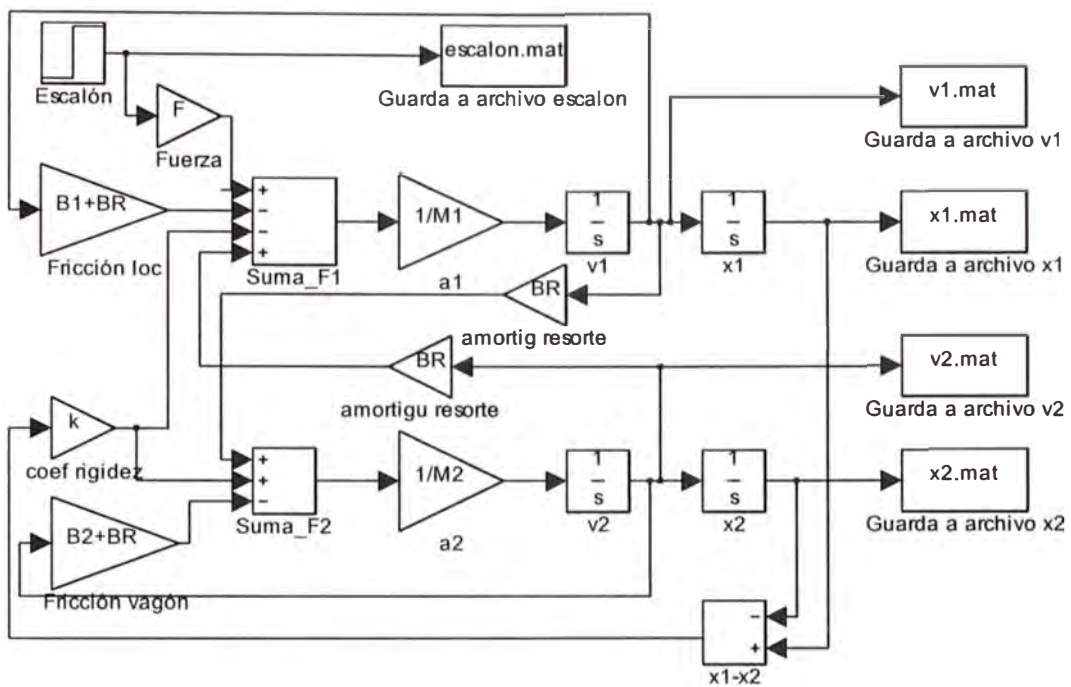


Fig. 2.9 Modelo del tren en SIMULINK.  
Monitoreo de las señales con almacenaje de datos

## CAPÍTULO III SISTEMA DE CONTROL ÓPTIMO

### 3.1 Diseño del controlador óptimo

Un Sistema de control óptimo se caracteriza por optimizar una función denominada función de costo  $J$ , la cual tiene la forma siguiente en el tiempo discreto:

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

Donde  $k$  es el tiempo discreto,  $N$  es el tiempo discreto final,  $x(k)$  es el vector de estado,  $r(k)$  es el vector de referencia (entrada al sistema de control) y  $u(k)$  es el vector de control óptimo. El comportamiento del sistema se hará óptimo eligiendo el vector de control  $u(k)$  de tal forma que la función de costo sea máxima o mínima.

La figura 3.1 muestra el diagrama en bloques del sistema de control óptimo con integrador debido a que el proceso es de tipo 0.

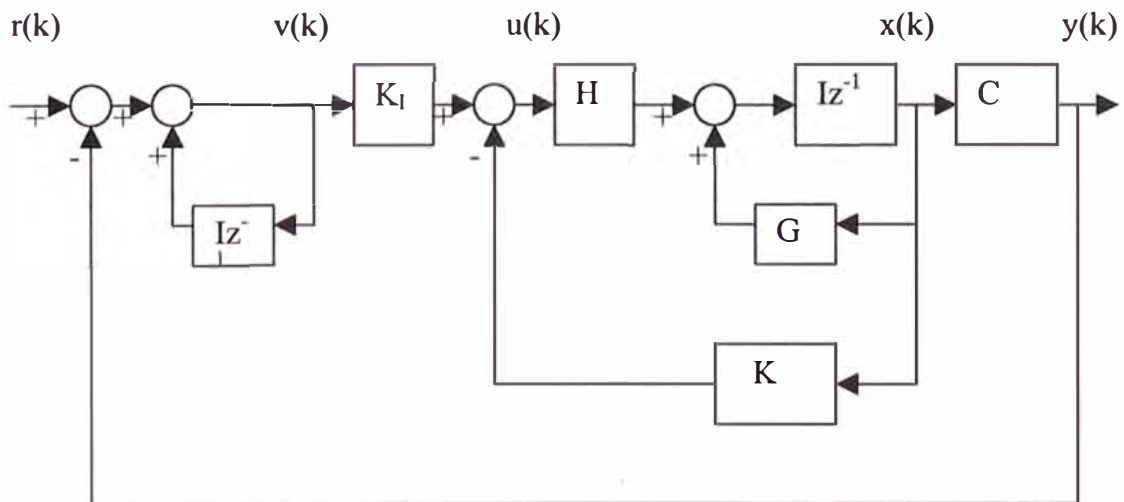


Figura 3.1 Regulador óptimo proporcional integral



Para nuestro proceso se considera un Control Óptimo Cuadrático Estacionario con un Regulador Óptimo Proporcional Integral llamado también Controlador Integral. Este regulador tiene por finalidad obtener un error nulo en estado estacionario mediante el integrador.

Donde:  $y(k)$  es la salida, representando la velocidad de la locomotora,  $r(k)$  es la entrada o señal de referencia. En nuestro caso se aplica un escalón unitario para simular el arranque del tren. El regulador óptimo está compuesto por un integrador con señal de salida  $v(k)$  y una ganancia integral  $K_I$ , los bloques restantes constituyen la planta con realimentación del estado. Las siguientes ecuaciones se deducen de la figura 3.1: ecuación de estado, la ecuación de salida, la señal de control, la matriz de ganancia del controlador  $K$ , el cual es un vector fila y la ecuación del integrador, las cuales se escriben a continuación, en orden respectivo:

$$x(k + 1) = Gx(k) + Hu(k) \quad (3.1)$$

$$y(k) = Cx(k) \quad (3.2)$$

$$u(k) = -Kx(k) + K_I v(k) \quad (3.3)$$

$$K = [K_1 \quad K_2 \quad K_3 \quad K_4] \quad (3.4)$$

$$v(k) = v(k - 1) + r(k) - y(k) \quad (3.5)$$

A continuación se deducen las ecuaciones que permiten obtener la solución al sistema de control óptimo.

La ecuación (3.5) se puede re-escribir como:

$$v(k + 1) = v(k) + r(k + 1) - y(k + 1)$$

Utilizando las ecuaciones (3.1) y (3.2):

$$v(k + 1) = v(k) + r(k + 1) - C[Gx(k) + Hu(k)]$$

Reemplazamos (3.3) y reordenamos términos se obtiene:

$$v(k+1) = (1 - CHK_I)v(k) + (-CG + CHK)x(k) + r(k+1) \quad (3.6)$$

Empleando las ecuaciones (3.1) y (3.3) se obtiene:

$$x(k+1) = Gx(k) + H[-Kx(k) + K_I v(k)]$$

$$x(k+1) = (G - HK)x(k) + HK_I v(k) \quad (3.7)$$

De las ecuaciones (3.6) y (3.7) y (3.2) se deduce:

$$\begin{bmatrix} x(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} G - HK & HK_I \\ -CG + CHK & 1 - CHK_I \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k+1) \quad (3.8)$$

$$y(k) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} \quad (3.9)$$

En el estado de equilibrio (para  $k \rightarrow \infty$ ), los valores de  $x(k)$ ,  $u(k)$  y  $v(k)$  toman valores estacionarios  $x(\infty)$ ,  $u(\infty)$  y  $v(\infty)$ . Así la ecuación (3.8) se convierte en:

$$\begin{bmatrix} x(\infty) \\ v(\infty) \end{bmatrix} = \begin{bmatrix} G - HK & HK_I \\ -CG + CHK & 1 - CHK_I \end{bmatrix} \begin{bmatrix} x(\infty) \\ v(\infty) \end{bmatrix} + \begin{bmatrix} 0 \\ r \end{bmatrix} \quad (3.10)$$

Se definen:

$$x(k) - x(\infty) = x_e(k) \quad (3.11)$$

$$v(k) - v(\infty) = v_e(k) \quad (3.12)$$

Restando la ecuación (3.10) de (3.8), y luego usando las relaciones (3.11) y (3.12) se obtiene:

$$\begin{bmatrix} x_e(k+1) \\ v_e(k+1) \end{bmatrix} = \begin{bmatrix} G - HK & HK_I \\ -CG + CHK & 1 - CHK_I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix}$$

$$= \begin{bmatrix} G & 0 \\ -CG & I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix} + \begin{bmatrix} H \\ -CH \end{bmatrix} \begin{bmatrix} -K & K_I \end{bmatrix} \begin{bmatrix} x_e(k) \\ v_e(k) \end{bmatrix} \quad (3.13)$$

La ecuación (3.13) puede ser escrita como:

$$\xi(k+1) = \tilde{G}\xi(k) + \tilde{H}w(k) \quad (3.14)$$

$$w(k) = -\tilde{K}\xi(k) \quad (3.15)$$

donde:

$$\tilde{G}(k) = \begin{bmatrix} G & 0 \\ -CG & I \end{bmatrix} \quad (3.16)$$

$$\tilde{H}(k) = \begin{bmatrix} H \\ -CH \end{bmatrix} \quad (3.17)$$

$$\tilde{K}(k) = \begin{bmatrix} K & -K_I \end{bmatrix} \quad (3.18)$$

La inclusión de un integrador en el sistema de control aumenta el orden del sistema en uno. En nuestro caso el orden del sistema original es 4, por lo que el nuevo orden del sistema con integrador es 5. Además la función de costo  $J$  se define del modo siguiente:

$$J = \frac{1}{2} \sum_{k=0}^{\infty} [\xi(k)^T (k) \tilde{Q} \xi(k) + w^2(k) \tilde{R}] \quad (3.19)$$

Se debe calcular entonces una matriz  $\tilde{K}$  adecuada de forma tal que minimice la función de costo  $J$  mencionada. Es posible formular la ecuación de Riccati y la ecuación de ganancia del controlador  $\tilde{K}$  para el regulador, como sigue:

$$\tilde{P} = \tilde{Q} + \tilde{G}^T \tilde{P} \tilde{G} - \tilde{G}^T \tilde{P} \tilde{H} [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.20)$$

$$\tilde{K} = [\tilde{R} + \tilde{H}^T \tilde{P} \tilde{H}]^{-1} \tilde{H}^T \tilde{P} \tilde{G} \quad (3.21)$$

El procedimiento de diseño consiste básicamente en calcular la matriz de ganancia óptima  $\tilde{K}$  de control.

### 3.2 Cálculo de la matriz de ganancia óptima

Para nuestro proceso, se calcula la matriz de ganancia óptima  $\tilde{K}$  realizando los siguientes pasos:

#### a) Cálculo de $\tilde{G}$ y $\tilde{H}$

En código MATLAB, utilizaremos la siguiente notación:

$$G1 = \tilde{G} \quad H1 = \tilde{H} \quad KK = \tilde{K}$$

Por lo tanto usando las ecuaciones (3.16) y (3.17), el cálculo se efectúa como sigue:

```
G1 = [G zeros(4,1); -C*G 1];
H1 = [H; -C*H];
```

Los resultados obtenidos son:

```
G1 = 0.2149    0.4025    0.7851    0.5581    0
      1.4567    0.1844   -1.4567    0.7371    0
      0.8220    0.5860    0.1780    0.3742    0
     -1.5326    0.7740    1.5326    0.1483    0
     -1.4567   -0.1844    1.4567   -0.7371    1.0000
```

```
H1 = 0.6141
      0.8305
      0.4100
      1.2093
     -0.8305
```

#### b) Asignación de las Matrices de Ponderación $\tilde{Q}$ y $\tilde{R}$

Partiendo de matrices identidad, determinamos los valores mas adecuados mediante varias iteraciones, asignando mayor importancia a la velocidad de la

locomotora. La matriz Q pondera el vector de estado  $x(k)$  mientras que la matriz R pondera la señal de control  $u$ . Las matrices de ponderación con las que se puede obtener la mejor solución finalmente son:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 200 \end{bmatrix};$$

$$R = [10];$$

### c) Ecuación de Riccati P y Matriz de Ganancia $\tilde{K}$

La ecuación de Riccati P y la matriz de Ganancia  $\tilde{K}$  se calculan usando el comando `dlqr` del MATLAB con la siguiente sintaxis:

$$[KK, P] = \text{dlqr}(G1, H1, Q, R);$$

Los valores obtenidos se indican a continuación:

$$P = 1.0e+005 * \begin{bmatrix} 0.0399 & 0.1704 & 0.3017 & 0.0035 & 0.3393 \\ 0.1704 & 1.9347 & 3.7720 & 0.0215 & 3.9420 \\ 0.3017 & 3.7720 & 7.4344 & 0.0403 & 7.7385 \\ 0.0035 & 0.0215 & 0.0403 & 0.0005 & 0.0436 \\ 0.3393 & 3.9420 & 7.7385 & 0.0436 & 8.0835 \end{bmatrix}$$

$$KK = \begin{bmatrix} 0.8585 & 0.0831 & -0.8550 & 0.8367 & -0.3432 \end{bmatrix}$$

De la matriz de ganancia se obtiene el valor de la ganancia de la matriz del integrador  $K_I$ , usando (3.18)

$$K_I = 0.3432$$

### 3.3 Diseño del observador óptimo cuadrático

En la sección 2.8 Discretización del modelo y simulación en tiempo discreto encontramos que el sistema no es observable. Por lo tanto requerimos hallar un vector de estado estimado  $\hat{x}(k)$ , lo que se logra mediante el empleo de un

observador discreto de estados. El diseño de dicho observador implica determinar la matriz de ganancia  $K_e$ , que puede ser calculada a partir de la minimización de una función de costo cuadrática.

