

**UNIVERSIDAD AUTONOMA DE NUEVO LEON**

**FACULTAD DE INGENIERIA MECANICA Y ELECTRICA**

**DIVISION DE ESTUDIOS SUPERIORES**



**ANALISIS Y SINTESIS DE SISTEMAS  
DE CONTROL LINEAL POR COMPUTADORA**

**T E S I S**

**QUE PARA OBTENER EL GRADO DE**

**MAESTRO EN CIENCIAS EN INGENIERIA ELECTRICA  
ESPECIALIDAD EN CONTROL**

**PRESENTA:**

**GERARDO ROMERO GALVAN**

**CIUDAD UNIVERSITARIA**

**ENERO DE 1993**



TM

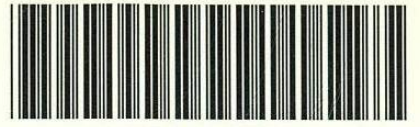
Z5853

.M2

FIME

1993

R6

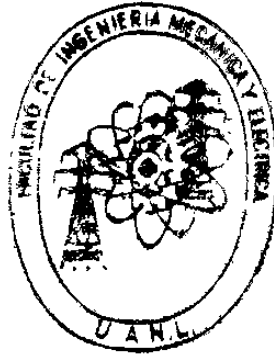


1020070660

**UNIVERSIDAD AUTONOMA DE NUEVO LEON**

FACULTAD DE INGENIERIA MECANICA Y ELECTRICA

DIVISION DE ESTUDIOS SUPERIORES



ANALISIS Y SINTESIS DE SISTEMAS  
DE CONTROL LINEAL POR COMPUTADORA

T E S I S

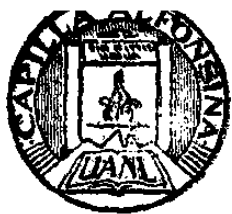
QUE PARA OBTENER EL GRADO DE  
MAESTRO EN CIENCIAS EN INGENIERIA ELECTRICA  
ESPECIALIDAD EN CONTROL

PRESENTA  
GERARDO ROMERO GALVAN

CIUDAD UNIVERSITARIA

ENERO DE 1993

TM  
Z5853  
.M2  
FINE  
1993  
R6



**FONDO TESIS**

24087

**U. A. N. L.**

**F. I. M. E.**

**Programa Doctoral en Ingeniería Eléctrica  
Control Automático y Robótica**

**Análisis y Síntesis de Sistemas de Control  
Lineal por Computadora**

**Autor: Gerardo Romero Galván**

**San Nicolás de los Garza**

**Enero de 1993**

# Índice

<b>Capítulo 1.</b>	<b>Introducción</b>	<b>1</b>
<b>Capítulo 2.</b>	<b>Preliminares</b>	<b>3</b>
<b>Capítulo 3.</b>	<b>Solución de Sistemas de Ecuaciones Lineales</b>	
3.1	Método Gauss-Jordan	9
3.2	Factorización LU	11
3.3	Descomposición en valores singulares	12
<b>Capítulo 4.</b>	<b>Análisis Cuantitativo y Cualitativo de Sistemas Lineales</b>	
4.1	Análisis Cuantitativo	18
4.1.1	Valores y Vectores Propios	18
4.1.2	Respuesta Transitoria	20
4.1.3	Evaluación numérica de $x(t)$	21
4.2	Análisis Cualitativo	24
4.2.1	Estabilidad	24
4.2.2	Controlabilidad y Observabilidad	25
4.2.3	Estabilizabilidad y Detectabilidad	27
4.2.4	Estabilidad Relativa	29
<b>Capítulo 5.</b>	<b>Asignación de Polos</b>	
5.1	Retroalimentación de Estado: Caso Monovariable	32
5.2	Retroalimentación de Estado: Caso Multivariable	34
5.3	Retroalimentación de Norma Mínima	37
5.4	Reguladores mediante asignación de polos	41
<b>Capítulo 6.</b>	<b>Técnicas de Control Moderno</b>	
6.1	Control Óptimo	50
6.1.1	Regulador Óptimo (LQR)	51
6.1.2	Estimadores de estado y Filtro de Kalman	55
6.2	Estabilidad Robusta	58
6.2.1	Margen de Robustez	60
6.2.2	Estabilidad $\ell_p$ -Robusta	65

<b>Capítulo 7.</b>	<b>Conclusiones</b>	<b>69</b>
<b>Apéndice A</b>	<b>Transformaciones Importantes</b>	<b>71</b>
<b>Apéndice B</b>	<b>Programas de Soporte</b>	<b>78</b>
<b>Apéndice C</b>	<b>Programas en MATLAB</b>	<b>82</b>
<b>Referencias</b>		<b>91</b>



## Índice de Figuras

Figura 3.1	La mejor aproximación por una línea recta	15
Figura 3.2	La mejor aproximación por una función senoidal	16
Figura 4.1	Regiones de Estabilidad	29
Figura 5.0-1	Sistema con retroalimentación de estado	32
Figura 5.4-1	Regulador por retroalimentación de estado	42
Figura 5.4-2	Gráfica de trayectoria de estado	44
Figura 5.4-3	Gráfica de ley de control	44
Figura 5.4-4	Trayectoria de estado de $(A + BK_1)$	47
Figura 5.4-5	Trayectoria de estado de $(A + BK_2)$	47
Figura 5.4-6	Gráfica de ley de control $u = K_1x$	48
Figura 5.4-7	Gráfica de ley de control $u = K_2x$	48
Figura 6.1-1	Gráfica de trayectoria de estado óptima	54
Figura 6.1-2	Gráfica de ley de control óptima	54
Figura 6.2-1	Curva de $P(j\omega)$	60

## *Introducción*

El objetivo principal de los métodos numéricos es encontrar soluciones con el menor error posible utilizando sólo las operaciones más simples de la aritmética; es decir, se trata de resolver problemas difíciles mediante muchos pasos fáciles; lo anterior ocasiona que muchos de los algoritmos que existen no sean tan efectivos y que en ocasiones se obtengan resultados erróneos. Esta es la razón principal por lo que es importante hacer una buena selección de los algoritmos o programas que sirvan para efectuar las operaciones en la computadora.

En este trabajo se presentan algunos métodos numéricos para efectuar cálculos en computadora que sirven para efectuar análisis y síntesis de sistemas de control lineal. El trabajo consistió principalmente en la búsqueda y selección de los programas más confiables y disponibles en utilerías de software como ORACLS [2], LINPACK [8], EISPACK [10] que contienen programas que en conjunto forman una librería de subrutinas que resuelven los problemas más comunes presentados en el análisis y síntesis de los sistemas de control lineales. En cada una de las secciones se presenta una breve descripción del problema a ser analizado así como el método de solución. La aportación personal más importante en cada fue la elaboración del programa que realiza el cálculo del problema descrito en cada una de las secciones mediante el uso de la computadora, el cual se presenta en cada sección como una subsección llamada referencias de cálculo numérico que contiene la información necesaria para acceder dichos programas. Es importante hacer indicar que todos los programas toman los datos de un archivo llamado "entrada.dat" y los resultados los deposita en un archivo llamado "salidad.dat" estos datos deberán estar en modo ASCII en el orden

y formato presentado en cada sección.

Por otro lado, el objetivo principal del presente trabajo fué crear una base de programas que efectuen los cálculos numéricos para resolver algunos problemas presentados dentro de los sistemas de control lineal; lo anterior con la finalidad de agilizar el análisis de dichos sistemas, además de crear una programoteca que pudiera auxiliar en el desarrollo de programas aun no elaborados. Hay que aclarar que algunos programas así como la totalidad de las gráficas presentadas en este trabajo fueron hechas con ayuda del lenguaje MATLAB por la facilidad que este presenta para su manejo. El trabajo se presenta como sigue:

En el capítulo 2 se presentan algunos conceptos básicos necesarios para la comprensión de los demas temas contenidos en este trabajo.

En el capítulo 3 se aborda el problema de la solución de sistemas de ecuaciones lineales mediante el uso de la computadora. Este es un tema muy interesante ya que existe una gran cantidad de problemas de ingeniería que pueden representarse como un conjunto de ecuaciones lineales.

El capítulo 4 describe una serie de problemas de los cuales se elaboraron los programas correspondientes para efectuar análisis cuantitativo y cualitativo de los sistemas de control lineal mediante el uso de la computadora.

En el capítulo 5 se presenta la mayor aportación personal a este trabajo, y que consiste en la obtención de la matriz de retroalimentación de estado de norma mínima, la cual representa algunas ventajas en la ley de control.

En el capítulo 6 se presentan algunas técnicas de control modernas como el caso del regulador óptimo (LQR), que consiste en encontrar la ley de control que minimiza un criterio de comportamiento. Otro punto importante es el Filtro de Kalman, que consiste en la estimación de los estados de un sistema de control en presencia de ruido en la salida. En otra de las secciones se presentan algunas técnicas para determinar los márgenes de estabilidad robusta en el dominio de la frecuencia.

## *Preliminares*

En este capítulo se presentan los preliminares matemáticos requeridos posteriormente.

### *2.1 Notación*

$\mathbb{N}$	Anillo de los números enteros
$\mathbb{R}$	Campo de los números reales
$\mathbb{C}$	Campo de los números complejos
$\mathbb{R}_+$	Números reales no negativos
$\mathbb{R}_-$	Números reales negativos
$\mathbb{C}_+$	Números complejos con parte real no negativa
$\mathbb{C}_-$	Números complejos con parte real negativa
$\mathbb{R}^n$	Vector de dimensión $n$ con elementos reales
$\mathbb{C}^n$	Vector de dimensión $n$ con elementos complejos
$\mathbb{R}^{n \times n}$	Matriz de dimensión $n \times n$ con elementos reales
$\mathbb{C}^{n \times n}$	Matriz de dimensión $n \times n$ con elementos complejos
$a_{ij}$	Elemento de la $i$ -ésima fila y la $j$ -ésima columna de la matriz $A$
$x_i$	$i$ -ésimo elemento del vector $x$
$A^T$	Matriz transpuesta de $A$
$A^H$	Matriz transpuesta conjugada de $A$
$A^{-1}$	Inversa de la Matriz $A$
$A^\dagger$	Pseudoinversa de $A$

$I_n$	Matriz identidad de dimensión $n \times n$
$\det(A)$	Determinante de $A$
$\lambda(A)$	Valores propios de $A$
Traza( $A$ )	Suma de elementos de la diagonal principal de la matriz $A$
$\partial I / \partial G$	Gradiente de $I$ con respecto a $G$
$\min\{f(x)\}$	Valor mínimo de $f(x)$ , donde $f(x)$ es una función continua
$\min\{x\}$	Componente mínimo del vector $x$ .
$\max\{f(x)\}$	Valor máximo de $f(x)$ , donde $f(x)$ es una función continua
$\max\{x\}$	Componente máximo del vector $x$

## 2.2 Definiciones Importantes

**Definición 2.2.1** Un *espacio vectorial*  $X$  sobre un campo  $F$  es un conjunto de elementos llamados vectores junto con dos operaciones. La primera operación es llamada suma vectorial que se define para todo vector  $x_1, x_2, x_3 \in X$ , como:

- a)  $(x_1 + x_2) \in X$
- b)  $x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3$
- c)  $x_1 + x_2 = x_2 + x_1$
- d) existe un vector  $\theta$  tal que  $x_1 + \theta = \theta + x_1 = x_1$
- e) para todo  $x_1 \in X$  existe un vector único  $-x_1$ , tal que:
 
$$x_1 + (-x_1) = (-x_1) + x_1 = \theta$$

La segunda operación es llamada multiplicación escalar que se define para todo  $\alpha, \beta \in F$  y  $x_1, x_2 \in X$ , como:

- f)  $\alpha x_1 \in X$
- g)  $(\alpha + \beta)x_1 = \alpha x_1 + \beta x_1$
- h)  $\alpha(\beta x_1) = (\alpha\beta)x_1$
- i)  $1 x_1 = x_1, 0 x_1 = \theta$



**Definición 2.2.2** Una *combinación lineal* de vectores  $x_1, x_2, \dots, x_n$  en un espacio vectorial es una suma de la forma:

$$\alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n$$

donde  $\alpha_i$  son escalares.

**Definición 2.2.3** Un vector  $x \in X$  es llamado *linealmente dependiente* de un conjunto de vectores  $S \in X$ , si  $x$  se puede expresar como una combinación lineal del conjunto de vectores  $S$ , de lo contrario se dice que es linealmente independiente. Una prueba necesaria y suficiente para determinar dependencia lineal es la siguiente:

Un conjunto de vectores  $x_1, x_2, \dots, x_n$  es linealmente independiente si y sólo si,  $\alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n = \theta$  implica que  $\alpha_k = 0$  para  $k=1, 2, \dots, n$ .

**Definición 2.2.4** Un espacio vectorial normado es un espacio vectorial  $X$  sobre el cual está definida una función valuada real que mapea todos los elementos de  $X$  en un número real  $\|x\|$  llamado la norma de  $x$ . Toda norma debe satisfacer los siguientes axiomas:

1.  $\|x\| \geq 0$  Para todo  $x \in X$  y  $\|x\| = 0$ , si y sólo si  $x = \theta$
2.  $\|\alpha x\| = |\alpha| \|x\|$  Para todo escalar  $\alpha$  y todo vector  $x \in X$
3.  $\|x+y\| \leq \|x\| + \|y\|$  Para todo  $x, y \in X$

Algunas normas comunes en  $\mathbb{R}^n$  se definen como sigue:

$$\|x\|_2^2 = x_1^2 + x_2^2 + \dots + x_n^2 \quad \text{norma 2 o norma euclideana}$$

$$\|x\|_1 = x_1 + x_2 + \dots + x_n \quad \text{norma 1.}$$

$$\|x\|_\infty = \max \{x_1, x_2, \dots, x_n\} \quad \text{norma infinito.}$$

Ahora, suponer que A una matriz de dimensión  $n \times n$ , entonces:

$$\|A\|_F^2 = \text{Traza}(A^T A) \quad \text{norma de Frobenius}$$

**Definición 2.2.5** Sea  $p$  un número real que cumple  $1 \leq p \leq \infty$ . El espacio  $\ell_p$  consiste de todas las secuencias de escalares  $\{\xi_1, \xi_2, \dots\}$  para los cuales:

$$\sum_{i=1}^{\infty} |\xi_i|^p < \infty$$

La norma de un elemento  $x = \{\xi_i\}$  en  $\ell_p$  está definida como:

$$\|x\|_p = \left( \sum_{i=1}^{\infty} |\xi_i|^p \right)^{\frac{1}{p}}$$

*NOTA: La norma es una generalización a un espacio vectorial del concepto de longitud o tamaño de un vector.*

### 2.3 Matrices y conceptos de cálculo numérico

**Definición 2.3.1** Una matriz  $A \in \mathbb{C}^{n \times n}$  es llamada *hermítica* si:

$$A^H = A$$

para el caso particular  $A \in \mathbb{R}^{n \times n}$  se le conoce como matriz *simétrica* y se reduce a:

$$A^T = A$$

**Definición 2.3.2** Una matriz  $A \in \mathbb{C}^{n \times n}$  es llamada *unitaria* si:

$$A^H A = I_n$$

para el caso particular  $A \in \mathbb{R}^{n \times n}$  es llamada matriz *ortogonal* y se reduce a:

$$A^T A = I_n$$

**Definición 2.3.3** Una matriz hermítica  $A \in \mathbb{C}^{n \times n}$  es *definida positiva* si  $x^H A x > 0$  para todo  $x \neq 0$ ,  $x \in \mathbb{C}^n$ . Si  $x^H A x \geq 0$ ,  $A$  es *semidefinida positiva*.

**Definición 2.3.4** El *rango* de una matriz es el número de columnas linealmente independientes de la misma matriz.

La forma más confiable de calcular el rango de una matriz es mediante la descomposición en valores singulares (Ver Apéndice A.6).

**Definición 2.3.5** Debido a que la computadora tiene un número de dígitos finito para representar cantidades, aparecen errores que se conocen como *errores de redondeo*

**Definición 2.3.6** En cálculos que requieren un determinado número de iteraciones es muy probable que nos encontremos con muchos errores de redondeo y cada uno de ellos desempeña el papel de un error de entrada para el resto del cálculo y cada uno tiene un efecto sobre la consiguiente salida. Los algoritmos en que es limitado el efecto de tales errores, de modo que se genera un resultado útil, se llaman *algoritmos estables*.

**Definición 2.3.7** Una matriz es llamada *mal condicionada* si cambios pequeños en sus elementos generan cambios grandes al realizar una operación determinada. Existen varias formas de comprobar si la matriz es mal condicionada ó no; una de las formas es mediante el uso de los valores singulares:

$$k(A) = \frac{\sigma_{\max}}{\sigma_{\min}}$$

donde:

$\sigma_{\max}$  - máximo valor singular de  $A$

$\sigma_{\min}$  - mínimo valor singular de  $A$

Si  $k(A)$  es un valor muy grande, entonces la matriz es mal condicionada.

## *Solución de Sistemas de Ecuaciones Lineales*

En este capítulo se analiza la solución de sistemas de ecuaciones lineales mediante el uso de métodos numéricos.

Sea el siguiente sistema de ecuaciones lineales

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\ \vdots + \quad \quad \quad \vdots + \dots + \quad \quad \quad \vdots &= \quad \quad \quad \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= b_m \end{aligned}$$

El sistema anterior puede ser representado matricialmente como:

$$Ax = b \tag{3.0-1}$$

donde:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \quad x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

Una gran cantidad de problemas pueden plantearse como un conjunto de ecuaciones

lineales, en el cual la solución a encontrar son precisamente los valores de  $x_1, \dots, x_n$ . Es necesario mencionar que existen tres casos muy importantes con respecto a la solución de sistemas de ecuaciones lineales, ver sección 4.3 de [21].

- a) - Solución única
- b) - Solución Múltiple
- c) - No existe solución

Algunas condiciones equivalentes a los casos anteriores son:

- a) Rango de  $[A \ b] = \text{Rango de } [A] = \text{dimensión de } x$
- b) Rango de  $[A \ b] = \text{Rango de } [A] < \text{dimensión de } x$
- c) Rango de  $[A \ b] > \text{Rango de } [A]$

Es importante obtener una forma de calcular la solución para los tres casos anteriores. Anteriormente se efectuaba el cálculo mediante aproximaciones de la inversa de la matriz  $A$  en (3.0-1), ver sección 4.1 de [13]; pero en la actualidad se emplean métodos más prácticos. En este capítulo presentaremos algunos métodos con los cuales podemos efectuar el cálculo de la solución de un sistema de ecuaciones lineales mediante el uso de la computadora.

### **3.1 Método Gauss-Jordan**

Mediante un ejemplo presentaremos cómo el método de Gauss-Jordan calcula la solución de un sistema de ecuaciones lineales.

Considérese el siguiente sistema de ecuaciones



$$0.0001x + 1.0000y = 1.0000 \quad (3.1.1)$$

$$1.0000x + 1.0000y = 2.0000 \quad (3.1-2)$$

El método de Gauss-Jordan resuelve el sistema anterior mediante una serie de operaciones elementales por filas (ver apéndice A.1); es decir haciendo el coeficiente de  $y$  en (3.1-2) y de  $x$  en (3.1-1) igual a cero y los coeficientes restantes igual a la unidad. Esto se logra multiplicando (3.1-1) por 10,000 y al resultado se le resta (3.1-2); sin embargo, supongase que sólo figuren tres dígitos significativos, entonces las nuevas ecuaciones nos quedan como sigue:

$$1.0000x + 0.0000y = 0.0000 \quad (3.1-3)$$

$$0.0000x - 10000 y = -10000 \quad (3.1-4)$$

Ahora, el siguiente paso es dividir la ecuación (3.1-4) por -10000 quedando el siguiente sistema de ecuaciones:

$$1.0000x + 0.0000y = 0.0000 \quad (3.1-5)$$

$$0.0000x + 1.0000y = 1.0000 \quad (3.1-6)$$

La cual tiene por solución  $x=0.0$ ,  $y=1.0$ ; es claro que no satisface el sistema de ecuaciones originales, ya que la solución correcta es  $x=1$ ,  $y=1$ .

Este tipo de problemas aparecen frecuentemente cuando se trata de resolver un sistema de ecuaciones lineales mediante un algoritmo numérico; sobre todo cuando la matriz  $A$  en (3.0-1) es del tipo mal condicionada ó de dimensión muy grande. En realidad éste ejemplo sirve para mostrar la importancia de seleccionar un buen método numérico para calcular la solución de un sistema de ecuaciones, ya que debido principalmente a redondeos y truncamientos numéricos realizados por la computadora podemos obtener soluciones

erróneas del sistema.