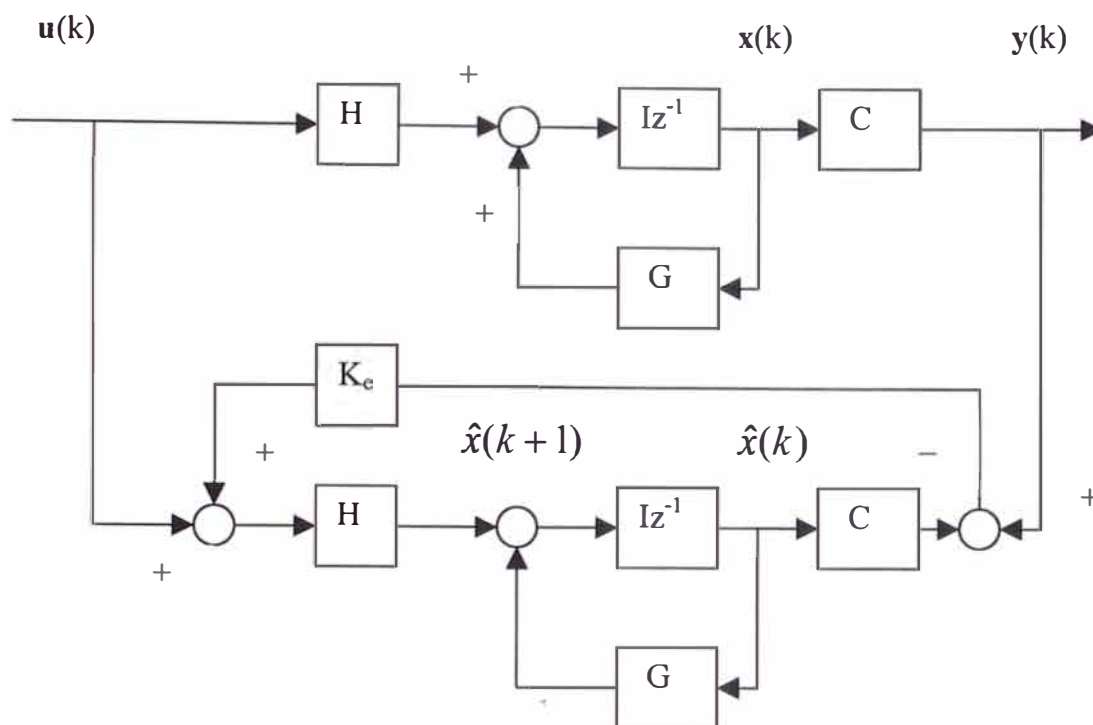


Figura 3.2 Diagrama de bloques del observador de estados

Observando la figura 3.2 se confirma las ecuaciones (3.1) y (3.2); asimismo formulando la ecuación del observador:

$$\hat{x}(k+1) = G\hat{x}(k) + Hu(k) + HK_e[\hat{y}(k) - C\hat{x}(k)] \quad (3.22)$$

donde  $\hat{x}(k)$  es el vector de estado estimado de dimensión  $n$ ,  $\hat{y}(k)$  representa el vector de salida estimado de dimensión  $m$ ,  $K_e$  es la matriz de ganancia de realimentación del observador con dimensión  $n \times m$  y  $C$  es la matriz de salida de dimensión  $m \times n$ . Reemplazando la ecuación (3.2) en (3.22) y restando la ecuación resultante de (3.1), podemos obtener la ecuación de error del observador:

$$e(k+1) = [G - K_e C]e(k); \quad e(k) = x(k) - \hat{x}(k) \quad (3.23)$$

mientras que su ecuación característica toma la forma:

$$\det[sI - G + K_e C] = 0 \quad (3.24)$$

Donde la matriz  $K_e$  debe ser escogida apropiadamente para que el error tienda a cero con una velocidad suficientemente alta. Las raíces de dicha ecuación característica deben posicionarse dentro del círculo unitario para una operación satisfactoria del observador.

El observador deseado debe permitir que el proceso sea completamente observable; es decir que cualquier estado inicial pueda determinarse a partir de la observación de los vectores de salida y de control, en un número finito de períodos de muestreo. El criterio para observabilidad completa se obtiene de la matriz  $N$  de acuerdo a la ecuación siguiente para el tiempo discreto:

$$N = [C^T \quad G^T C^T \quad \dots \quad (G^T)^{n-1} C^T] \quad (3.25)$$

La condición necesaria y suficiente para que el sistema sea completamente observable es que la matriz  $N$  posea rango  $n$  (rango completo). Es decir el rango de  $n$  debe ser igual al rango del proceso.

Para calcular la matriz  $K_e$  se procede de la misma forma en que se hizo con la matriz de ganancia  $K$  del controlador; es decir; podemos utilizar las ecuaciones que describen al sistema de control óptimo, conveniente modificadas para que nos permitan calcular  $K_e$ . Dado que el determinante de una matriz y el de su transpuesta son iguales, podemos reformular la ecuación (3.24) como sigue:

$$\det[sI - G + K_e C] = \det[sI - G + K_e C]^T = \det[sI - G^T + K_e^T C^T] \quad (3.26)$$

La ecuación característica del controlador de estados es la siguiente:

$$\det[zI - G + HK] = 0 \quad (3.27)$$

Comparando las dos últimas ecuaciones (3.26) y (3.27), se deduce que se deben hacer las siguientes modificaciones:

$$G \rightarrow G^T; \quad H \rightarrow C^T; \quad K \rightarrow K^T \quad (3.28)$$

Empleando estas modificaciones en las estructuras de la ecuación de estado del proceso  $x(k+1)=Gx(k)+Hu(k)$ , de la ecuación de salida  $y(k)=Cx(k)$  y de la ley de control  $u(k)=-Kx(k)$ , se obtiene la siguiente ecuación de estado:

$$\alpha(k+1) = G^T \alpha(k) + C^T \beta(k) \quad (3.29)$$

con una ley de control que posee la forma siguiente:

$$\beta(k) = -K_e^T \alpha(k) \quad (3.30)$$

Empleando las ecuaciones (3.29) y (3.30) en la función de costo siguiente:

$$J = \frac{1}{2} \sum_{k=0} [\alpha^T(k) Q_e \alpha(k) + \beta^T(k) R_e \beta(k)] \quad (3.31)$$

entonces la correspondiente ecuación de Riccati toma la forma:

$$P_e = Q_e + GP_e G^T - GPC^T [R_e + CP_e C^T]^{-1} CP_e G^T \quad (3.32)$$

y la matriz de ganancia  $K_e$  viene dada por:

$$K_e = [R_e + CP_e C^T]^{-1} CP_e G^T \quad (3.33)$$

Para determinar  $P_e$  a partir de la ecuación (3.32), empleamos el mismo procedimiento que para determinar  $P$ . Es decir, para determinar  $P_e$  utilizaremos la siguiente ecuación recursiva:

$$P_e(k+1) = Q_e + GP_e(k)G^T - GP_e(k)C^T [R_e + CP_e(k)C^T]^{-1} CP_e(k)G^T \quad (3.34)$$

El cálculo del observador se escribe en MATLAB del modo siguiente:

```
% Cálculo de la Matriz de Ganancia del Observador
```



```

NO=[C' G'*C' G'^2*C' G'^3*C' G'^4*C'];
rnN=rank(NO); %rnN=3 Por lo tanto no es observable
Qe=[1 0 0 0
     0 1000 0 0
     0 0 1 0
     0 0 0 1];
Re=[10];
Pe=zeros(4,4);
for i=1:40
    Pe=Qe+G*Pe*G'-G*Pe*C'*inv(Re+C*Pe*C')*C*Pe*G';
end
KeT=inv(Re+C*Pe*C')*C*Pe*G';
Pe =
    1.0e+003 *
    4.5404    0.1109    4.5243    0.0043
    0.1109    1.1765    0.0921   -0.0097
    4.5243    0.0921    4.5201   -0.0071
    0.0043   -0.0097   -0.0071    0.2328

KeT    0.4756    0.1999    0.6687    0.7420

```

En el capítulo II se calculó la matriz de Observabilidad N y se obtuvo que el rango de dicha matriz tanto en tiempo continuo como en tiempo discreto es 3, es decir el sistema no es completamente observable. Sin embargo, luego de obtener la matriz de ganancia Ke, el sistema es observable en estado completo.

### 3.4 Simulación del sistema de control óptimo

El programa `optimo.m` del anexo A.4 simula el sistema de control óptimo empleando el regulador óptimo proporcional integral y el modelo del proceso descrito anteriormente.

La figura 3.2 ilustra el resultado de simulación. Consta de dos gráficas. El gráfico superior muestra la velocidad de la locomotora con el flanco de subida para el arranque y el flanco de bajada para la parada. La gráfica inferior muestra la señal de control  $u$

El tiempo de ascenso 10%-90%  $T_{r1}$  es de 5 segundos, considerando que alcanza los 0.2851 m/s (10%) al 1 segundo y 0.9183 m/s (90%) a los 6 segundos.

El tiempo de estabilización  $T_s$  es de 9 segundos, pues alcanza la velocidad de 0.9755 m/s en ese tiempo, considerando un  $\delta = \pm 2\%$  para una velocidad estabilizada en 0.9901 m/s.

No presenta ningún sobrepico.

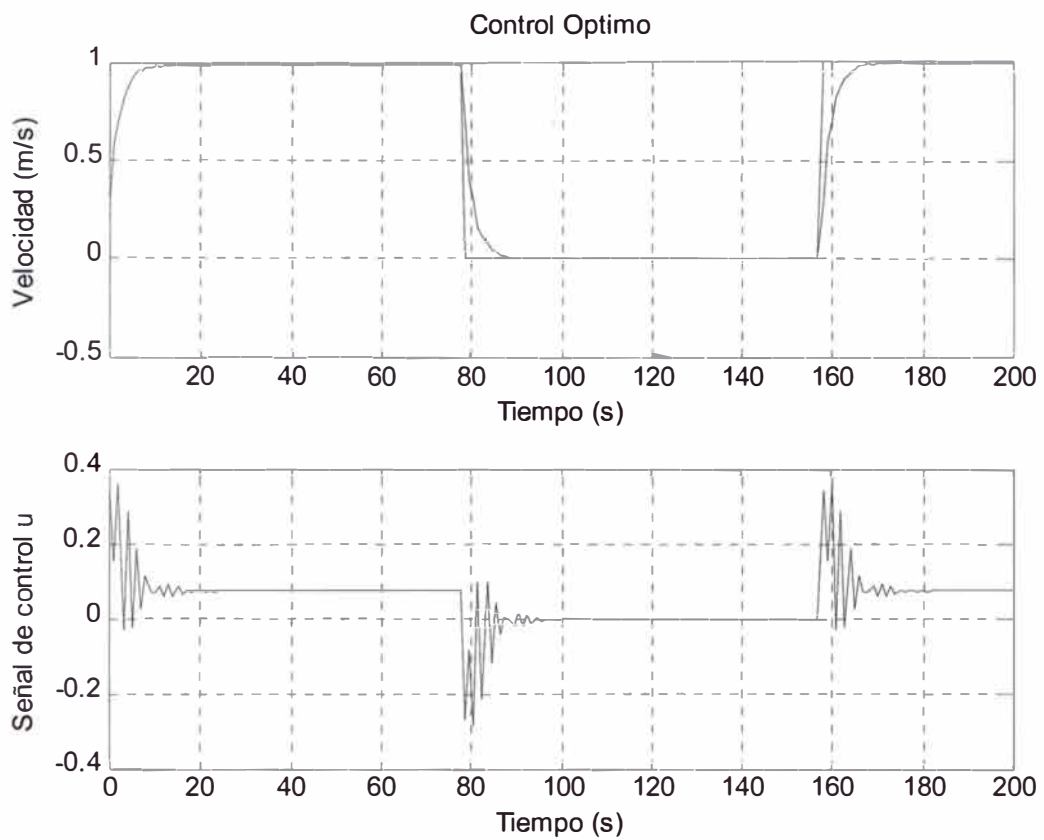


Fig. 3.3 Simulación del sistema de control óptimo

## CAPÍTULO IV

### SISTEMA DE CONTROL ADAPTIVO

#### 4.1 Diseño del controlador adaptivo

Un sistema de control adaptivo es aquel sistema que continua y automáticamente mide las características dinámicas de la planta, las compara con las características dinámicas deseadas y usa la diferencia para variar parámetros ajustables del sistema o para generar una señal de accionamiento de modo que se pueda mantener el funcionamiento óptimo con independencia de las variaciones ambientales.

Para nuestro proceso empleamos un sistema de control adaptivo con autosintonización, el cual combina en el diseño un método de estimación de parámetros denominado el de los “mínimos cuadrados recursivo mejorado”, una técnica de estimación de estados (el filtro de Kalman), una representación lineal del modelo del proceso y una ley de control (el controlador proporcional-integral óptimo cuadrático con realimentación de estados).

El objetivo del control es determinar una fuerza de control capaz de minimizar la diferencia entre la salida del proceso y la señal de referencia. La figura 4.1 muestra un sistema de control adaptivo con autosintonización que opera como sigue: luego de cada tiempo de muestreo, el vector estimado de parámetros  $\theta$  se actualiza empleando los datos proporcionados por la entrada  $U$  del proceso y por la salida  $Y$  del mismo. Luego los elementos de  $\theta$  se usan para recuperar el modelo lineal del proceso, lo cual permite estimar el vector de estado del modelo del proceso  $\hat{x}$

(empleando un filtro de Kalman) y el valor de equilibrio  $\bar{U}$  de la ley de control actual  $U$ . Tales resultados se utilizan luego para calcular la ley de control residual  $u$  y actualizar la ley de control  $U$  a partir de la relación  $U = \bar{U} + u$ .

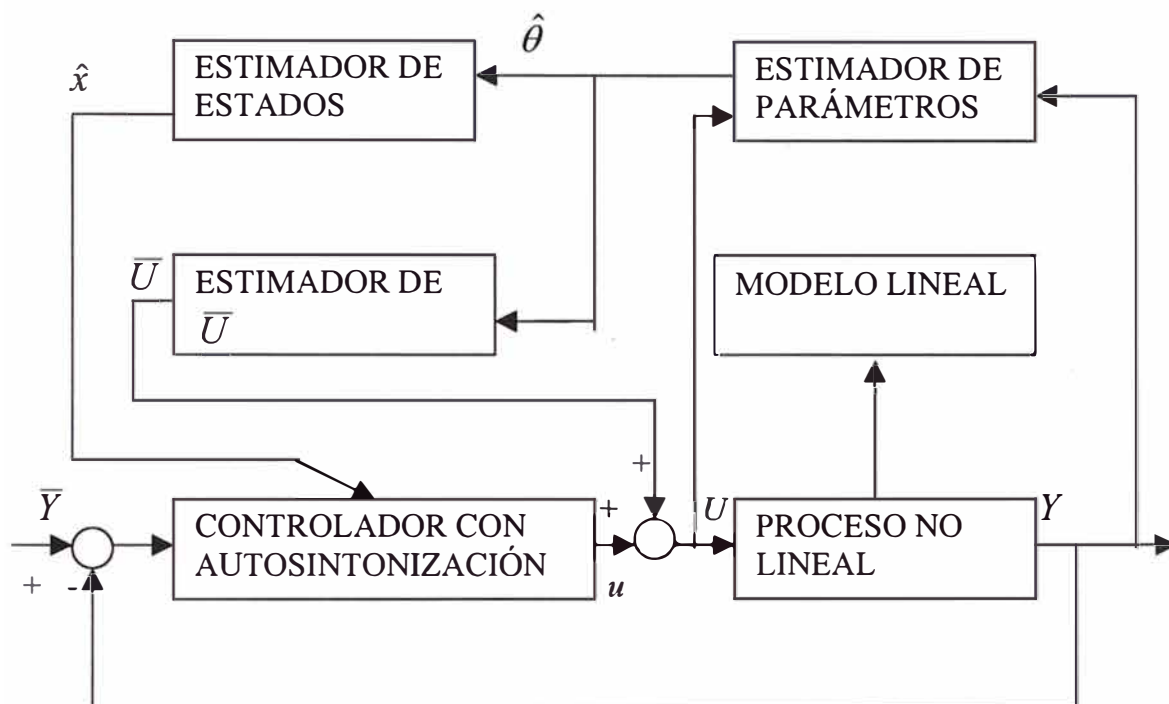


Fig. 4.1 Sistema de control adaptivo con autosintonización

## 4.2 Estimación de parámetros

Para estimar los parámetros del modelo del proceso, empleamos el método de los mínimos cuadrados recursivo mejorado. Como primer paso se calculan los valores de  $a_1, a_2, a_3, a_4, b_1, b_2, b_3$  y  $b_4$  desde G y H para el vector inicial de parámetros.

El modelo discreto en código MATLAB es el siguiente:

```
% MODELO LINEAL DISCRETO
T = 1; % PERIODO DE MUESTREO
[G, H, C, D]=c2dm(A, B, C, D, T, 'zoh');
[num, den]=ss2tf(G, H, C, D);
a1 = den(2); a2 = den(3); a3=den(4); a4=den(5);
b1 = num(2); b2 = num(3); b3=num(4); b4=num(5);
```

El resultado obtenido es:

```
num = [0      0.8305      0.7393      -0.8200      -0.7498]
den = [1.0000      -0.7256      -0.4703      -0.6402      0.8361]
```

Por lo tanto los valores de  $a_1, a_2, a_3, a_4, b_1, b_2, b_3$  y  $b_4$  son:

```
a1= -0.7256  a2=-0.4703      a3=-0.6402  a4=0.8361
b1= 0.8305  b2= 0.7393  b3=-0.8200      b4=-0.7498
```

Luego el vector inicial de parámetros es:

$$\theta(0) = [-a_1 \quad -a_2 \quad -a_3 \quad -a_4 \quad b_1 \quad b_2 \quad b_3 \quad b_4 \quad C(0)]^T \quad (4.1)$$

donde el valor inicial del  $C(0)$  considerando  $\bar{U} = 0$  e  $\bar{Y}$  igual a la señal de referencia es:

$$C = (1 + a_1 + a_2 + a_3 + a_4)\bar{Y} \quad (4.2)$$

En MATLAB se escribe la ecuación (4.1) como:

```
thi = [-a1 -a2 -a3 -a4 b1 b2 b3 b4 Ce]'
```

y la ecuación (4.2) como:

```
Ce = (1 + a1 + a2 + a3 + a4)*Ref;
```

Los resultados son los siguientes:

```
thi = [ 0.7256
        0.4703
        0.6402
       -0.8361
        0.8305
        0.7393
       -0.8200
       -0.7498
       -0.0000]
```

```
Ce = -1.4791e-018
```

Como siguiente paso se fija la matriz de covarianza inicial  $P$  de orden 9, la cual se escribe en MATLAB con  $NN=9$  así:

```
alfa=1000; Pi = alfa*eye(NN); % MATRIZ P INICIAL
```

En el siguiente paso se obtiene las nuevas mediciones de  $Y(k)$  y de  $U(k)$  para obtener el vector de información  $\psi^T$  usando el vector de estado  $\theta(k)$ :

$$Y(k) = \psi^T(k) \hat{\theta}(k) \quad (4.3)$$

Donde:

$$\psi^T(k) = \begin{bmatrix} Y(k-1) & Y(k-2) & Y(k-3) & Y(k-4) & \dots & \dots & U(k-1) & U(k-2) & U(k-3) & U(k-4) & 1 \end{bmatrix} \quad (4.4)$$

$$\hat{\theta}(k) = [-a_1 \quad -a_2 \quad -a_3 \quad -a_4 \quad b_1 \quad b_2 \quad b_3 \quad b_4 \quad C(k)]^T \quad (4.5)$$

Los que se escriben en MATLAB de la siguiente manera:

```
Y(t+4) = [Y(t+3) Y(t+2) Y(t+1) Y(t) ...
           U(t+3) U(t+2) U(t+1) U(t)] * [-a1  -a2  -a3  -a4 ...
           b1 b2 b3 b4]';
psi = [Y(t+3) Y(t+2) Y(t+1) Y(t) ...
       U(t+3) U(t+2) U(t+1) U(t) 1]';
```

El valor final del vector de información  $\psi$  calculado al finalizar el proceso de simulación es:

```
psi = 0.9959
      1.0013
      0.9983
      0.9962
      0.0769
      0.0766
      0.0770
      0.0768
      1.0000
```

El procedimiento de identificación o estimación de parámetros se continúa realizando los siguientes pasos:

$$\text{Cálculo de } \rho(k) = \max(1, \|\psi(k)\|) \quad \psi_n = \psi \rho(k)$$

Se determina la matriz  $N(k)$  aplicando la factorización Cholesky en:

$$N(k)N(k)^T = P(k) \quad (4.6)$$

La matriz diagonal de escalamiento  $S(k)$   $s_{jj}$ , se determina de tal modo que minimice el número condicional del producto  $S(k) N(k)$  siempre que  $s_{jj} = 1/n_{jj}$  donde cada  $n_{jj}$  se obtiene tomando el valor absoluto de la suma de los elementos de la fila  $j$  de  $N(k)$ . Finalmente se implementa el criterio para la estimación y para encontrar la nueva matriz de escalamiento  $N_{new}$ .

El código MATLAB para la estimación de parámetros es el siguiente:

```

rho = max(1,norm(psi));
psin = psi/rho;
Nn = chol(Pi'); % Nn'*Nn = Pi => Nn*N' = Pi'
S = inv(diag(Nn*ones(NN,1),0));
Ps = S*Pi*S;
psins = inv(S)*psin;
rt = 1 + psins'*Ps*psins;
lamb = 1 - (rt-sqrt(rt^2-
4*norm(Ps*psins)^2/trace(Ps)))/2;
e = Y(t+4)/rho - psin'*thi;
j = psins'*Ps*psins + lamb;
th = thi + e*inv(S)*Ps*psins/j;
Hns = Ps*psins/j;
Ps= (Ps - Hns*psins'*Ps)/lamb;
tt = abs(max(eig(Ps))/min(eig(Ps)) );
cmax = 100; cmin = 15;
if tt >= cmin,
    Pi = Ps; thi = th;
elseif tt >= cmax,
    Nnew = chol(Ps'); % Nnew*Nnew' = Ps'
    Snew = inv(diag(Nnew*ones(NN,1),0));
    Psnew = Snew*Ps*Snew;
    Pi = Psnew; thi = th;
end

```

### 4.3 Estimación de estados

El procedimiento de estimación de parámetros anterior proporciona el vector de parámetros estimado del proceso, a saber:

$$\hat{\theta}(k) = [-a_1 \quad -a_2 \quad -a_3 \quad -a_4 \quad b_1 \quad b_2 \quad b_3 \quad b_4 \quad C(k)]^T$$

Usando los elementos de dicho vector de parámetros, se reconstruye las matrices estimadas  $\hat{G}(k)$ ,  $\hat{H}(k)$  y  $\hat{C}(k)$  correspondientes a la representación canónica controlable:

$$\hat{G}(k) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_4(k) & -a_3(k) & -a_2(k) & -a_1(k) \end{bmatrix} \quad \hat{H}(k) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (4.7)$$

$$\hat{C}(k) = [b_4(k) \quad b_3(k) \quad b_2(k) \quad b_1(k)] \quad \hat{D}(k) = 0 \quad (4.8)$$

El resultado obtenido con MATLAB, considerando que  $GE = \hat{G}(k)$ ,  $HE = \hat{H}(k)$  y  $CE = \hat{C}(k)$  es el siguiente:

$$\begin{array}{r} GE = \\ \begin{array}{cccc} 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 1.0000 \\ -0.8361 & 0.6402 & 0.4703 & 0.7256 \end{array} \\ \\ HE = \\ \begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \end{array} \\ \\ CE = \\ \begin{array}{cccc} -0.7498 & -0.8200 & 0.7393 & 0.8305 \end{array} \\ \\ DE = 0 \end{array}$$

El vector de estado estimado  $\hat{x}(k)$  puede ser obtenido empleando el filtro de Kalman, con ecuación de observación:

$$\hat{x}(k) = \bar{x}(k) + K_o(k)[y(k) - \hat{C}(k)\bar{x}(k)] \quad (4.9)$$

y ecuación de actualización de estados:

$$\bar{x}(k+1) = \hat{G}(k)\hat{x}(k) + \hat{H}(k)u(k) \quad (4.10)$$



La matriz de ganancia del Filtro de Kalman  $K_0(k)$  se calcula de:

$$K_0(k) = P_0(k)\hat{C}^T(k)[\hat{C}(k)P_0(k)\hat{C}^T(k) + R_0]^{-1} \quad (4.11)$$

Donde  $P_0(k)$  es la matriz definida positiva, solución única de la siguiente ecuación matricial discreta asociada de Riccati:

$$P_0(k+1) = Q_0 + \hat{G}(k)P_0(k)\hat{G}(k) - \hat{G}^T(k)K_0(k)\hat{C}(k)P_0(k)\hat{G}(k) \quad (4.12)$$

Donde  $Q_0$  y  $R_0$  son matrices de covarianza definidas positivas correspondiente a los disturbios  $v$  y  $w$ , respectivamente.

El cálculo de la ganancia del estimador de Kalman es obtenido en MATLAB partiendo de las matrices estimadas de la siguiente manera:

```
% CALCULO DE LA GANANCIA F DEL ESTIMADOR DE KALMAN
F=dlqe(GE,HE,CE,Sigma,Xi);

F = 0.1142
    -0.1481
     0.1872
     0.9831
```

donde Sigma es igual a 1 y Xi es igual a 100. Parámetros obtenidos luego de realizar varias pruebas que finalmente permiten optimizar la ganancia del filtro de Kalman.

#### 4.4 Ley de control

El controlador proporcional óptimo es una matriz de ganancia  $K_x$  que permite formular la siguiente ley de control  $u = -K_x x(k)$ . Dicha matriz es el resultado de la minimización de la función de costo:

$$I = \sum_{k=0}^{\infty} [x^T(k)Qx(k) + u^2(k)R] \quad (4.13)$$

sujeto a la ecuación de restricción:

$$x(k+1) = Gx(k) + Hu(k)$$

donde la matriz  $Q=Q^T$  es semidefinida positiva y la matriz  $R$  es definida positiva. La ganancia  $K_x$  se obtiene de:

$$K_x = [R + H^T S H]^{-1} H^T S G \quad (4.14)$$

Donde  $S$  es la única matriz definida positiva solución de la siguiente ecuación matricial discreta asociada de Riccati:

$$0 = S - G^T S G + G^T S H K_x \quad (4.15)$$

Para mejorar el rendimiento del controlador proporcional es posible adicionar una acción integral al controlador. Definiendo la variable  $z(k)$  como la sumatoria del error  $[\bar{Y}(k) - Y(k)]$ :

$$z(k) = \sum_{i=0}^{k-1} [\bar{Y}(k) - Y(k)] = \sum_{i=0}^{k-1} [-y(k)]; \quad z(k+1) = \sum_{i=0}^k [-y(k)]$$

entonces:

$$z(k+1) = z(k) - y(k) = z(k) - Cx(k) \quad (4.16)$$

de modo que la representación en el espacio de estado del sistema aumentado toma la forma siguiente:

$$x^a(k+1) = G^a x^a(k) + H^a u(k) \quad (4.17)$$

$$y(k) = C^a x^a(k) \quad (4.18)$$

donde el superíndice  $a$  significa aumentado. El vector  $x^a$  y las matrices  $A^a$ ,  $B^a$ ,  $C^a$  poseen la forma:

$$x^a = \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} \quad (4.19)$$

$$G^a(k) = \begin{bmatrix} G & 0 \\ -C & I \end{bmatrix} \quad (4.20)$$

$$H^a(k) = \begin{bmatrix} H \\ 0 \end{bmatrix} \quad (4.21)$$

$$C^a = [C \quad 0] \quad (4.22)$$

Por consiguiente, el problema consiste en determinar la matriz de ganancia  $K^a$  como resultado de la minimización de la siguiente función de costo:

$$I^a = \sum_{k=0}^{\infty} [(x^a)^T(k) Q^a x^a(k) + u^2(k) R] \quad (4.23)$$

donde la matriz  $Q^a = [Q^a]^T$  es semidefinida positiva. La ganancia  $K^a$  se obtiene de:

$$K^a = \{R + [H^a]^T P^a H^a\}^{-1} [H^a]^T P^a G^a \quad (4.24)$$

En la ecuación anterior,  $P^a$  es la única matriz definida positiva, solución de la siguiente ecuación matricial discreta asociada de Riccati con los parámetros de ponderación de rendimiento  $R$  y  $Q^a$  :

$$P^a = Q^a + [G^a]^T P^a G^a - [G^a]^T P^a H^a K^a \quad (4.25)$$

Por consiguiente, la ley de control del controlador óptimo cuadrático proporcional integral viene a ser:

$$u(k) = -K^a x^a(k) \quad (4.26)$$

La matriz de ganancia  $K^a$  de nuestro sistema de control adaptivo es calculada en MATLAB hallando inicialmente las matrices aumentadas  $G^a$ ,  $H^a$  y  $C^a$  en forma consecutiva, y luego usando el comando `dlqr` obtenemos la matriz de ganancia  $K^a$  :

```
nn = 4; % nn: orden del proceso;
```

```

r = 1;      % r: número de salidas,
Ga = [GE zeros(nn,r);-CE eye(r,r)];
Ha = [HE;zeros(r,r)];
Ca = [CE zeros(r,r)];
Ka = dlqr(Ga, Ha, Qa, R);

```

Los resultados son:

```

Ga =      0      1.0000      0      0      0
      0      0      1.0000      0      0
      0      0      0      1.0000      0
    -0.8361  0.6402  0.4703  0.7256      0
      0.7498  0.8200 -0.7393 -0.8305  1.0000

Ha = 0
      0
      0
      1
      0

Ca =  -0.7498  -0.8200  0.7393  0.8305      0

Ka =  -0.3619  -0.0761  0.1227  0.4002  -0.0669

```

#### 4.5 Ley de control residual

Para el sistema de control en lazo cerrado se toma la señal  $\bar{Y}(k)$  igual a la referencia para todo  $k$ . Así se calcula la ley de control residual empleando la relación  $u(k) = U(k) - \bar{U}(k)$ , donde  $\bar{U}(k)$  se halla a partir de la siguiente ecuación:

$$\bar{U}(k) = \left[ \hat{b}_1(k) + \hat{b}_2(k) + \hat{b}_3(k) + \hat{b}_4(k) \right]^{-1} \left[ 1 + \hat{a}_1(k) + \hat{a}_2(k) + \hat{a}_3(k) + \hat{a}_4(k) \right] \dots \dots \bar{Y}(k) - \hat{C}(k) \quad (4.27)$$

Finalmente actualizamos la ley de control actual:  $U(k) = u(k) + \bar{U}(k)$

#### 4.6 Simulación del sistema de control adaptivo

El programa adaptivo.m, que simula el sistema de control adaptivo está listado en el anexo A.5, utiliza el método de los mínimos cuadrados recursivo mejorado

(MCRM). Se incluye una limitación a la señal de control a 1 V para no saturar el amplificador.

El tiempo de ascenso 10%-90%  $T_{r1}$  es de 3 segundos, considerando que alcanza los 0.2843 m/s (10%) a los 6 segundos y 0.9172 m/s (90%) a los 9 segundos.

El tiempo de estabilización  $T_s$  es de 20 segundos, pues alcanza la velocidad de 1.0138 m/s en ese tiempo, considerando un  $\delta = \pm 2\%$ .

El tiempo del pico  $T_p$  es alcanzado a los 14 segundos con un sobre nivel porcentual de 6.47%

La figura 4.2 muestra el resultado del programa. Consta de dos gráficas. El gráfico superior muestra la velocidad de la locomotora con el flanco de subida para el arranque y el flanco de bajada para la parada. La gráfica inferior muestra la señal de control  $u$ .