### 3.2 Solución por Factorización LU

Un método más efectivo para resolver sistemas de ecuaciones lineales consiste en factorizar la matriz A de (3.0-1) como sigue:

$$A = LU = \begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ l_{n1} & l_{n2} & \dots & \dots & l_{nn} \end{pmatrix} \begin{pmatrix} 1 & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & 1 & u_{23} & \dots & u_{2n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \quad (3.2-1)$$

suponiendo que la factorización anterior puede ser realizada (ver apéndice A.5), entonces podemos sustituir en la ecuación (3.0-1) la factorización anterior:

$$LUx = b \quad (3.2-2)$$

Haciendo  $Ux = y$ , obtenemos:

$$Ly = b \quad (3.2-3)$$

Es claro que debido a la forma de la matriz L en (3.2-1) podemos obtener la solución de (3.2-3) haciendo  $y_1 = b_1 / l_{11}$  y después la segunda ecuación involucra a  $y_1$  y  $y_2$  de tal forma que podemos obtener el valor de  $y_2$  y así sucesivamente hasta  $y_n$ . Este procedimiento es llamado sustitución hacia adelante y la ecuación general para la obtención de la solución es la siguiente:

$$y_1 = \frac{b_1}{l_{11}}$$

$$y_i = \frac{b_i - \sum_{j=1}^{i-1} l_{ij} y_j}{l_{ii}}$$

donde:  $i=2, \dots, n$

Una vez obtenida la solución para  $y$ ; podemos obtener la solución para  $x$  de la ecuación  $Ux = y$ . Debido a la estructura triangular superior de la matriz  $U$ , es evidente que  $x_n = y_n$ ; después la ecuación anterior involucra a  $x_n$  y  $x_{n-1}$  de tal forma que se puede obtener la solución de  $x$  como:

$$x_n = y_n$$

$$x_i = y_i - \sum_{j=i+1}^n u_{ij} x_j$$

donde:  $j=n-1, n-2, \dots, 1$

Este método de cálculo es más efectivo que el Gauss-Jordan pero sigue siendo restrictivo ya que sólo es válido para matrices  $A$  en (3.0-1) que son no singulares, es decir, sistemas de ecuaciones lineales que tienen solución única.

### 3.3 Solución por Descomposición en Valores Singulares

El método más confiable en la actualidad para resolver un sistema de ecuaciones lineales es mediante la descomposición en valores singulares. Este método consiste en factorizar la matriz  $A$  de (3.0-1) como  $A = U\Sigma V^T$  donde  $\Sigma$  es una matriz diagonal compuesta por los valores singulares (ver apéndice A.6). Este método es el más general ya que toda matriz puede descomponerse en valores singulares y por lo tanto no existe ninguna restricción para su aplicación. Entonces, efectuando dicha factorización y sustituyendo en la ecuación (3.0-1) tenemos:

$$U\Sigma V^T x = b \tag{3.3-1}$$

siendo la solución para el caso general:

$$x = V\Sigma^+ U^T b \tag{3.3-2}$$

donde  $\Sigma^+$  es la pseudoinversa de  $\Sigma$  como se definió en el apéndice A.6. El método anterior resuelve los dos casos existentes para la solución de un sistema de ecuaciones lineales; esto

es, cuando existe solución única ó solución múltiple. Además de lo anterior, cuando no existe solución, podemos obtener la mejor aproximación en el sentido de mínimos cuadrados como veremos a continuación.

### *Mínimos Cuadrados*

Como se mostró anteriormente, existen casos cuando el sistema de ecuaciones no tiene solución (sistema inconsistente), por lo que se requiere calcular la mejor aproximación en cierto sentido. Mínimos cuadrados obtiene la mejor aproximación en norma euclídeana. El problema de mínimos cuadrados consiste en determinar las variables  $x$ , tales que  $Ax$  sea lo más cercano posible a  $b$  en la ecuación (3.0-1). El problema de mínimos cuadrados busca encontrar la  $x$  que minimiza el error  $\|Ax - b\|_2$  entre todas las  $x$  posibles. Supongase que la matriz  $A$  se descompone en valores singulares; entonces, la ecuación (3.0-1) nos queda como:

$$U\Sigma V^T x = b \quad (3.3-3)$$

haciendo  $y = V^T x$  y calculando la norma euclídeana del error, obtenemos:

$$\|\Sigma y - U^T b\|_2 \quad (3.3-4)$$

en donde ahora la variable con respecto a la cual se está minimizando es  $y$ . Por lo tanto,  $x$  minimiza a  $\|Ax - b\|_2$  si y sólo si,  $y$  minimiza a  $\|\Sigma y - U^T b\|_2$ . Haciendo  $p = U^T b$  tenemos que:

$$\begin{aligned} \|\Sigma y - p\|_2^2 &= |\sigma_1 y_1 - p_1|^2 + \dots + |\sigma_r y_r - p_r|^2 \\ &+ |p_{r+1}|^2 + \dots + |p_m|^2 \end{aligned} \quad (3.3-5)$$

donde:

- $r$  - rango de  $A$
- $\sigma_i$  -  $i$ -ésimo valor singular
- $y_i$  -  $i$ -ésimo elemento de  $y$
- $p_i$  -  $i$ -ésimo elemento de  $p$

Esta ecuación obtiene su valor mínimo haciendo cero tantos términos como sea posible, el detalle es que sólo podemos manipular los elementos de  $y$ . Entonces haciendo  $y_i = p_i / \sigma_i$  para  $1 \leq i \leq k$  y  $y_i$  arbitrario para  $k+1 \leq i \leq m$  obtenemos el valor mínimo. Como

$\|x\|_2 = \|Vy\|_2 = \|y\|_2$  ya que  $V$  es unitaria; por lo que la  $x$  que minimiza el error esta dada por  $x = Vy$ . Como podemos observar la solución al problema de mínimos cuadrados es precisamente la solución de un sistema de ecuaciones lineales calculada mediante la descomposición en valores singulares. La prueba anterior puede hacerse mediante otro enfoque utilizando el teorema de la proyección, ver sección 3.3 de [19]. Es fácil notar que para el caso en que el rango de la matriz  $A$  en (3.0-1) es completo, el error es igual a cero, o bien el sistema tiene solución única. En seguida presentaremos un ejemplo donde obtenemos la mejor aproximación en el sentido de mínimos cuadrados.

*Ejemplo 1:* Se requiere encontrar un linea recta que mejor aproxime al siguiente conjunto de puntos:

x	1	2	10	-1	5	7
y	.1	.3	.9	0	.4	.8

En otras palabras se requiere encontrar los parámetros de una recta  $y = ax + b$  que mejor aproximen al conjunto de puntos; es decir, encontrar la mejor aproximación en norma euclídeana de el siguiente sistema de ecuaciones.

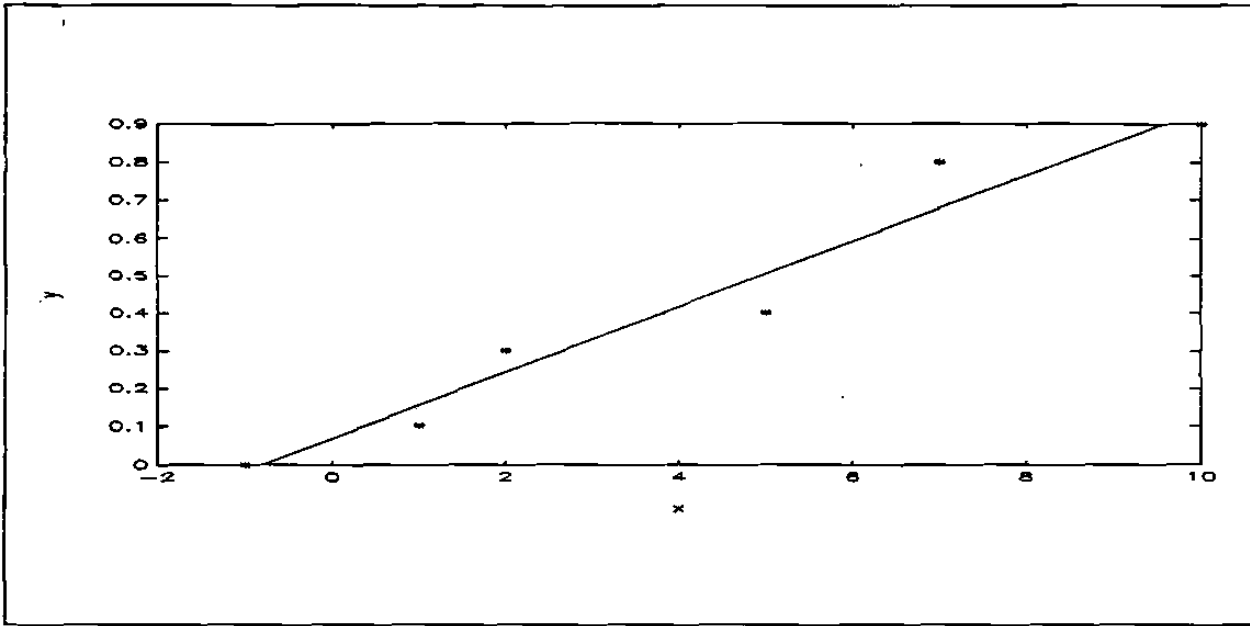
$$\begin{aligned}
 a + b &= .1 \\
 2a + b &= .3 \\
 10a + b &= .9 \\
 -a + b &= 0 \\
 5a + b &= .4 \\
 7a + b &= .8
 \end{aligned}$$

Aplicando la descomposición en valores singulares obtenemos la solución:

$$\begin{aligned}
 a &= 0.0869 \\
 b &= 0.0690
 \end{aligned}$$

Por lo tanto  $y = .0869x + .069$ , la figura 3.1 presenta la gráfica que muestra los resultados obtenidos.





*Figura 3.1 La Mejor Aproximación por una línea recta*

El método anterior puede aplicarse igualmente a relaciones no-lineales en los cuales los parámetros entran linealmente en la ecuación; como se mostrará en el siguiente ejemplo.