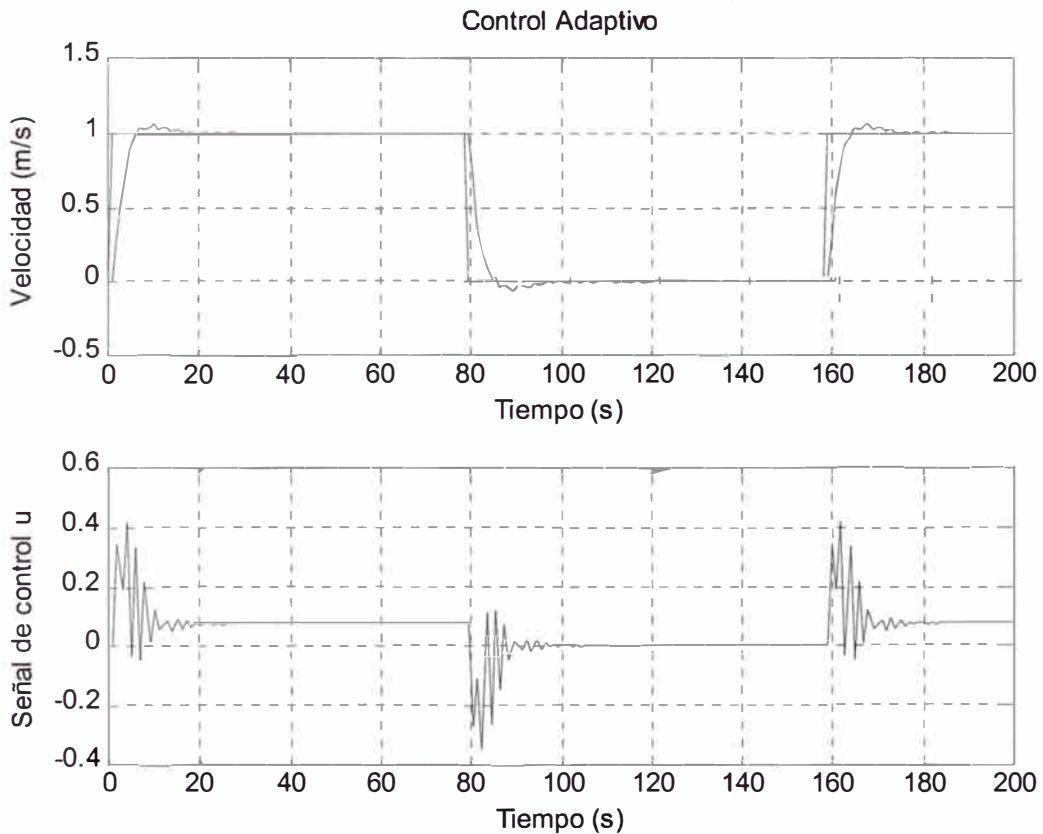


Fig. 4.2 Simulación del sistema de control adaptivo

## **CAPÍTULO V**

### **SISTEMA DE CONTROL PREDICTIVO**

#### **5.1 Diseño del controlador predictivo**

El control predictivo es un método que usa el modelo del proceso para predecir las acciones de control y de la salida futuras del proceso controlado, usando para ello parámetros de peso, filtraje, horizontes de predicción y horizontes de control. La estrategia comprende las siguientes partes:

- Proceso de predicción a través de un modelo del proceso
- La especificación de una trayectoria de referencia
- La estructuración de la futura ley de control
- La definición de la función de costo y sus restricciones
- Cálculo del escenario de control óptimo

El control predictivo pronostica la salida de la planta en un escenario de tiempo de duración  $N_2$  (el horizonte de tiempo  $N_2$ ). Tal predicción depende de las salidas y entradas pasadas, pero también del escenario del control futuro. La obtención del modelo que refleje lo más fielmente posible la evolución dinámica de la planta es la clave del éxito en las aplicaciones. La notación usada para deducir los principios del control predictivo es la siguiente:

- $t$ , representa el tiempo discreto ( $t=0,1,2,\dots$ )
- $u(t)$ , denota la entrada al proceso y constituye la variable manipulada o señal de control

- $y(t)$ , es la salida del proceso y constituye la variable controlada
- $w(t)$ , representa la trayectoria deseada o “setpoint”
- $r(t)$ , denota la trayectoria de referencia
- $u(t+k/t)$ , indica los valores futuros de la entrada en el tiempo  $t+k$  para un tiempo dado  $t$
- $y(t+k/t)$ , denota los valores futuros de la salida basada en las mediciones pasadas disponibles en el tiempo  $t$ :

$$\{y(t), y(t-1), \dots, u(t-1), u(t-2), \dots\}$$

y en los valores futuros de la entrada postulados en el tiempo  $t$ :

$$\{u(t/t), u(t+1/t), \dots\}$$

La figura 5.1 muestra el diagrama de bloques que visualiza la estrategia del control predictivo.

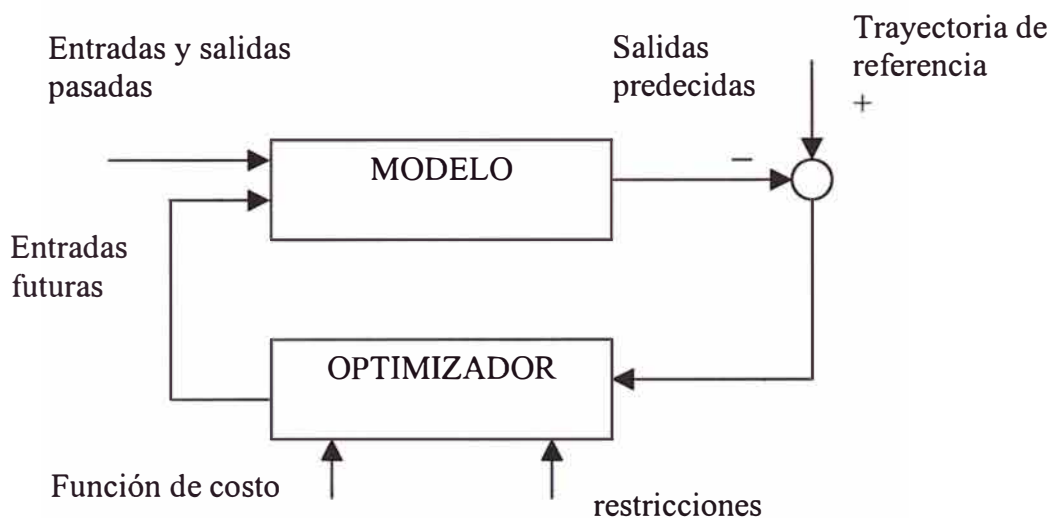


Fig. 5.1. Estrategia del control predictivo

El objetivo de controlador predictivo es determinar el vector de control  $\mathbf{u}(t+k/t)$ , para  $k=0, \dots, N_2-1$ , que minimice la siguiente función de costo:

$$J = \sum_{k=N_1}^{N_2} [r(t+k/t) - y(t+k/t)]^2 + \lambda \sum_{k=0}^{N_u-1} [\Delta u(k+t/t)]^2 \quad (5.1)$$

Donde:

$$\Delta u(t+k/t) = u(t+k/t) - u(t+k-1/t) \quad (5.2)$$

con  $\Delta u(t+k/t)=0$  para  $k \geq N_u$ . La trayectoria de referencia está representada por la ecuación :

$$r(t+k/t) = \alpha r(t+k-1/t) + (1-\alpha)w(t+k/t) \quad (5.3)$$

evaluada para  $k=1, \dots, N_2$ , con  $r(t/t)=y(t)$ .

Los parámetros de diseño son:

- $N_1$ : Horizonte de predicción mínimo.
- $N_2$ : Horizonte de predicción máximo, por defecto podemos considerar  $N_2 = N_1+1, \dots, N_1+10$
- $N_u$ : Horizonte de control; por defecto considerar  $N_u = 1$
- $N_1, \dots, N_2$  : Horizonte de coincidencia; por defecto considerar  $N_1$  (tiempo muerto)
- $\lambda$ : Parámetro de ponderación; por defecto considerar cero
- $\alpha$ : Parámetro de filtraje; por defecto considerar cero. En nuestro caso el valor es de 0.3

Si la trayectoria deseada  $w(t)$  se programa previamente, entonces los valores futuros  $w(t+k)$  se pueden emplear para determinar  $w(t+k/t)$ . Es decir,  $w(t+k/t) = w(t+k)$  para  $k=1, \dots, N_2$ . Si no deseamos una estrategia de control predictivo en avance, el valor actual  $w(t)$  se puede usar como un valor de predicción  $w(t+k/t) = w(t)$  para  $k=1, \dots, N_2$



## 5.2 Determinación del modelo CARIMA del proceso

Para el modelo del proceso que se indicó en la figura 5.1 se usará el modelo CARIMA, del inglés “Controlled AutoRegressive Integrated Moving Average” que significa Promedio Móvil Integrado Autoregresivo Controlado. Este es un modelo lineal básico que puede ser representado por:

$$A(z^{-1})y(t) = B(z^{-1})u(t) + \frac{C(z^{-1})}{D(z^{-1})}e(t) \quad (5.4)$$

Por ser una planta de orden 4 en nuestro caso:

$$A(z^{-1}) = 1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + a_4z^{-4} \quad (5.5)$$

$$B(z^{-1}) = b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + b_4z^{-4} \quad (5.6)$$

El origen del término CARIMA se debe a que la ecuación de diferencias del modelo del proceso presenta tres componentes, los cuales para nuestro caso son:

\* Componente de salida autoregresiva

$$y(k) = -a_1y(k-1) - a_2y(k-2) - a_3y(k-3) - a_4y(k-4)$$

\* Componente promedio temporal de control:

$$+ b_0u(k-1) + b_1u(k-2) + b_2u(k-3) + b_3u(k-4) + b_4u(k-5)$$

\* Componente temporal integrado del error  $e(t)$ :

$$+ \frac{e(k) + c_1e(k-1) + c_2e(k-2) + c_3e(k-3) + c_4e(k-4)}{1 - z^{-1}}$$

En este caso consideraremos que el error  $e(t)$  es despreciable, por lo tanto la ecuación diferencia que describe el modelo de la planta es:

$$y(k) = -a_1y(k-1) - a_2y(k-2) - a_3y(k-3) - a_4y(k-4) \dots \dots \dots + b_0u(k-1) + b_1u(k-2) + b_2u(k-3) + b_3u(k-4) + b_4u(k-5) \quad (5.7)$$

Los valores de  $a_1, a_2, a_3, a_4, b_0, b_1, b_2, b_3$  y  $b_4$  se obtienen desde  $G$  y  $H$ , tal como se mostró en el capítulo IV, en la sección 4.2. Adicionalmente se muestra la ecuación diferencia en código MATLAB:

```
% MODELO DEL PROCESO EN TIEMPO DISCRETO
tm=1;      % TIEMPO DE MUESTREO
[G,H,C,D]=c2dm(A,B,C,D,tm,'zoh');
[numd,dend]=ss2tf(G,H,C,D);
a1=dend(2); a2=dend(3); a3=dend(4); a4=dend(5);
b1=numd(2); b2=numd(3); b3=numd(4); b4=numd(5);
```

El resultado obtenido es:

```
num = [0 0.2976 0.4931 0.4746 0.2613]
den = [1.0000 -0.7256 -0.4703 -0.6402 0.8361]
```

Por lo tanto los valores de  $a_1, a_2, a_3, a_4, b_1, b_2, b_3$  y  $b_4$  son:

```
a1= -0.7256 a2= -0.4703 a3=-0.6402 a4=0.8361
b1= 0.2976 b2= 0.4931 b3=0.4746 b4=0.2613

%PROCESO
y(t+1)=-a1*y(t)-a2*y(t-1)-a3*y(t-2)-a4*y(t-3)...
      +b1*u(t)+b2*u(t-1)+b3*u(t-2)+b4*u(t-3);
```

### 5.3 Cálculo de la respuesta libre del proceso

La respuesta futura o pronosticada es el resultado de dos efectos:

$$y(t+k/t) = y_{libre}(t+k/t) + y_{forz}(t+k/t) \quad (5.8)$$

donde  $y_{libre}$  es la respuesta que se obtiene cuando no se aplica ninguna ley de control y aparece como consecuencia de dos factores:

\* El efecto debido a las señales de control pasadas  $u(t-1), u(t-2), \dots$

\* El efecto debido al escenario de control futuro por defecto se tiene  $u(t/t)=u(t-1), u(t+1/t)=u(t-1), u(t+2/t)=u(t-1), \dots$  esto es  $\Delta u(t/t)=\Delta u(t+1/t)=\dots=\Delta u(t+N_u-1/t)=0$

\* El efecto debido a las perturbaciones futuras del ruido  $n(t+k/t)$

La componente  $y_{\text{forz}}(t+k/t)$  aparece debido al resultado de dos efectos. El primero causado por la acción del control futuro; es decir, causado por las señales  $\Delta u(t/t), \Delta u(t+1/t), \dots, \Delta u(t+N_u-1/t)$ . El segundo efecto se debe a las respuestas del proceso a una secuencia de entradas escalón.

El siguiente programa determina recursivamente la trayectoria deseada  $W(t)$ , la trayectoria de referencia  $r(t)$  y al respuesta libre  $y_{\text{libre}}$ , la cual será denotada por  $y_f$ .

```
%CONDICIONES INICIALES
for k=1:5
    yf(k)=0; y(k)=0; r(k)=0; u(k)=0;
end
alf=0.3; lambda=0; MM=400;
%LAZO DE CONTROL
for t=5:MM+4;
%TRAYECTORIA DESEADA, REFERENCIA r Y RESPUESTA LIBRE yf
    for k=N1:N2,
        W(t+k)=0.5+0.5*sign(sin(0.02*t));
        r(t+k)=alf*r(t+k-1)+(1-alf)*W(t+k);
        yf(t+k)=-a1*yf(t+k-1)-a2*yf(t+k-2)-a3*yf(t+k-3)...
            -a4*yf(t+k-4)+(b1+b2+b3+b4)*u(t-1);
    end
```

#### 5.4 Cálculo de la respuesta del proceso al escalón

Los coeficientes  $g_k$  de la respuesta al escalón se determinan partiendo de la ecuación de diferencias del proceso señalada por la ecuación (5.4). Teniendo en cuenta que una entrada escalón es  $u(k)=1$  para todo  $k \geq 0$ , entonces  $y(k) = g(k)$  para todo  $k \geq 0$ . Luego los coeficientes  $g(k)$  se determinan de:

$$g(k) = -a_1g(k-1) - a_2g(k-2) - a_3g(k-3) - a_4g(k-4) \dots \dots \dots + b_0u(k-1) + b_1u(k-2) + b_2u(k-3) + b_3u(k-4) + b_4u(k-5) \quad (5.9)$$

De donde obtenemos reemplazando desde  $k=0$  hasta 10:

$$g(0) = 0, \quad g(1) = b_1$$

$$g(2) = -a_1b_1 + b_1 + b_2$$

$$g(3) = -a_1g(2) - a_2g(1) + b_1 + b_2 + b_3$$

$$g(4) = -a_1g(3) - a_2g(2) - a_3g(1) + b_1 + b_2 + b_3 + b_4$$

Para  $k$  mayor o igual que 5 y hasta el horizonte de predicción máximo  $N_2=10$

$$g(k) = -a_1g(k-1) - a_2g(k-2) - a_3g(k-3) - a_4g(k-4) + b_1 + b_2 + b_3 + b_4$$

En código MATLAB, el cálculo se efectúa como sigue:

```
% HORIZONTES
N1=1; N2=10; Nu=1;
% RESPUESTA AL ESCALON
g(1)=b1; g(2)=-a1*b1+b1+b2;
g(3)=-a1*g(2)-a2*g(1)+b1+b2+b3;
g(4)=-a1*g(3)-a2*g(2)-a3*g(1)+b1+b2+b3+b4;
for k=5:N2
    g(k)=-a1*g(k-1)-a2*g(k-2)-a3*g(k-3)-a4*g(k-4)...
        +b1+b2+b3+b4;
end
```

Los resultados de la evaluación de  $g$  son los siguientes:

```
g =

Columns 1 through 7
    0.2976    1.0066    2.1357    3.7403    5.6406
  7.9042   10.5237

Columns 8 through 10
  13.3640   16.5171   19.9253
```

## 5.5 Determinación de la ley de control

La ley de control a emplear es del tipo escalar ( $Nu=1$ ) con  $N_1=1$  y  $N_2=12$ . La forma de la ley de control es:

$$\Delta u = \frac{\sum_{k=N_1}^{N_2} g(k) [r(t+k/t) - y_{libre}(t+k/t)]}{\sum_{k=N_1}^{N_2} g^2(k) + \lambda} \quad (5.10)$$

El código MATLAB se expresa así:

```

%LEY DE CONTROL
du(t)=(g(1)*(r(t+1)-yf(t+1))+g(6)*(r(t+6)-yf(t+6))+...
      g(2)*(r(t+2)-yf(t+2))+g(7)*(r(t+7)-yf(t+7))+...
      g(3)*(r(t+3)-yf(t+3))+g(8)*(r(t+8)-yf(t+8))+...
      g(4)*(r(t+4)-yf(t+4))+g(9)*(r(t+9)-yf(t+9))+...
      g(5)*(r(t+5)-yf(t+5))+g(10)*(r(t+10)-
yf(t+10)))/...
      (g(1)^2+g(2)^2+g(3)^2+g(4)^2+g(5)^2+g(6)^2+...
      g(7)^2+g(8)^2+g(9)^2+g(10)^2+lambda);
u(t)=u(t-1)+du(t);

```

### 5.6 Simulación del sistema de control predictivo

El programa predictivo.m que simula el sistema de control predictivo está listado en el anexo A6. Estamos empleando restricción en la fuerza de control en el rango de  $-1$  hasta  $+1$  para evitar la saturación del amplificador de potencia.