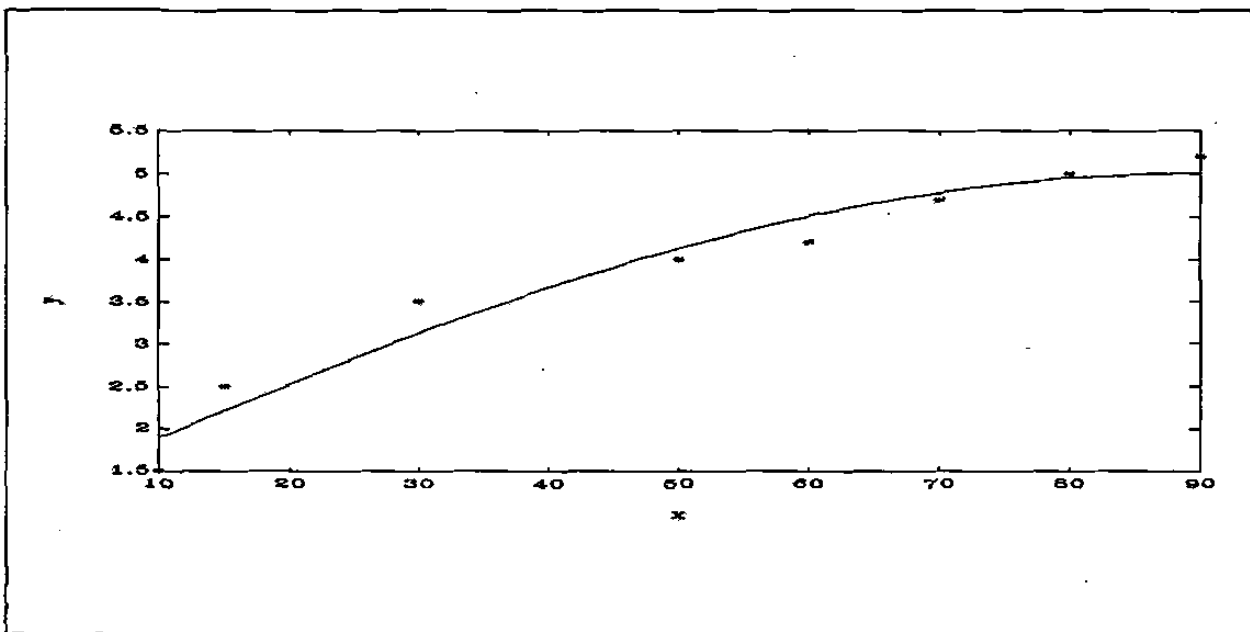
*Ejemplo 2:* Aproximar el conjunto de puntos presentado en la siguiente tabla, por la ecuación  $y = a \sin x + b$ .

x	10	15	30	50	60	70	80	90
y	1.5	2.5	3.5	4	4.2	4.7	5	5.2

El problema se puede plantear como encontrar la mejor aproximación a la solución del siguiente sistema de ecuaciones lineales.

$$\begin{aligned}
0.1736a + b &= 1 \\
0.2588a + b &= 2 \\
0.5000a + b &= 2.5 \\
0.7660a + b &= 3 \\
0.8660a + b &= 3.5 \\
0.9397a + b &= 5 \\
0.9848a + b &= 6 \\
1.0000a + b &= 6.5
\end{aligned}$$

El cual tiene por solución,  $y = 3.7682 \sin(x) + 1.2396$  cuya gráfica es:



*Figura 3.2 La Mejor Aproximación por una función senoidal*

## Referencias de Cálculo Numérico

*Para el cálculo de la solución de un sistema de ecuaciones lineales se elaboró el programa AXB.EXE el cual se apoya principalmente en la subrutina ZSVDC de LINPACK [8] que realiza la decomposición en valores singulares de una matriz. Es importante mencionar que*

el programa *AXB.EXE* resuelve los tres casos que se presentan en sistemas de ecuaciones lineales. El formato de los datos de entrada y salida es el siguiente:

Datos de Entrada

*na*     $na \in \mathbb{R}^2$ , dimensión de la matriz *A* de (3.0-1)

$na(1) = m =$  numero de filas

$na(2) = n =$  numero de columnas

*A*     $A \in \mathbb{R}^{m \times n}$ , elementos de la matriz *A* de (3.0-1) ordenados por filas

Datos de Salida

*x*     $x \in \mathbb{R}^{n \times 1}$ , solución del sistema de ecuaciones lineales

## *Análisis Cuantitativo y Cualitativo de Sistemas Lineales*

---

### *4.1 Análisis Cuantitativo*

Existen dos tipos de análisis que se llevan a cabo dentro de los sistemas de control lineal: el análisis cuantitativo y el análisis cualitativo. El primero consiste en resolver explícitamente un sistema de ecuaciones diferenciales lineales y obtener mediante esta forma, la respuesta dinámica de dicho sistema. El segundo consiste en obtener propiedades de las ecuaciones diferenciales y algebraicas del sistema, como controlabilidad, observabilidad, estabilidad, etc; las cuales se presentarán en la sección 4.2.

#### *4.1.1 Valores y Vectores Propios*

Los valores y vectores propios de matrices tienen una importancia fundamental en el análisis de sistemas lineales. Estos conceptos básicos se estudian desde una gran diversidad de puntos de vista, como es frecuente el caso en que el modelo depende de un parámetro desconocido, cuyo valor debe seleccionarse para hacer singular una matriz determinada, situación presente al modelar ciertos fenómenos oscilatorios, ver sección 7.1 de [21]. Por otro lado los valores propios de la matriz que compone el sistema dinámico de ecuaciones

lineales, son los que determinan la estabilidad de dicho sistema; propiedad muy importante en sistemas de control automático.

*Definición* Los *valores propios* de una matriz  $A \in \mathbb{C}^{n \times n}$  son los  $\lambda_i \in \mathbb{C}$ , para los cuales existe un vector  $v_i \neq 0$  tal que  $Av_i = \lambda_i v_i$ .

*Definición* Los *vectores propios derechos* de una matriz  $A \in \mathbb{C}^{n \times n}$  son los vectores  $v_i \neq 0$ , para los que existe un valor de  $\lambda_i \in \mathbb{C}$  tal que  $Av_i = \lambda_i v_i$ .

*Definición* Los *vectores propios izquierdos* de una matriz  $A \in \mathbb{C}^{n \times n}$  son los vectores  $w_i \neq 0$ , para los que existe un valor de  $\lambda_i \in \mathbb{C}$  tal que  $w_i^H A = \lambda_i w_i^H$ .

*Definición* El *polinomio característico* de una matriz  $A$  se define como el polinomio cuyas raíces son los valores propios de  $A$  y se calcula como el  $\det(\lambda I - A)$ . El polinomio característico se representa como:

$$P(\lambda) = \lambda^n + \alpha_{n-1} \lambda^{n-1} + \dots + \alpha_0$$

## Referencias de cálculo numérico

El cálculo numérico de los valores propios de una matriz, es realizado mediante el programa VALPRO.EXE el cual es auxiliado de las subrutinas BALANC, ORTHES, HQR de EISPACK, [10]. El formato de los datos de entrada y salida es el siguiente:

### Datos de Entrada

$n$   $n \in \mathbb{N}$ , numero de filas de la matriz a analizar

$A$   $A \in \mathbb{R}^{n \times n}$ , matriz cuyos valores propios queremos obtener y debera de estar ordenada por filas

### Datos de Salida

$vp$   $vp \in \mathbb{R}^{n \times 2}$  que contiene la parte real e imaginaria de los valores propios de la matriz  $A$

$vp(i,1) = \text{parte real}$

$vp(i,2) = \text{parte imaginaria}$

$i = 1, \dots, n$

#### 4.1.2 Respuesta Transitoria

La respuesta transitoria de un sistema dinámico es una propiedad cuantitativa muy importante dentro de los sistemas de control ya que nos proporciona la evolución del sistema conforme transcurre el tiempo. Antes de caracterizar la respuesta transitoria de un sistema lineal invariante en el tiempo, presentaremos algunas definiciones importantes para su comprensión.

*Concepto de Estado:* El estado de un sistema es la estructura matemática que contiene un conjunto de variables  $x_1(t_0), x_2(t_0), \dots, x_n(t_0)$ , llamadas variables de estado, tal que los valores iniciales  $x_i(t_0)$  de éste conjunto y las entradas del sistema  $u_i(t)$  son suficientes para describir unívocamente la respuesta futura del sistema para todo  $t \geq t_0$ .

*Espacio de estado:* Se define como un espacio n-dimensional en el cual los componentes del vector de estado representan los ejes coordenados.

*Trayectoria de estado:* Trayectoria de estado es definida como el comportamiento producido en el espacio de estado  $x(t)$  cuando éste cambia con respecto al tiempo. Espacio de estado y trayectoria de estado en el caso de dos dimensiones es llamado plano de fase y trayectoria de fase respectivamente.

*Ecuación dinámica:* Es el conjunto de ecuaciones que describen la relación única entre la entrada, salida y el estado. En nuestro caso sólo se analizarán las ecuaciones dinámicas lineales, y se representan como sigue:

caso continuo:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{4.1.2-1}$$

$$x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^p, y(t) \in \mathbb{R}^q, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times p}, C \in \mathbb{R}^{q \times n}, D \in \mathbb{R}^{q \times p}$$

caso discreto:

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k)\end{aligned}\tag{4.1.2-2}$$

$$x(k) \in \mathbb{R}^n, u(k) \in \mathbb{R}^p, y(k) \in \mathbb{R}^q, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times p}, C \in \mathbb{R}^{q \times n}, D \in \mathbb{R}^{q \times p}$$

El sistema de ecuaciones (4.1.2-1) tiene solución única y se puede obtener mediante la siguiente fórmula, ver sección 4.2 de [5]:

$$x(t) = e^{At}x(t_0) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau\tag{4.1.2-3}$$

donde evaluando el valor de  $x(t)$  para un determinado intervalo de tiempo  $[t_0, t]$  podemos obtener la trayectoria de estado del sistema que nos muestra su comportamiento durante ese intervalo de tiempo. A continuación se presenta la forma de obtener numéricamente el valor de  $x(t)$ .

### 4.1.3 Evaluación numérica de $x(t)$

a) - Calcular  $e^{At}$  para un tiempo determinado  $t_a$  mediante el método de series:

$$e^{At_a} = I + \frac{At_a}{1!} + \frac{(At_a)^2}{2!} + \frac{(At_a)^3}{3!} + \dots + \frac{(At_a)^k}{k!} + \dots\tag{4.1.2-4}$$

b) - Calcular la integral de  $e^{At}$  como:

$$\int_0^t e^{A\tau} d\tau = It + \frac{At^2}{2!} + \frac{A^2t^3}{3!} + \dots \quad (4.1.2-5)$$

c) - Efectuar el cálculo de  $x(t_a)$

$$x(t_a) = e^{At}x(t_0) + \left( \int_0^t e^{A\tau} d\tau \right) Bu$$

de esta forma obtenemos el valor de  $x(t_a)$  para un tiempo determinado  $t_a$ . Procediendo con la misma secuencia a,b,c con diferentes valores de  $t_a$  se obtendrá la respuesta total del sistema. Es necesario hacer notar que el método anterior sólo funciona con valores de  $u(\tau)$  constantes ya que de lo contrario no sería posible sacar fuera del integrando el término  $Bu(\tau)$ .

Por otro lado es importante mencionar que tanto los valores propios como los vectores propios de la matriz A en (4.1.2-1) determinan la respuesta transitoria de un sistema lineal invariante en el tiempo; la salida de un sistema de control, cuando los valores propios son diferentes (caso genérico), podemos obtenerla de la siguiente ecuación, ver sección 18.8 de [7].

$$y(t) = \sum_{j=1}^n (Cv_j)e^{\lambda_j t} w_j^T x(0) + \sum_{j=1}^m \sum_{i=1}^n (Cv_j)(w_i^T b_j) \int_0^t e^{\lambda_i \tau} u_j(t-\tau) d\tau \quad (4.1.2-6)$$

De lo anterior podemos observar claramente que la salida  $y(t)$  es una combinación lineal de n funciones de la forma:

$$v_i e^{\lambda_i t} \quad (4.1.2-7)$$

Por consiguiente se puede concluir que la respuesta transitoria de un sistema lineal invariante en el tiempo queda determinada por los valores propios quienes determinan la rapidez de respuesta del sistema, y los vectores propios quienes determinan la forma de respuesta del sistema. Si queremos modificar la respuesta transitoria de un sistema lineal invariante en el tiempo, lo obtendremos variando estos parámetros.