El tiempo de ascenso 10-90%  $T_{r1}$  es de 12 segundos, considerando que alcanza los 0.1325 m/s (10%) a los 8 segundos y 0.9189 m/s (90%) a los 20 segundos.

El tiempo de estabilización  $T_s$  es de 33 segundos, pues alcanza la velocidad de 1.0194 m/s en ese tiempo, considerando un  $\delta = \pm 2\%$  para una velocidad estabilizada de 1 m/s.

El tiempo de pico  $T_p$  es alcanzado a los 29 segundos con un sobre nivel porcentual de 2.37%

La figura 5.2 muestra el resultado de la simulación mostrando la velocidad de salida así como la fuerza de control tanto para el arranque como para la parada.

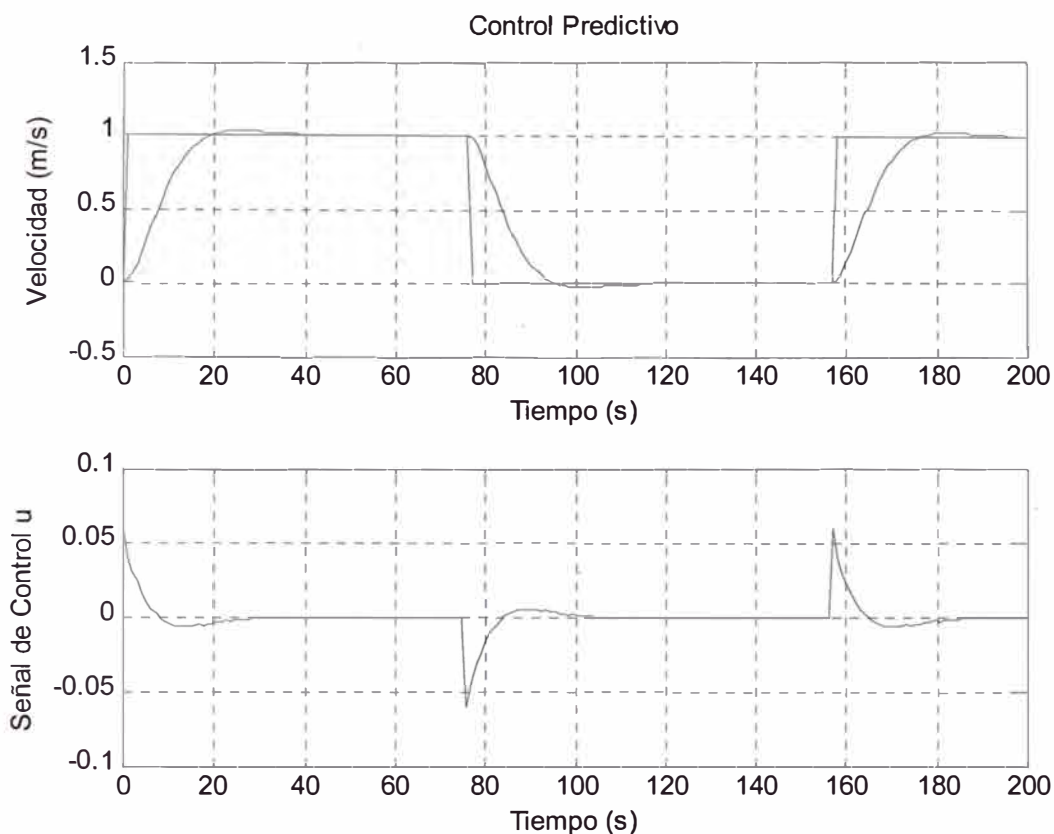


Fig. 5.2 Simulación del sistema de control predictivo

Si se considera una restricción a la salida  $y(t)=r(t)$ , se obtienen los siguientes resultados:

El tiempo de ascenso 10%-90%  $T_{r1}$  es de 1 segundo, considerando que alcanza los 0.7000 m/s (10%) a los 6 segundos y 0.9100 m/s (90%) a los 7 segundos.

El tiempo de estabilización  $T_s$  es de 9 segundos, pues alcanza la velocidad de 0.9919 m/s en ese tiempo, considerando un  $\delta = \pm 2\%$  para una velocidad estabilizada de 1 m/s.

No tiene sobrepico.

La figura 5.3 muestra el resultado de la simulación mostrando la velocidad de salida así como la fuerza de control tanto para el arranque como para la parada.

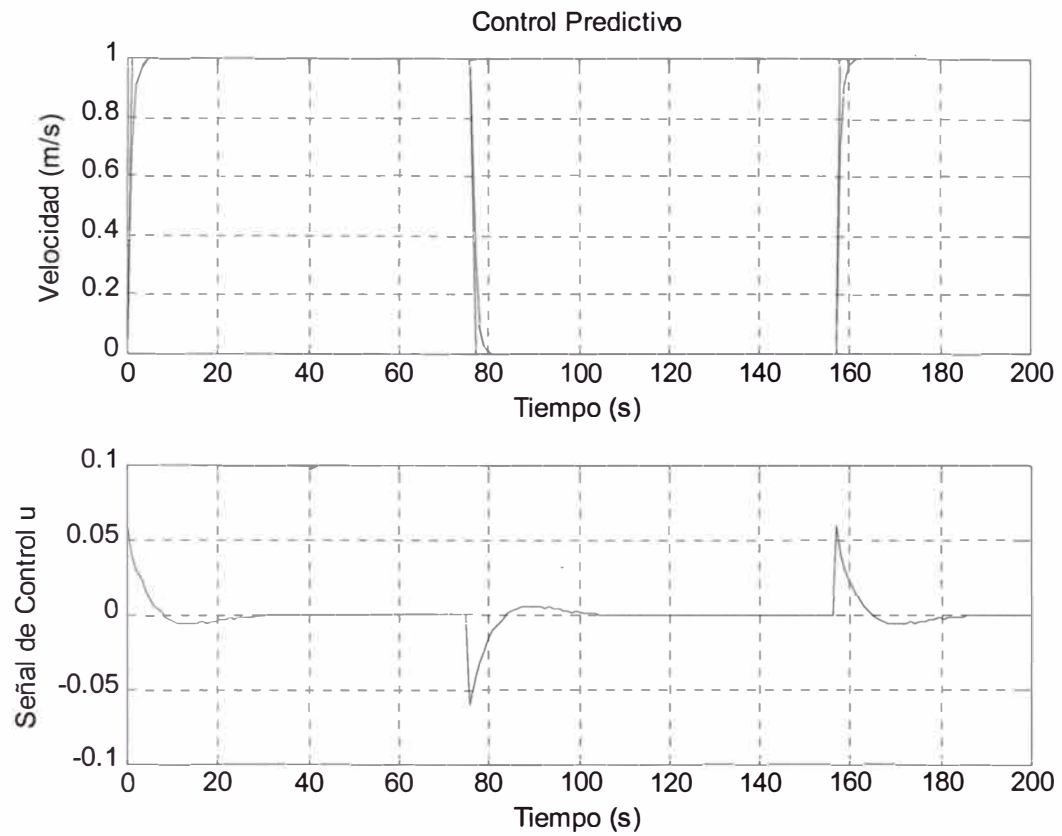


Fig. 5.3 Simulación del sistema de control predictivo con restricción a la salida

## **CAPITULO VI COMPARACIÓN DE RESULTADOS**

Se ha utilizado los siguientes parámetros con el mismo valor en las cuatro simulaciones para que las comparaciones sean válidas:

Tiempo de muestreo: 1 segundo

Señal de referencia:  $0.5 + 0.5 \text{ sign}(\sin(0.04t))$

La señal de referencia simula una onda cuadrada con la finalidad de mostrar el arranque y la parada del tren. El flanco de subida simula el arranque y el flanco de bajada simula la parada.

### **6.1 Tiempo de ejecución**

Medimos el tiempo en que la Computadora Personal PC que utilizamos tardó en la ejecución de cada uno de los Sistemas de Control, sin incluir las gráficas, con el siguiente resultado:

- Modelo de la Planta: 0.66 segundos
- Control Óptimo: 0.99 segundos
- Control Adaptivo: 14.45 segundos
- Control Predictivo: 0.77 segundos
- Control Predictivo con restricción a la salida: 0.77 segundos



## 6.2 Tiempo de ascenso

El tiempo de ascenso 10%-90%,  $T_{r1}$  es el tiempo que tarda el sistema en llegar del 10% del valor final al 90% del valor final. Los valores obtenidos son los que se indican a continuación:

- Modelo de la Planta: 28 segundos
- Control Óptimo: 5 segundos
- Control Adaptivo: 3 segundos
- Control Predictivo: 12 segundos
- Control Predictivo con restricción a la salida 1 segundo

## 6.3 Tiempo de estabilización

El tiempo en el que el tren alcanza la velocidad final deseada para cada uno de los Sistemas de Control es el siguiente:

- Modelo de la Planta: 50 segundos
- Control Óptimo: 9 segundos
- Control Adaptivo: 20 segundos
- Control Predictivo: 33 segundos
- Control Predictivo con restricción a la salida 9 segundos

## 6.4 Sobrepico

El sobrepico se presenta en el caso de control adaptivo y en el caso de control predictivo sin señal de restricción a la salida, así:

- Control Adaptivo: 6.47% a los 14 segundos
- Control Predictivo: 2.37% a los 29 segundos

## 6.5 Seguimiento a la señal de referencia

Control Óptimo: La velocidad de salida sigue a la señal de referencia con un error en estado estacionario menor a 2 %.

Control Adaptivo: La velocidad de salida sigue a la señal de referencia con un error en estado estacionario menor a 2%.

Control Predictivo: Observamos que la velocidad de salida sigue a la señal de referencia con error en estado estacionario menor a 2 %.

Control Predictivo con restricción a las salida: Observamos que la velocidad de salida sigue a la señal de referencia con error en estado estacionario igual a cero a partir del 14° segundo.

## CONCLUSIONES

Considerando los resultados obtenidos para cada una de las estrategias de control tratadas en este trabajo, se puede sacar las siguientes conclusiones:

1. El sistema de control óptimo tiene uno de los tiempos de estabilización más cortos y es el más simple de implementar. Su señal de control produce oscilaciones amortiguadas y tiende hacia un valor cercano a cero en tiempo estable, es decir hasta lograr la estabilización de la velocidad del tren.
2. El sistema de control adaptivo es el más lento en ejecutarse de los sistemas de control debido a los numerosos cálculos necesarios en su algoritmo de control. La señal de control es parecida a la del control óptimo, también tienden hacia un valor cercano a cero en tiempo estable.
3. El sistema de control predictivo sin restricción a la salida tiene el mayor tiempo de ascenso y el mayor tiempo de estabilización. Sin embargo es uno de los más veloces en ejecutarse. Su señal de control no tiene oscilaciones y tiende hacia cero una vez alcanzado el estado estable.
4. El sistema de control predictivo con restricción a la salida tiene el menor tiempo de ascenso y el menor tiempo de estabilización. Este sistema de control es el mejor de los tres sistemas comparados pues el tiempo de ascenso y tiempo de estabilización es el menor a todos los demás y el error en estado estacionario es

cero. La señal de control es similar a la de control óptimo sin oscilaciones, y tiende a cero en tiempo estable.

5. La utilización de cualquiera de los sistemas de control se justifica plenamente ya que mejora el desempeño del sistema, tal como el tiempo de ascenso en por lo menos a la mitad en el peor caso y casi en treinta veces menos en el mejor caso y el tiempo de estabilización en casi la mitad en el peor caso y a la quinta parte en el mejor sistema de control.
6. Considerando que en el presente trabajo el frenado se realiza solamente por fricción y teniendo en cuenta que el frenado es más importante que el arranque en un tren, estimamos que este informe de suficiencia constituye una base para el planteamiento de un sistema de control para el frenado dinámico de un tren.