## Referencias de cálculo numérico

El cálculo numérico de la respuesta al escalón de un sistema de ecuaciones diferenciales lineales invariante en el tiempo es realizada mediante el programa RTRANS.EXE el cual realiza el cálculo aplicando el método presentado en la sección 4.1.3 apoyándose principalmente en las subrutinas EXPSER, EXPINT de ORACLS [2]. El formato de los datos de entrada es el siguiente:

### Datos de Entrada

- $n$      $n \in \mathbb{N}$  número de filas de la matriz  $A$  de (4.1.2-1)
- $A$      $A \in \mathbb{R}^{n \times n}$ , matriz  $A$  de (4.1.2-1)
- $p$      $m \in \mathbb{N}$  número de columnas de la matriz  $B$  de (4.1.2-1)
- $B$      $B \in \mathbb{R}^{n \times m}$ , matriz  $B$  de (4.1.2-1)
- $nc$      $nc \in \mathbb{N}^2$ , dimensión de la matriz  $C$  de (4.1.2-1)
- $nc(1) = q$   
           $nc(2) = n$
- $C$      $C \in \mathbb{R}^{q \times m}$ , matriz  $C$  de (4.1.2-1)
- $t_{max}$      $t_{max} \in \mathbb{R}$ , tiempo máximo de simulación

### Datos de Salida

- $y$      $y \in \mathbb{R}^{100 \times q}$ , matriz que contiene la simulación de la salida  $y = Cx$ . El valor de  $y$  es evaluada en 100 puntos equidistantes partiendo de 0 hasta  $t_{max}$

$\in \mathbb{R}^{100}$ , contiene los puntos donde  $x(t)$  fue evaluado

## 4.2 *Análisis Cualitativo*

El análisis cualitativo de sistemas lineales nos proporciona algunas propiedades importantes del sistema, sin necesidad de obtener la solución del mismo. Lo importante es que en base a éstas propiedades podemos hacer algunas conclusiones sobre el sistema, tales como si es posible controlar todos los modos de un sistema mediante cierta entrada o si es posible obtener una estimación de las variables de estado de un sistema de ecuaciones lineales, dichas propiedades son controlabilidad y observabilidad respectivamente. El análisis cualitativo incluye también la propiedad de estabilidad; la más importantes en un sistema de control. En seguida definiremos estos conceptos.

### 4.2.1 *Estabilidad*

Una de las propiedades cualitativas más importantes de los sistemas de control es la estabilidad. Para el caso de sistemas lineales invariantes en el tiempo, la estabilidad está determinada por los valores propios de la matriz  $A$  en (4.1.2-1). Presentaremos a continuación el concepto formal.

*Definición* Un estado  $x^*$  es un *punto de equilibrio* de un sistema, si una vez que la trayectoria de estado  $x(t) = x^*$ , ésta permanece igual a  $x^*$  para todo tiempo futuro, ver sección 2.1.2 de [27].

*Definición* Un sistema de ecuaciones dinámicas lineales es *estable* si todos los estados convergen a los puntos de equilibrio. El sistema de ecuaciones dinámicas (4.1.2-1) es estable si y sólo si, los valores propios de  $A$  tienen parte real negativa; es decir  $\lambda(A) \in \mathbb{C}_-$ .

## Referencias de cálculo numérico

El cálculo numérico que sirve para verificar la estabilidad de un sistema lineal invariante en el tiempo es realizado mediante el programa ESTABLE.EXE que consiste en obtener los valores propios de la matriz  $A$  y verifica si estos tienen parte real negativa o valor absoluto menor a 1 para el caso continuo y discreto respectivamente. El formato de los datos es el siguiente:

### Datos de Entrada

- $n$      $n \in \mathbb{N}$ , número de filas de la matriz a analizar
- $A$      $A \in \mathbb{R}^{n \times n}$ , matriz cuya estabilidad será analizada

### Datos de Salida

- $est$      $est \in \mathbb{N}$
- $est = 1$       el sistema es estable
- $est = 0$       el sistema inestable

## 4.2.2 Controlabilidad y Observabilidad

**Definición** El sistema de ecuaciones dinámicas (4.1.2-1) es completamente *controlable* si para cualquier tiempo  $t_0$ , cada estado inicial  $x(t_0)$  puede transferirse a cualquier estado final  $x(t_f)$  en un tiempo finito  $t_f > t_0$  por medio de un vector de entrada  $u(t)$   $t \in [t_0, t_f]$ .

El sistema de ecuaciones dinámicas (4.1.2-1) es controlable si y sólo si, la matriz de controlabilidad  $U = [B \ AB \ \dots \ A^{n-1}B]$  tiene rango  $n$ . Es muy común referirse a la controlabilidad del sistema de ecuaciones dinámicas como el par  $(A,B)$ .

*Definición* El sistema de ecuaciones dinámicas (4.1.2-1) es completamente *observable* si cada estado inicial  $x(t_0)$  puede determinarse exactamente mediante las mediciones de la salida  $y(t)$  y la entrada  $u(t)$ , en un intervalo finito de tiempo  $[t_0, t_f]$ .

El sistema de ecuaciones (4.1.2-1) es observable si y sólo si, la matriz de observabilidad  $V = [C^T \ A^T C^T \ \dots \ (A^T)^{n-1} C^T]^T$  tiene rango  $n$ . De nuevo, con frecuencia se refiere a la observabilidad de un sistema como el par  $(A,C)$ .

### *Dualidad*

Sea el siguiente sistema de ecuaciones dinámicas:

$$\begin{aligned}\dot{z} &= A^T z + C^T u \\ w &= B^T z\end{aligned}\tag{4.2.2-1}$$

*Teorema de Dualidad:* La ecuación dinámica (4.1.2-1) es controlable (observable) si y sólo si, la ecuación dinámica (4.2.2-1) es observable (controlable), ver sección 3.4 de [5].

## Referencias de cálculo numérico

*El cálculo numérico de la controlabilidad de un sistema de control lineal invariante en el tiempo es realizado con el programa CTRL.EXE basada en la subrutina CTROL de ORACLS [2]. El formato de los datos es el siguiente:*

### Datos de Entrada

- $n$       $n \in \mathbb{N}$ , número de filas de la matriz  $A$  de (4.1.2-1)
- $A$       $A \in \mathbb{R}^{nm}$ , matriz  $A$  de (4.1.2-1) ordenada por columnas

$m$   $m \in \mathbb{N}$  número de columnas la matriz  $B$  de (4.1.2-1)

$B$   $B \in \mathbb{R}^{n \times m}$ , matriz  $B$  de (4.1.2-1) ordenada por columnas

#### Datos de Salida

$ctrl$   $ctrl \in \mathbb{N}$

$ctrl = 1$  el sistema es controlable

$ctrl = 0$  el sistema es no controlable

*Nota: la prueba de observabilidad se puede realizar con este mismo programa aplicando el teorema de dualidad.*

### 4.2.3 Estabilizabilidad y Detectabilidad

En ésta sección introduciremos conceptos más generales que los mencionados anteriormente; estabilizabilidad y detectabilidad. Estos dos nuevos conceptos son importantes al aplicar técnicas de control óptimo ya que para que éste problema tenga solución, es necesario que el sistema sea estabilizable ó detectable segun sea el caso.

**Definición** El sistema de ecuaciones dinámica (4.1.2-1) es *estabilizable* si sus modos no controlables son estables.

Una forma de probar estabilizabilidad es la siguiente:

$$U^* \triangleq [\lambda I - A : B]$$

Se dice que el sistema de ecuaciones dinámicas (4.1.2-1) es estabilizable si y sólo si, el rango de  $U^* = n$ , para todo  $\lambda_i \in \mathbb{C}_+$ .

**Definición** El sistema de ecuaciones (4.1.2-1) es *detectable* si sus modos no

observables son estables.

Una vez más una forma de probar detectabilidad del sistema (4.1.2-1) es como sigue:

$$V^* \triangleq [\lambda I - A^T : C^T]$$

Se dice que el sistema de ecuaciones dinámicas (4.2.0-1) es detectable si y sólo si, el rango de  $V^*=n$ , para todo  $\lambda_i \in \mathbb{C}_+$ .

En otras palabras el concepto de estabilizabilidad y detectabilidad significa que es posible ya sea aplicar una ley de control para estabilizar un sistema o bien estimar las variables de estado de un sistema si dicho sistema es estabilizable o detectable respectivamente; ya que los modos que no se pueden controlar u observar son estables.

## Referencias de cálculo numérico

*El cálculo numérico de la estabilizabilidad de un sistema de control lineal invariante en el tiempo es realizado con el programa ESTBD.EXE que construye la matriz  $U^*$  y después se verifica su rango mediante la descomposición en valores singulares. El formato de los datos es el siguiente:*

### Datos de Entrada

- $n$      $n \in \mathbb{N}$  número de filas de la matriz  $A$  de (4.1.2-1)
- $A$      $A \in \mathbb{R}^{n \times n}$ , matriz  $A$  de (4.1.2-1) ordenada por columnas
- $m$      $m \in \mathbb{N}$  número de columnas la matriz  $B$  de (4.1.2-1)
- $B$      $B \in \mathbb{R}^{n \times m}$ , matriz  $B$  de (4.1.2-1) ordenada por columnas

### Datos de Salida

*est*     $est \in \mathbb{N}$

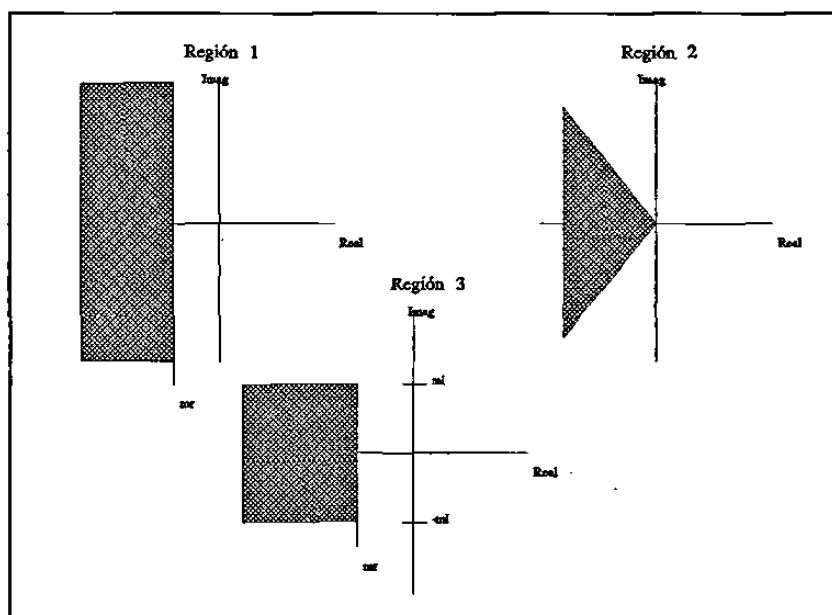
*est* = 1    *el sistema es estabilizable*

*est* = 0    *el sistema no es estabilizable*

*Nota: al igual que el caso de observabilidad la prueba de detectabilidad se puede realizar mediante este programa.*

#### 4.2.4 Estabilidad Relativa

Algunas veces es necesario definir regiones de estabilidad dentro del semiplano izquierdo del plano complejo y poder determinar mediante alguna forma, si el sistema de control se encuentra dentro de esa región definida; a este problema se le conoce como estabilidad relativa. En esta sección presentaremos tres tipos de regiones que pueden ser analizadas en computadora.



*Figura 4.1 Regiones de Estabilidad*

Nótese que para garantizar estabilidad relativa es necesario que el sistema sea estable, de lo contrario no podrá ser relativamente estable.

## Referencias de cálculo numérico

El cálculo numérico es realizado mediante el programa ESTREL.EXE al cual se le indica la región de estabilidad y el programa se encarga de verificar si el sistema es relativamente estable o no. El formato de los datos es el siguiente:

### Datos de Entrada

$n$      $n \in \mathbb{N}$  número de filas de la matriz a analizar

$A$      $A \in \mathbb{R}^{n \times n}$ , matriz a analizar ordenada por filas

$mr$      $mr \in \mathbb{R}$ , máximo valor real permitido (ver figura 4.1)

$mi$      $mi \in \mathbb{R}$ , máximo valor imaginario permitido (ver figura 4.1)

$tp$      $tp \in \mathbb{N}$  tipo de región a analizar

$tp=1$     región 1 (ver figura 4.1)

$tp=2$     región 2 (ver figura 4.1)

$tp=3$     región 3 (ver figura 4.1)

### Datos de Salida

$er$      $er \in \mathbb{N}$

$er = 1$     el sistema es relativamente estable

$er = 0$     el sistema es relativamente inestable



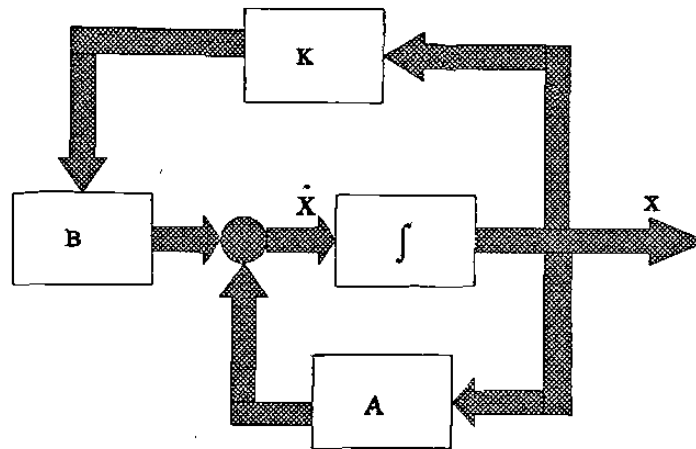
## *Asignación de Polos*

Dentro de un sistema de control, con frecuencia es necesario alterar su respuesta transitoria con el objeto de obtener el comportamiento deseado de dicho sistema. En el caso de análisis entrada-salida, se utilizan algunas técnicas de compensación para modificar la respuesta transitoria del sistema, ver [22], [17]. Para el caso de análisis en el dominio del tiempo, una de las técnicas más comunes de alterar la respuesta transitoria de un sistema de control es mediante la aplicación de una retroalimentación de estado, como se presenta en seguida (ver también figura 5.0-1):

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (5.0-1)$$

$$u(t) = Kx(t) \quad (5.0-2)$$

$$A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, K \in \mathbb{R}^{m \times n}, x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m$$



*Figura 5.0-1 Sistema con retroalimentación de estado*

Como se mencionó en el capítulo anterior la respuesta dinámica de un sistema lineal invariante en el tiempo está determinada únicamente por los valores y vectores propios de la matriz  $A$  en (5.0-1), de tal forma que para modificar su respuesta transitoria es necesario modificar dichos parámetros mediante de la entrada; lo anterior puede lograrse solamente si el sistema (5.0-1) es controlable. Si el sistema de ecuaciones dinámicas (5.0-1) es controlable, entonces podemos encontrar un valor  $K$ , de tal forma que al sustituir (5.0-2) en (5.0-1) el sistema resultante  $\dot{\hat{x}} = (A + BK)x$  tenga los valores propios deseados y por lo tanto la respuesta dinámica requerida. En este capítulo se presentará la forma de obtener la matriz de retroalimentación de estado  $K$ ; tanto para sistemas monovariantes como para sistemas multivariantes. Se tratará además el problema de minimizar la norma Frobenius de  $K$  en el caso de sistemas multivariantes.

### **5.1 Retroalimentación de Estado: Caso Monovariante**

Cuando  $m=1$  en la ecuación (5.0-1) se dice que el sistema es monovariante. Para el caso anterior, si el sistema de ecuaciones dinámicas es controlable; existe una solución única al problema de asignación de polos mediante retroalimentación de estado, ver capítulo 7 de

[5]. Para este caso existe un algoritmo con el cual se puede obtener su cálculo.

**Algoritmo 5.1-1**

1. Encontrar el polinomio característico de A:  $\det(\lambda I - A)$ 

$$\lambda^n + \alpha_{n-1} \lambda^{n-1} + \dots + \alpha_0$$
2. Calcular el polinomio característico deseado
 
$$\lambda^n + \bar{\alpha}_{n-1} \lambda^{n-1} + \dots + \bar{\alpha}_0$$
3. Calcular  $k^* = [ \alpha_0 - \bar{\alpha}_0 \quad \alpha_1 - \bar{\alpha}_1 \quad \dots \quad \alpha_{n-1} - \bar{\alpha}_{n-1} ]$
4. Calcular  $q_{n-i} = A q_{n-i} + \alpha_{n-i} q_n$ , para  $i=1, 2, \dots, n-1$ ,  $q_n = B$
5. Formar la matriz
 
$$Q = [ q_1 \quad q_2 \quad \dots \quad q_n ]$$
6. Calcular la retroalimentación de estado que asigna los polos deseados
 
$$k = k^* Q^{-1}$$

## Referencias de cálculo numérico

*El programa que calcula la matriz de retroalimentación de estado se llama RETRO.EXE y se apoya principalmente en la subrutina POLE de ORACLS [2]. Este programa calcula la matriz de retroalimentación de estado así como la respuesta transitoria del sistema retroalimentado y también la respuesta transitoria de la ley de control  $u = kx$ . El formato de los datos de entrada es el siguiente:*

Datos de Entrada

- $n$      $n \in \mathbb{N}$  número de filas de la matriz A del sistema (5.0-1)
- $A$      $A \in \mathbb{R}^{m \times n}$ , matriz A del sistema (5.0-1) ordenada por columnas
- $b$      $b \in \mathbb{R}^n$ , vector B del sistema (5.0-1) cuando el sistema es monovariable

$vdp$   $vdp \in \mathbb{N}^{n \times 2}$ , valores propios deseados para el sistema retroalimentado

$vdp(i,1)$  = parte real

$vdp(i,2)$  = parte imaginaria

### Datos de Salida

$k$   $k \in \mathbb{R}^n$ , matriz de retroalimentación de estado

$AR$   $AR \in \mathbb{R}^{n \times n}$ , matriz del sistema retroalimentado

$$AR = A + BK$$

$x$   $x \in \mathbb{R}^{100n}$ , respuesta transitoria del sistema retroalimentado

$u$   $u \in \mathbb{R}^{100}$ , respuesta transitoria de la ley de control  $u = kx$

## 5.2 Retroalimentación de Estado: Caso Multivariable

Cuando el sistema de control (5.0-1) y (5.0-2) es multivariable, es decir  $m > 1$ , la matriz  $K$  de retroalimentación de estado está determinada por la solución para  $L$  y  $K$  de las siguientes ecuaciones:

$$AL - L\tilde{A} = -BG \tag{5.2-1}$$

$$KL = G \tag{5.2-2}$$

$$B \in \mathbb{R}^{n \times m}, L \in \mathbb{R}^{n \times n}, G \in \mathbb{R}^{m \times n}, \tilde{A} \in \mathbb{R}^{n \times n}$$

donde:

$A, B, K$  - Matrices del sistema de ecuaciones (5.0-1) y (5.0-2)

- $\tilde{A}$  - Matriz cuyos valores propios son los deseados para el sistema retroalimentado
- $G$  - Matriz arbitraria

Obteniendo el valor de  $K$  en (5.2-2) obtendremos también la matriz de retroalimentación de estado que asigna los valores propios deseados; que a su vez serán los polos del sistema retroalimentado.

*Prueba:*

Sustituyendo (5.2-2) en (5.2-1)

$$AL - L\tilde{A} = -BKL \quad (5.2-3)$$

$$AL + BKL = L\tilde{A} \quad (5.2-4)$$

$$(A + BK)L = L\tilde{A} \quad (5.2-5)$$

Si  $L$  es invertible

$$(A + BK) = L\tilde{A}L^{-1} \quad (5.2-6)$$

Debido a que la ecuación (5.2-6) representa una transformación de similitud entre  $(A+BK)$  y  $\tilde{A}$  entonces  $\lambda(A+BK) = \lambda(\tilde{A})$  (ver apéndice A.2); Por lo tanto el sistema retroalimentado  $(A+BK)$  tendrá la dinámica requerida. Las condiciones para la solución de las ecuaciones (5.2-1) y (5.2-2) son:

**C5.2-1**  $(A,B)$  controlable

**C5.2-2**  $\lambda(A) \cap \lambda(\tilde{A}) = \emptyset$

**C5.2-3**  $(\tilde{A},G)$  observable

La condición (C5.2-1) implica que el problema está bien planteado, es decir que podemos asignar la totalidad de los polos mediante la aplicación de una entrada. La condición (C5.2-2) garantiza la solución única de la ecuación (5.2-1) (ver sección 12.3 de

[18]) y por último la condición (C5.2-3) garantiza la invertibilidad de la matriz L, ver [9].