## **ANEXO A**

### **LISTADO DE LOS PROGRAMAS**

A1. Simulación del sistema en lazo abierto	60
A2. Modelo del tren en Simulink	63
A3. Gráfica de respuestas en Simulink	76
A4. Simulación del sistema con control óptimo	77
A5. Simulación del sistema con control adaptivo	80
A6. Simulación del sistema con control predictivo	85

## ANEXO A

### LISTADO DE LOS PROGRAMAS

#### A1. Simulación del sistema en lazo abierto

```
% modelo.m
% Sistema de control de arranque y parada de un tren
% Simulación del sistema en lazo abierto (sin ningún
control)
%
%
clear all;
ti=clock;
% Parámetros del proceso
M1=126000;      % Masa de la locomotora (kg)
M2=120000;     % Masa del vagón (kg)
k=1000000;     % Coeficiente de rigidez del resorte (N/m)
B1=10000;     % Coeficiente de fricción estática de
              % la locomotora (N-s/m)
B2=10000;     % Coeficiente de fricción estática del
              % vagón (N-s/m)
BR=1000;      % Coeficiente de fricción viscosa del
              % resorte (N-s/m)
F=260000;     % Fuerza (N)
% Modelo del tren en tiempo continuo
A=[0 1 0 0;
   -k/M1  -(B1+BR)/M1  k/M1  BR/M1;
   0  0  0  1;
   k/M2  BR/M2  -k/M2  -(B2+BR)/M2];
B=[ 0;
   F/M1;
   0;
   0];
C=[0 1 0 0];
D=[0];
%Controlabilidad del Proceso
M=[B A*B A^2*B A^3*B];
rangoM=rank(M); %rangoM = 4 => Es completamente
controlable
N=[C
```

```

C*A
C*A^2
C*A^3];
rangoN=rank(N); %rangoNd = 3 => No es completamente
Observable
rAB=rank(ctrb(A,B)); %rAB=4 Completamente controlable
rAC=rank(observ(A,C)); %rAC=3 No es completamente
observable
%Calculo de los valores propios
eigA=eig(A);
% Conversión al tiempo discreto
T=1; %Tiempo de muestreo
[G,H,C,D]=c2dm(A,B,C,D,T,'zoh');
%Matriz de Controlabilidad
Md=[H G*H G^2*H G^3*H];
rangoMD=rank(Md); %rangoMD = 4 => Es completamente
controlable
%Matriz de Observabilidad
Nd=[C
C*G
C*G^2
C*G^3];
rangoNd=rank(Nd);
rGC=rank(observ(G,C)); %rGC = 3 =>No es completamente
Observable
eigG=eig(G); %Cálculo de los valores propios
etime(clock, ti)
%Respuesta al escalon en tiempo continuo y discreto
figure(1);
TT=0:1:200;
[Y,X]=step(A,B,C,D,1,TT);
[YY,XX]=dstep(G,H,C,D,1,201);
subplot(211)
plot(TT,Y,'k'); grid on;
title('Planta en lazo abierto')
xlabel('Tiempo (s)')
ylabel('Velocidad (m/s) - t continuo')
subplot(212)
plot(TT,YY,'k:'); grid on;
xlabel('Tiempo (s)')
ylabel('Velocidad (m/s)- t discreto')
%
% Resultados del Proceso
%
%A =
% 0 1.0000 0 0
% -7.9365 -0.0873 7.9365 0.0079
% 0 0 0 1.0000
% 8.3333 0.0083 -8.3333 -0.0917

```

```

%M =
%      0      2.0635     -0.1801     -16.3611
%     2.0635     -0.1801     -16.3611      3.1310
%      0      0      0.0172      17.1927
%      0      0.0172      17.1927     -3.3569
%
%rangoM =
%      4  => Es completamente controlable
%
%N =
%      0      1.0000      0      0
%    -7.9365     -0.0873      7.9365      0.0079
%     0.7590     -7.9288     -0.7590      7.9351
%   129.0529      1.5173   -129.0529     -1.5493
%
%rangoN =
%      3  => No es completamente Observable
%
%eigA =
%   -0.0488 + 4.0333i
%   -0.0488 - 4.0333i
%   -0.0813
%   -0.0000
%
%G =
%     0.2149      0.4025      0.7851      0.5581
%     1.4567      0.1844     -1.4567      0.7371
%     0.8220      0.5860      0.1780      0.3742
%    -1.5326      0.7740      1.5326      0.1483
%
%H =
%     0.6141
%     0.8305
%     0.4100
%     1.2093
%
%Md =
%     0.6141      1.4631      2.3298      3.3110
%     0.8305      1.3419      0.5443      0.8080
%     0.4100      1.5170      2.4497      3.1180
%     1.2093      0.5093      1.1969      0.7825
%
%rangoMD = 4 => Es completamente controlable
%
%Nd =
%      0      1.0000      0      0
%     1.4567      0.1844     -1.4567      0.7371
%    -1.7455      0.3372      1.7455      0.5132
%     0.7644      0.7798     -0.7644      0.0036

```



```

%
% rangoNd = 3 => No es completamente Observable
%
% eigG =
% -0.5982 + 0.7411i
% -0.5982 - 0.7411i
% 1.0000
% 0.9219
%
%
% Velocidad final = 13.0000 m/s
% Tiempo de ejecución 0.66 segundos
% Tiempo de ascenso 10%-90% = 28 segundos [30 (11.7296
m/s) - 3 (2.1724 m/s)]
% Tiempo de estabilización = 50 segundos (12.7645 m/s -
error menor a 2%)
% No tiene sobrepico

```

## A2. Modelo del tren en SIMULINK

```

Model {
  Name          "modelotren"
  Version       3.00
  SimParamPage  "Solver"
  SampleTimeColors off
  InvariantConstants off
  WideVectorLines off
  ShowLineWidths off
  ShowPortDataTypes off
  StartTime     "0.0"
  StopTime      "200"
  SolverMode    "SingleTasking"
  Solver        "ode45"
  RelTol        "1e-3"
  AbsTol        "1e-6"
  Refine        "1"
  MaxStep       "auto"
  InitialStep   "auto"
  FixedStep     "auto"
  MaxOrder      5
  OutputOption  "RefineOutputTimes"
  OutputTimes   "[]"
  LoadExternalInput off
  ExternalInput "[t, u]"
  SaveTime      on
  TimeSaveName  "tout"

```

```

SaveState          off
StateSaveName      "xout"
SaveOutput         on
OutputSaveName     "yout"
LoadInitialState   off
InitialState       "xInitial"
SaveFinalState     off
FinalStateName     "xFinal"
SaveFormat         "Matrix"
LimitMaxRows       off
MaxRows            "1000"
Decimation         "1"
AlgebraicLoopMsg   "warning"
MinStepSizeMsg     "warning"
UnconnectedInputMsg "warning"
UnconnectedOutputMsg "warning"
UnconnectedLineMsg "warning"
InheritedTsInSrcMsg "warning"
IntegerOverflowMsg "warning"
UnnecessaryDatatypeConvMsg "none"
Int32ToFloatConvMsg "warning"
SignalLabelMismatchMsg "none"
ConsistencyChecking "off"
ZeroCross          on
SimulationMode     "normal"
BlockDataTips      on
BlockParametersDataTip on
BlockAttributesDataTip off
BlockPortWidthsDataTip off
BlockDescriptionStringDataTip off
BlockMaskParametersDataTip off
ToolBar            off
StatusBar          off
BrowserShowLibraryLinks off
BrowserLookUnderMasks off
OptimizeBlockIOStorage on
BufferReuse        on
BooleanDataType    off
RTWSystemTargetFile "grt.tlc"
RTWInlineParameters off
RTWRetainRTWFile   off
RTWTemplateMakefile "grt default tmf"
RTWMakeCommand     "make rtw"
RTWGenerateCodeOnly off
ExtModeMexFile     "ext comm"
ExtModeBatchMode   off
ExtModeTrigType    "manual"
ExtModeTrigMode    "oneshot"
ExtModeTrigPort    "1"

```

```

ExtModeTrigElement      "any"
ExtModeTrigDuration     1000
ExtModeTrigHoldOff     0
ExtModeTrigDelay        0
ExtModeTrigDirection   "rising"
ExtModeTrigLevel        0
ExtModeArchiveMode     "off"
ExtModeAutoIncOneShot  off
ExtModeIncDirWhenArm   off
ExtModeAddSuffixToVar  off
ExtModeWriteAllDataToWs off
ExtModeArmWhenConnect  off
Created                 "Mon Mar 11 18:59:18 2002"
UpdateHistory           "UpdateHistoryNever"
ModifiedByFormat        "%<Auto>"
ModifiedDateFormat     "%<Auto>"
LastModifiedDate       "Tue Dec 03 22:17:24 2002"
ModelVersionFormat     "1.%<AutoIncrement:19>"
ConfigurationManager   "none"
BlockDefaults {
  Orientation           "right"
  ForegroundColor       "black"
  BackgroundColor      "white"
  DropShadow            off
  NamePlacement         "normal"
  FontName              "Helvetica"
  FontSize              10
  FontWeight            "normal"
  FontAngle             "normal"
  ShowName              on
}
AnnotationDefaults {
  HorizontalAlignment   "center"
  VerticalAlignment     "middle"
  ForegroundColor      "black"
  BackgroundColor      "white"
  DropShadow            off
  FontName              "Helvetica"
  FontSize              10
  FontWeight            "normal"
  FontAngle             "normal"
}
LineDefaults {
  FontName              "Helvetica"
  FontSize              9
  FontWeight            "normal"
  FontAngle             "normal"
}
System {

```

```

Name          "modelotren"
Location      [102, 250, 684, 604]
Open         on
ModelBrowserVisibility off
ModelBrowserWidth 212
ScreenColor   "white"
PaperOrientation "landscape"
PaperPositionMode "auto"
PaperType     "usletter"
PaperUnits    "inches"
ZoomFactor    "100"
AutoZoom      on
ReportName    "simulink-default.rpt"
Block {
  BlockType    Step
  Name         "Escalón"
  Position     [25, 15, 55, 45]
  Time        "100"
  Before      "1"
  After       "0"
  SampleTime  "0"
}
Block {
  BlockType    Gain
  Name         "Fricción loc"
  Position     [25, 83, 85, 127]
  Gain        "B1+BR"
  SaturateOnIntegerOverflow on
}
Block {
  BlockType    Gain
  Name         "Fricción_vagón"
  Position     [30, 243, 90, 287]
  Gain        "B2+BR"
  SaturateOnIntegerOverflow on
}
Block {
  BlockType    Gain
  Name         "Fuerza"
  Position     [80, 45, 110, 75]
  Gain        "F"
  SaturateOnIntegerOverflow on
}
Block {
  BlockType    Sum
  Name         "Suma_F1"
  Ports        [4, 1, 0, 0, 0]
  Position     [125, 89, 165, 131]
  IconShape    "rectangular"
}

```

```

    Inputs          "+---+"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType          Sum
    Name                "Suma_F2"
    Ports              [3, 1, 0, 0, 0]
    Position           [125, 219, 155, 251]
    IconShape          "rectangular"
    Inputs             "++-"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType          Scope
    Name                "Vista escalón"
    Ports              [1, 0, 0, 0, 0]
    Position           [265, 15, 295, 45]
    Floating            off
    Location           [18, 116, 342, 356]
    Open                on
    NumInputPorts      "1"
    TickLabels         "on"
    ZoomMode           "xonly"
    List {
ListType          AxesTitles
axes1             "%<SignalLabel>"
    }
    Grid                "on"
    TimeRange          "100"
    YMin                "0"
    YMax                "1"
    SaveToWorkspace    off
    SaveName            "ScopeData"
    DataFormat          "Matrix"
    LimitMaxRows       on
    MaxRows             "5000"
    Decimation          "1"
    SampleInput        off
    SampleTime         "0"
}
Block {
    BlockType          Scope
    Name                "Vista_v1"
    Ports              [1, 0, 0, 0, 0]
    Position           [440, 35, 470, 65]
    Floating            off
    Location           [684, 73, 1008, 312]
    Open                on
    NumInputPorts      "1"
}

```

```

    TickLabels      "on"
    ZoomMode        "yonly"
    List {
ListType      AxesTitles
axes1        "%<SignalLabel>"
    }
    Grid            "on"
    TimeRange       "auto"
    YMin            "-5"
    YMax            "5"
    SaveToWorkspace off
    SaveName        "ScopeData"
    DataFormat      "Matrix"
    LimitMaxRows    on
    MaxRows         "5000"
    Decimation      "1"
    SampleInput     off
    SampleTime      "0"
}
Block {
    BlockType      Scope
    Name           "Vista_v2"
    Ports          [1, 0, 0, 0, 0]
    Position       [440, 155, 470, 185]
    Floating       off
    Location       [678, 339, 1002, 578]
    Open           on
    NumInputPorts  "1"
    TickLabels     "on"
    ZoomMode       "on"
    List {
ListType      AxesTitles
axes1        "%<SignalLabel>"
    }
    Grid            "on"
    TimeRange       "auto"
    YMin            "-5"
    YMax            "5"
    SaveToWorkspace off
    SaveName        "ScopeData"
    DataFormat      "Matrix"
    LimitMaxRows    on
    MaxRows         "5000"
    Decimation      "1"
    SampleInput     off
    SampleTime      "0"
}
Block {
    BlockType      Scope

```

```

Name          "Vista_x1"
Ports         [1, 0, 0, 0, 0]
Position      [440, 95, 470, 125]
Floating      off
Location      [390, 443, 714, 682]
Open          on
NumInputPorts "1"
TickLabels    "on"
ZoomMode      "on"
List {
ListType      AxesTitles
axes1         "%<SignalLabel>"
}
Grid          "on"
TimeRange     "auto"
YMin          "-5"
YMax          "5"
SaveToWorkspace off
SaveName      "ScopeData"
DataFormat    "Matrix"
LimitMaxRows  on
MaxRows       "5000"
Decimation    "1"
SampleInput   off
SampleTime    "0"
}
Block {
BlockType     Scope
Name          "Vista_x2"
Ports         [1, 0, 0, 0, 0]
Position      [440, 220, 470, 250]
Floating      off
Location      [56, 437, 380, 676]
Open          on
NumInputPorts "1"
TickLabels    "on"
ZoomMode      "on"
List {
ListType      AxesTitles
axes1         "%<SignalLabel>"
}
Grid          "on"
TimeRange     "auto"
YMin          "-5"
YMax          "5"
SaveToWorkspace off
SaveName      "ScopeData"
DataFormat    "Matrix"
LimitMaxRows  on

```

```

    MaxRows          "5000"
    Decimation       "1"
    SampleInput      off
    SampleTime       "0"
}
Block {
    BlockType        Gain
    Name             "a1"
    Position         [195, 87, 245, 133]
    Gain             "1/M1"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType        Gain
    Name             "a2"
    Position         [195, 211, 245, 259]
    Gain             "1/M2"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType        Gain
    Name             "amortig resorte"
    Position         [240, 130, 270, 160]
    Orientation      "left"
    Gain             "BR"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType        Gain
    Name             "amortigu resorte"
    Position         [185, 165, 215, 195]
    Orientation      "left"
    Gain             "BR"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType        Gain
    Name             "coef. rigidez"
    Position         [30, 195, 60, 225]
    Gain             "k"
    SaturateOnIntegerOverflow on
}
Block {
    BlockType        Integrator
    Name             "v1"
    Ports            [1, 1, 0, 0, 0]
    Position         [275, 95, 305, 125]
    ExternalReset    "none"
    InitialConditionSource "internal"
}

```



```

InitialCondition      "0"
LimitOutput           off
UpperSaturationLimit "inf"
LowerSaturationLimit "-inf"
ShowSaturationPort   off
ShowStatePort        off
AbsoluteTolerance    "auto"
}
Block {
  BlockType           Integrator
  Name                 "v2"
  Ports                [1, 1, 0, 0, 0]
  Position             [275, 220, 305, 250]
  ExternalReset        "none"
  InitialConditionSource "internal"
  InitialCondition     "0"
  LimitOutput          off
  UpperSaturationLimit "inf"
  LowerSaturationLimit "-inf"
  ShowSaturationPort   off
  ShowStatePort        off
  AbsoluteTolerance    "auto"
}
Block {
  BlockType           Integrator
  Name                 "x1"
  Ports                [1, 1, 0, 0, 0]
  Position             [340, 95, 370, 125]
  ExternalReset        "none"
  InitialConditionSource "internal"
  InitialCondition     "0"
  LimitOutput          off
  UpperSaturationLimit "inf"
  LowerSaturationLimit "-inf"
  ShowSaturationPort   off
  ShowStatePort        off
  AbsoluteTolerance    "auto"
}
Block {
  BlockType           Sum
  Name                 "x1-x2"
  Ports                [2, 1, 0, 0, 0]
  Position             [340, 292, 370, 323]
  Orientation          "left"
  IconShape            "rectangular"
  Inputs               "-+"
  SaturateOnIntegerOverflow on
}
Block {

```

```

BlockType      Integrator
Name           "x2"
Ports         [1, 1, 0, 0, 0]
Position      [340, 220, 370, 250]
ExternalReset  "none"
InitialConditionSource "internal"
InitialCondition "0"
LimitOutput   off
UpperSaturationLimit "inf"
LowerSaturationLimit "-inf"
ShowSaturationPort off
ShowStatePort  off
AbsoluteTolerance "auto"
}
Line {
  SrcBlock      "Fricción loc"
  SrcPort       1
  DstBlock      "Suma_F1"
  DstPort       2
}
Line {
  SrcBlock      "Fricción_vagón"
  SrcPort       1
  Points        [0, -20]
  DstBlock      "Suma_F2"
  DstPort       3
}
Line {
  SrcBlock      "coef. rigidez"
  SrcPort       1
  Points        [20, 0]
  Branch {
Points        [0, -95]
DstBlock      "Suma_F1"
DstPort       3
  }
  Branch {
Points        [0, 25]
DstBlock      "Suma_F2"
DstPort       2
  }
}
Line {
  SrcBlock      "Suma_F1"
  SrcPort       1
  DstBlock      "a1"
  DstPort       1
}
Line {

```

```

    SrcBlock      "v2"
    SrcPort      1
    Points       [5, 0]
    Branch {
Points       [0, 55; -300, 0]
DstBlock     "Fricción_vagón"
DstPort      1
    }
    Branch {
Points       [5, 0]
Branch {
    DstBlock     "x2"
    DstPort      1
}
}
Branch {
    Points       [0, -55]
    Branch {
        Points       [0, -10]
        DstBlock     "Vista_v2"
        DstPort      1
    }
    Branch {
        Labels       [1, 0]
        DstBlock     "amortigu resorte"
        DstPort      1
    }
}
}
}
}
Line {
    SrcBlock      "x1-x2"
    SrcPort      1
    Points       [-325, 0; 0, -100]
    DstBlock     "coef. rigidez"
    DstPort      1
}
Line {
    SrcBlock      "x2"
    SrcPort      1
    Points       [10, 0]
    Branch {
DstBlock     "x1-x2"
DstPort      1
    }
    Branch {
DstBlock     "Vista_x2"
DstPort      1
    }
}
}

```

```

Line {
  SrcBlock      "a2"
  SrcPort       1
  DstBlock      "v2"
  DstPort       1
}
Line {
  SrcBlock      "Suma_F2"
  SrcPort       1
  DstBlock      "a2"
  DstPort       1
}
Line {
  SrcBlock      "amortig resorte"
  SrcPort       1
  Points        [-125, 0]
  DstBlock      "Suma_F2"
  DstPort       1
}
Line {
  SrcBlock      "amortigu resorte"
  SrcPort       1
  Points        [-80, 0; 0, -55]
  DstBlock      "Suma_F1"
  DstPort       4
}
Line {
  SrcBlock      "x1"
  SrcPort       1
  Points        [25, 0]
  Branch {
Points      [0, 205]
DstBlock    "x1-x2"
DstPort     2
  }
  Branch {
DstBlock    "Vista_x1"
DstPort     1
  }
}
Line {
  SrcBlock      "v1"
  SrcPort       1
  Points        [5, 0]
  Branch {
Points      [5, 0]
Branch {
  Points      [5, 0]
Branch {

```

```

        DstBlock      "x1"
        DstPort      1
    }
    Branch {
        Points        [0, -60]
        DstBlock      "Vista_v1"
        DstPort      1
    }
}
Branch {
    Points          [0, 35]
    DstBlock        "amortig resorte"
    DstPort         1
}
    Branch {
Points          [0, -100; -305, 0]
DstBlock        "Fricción loc"
DstPort         1
    }
}
Line {
    SrcBlock        "a1"
    SrcPort         1
    DstBlock        "v1"
    DstPort         1
}
Line {
    SrcBlock        "Fuerza"
    SrcPort         1
    Points          [0, 35]
    DstBlock        "Suma_F1"
    DstPort         1
}
Line {
    SrcBlock        "Escalón"
    SrcPort         1
    Points          [5, 0]
    Branch {
DstBlock        "Vista escalón"
DstPort         1
    }
    Branch {
DstBlock        "Fuerza"
DstPort         1
    }
}
}
}
}

```

### A3. Gráfica de respuestas en Simulink

```

%Grafica de respuesta en simulink
clear all;
%escalón
load escalon.mat
figure(10);
t=1:1:300;
y=escalon(1,t);
r=escalon(2,t);
plot(y,r,'k-');
grid on;
axis([0,200,0,1.2]);
xlabel('Tiempo (s)')
ylabel('Velocidad (m/s)')
title('Escalón de arranque y parada del proceso')
% velocidad v1
load v1.mat
figure(11);
t=1:1:600;
y=v1(1,t);
r=v1(2,t);
plot(y,r,'k-');
grid on
axis([0,200,0,15])
xlabel('Tiempo (s)')
ylabel('Velocidad v1(m/s)')
title('Arranque y parada del proceso')
%velocidad v2
load v2.mat
figure(12);
t=1:1:600;
y=v2(1,t);
r=v2(2,t);
plot(y,r,'k-');
grid on
axis([0,200,0,15])
xlabel('Tiempo (s)')
ylabel('Velocidad v2(m/s)')
title('Arranque y parada del proceso')
%distancia x1
load x1.mat
figure(13);
t=1:1:600;
y=x1(1,t);
r=x1(2,t);
plot(y,r,'k-');
grid on

```

```

axis([0,200,0,1500])
xlabel('Tiempo (s)')
ylabel('Distancia x1(m)')
title('Arranque y parada del proceso')
%distancia x2
load x2.mat
figure(14);
t=1:1:600;
y=x2(1,t);
r=x2(2,t);
plot(y,r,'k-');
grid on
axis([0,200,0,1500])
xlabel('Tiempo (s)')
ylabel('Distancia x2(m)')
title('Arranque y parada del proceso')

```

#### A4. Simulación del sistema con control óptimo

```

% optimo.m
% Sistema de Control de Arranque y Parada de un Tren
% Simulación del Sistema de Control Optimo
%
clear all;
ti=clock;
% Parámetros del proceso
M1=126000;      % Masa de la locomotora (kg)
M2=120000;     % Masa del vagón (kg)
k=1000000;     % Coeficiente de rigidez del resorte (N/m)
B1=10000;     % Coeficiente de fricción estática de
              % la locomotora (N-s/m)
B2=10000;     % Coeficiente de fricción estática del
              % vagón (N-s/m)
BR=1000;      % Coeficiente de fricción viscosa del
              % resorte (N-s/m)
F=260000;     % Fuerza (N)
% Modelo del tren en tiempo continuo
A=[0 1 0 0;
   -k/M1 -(B1+BR)/M1 k/M1 BR/M1;
   0 0 0 1;
   k/M2 BR/M2 -k/M2 -(B2+BR)/M2];
B=[ 0;
   F/M1;
   0;
   0];
C=[0 1 0 0];

```

```

D=[0];
% Controlabilidad del Proceso
rAB=rank(ctrb(A,B));      %4 Completamente controlable
rAC=rank(observ(A,C));   %3 No es completamente
observable
% Modelo del Tren en Tiempo Discreto
T = 1;      % Período de Muestreo
[G,H,C,D]=c2dm(A,B,C,D,T,'zoh');
% Adicionamos la Acción Integral
G1 = [G zeros(4,1);-C*G 1];
H1 = [H;-C*H];
% Matrices de Ponderación
Q = [1 0 0 0 0      %x(k)
     0 1000 0 0 0
     0 0 1 0 0
     0 0 0 1 0
     0 0 0 0 200];
R = [10];          %u(k)
% Obtenemos la Ganancia KK del Controlador aplicando dlqr
[KK,P]=dlqr(G1,H1,Q,R);
% Separamos la acción integral
K = [KK(1) KK(2) KK(3) KK(4)];
KI = -KK(5);
% Simulación
N = 200; x=[0;0;0;0]; v=0; Y=0;
for k=1:N
    r(k)=0.5+0.5*sign(sin(0.04*k)); Ref=r(k);
    v = v + Ref - Y;
    U = -K*x + KI*v;
    x = G*x + H*U;
    Y = C*x;
    y(k) = Y; u(k) = U;
end
% Calculo de la Matriz de Ganancia del Observador
NO=[C' G'*C' G'^2*C' G'^3*C' G'^4*C'];
rnN=rank(NO); %rnN=3 Por lo tanto no es observable
Qe=[1 0 0 0
    0 1000 0 0
    0 0 1 0
    0 0 0 200];
Re=[10];
Pe=zeros(4,4);
for i=1:100
    Pe=Qe+G*Pe*G'-G*Pe*C'*inv(Re+C*Pe*C')*C*Pe*G';
end
KeT=inv(Re+C*Pe*C')*C*Pe*G';
%Pe =
% 1.0e+003 *
% 4.5404 0.1109 4.5243 0.0043

```



```

%      0.1109      1.1765      0.0921      -0.0097
%      4.5243      0.0921      4.5201      -0.0071
%      0.0043      -0.0097      -0.0071      0.2328
%
%KeT =
%      0.4756      0.1999      0.6687      0.7420
% Comprobación de KK
%Ecuación de Riccati
Ptilde=diag(0,4);
for i=1:100
    Ptilde=Q+G1'*Ptilde*G1-
G1'*Ptilde*H1*inv(R+H1'*Ptilde*H1)*H1'*Ptilde*G1;
end
%Cálculo de la ganancia KK del controlador
KKtilde=inv(R+H1'*Ptilde*H1)*H1'*Ptilde*G1;
etime(clock,ti)
%Gráficos
figure(2);
t=linspace(0,N*T,N);
subplot(2,1,1)
plot(t,y(1:N),t,r(1:N)), grid
title('Control Optimo')
ylabel('Velocidad (m/s)')
xlabel('Tiempo (s)')
subplot(2,1,2)
plot(t,u(1:N)), grid
xlabel('Tiempo (s)')
ylabel('Señal de control u')
%
% Resultados del Proceso
%
%
%G =
%      0.2149      0.4025      0.7851      0.5581
%      1.4567      0.1844      -1.4567      0.7371
%      0.8220      0.5860      0.1780      0.3742
%      -1.5326      0.7740      1.5326      0.1483
%
%H =
%      0.6141
%      0.8305
%      0.4100
%      1.2093
%
%G1 =
%      0.2149      0.4025      0.7851      0.5581      0
%      1.4567      0.1844      -1.4567      0.7371      0
%      0.8220      0.5860      0.1780      0.3742      0
%      -1.5326      0.7740      1.5326      0.1483      0

```

```

%   -1.4567   -0.1844   1.4567   -0.7371   1.0000
%
%H1 =
%   0.6141
%   0.8305
%   0.4100
%   1.2093
%  -0.8305
%
%P
% 1.0e+005 *
%   0.0399   0.1704   0.3017   0.0035   0.3393
%   0.1704   1.9347   3.7720   0.0215   3.9420
%   0.3017   3.7720   7.4344   0.0403   7.7385
%   0.0035   0.0215   0.0403   0.0005   0.0436
%   0.3393   3.9420   7.7385   0.0436   8.0835
%
%KK =
%   0.8585   0.0831  -0.8550   0.8367  -0.3432
%
%Ptilde =
% 1.0e+003 *
%   2.5448   0.3684  -2.5336   0.1628  -0.2259
%   0.3684   1.1164  -0.2724   0.0116   0.0579
%  -2.5336  -0.2724   2.7205  -0.1616   0.4172
%   0.1628   0.0116  -0.1616   0.0257  -0.0146
%  -0.2259   0.0579   0.4172  -0.0146   0.7703
%
%
%KKtilde =
%   0.8585   0.0831  -0.8550   0.8367  -0.3432
%
%KI =
%   0.3432
%
%
% Tiempo de ejecucion = 0.99 segundos
% Tiempo de ascenso 10%-90% = 5 segundos [6 (0.9183) - 1
(0.2851)]
% Tiempo de estabilización = 9 segundos (0.9755 error
menor a 2%) a un valor de 0.9901
% No tiene sobrepico

```

## A5. Simulación del sistema con control adaptivo

```
% adaptivo.m
```

```

% Sistema de Control de Arranque y Parada de un Tren
% Simulación del Sistema de Control Adaptivo
%
%
clear all;
ti=clock;
% Parámetros del proceso
M1=126000;      % Masa de la locomotora (kg)
M2=120000;      % Masa del vagón (kg)
k=1000000;      % Coeficiente de rigidez del resorte (N/m)
B1=10000;       % Coeficiente de fricción estática de
                % la locomotora (N-s/m)
B2=10000;       % Coeficiente de fricción estática del
                % vagón (N-s/m)
BR=1000;        % Coeficiente de fricción viscosa del
                % resorte (N-s/m)
F=260000;       % Fuerza (N)
% Modelo del tren en tiempo continuo
A=[0 1 0 0;
   -k/M1  -(B1+BR)/M1  k/M1  BR/M1;
   0  0  0  1;
   k/M2  BR/M2  -k/M2  -(B2+BR)/M2];
B=[ 0;
   F/M1;
   0;
   0];
C=[0 1 0 0];
D=[0];
%PARAMETROS GENERALES
nn = 4; r=1; % nn: orden del proceso; r: NUMERO DE
SALIDAS
NN = 9; % ORDEN DEL VECTOR ESTIMADO DE PARAMETROS
% cmax = 100; cmin = 15; % PARA EL ESTIMADOR DE
PARAMETROS
% PARAMETROS PARA OPTIMIZAR GANANCIA DEL FILTRO DE KALMAN
Xi = 1 ; Sigma = 100;
% PARAMETROS DE OPTIMIZACION DEL CONTROLADOR Ke
R = 100; Qa = 1*eye(r+nn,r+nn);
T = 1; % PERIODO DE MUESTREO
% MODELO LINEAL DISCRETO
[G,H,C,D]=c2dm(A,B,C,D,T,'zoh');
[num,den]=ss2tf(G,H,C,D);
a1 = den(2); a2 = den (3); a3=den(4); a4=den(5);
b1 = num(2); b2 = num (3); b3=num(4); b4=num(5);
% CONDICIONES INICIALES
Ref = 1; % MAGNITUD DE LA SEÑAL DE REFERENCIA: Yo = Ref
Ce = (1 + a1 + a2 +a3 + a4)*Ref; % Uo = 0 (ARBITRARIO)
thi = [-a1 -a2 -a3 -a4 b1 b2 b3 b4 Ce]';
alfa=1000; Pi = alfa*eye(NN); % MATRIZ P INICIAL

```

```

lamb=0.98; lamb0 = 0.99;          % FACTOR DE MEMORIA
Y(1)=0; Y(2)=0; Y(3)=0; Y(4)=0;
y(1)=0; y(2)=0; y(3)=0; y(4)=0; % salida
U(1)=0; U(2)=0; U(3)=0; U(4)=0;
u(1)=0; u(2)=0; u(3)=0; u(4)=0; % control
xmenos = zeros(nn,1); %VECTOR DE OBSERVACION INICIAL
z=0; %ACCION INTEGRAL INICIAL
% INICIO DEL LAZO DE CONTROL *****
MM = 200;
for t = 1:MM
    % SEÑAL DE REFERENCIA
    g(t+4)=0.5+0.5*sign(sin(0.04*t));
    % MODELO LINEAL DEL PROCESO
    Y(t+4) = [Y(t+3) Y(t+2) Y(t+1) Y(t)...
    U(t+3) U(t+2) U(t+1) U(t)]*[-a1  -a2  -a3  -a4  b1 b2 b3 b4]'
    y(t+4) = Y(t+4) - g(t+4);%salida residual
(desviacion)
    %ESTIMACION DE PARAMETROS (METODO MCRM)
    psi = [Y(t+3) Y(t+2) Y(t+1) Y(t) ...
           U(t+3) U(t+2) U(t+1) U(t) 1]';
    rho = max(1,norm(psi));
    psin = psi/rho;
    Nn = chol(Pi'); % Nn'*Nn = Pi => Nn*N' = Pi'
    S = inv(diag(Nn*ones(NN,1),0));
    Ps = S*Pi*S;
    psins = inv(S)*psin;
    rt = 1 + psins'*Ps*psins;
    lamb = 1 - (rt-sqrt(rt^2-
4*norm(Ps*psins)^2/trace(Ps)))/2;
    e = Y(t+4)/rho - psin'*thi;
    j = psins'*Ps*psins + lamb;
    th = thi + e*inv(S)*Ps*psins/j;
    Hns = Ps*psins/j;
    Ps= (Ps - Hns*psins'*Ps)/lamb;
    tt = abs(max(eig(Ps))/min(eig(Ps)));
    cmax = 100; cmin = 15;
    if tt >= cmin,
        Pi = Ps; thi = th;
    elseif tt >= cmax,
        Nnew = chol(Ps'); % Nnew*Nnew' = Ps'
        Snew = inv(diag(Nnew*ones(NN,1),0));
        Psnew = Snew*Ps*Snew;
        Pi = Psnew; thi = th;
    end
    % RECUPERACION DE LA ECUACION CANONICA CONTROLABLE
    ae1=-th(1); ae2=-th(2); ae3=-th(3);ae4=-th(4);
    be1=th(5); be2=th(6); be3=th(7);be4=th(8); Ce=th(9);
    ae11=0; ae12=1; ae13=0; ae14=0;
    ae21=0; ae22=0; ae23=1; ae24=0;