## Referencias de cálculo numérico

*La obtención de la matriz de retroalimentación de estado para sistemas de control multivariantes se realiza; primeramente obteniendo el valor de L en (5.2-1) mediante un método numérico confiable (ver apéndice B.2) y posteriormente obteniendo el valor de K mediante la descomposición en valores singulares (Ver Apéndice A.6). El programa que realiza el cálculo numérico de la matriz de retroalimentación de estado multivariable es RETROMUL.EXE. El formato de los datos es el siguiente:*

### Datos de Entrada

- n*      $n \in \mathbb{N}$  número de filas de la matriz A del sistema (5.0-1)
- A*      $A \in \mathbb{R}^{n \times n}$ , matriz A del sistema (5.0-1) ordenada por columnas
- B*      $B \in \mathbb{R}^{n \times m}$ , vector B del sistema (5.0-1) cuando el sistema es multivariable
- AT*     $AT \in \mathbb{R}^{n \times n}$ , matriz cuyos valores propios son los deseados para el sistema retroalimentado

### Datos de Salida

- K*      $K \in \mathbb{R}^{n \times m}$ , matriz de retroalimentación de estado
- AR*     $AR \in \mathbb{R}^{n \times n}$ , matriz del sistema retroalimentado
- $AR = A + BK$
- x*      $x \in \mathbb{R}^{100m}$ , respuesta transitoria del sistema retroalimentado

$u \in \mathbb{R}^{100m}$ , respuesta transitoria de la ley de control  $u = Kx$

### 5.3 Retroalimentación de Norma Mínima

La libertad que se tiene para seleccionar la matriz  $G$  de la ecuación (5.2-1) nos da la oportunidad de obtener un número indeterminado de soluciones al problema de asignación de polos, lo cual a su vez nos proporciona libertad para agregar condiciones a las ecuaciones (5.2-1) y (5.2-2) para obtener una solución única. En algunos trabajos se reporta el uso de este grado de libertad para seleccionar además de los valores propios también los vectores propios del sistema retroalimentado [20], [12], [24]. En otros se busca obtener un sistema retroalimentado robusto ante perturbaciones en los elementos de la matriz retroalimentada [15]. El objetivo de ésta sección es aprovechar el grado de libertad mencionado anteriormente para minimizar la norma de Frobenius de la matriz de retroalimentación de estado obteniéndose con esto diversas ventajas. El algoritmo consiste en lo siguiente:

Considérese el siguiente sistema de control multivariable

$$\dot{x} = Ax + Bu \quad (5.3-1)$$

$$u = Kx \quad (5.3-2)$$

$$A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, K \in \mathbb{R}^{m \times n}, u \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n$$

Entonces si las condiciones C5.2-1, C5.2-2, C5.2-3 se cumplen, el problema de minimizar la norma de Frobenius de  $K$  se puede plantear como:

minimizar

$$I = \text{traza}\{K^T K\}$$

Sujeto a

$$AL - L\tilde{A} = -BG$$

$$KL = G$$

La solución de éste problema es realizada mediante el siguiente algoritmo cuya deducción se presenta en [16].

*Algoritmo 5.3-1*

Paso 1. Hacer  $k=1$ ,  $G_1 \in \mathbb{R}^{m \times n}$

Paso 2. Resolver para  $L_k$

$$AL_k - L_k\tilde{A} = -BG$$

Paso 3. Obtener  $F_k$  y  $H_k$  de

$$F_k L_k = G_k$$

$$H_k L_k = F_k$$

Paso 4. Calcular  $U_k$  de

$$\tilde{A}U_k - U_k A = H_k^T F_k$$

Paso 5. Calcular

$$\frac{\partial I}{\partial G} = 2(H_k - B^T U_k^T)$$

Si  $\|\partial I / \partial G\|_F > \epsilon$ , entonces efectuar paso 6

Si no, hacer  $K = F_k$  y parar.

Paso 6. Obtener un nuevo valor de  $G_{k+1}$  usando el método basado en el gradiente. Hacer  $k = k+1$  y regresar el paso 2.



### Implementación del Algoritmo 5.3-1

En los pasos 2 y 4 del algoritmo anterior no existe ningún problema para efectuarlos ya que las ecuaciones son idénticas a las ecuaciones (5.2-1) y (5.2-2) por lo tanto se puede usar el mismo procedimiento. Para el paso 3 únicamente es necesario calcular inversas y multiplicaciones de matrices. El paso 5 es completamente sencillo ya que una vez obtenidas las matrices en los pasos anteriores, únicamente se requerirán operaciones suma y multiplicación de matrices. El paso 6 es el que representa un poco más de problema por lo tanto detallaremos la manera de efectuar su cálculo.

*El método de descenso acelerado consiste en lo siguiente:*

Sea el gradiente negativo de una función  $f(x)$  de  $n$  variables evaluado en  $x=x_k$ , definido como:

$$-g(x_k) = - \left| \frac{\partial f(x)}{\partial x} \right|_{x=x_k} = \frac{x_{k+1} - x_k}{\alpha}$$

$x_{k+1}, x_k \in \mathbb{R}^n$

Como es bien conocido  $-g(x_k)$  nos proporciona la dirección hacia donde la función  $f(x)$  disminuye su valor, ver sección 7.2.3 de [25].

Si ordenando términos obtenemos en la ecuación anterior:

$$x_{k+1} = x_k - \alpha g(x_k) \tag{5.3-3}$$

Nótese que si reemplazamos  $x_{k+1}$  en  $f(x_k)$  nos queda una función  $f(x_k - \alpha g(x_k))$ ; que es una función de una sola variable ( $\alpha$ ), entonces el valor de  $x_{k+1}$  será calculado con el valor de  $\alpha$  que satisfaga la siguiente ecuación.

$$\frac{\partial f [x_k - \alpha g(x_k)]}{\partial \alpha} = 0$$

De ésta forma garantizamos que el valor de  $\alpha$  calculado es el óptimo, ver sección 6.6 de [23]. Lo anterior se continúa hasta que  $\|g(x_k)\| < \epsilon$ .

### Aplicación del método de descenso acelerado al problema de norma mínima

Para calcular el nuevo valor de  $G_{k+1}$  del paso 6 en el *algoritmo 5.3-1* consideraremos lo siguiente:

$$I = \text{Traza}(K^T K) = \text{Traza}((GL^{-1})^T (GL^{-1}))$$

Entonces, debido a que se tiene información del valor del gradiente (ver paso 5) podemos aplicar el método de descenso acelerado para encontrar un nuevo valor  $G_{k+1}$  que nos conduzca al punto mínimo. Para facilitar la aplicación del método nos apoyaremos en la siguiente definición:

*Definición* Sea  $A \in \mathbb{R}^{n \times m}$  la siguiente matriz:

$$A = [a_1 \ a_2 \ \dots \ a_m]$$
$$a_i \in \mathbb{R}^n$$

entonces la operación  $\text{vec}(A) \in \mathbb{R}^{nm}$  consiste en lo siguiente:

$$\text{vec}(A) = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix}$$

Nótese que la operación anterior, consiste en formar un vector  $\text{vec}(A)$  de las columnas de la matriz  $A$ . Ahora aplicando la operación anterior a nuestro problema tenemos

$$\text{vec}\left(\frac{\partial I}{\partial G}\right) = \text{vec}(2(H_k - B^T U_k^T))$$

$$\text{vec}(G_{k+1}) = \text{vec}(G_k) - \alpha \text{vec}\left(\frac{\partial I}{\partial G}\right)$$

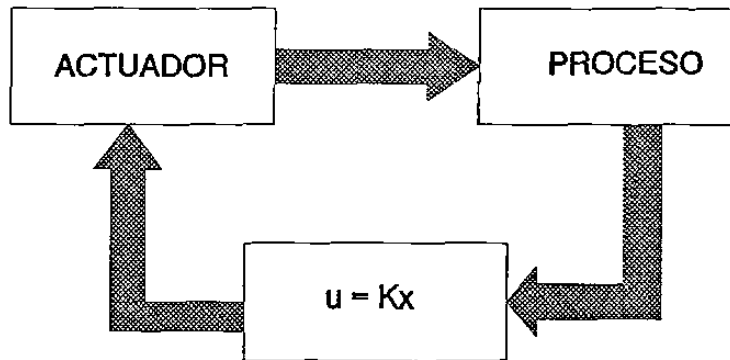
Donde el valor de  $\alpha$  se determina como lo mostramos anteriormente usando el método del gradiente

$$\frac{\partial I}{\partial \alpha} = \frac{\partial [\text{Traza}((G_{k+1} L^{-1})^T (G_{k+1} L^{-1}))]}{\partial \alpha}$$

Se puede utilizar para encontrar el valor de  $\alpha$ , los métodos de búsqueda unidireccional (ver sección 7.2 de [25]) como el de la sección dorada, Fibonacci, etc.

#### 5.4 Reguladores mediante asignación de polos

En esta sección presentaremos algunos ejemplos donde será necesario calcular la matriz de retroalimentación de estado con el fin de asignar los polos de un sistema de control dado. Antes de presentar estos ejemplos es necesario mencionar que un regulador consiste en la aplicación de una ley de control (Ver figura 5.4-1) con la finalidad de llevar a un sistema de un estado diferente de cero al estado cero del mismo, pudiéndose lograr lo anterior mediante una retroalimentación de estado. Uno de los objetivos de diseñar un regulador para un sistema es que una vez retroalimentado el sistema, este tendrá la capacidad de eliminar perturbaciones que puedan reflejarse como cambios de condiciones iniciales.



*Figura 5.4-1 Regulador por retroalimentación de estado*

**Ejemplo 1:** Se desea encontrar la ley de control  $u = kx$ , que permita asignar los polos del sistema retroalimentado en  $\{-1, -2, -1 \pm j\}$  del siguiente sistema:

$$\dot{x} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 5 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \\ 0 \\ -2 \end{pmatrix} u$$

Entonces el problema consiste en encontrar el valor de  $k$ , de tal forma que  $(A + bk)$  tenga los polos deseados. Utilizando el programa mencionado en las referencias de cálculo numérico para el problema de retralimentación de estado monovariable obtenemos el siguiente valor de  $k$ .

$$k = (1.3333 \quad 3.3333 \quad 8.1667 \quad 4.1667)$$

Dónde podemos observar el sistema retroalimentado tiene los valores propios deseados:

$$A + bk = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1.3333 & 3.3333 & 7.1667 & 4.1667 \\ 0 & 0 & 0 & 1 \\ -2.6667 & -6.6667 & -11.3333 & -8.3333 \end{pmatrix}$$

$$\lambda(A + bk) = \{-1, -2, -1 \pm j\}$$

Como observamos, hemos obtenido una ley de control que asigne los valores propios en un lugar previamente establecido. Nótese que los polos del sistema retroalimentado se encuentran en el semiplano izquierdo del plano complejo, por lo tanto el sistema retroalimentado es estable, lo que implica que a partir de condiciones iniciales diferentes de cero el sistema podrá llevar los estados al estado cero. En seguida presentamos algunas simulaciones del sistema retroalimentado con la ley de control calculada en el que mostraremos las gráficas de trayectoria de estado y de la ley de control respectivamente partiendo de condiciones iniciales  $x(0) = [5 \ 5 \ 5]^T$ .