```

```

ae31=0; ae32=0; ae33=0; ae34=1;
ae41=-ae4; ae42=-ae3; ae43=-ae2; ae44=-ae1;
GE = [ae11 ae12 ae13 ae14
      ae21 ae22 ae23 ae24
      ae31 ae32 ae33 ae34
      ae41 ae42 ae43 ae44];
be11=0; be21=0; be31=0; be41=1;
HE=[be11;be21;be31;be41];
ce11=be4; ce12=be3; ce13=be2; ce14=be1;
CE=[ce11 ce12 ce13 ce14]; DE=[0];
% CALCULO DE LA GANANCIA F DEL ESTIMADOR DE KALMAN
F=dlqe(GE,HE,CE,Sigma,Xi);
xmas = xmenos + F*(y(t+4) - CE*xmenos); %
OBSERVACION
z = z -CE*xmas; % ACCION
INTEGRAL
% CALCULO DE LA GANANCIA Ka DEL CONTROLADOR
Ga = [GE zeros(nn,r);-CE eye(r,r)];
Ha = [HE;zeros(r,r)];
Ca = [CE zeros(r,r)];
Ka = dlqr(Ga, Ha, Qa, R);
% CALCULO DE LA LEY DE CONTROL RESIDUAL
u(t+4) = -Ka*[xmas;z];
% ACTUALIZANDO ESTADOS
xmenos = GE*xmas + HE*u(t+4);
AE = 1 + ae1 + ae2 + ae3 + ae4; BE = be1 + be2
+ be3 + be4;
Uin = inv(BE)*(AE*g(t+4) - Ce);
% CALCULO DE LA SEÑAL DE CONTROL U (t)
U(t+4) = u(t+4) + Uin;
%LIMITANDO LA SEÑAL U(t)
if(U(t+4) > 1)
    U(t+4) = 1;
elseif(U(t+4) <-1)
    U(t+4) = -1;
else
    U(t+4) = U(t+4);
end
end % FIN DEL LAZO DE CONTROL
etime(clock,ti)
% GRAFICOS
figure(3);
ejex = linspace(0,MM*T,MM);
subplot(2,1,1)
plot(ejex,Y(4:MM+3),ejex,g(4:MM+3)), grid
title('Control Adaptivo')
ylabel('Velocidad (m/s)')
xlabel('Tiempo (s)')
subplot(2,1,2), grid

```

```

        plot(ejex,U(3:MM+2)), grid
        xlabel('Tiempo (s)')
        ylabel('Señal de control u')
%
% Resultados del Proceso
%
%num =
%      0      0.8305      0.7393      -0.8200      -0.7498
%den =
%      1.0000     -0.7256     -0.4703     -0.6402      0.8361
%
%thi =
%      0.7256
%      0.4703
%      0.6402
%     -0.8361
%      0.8305
%      0.7393
%     -0.8200
%     -0.7498
%     -0.0000
%
%Ce =
% -1.4791e-018
%
%psi =
%      0.9959
%      1.0013
%      0.9983
%      0.9962
%      0.0769
%      0.0766
%      0.0770
%      0.0768
%      1.0000
%
%GE =
%      0      1.0000      0      0
%      0      0      1.0000      0
%      0      0      0      1.0000
%     -0.8361      0.6402      0.4703      0.7256
%HE =
%      0
%      0
%      0
%      1
%CE =
%     -0.7498     -0.8200      0.7393      0.8305
%

```

```

%F =
%   0.1142
%  -0.1481
%   0.1872
%   0.9831
%
%Ga =
%       0       1.0000       0       0       0
%       0       0       1.0000       0       0
%       0       0       0       1.0000       0
%  -0.8361  0.6402  0.4703  0.7256       0
%   0.7498  0.8200 -0.7393 -0.8305  1.0000
%
%Ha =
%   0
%   0
%   0
%   1
%   0
%
%Ca =
%  -0.7498  -0.8200  0.7393  0.8305       0
%
%Ka =
%  -0.3619  -0.0761  0.1227  0.4002  -0.0669
%
%
%
% Tiempo de ejecución = 14.45 segundos.
% Tiempo de ascenso 10%-90% = 3 segundos [9 (0.9172) - 6
(0.2843)]
% Tiempo de estabilización = 20 segundos (1.0138 error
menor a 2%)
% Sobrepico de 6.47% a los 14 segundos

```

## A6. Simulación del sistema con control predictivo

```

% predictivo.m
% Sistema de Arranque y Parada de un Tren
% Simulación del Sistema con Control Predictivo
%
%
clear all;
ti=clock;
% Parámetros del proceso
M1=126000;          % Masa de la locomotora (kg)

```

```

M2=120000;          % Masa del vagón (kg)
k=1000000;         % Coeficiente de rigidez del resorte (N/m)
B1=10000;          % Coeficiente de fricción estática de
                   % la locomotora (N-s/m)
B2=10000;          % Coeficiente de fricción estática del
                   % vagón (N-s/m)
BR=1000;           % Coeficiente de fricción viscosa del
                   % resorte (N-s/m)
F=260000;          % Fuerza (N)
% Modelo del tren en tiempo continuo
A=[0 1 0 0;
   -k/M1 -(B1+BR)/M1 k/M1 BR/M1;
   0 0 0 1;
   k/M2 BR/M2 -k/M2 -(B2+BR)/M2];
B=[ 0;
   F/M1;
   0;
   0];
C=[1 0 0 0];
D=[0];
% MODELO DEL PROCESO EN TIEMPO DISCRETO
tm=1;              % TIEMPO DE MUESTREO
[G,H]=c2d(A,B,tm);
[numd,dend]=ss2tf(G,H,C,D);
a1=dend(2); a2=dend(3); a3=dend(4); a4=dend(5);
b1=numd(2); b2=numd(3); b3=numd(4); b4=numd(5);
% HORIZONTES
N1=1; N2=10; Nu=1;
% RESPUESTA AL ESCALON
g(1)=b1; g(2)=-a1*b1+b1+b2;
g(3)=-a1*g(2)-a2*g(1)+b1+b2+b3;
g(4)=-a1*g(3)-a2*g(2)-a3*g(1)+b1+b2+b3+b4;
for k=5:N2
    g(k)=-a1*g(k-1)-a2*g(k-2)-a3*g(k-3)-a4*g(k-4)...
        +b1+b2+b3+b4;
end
% CONDICIONES INICIALES
for k=1:5
    yf(k)=0; y(k)=0; r(k)=0; u(k)=0;
end
alf=0.3; lambda=0; MM=200;
% LAZO DE CONTROL
for t=5:MM+4;
    % TRAYECTORIA DESEADA, REFERENCIA r Y RESPUESTA LIBRE
yf
    for k=N1:N2
        W(t+k)=0.5+0.5*sign(sin(0.04*t));
        r(t+k)=alf*r(t+k-1)+(1-alf)*W(t+k);
        yf(t+k)=-a1*yf(t+k-1)-a2*yf(t+k-2)-a3*yf(t+k-3)...

```



```

    -a4*yf(t+k-4)+(b1+b2+b3+b4)*u(t-1);
end
%Restriccion a la salida
%
y(t)=r(t);
%LEY DE CONTROL
du(t)=(g(1)*(r(t+1)-yf(t+1))+g(6)*(r(t+6)-
yf(t+6))+...
    g(2)*(r(t+2)-yf(t+2))+g(7)*(r(t+7)-yf(t+7))+...
    g(3)*(r(t+3)-yf(t+3))+g(8)*(r(t+8)-yf(t+8))+...
    g(4)*(r(t+4)-yf(t+4))+g(9)*(r(t+9)-yf(t+9))+...
    g(5)*(r(t+5)-yf(t+5))+g(10)*(r(t+10)-
yf(t+10)))/...
    (g(1)^2+g(2)^2+g(3)^2+g(4)^2+g(5)^2+g(6)^2+...
    g(7)^2+g(8)^2+g(9)^2+g(10)^2+lambda);
u(t)=u(t-1)+du(t);
%LIMITANDO EL VALOR DE u(t)
if(u(t)>0.2)
    u(t)=0.2;
elseif(u(t)<-0.2)
    u(t)=-0.2;
end
%PROCESO
y(t+1)=-a1*y(t)-a2*y(t-1)-a3*y(t-2)-a4*y(t-3)...
    +b1*u(t)+b2*u(t-1)+b3*u(t-2)+b4*u(t-3);
end %FIN DEL LAZO DE CONTROL
etime(clock,ti)
%GRAFICOS
figure(5);
ejex=linspace(0,MM*tm,MM-4);
subplot(2,1,1)
plot(ejex,y(5:MM),ejex,W(5:MM));
grid on
title('Control Predictivo')
xlabel('Tiempo (s)')
ylabel('Velocidad (m/s)')
subplot(2,1,2)
plot(ejex,u(5:MM));
grid on
xlabel('Tiempo (s)')
ylabel('Señal de Control u')
%
% Resultados del proceso
%
%numd =
%      0      0.2976      0.4931      0.4746      0.2613
%
%dend =
%      1.0000     -0.7256     -0.4703     -0.6402      0.8361

```

```
%  
%g =  
% Columns 1 through 7  
% 0.2976 1.0066 2.1357 3.7403 5.6406  
7.9042 10.5237  
%  
% Columns 8 through 10  
% 13.3640 16.5171 19.9253  
%  
% Resultados sin restriccion a la salida  
% Tiempo de ejecución = 0.77 segundos,  
% Tiempo de ascenso 10%-90% = 12 segundos [ 20(0.9189)  
8(0.1325)]  
% Tiempo de estabilización = 33 segundos (1.0194 error  
menor a 2%)  
% Sobrepico 2.37% a los 29 segundos.  
%  
% Resultados con restriccion a la salida  
% Tiempo de ejecución = 0.77 segundos,  
% Tiempo de ascenso 10%-90% = 1 segundo 7(0.9100)  
6(0.7000)]  
% Tiempo de estabilización = 9 segundos (0.9919 error  
menor a 2%)  
% No tiene sobrepico
```

## **ANEXO B LISTADO DE FIGURAS**

### **CAPÍTULO II**

#### **MODELADO DEL PROCESO Y SIMULACIÓN EN TIEMPO CONTÍNUO**

Figura 2.1	Representación del tren	9
Figura 2.2	Diagrama de cuerpo libre del tren	10
Figura 2.3	Respuesta al escalón en tiempo continuo y discreto. Modelo del tren en lazo abierto sin controlador	18
Figura 2.4	Modelo del tren en SIMULINK. Monitoreo de las señales con osciloscopios	19
Figura 2.5	Escalón de entrada en lazo abierto	20
Figura 2.6	Respuesta del proceso al escalón. Velocidad de la locomotora	21
Figura 2.7	Respuesta del proceso al escalón. Velocidad del vagón	22
Figura 2.8	Distancia recorrida por el tren	23
Figura 2.9	Modelo del tren en SIMULINK. Monitoreo de las señales con almacenaje de datos	24

### **CAPÍTULO III**

#### **SISTEMA DE CONTROL ÓPTIMO**

Figura 3.1	Regulador óptimo proporcional integral	25
Figura 3.3	Diagrama de bloques del observador de estados	31
Figura 3.3	Simulación del sistema de control óptimo	35

**CAPÍTULO IV****SISTEMA DE CONTROL ADAPTIVO**

Figura 4.1 Sistema de control adaptivo con autosintonización 37

Figura 4.2 Simulación del sistema de control adaptivo 46

**CAPÍTULO V****SISTEMA DE CONTROL PREDICTIVO**

Figura 5.1 Estrategia del control predictivo 48

Figura 5.2 Simulación del sistema de control predictivo 55

Figura 5.3 Simulación del sistema de control predictivo con restricción a la salida 56

## BIBLIOGRAFÍA

- [1] Arturo Rojas Moreno. *Control Avanzado. Diseño y Aplicaciones en Tiempo Real*. Universidad Nacional de Ingeniería. Lima 2001
- [2] Dolores M. Etter. *Solución de Problemas de Ingeniería con MATLAB*. Segunda Edición. Prentice Hall Hispanoamericana S.A. México 1998.
- [3] *Using Matlab Version 6* The Mathworks, Inc., Estados Unidos 2000
- [4] Andrew Grace, Allan J. Laub, John N. Little, Clay M. Thompson. *Control System Toolbox – User’s Guide* The Mathworks, Inc. Estados Unidos 1994
- [5] Robert H. Bishop. *Modern Control Systems Analysis & Design using Matlab & Simulink*. Addison Wesley. Estados Unidos 1997
- [6] Benjamin C. Kuo. *Sistemas de Control Automático*. Séptima edición. Prentice Hall Hispanoamericana S.A. México 1996
- [7] Katsuhiko Ogata. *Sistemas de Control en Tiempo Discreto*. Segunda Edición. Prentice Hall Hispanoamericana S.A. México 1996
- [8] Katsuhiko Ogata. *Ingeniería de Control Moderna*. Prentice Hall Hispanoamericana S.A. México 1980
- [9] Raúl Benites Saravia. *Quick Training MATLAB 5.3 Interactivo*. Universidad Nacional de Ingeniería. Lima 2001
- [10] University of Michigan. University of Carnegie Mellon *Control Tutorials for Matlab*. <http://www.engin.umich.edu/group/ctm/home.text.html>
- [11] Bill Messner, Dawn Tilbury. *Control Tutorials for Matlab and Simulink*. Addison-Wesley. Estados Unidos 1998  
<http://wolfman.eos.uoguelph.ca/~jzelek/matlab/ctms/simulink/model/model.htm>
- [12] University of Idaho. *Intelligent Power Management System*. 1997  
<http://www.ee.idaho.edu/trainnet/reports/interim/intrep.pdf>
- [13] *Diesel locomotive FAQ*. 2002

<http://exotic.railfan.net/dieselfaq.htm>

- [14] Steve Sconfienza. *Information on Diesel-Electric and Electric locomotives*. Jun 2002.  
[http://www.crisny.org/not-for-ptofit/railned/text/en\\_info.htm](http://www.crisny.org/not-for-ptofit/railned/text/en_info.htm)
- [15] Urban Fredriksson. *rec.models.railroad FAQ PROTOTYPE part 2 of 2*.  
[http://www.northeast.railfan.net/pro\\_faq2.html](http://www.northeast.railfan.net/pro_faq2.html)
- [16] TrainWeb. *Diesel Locomotive Technology*. 24 Noviembre 2002  
<http://www.trainweb.org/railwaytechnical/diesel.html>
- [17] General Motors. *JT42CWR.Aug,31 2001* Aug. 31, 2002  
<http://www.gmemd.com/en/locomotive/international/gt42cwr/index.htm>
- [18] General Electric. *Box Cars*. 2000  
<http://www.ge.com/capital/rail/equipment/boxcar/bcgeinfo.shtml>