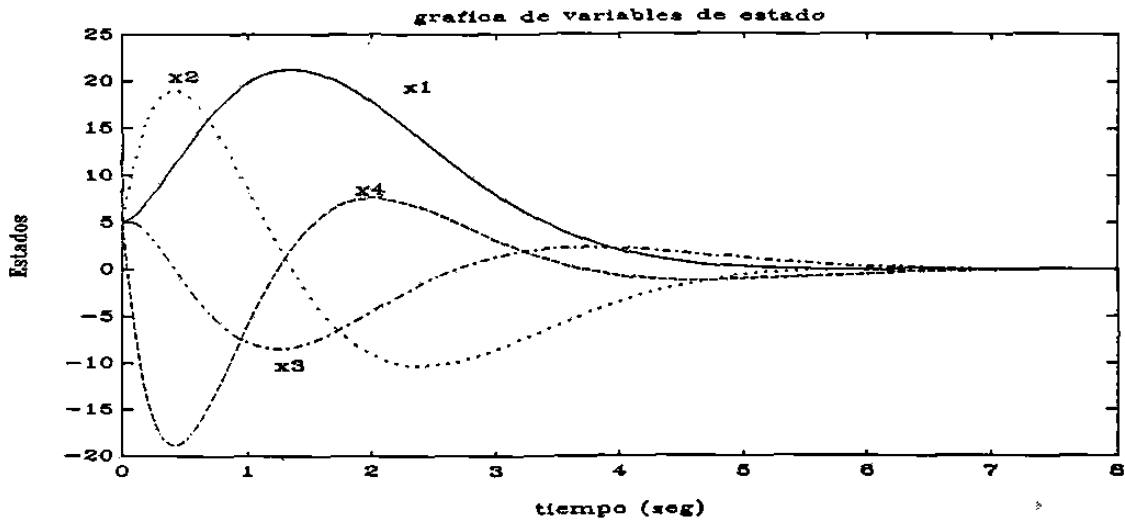


Figura 5.4-2 Gráfica de Trayectoria de Estados

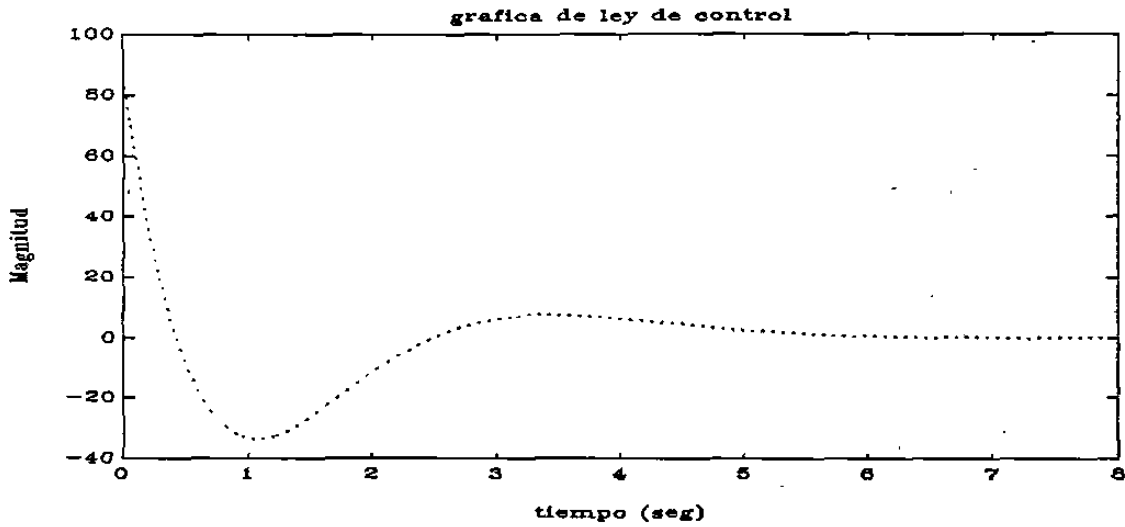


Figura 5.4-3 Gráfica de Ley de Control

**Ejemplo 2** En este ejemplo presentaremos un sistema de control multivariable, donde aplicaremos el método para minimizar la norma Frobenius de la matriz de retroalimentación de estado.

Considerar el siguiente sistema de control:

$$\dot{x} = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 2 & 1 & -1 \end{pmatrix} x + \begin{pmatrix} 0 & 0 \\ 1.5 & 1 \\ -1 & 1 \end{pmatrix} u$$

Es claro que el sistema anterior es inestable ya que tiene sus polos en  $\lambda(A) = \{-1, 1 \pm j\}$ . Entonces se requiere ubicar los polos del sistema retroalimentado en  $\lambda(A + BK) = \{-1.5, -2.5, -3.5\}$ ; para esto se deberá calcular la matriz de retroalimentación de estado K de tal forma que asigne dichos polos. Recordando que para sistemas multivariables la matriz K no es única, a manera de ejemplificación se obtendrá primero una matriz K arbitraria y posteriormente una de norma Frobenius mínima.

a) Resolviendo las ecuaciones (5.2-1) y (5.2-2) cuando:

$$\tilde{A} = \begin{pmatrix} -1.5 & 0 & 0 \\ 0 & -2.5 & 0 \\ 0 & 0 & -3.5 \end{pmatrix} \quad G = \begin{pmatrix} 1 & 1 & -1 \\ 0 & 1.5 & 1 \end{pmatrix}$$

se obtienen los valores:

$$K_1 = \begin{pmatrix} 39.4123 & -12.0481 & 0.7567 \\ -49.0150 & 11.4247 & -1.0958 \end{pmatrix}$$

$$A + BK_1 = \begin{pmatrix} 1 & -1 & 0 \\ 11.1026 & -5.6475 & 0.0393 \\ -86.4282 & 24.4728 & -2.8525 \end{pmatrix}$$

$$\|K_1\|_F = 65.0641$$

donde los polos del sistema retroalimentado son:

$$\lambda(A + BK_1) = \{-1.5, -2.5, -3.5\}$$

b) Ahora calculando la matriz  $K$  de norma Frobenius mínima obtenemos:

$$K_2 = \begin{pmatrix} 3.1138 & -2.0280 & 0.9607 \\ 3.7396 & -3.3580 & -1.1394 \end{pmatrix}$$

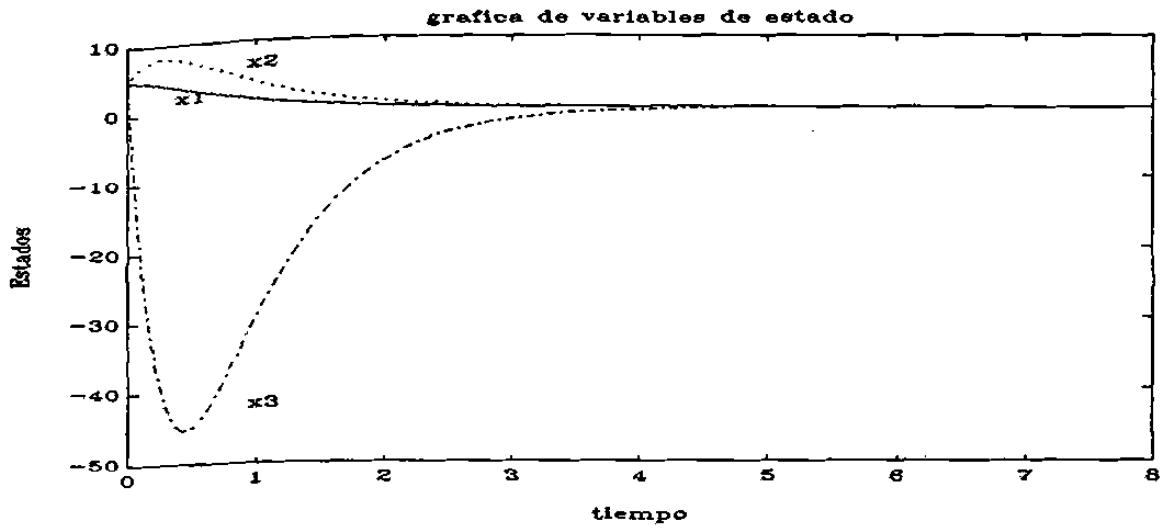
$$A + BK_2 = \begin{pmatrix} 1 & -1 & 0 \\ 9.4103 & -5.3999 & 0.3017 \\ 2.6258 & -0.3300 & -3.1001 \end{pmatrix}$$

$$\|K_2\|_F = 6.4257$$

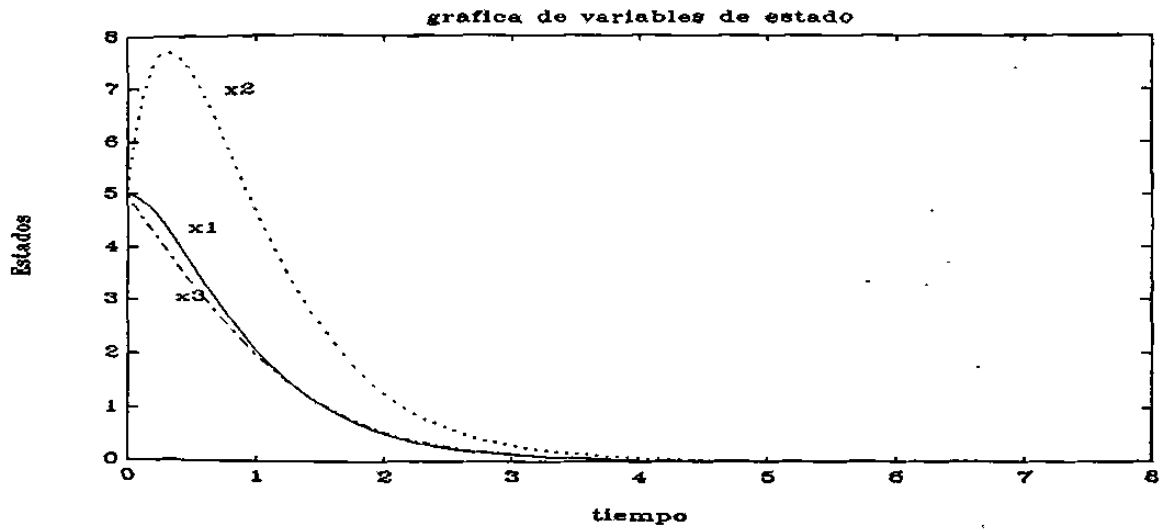
$$\lambda(A + BK_2) = \{-1.5, -2.5, -3.5\}$$

En seguida presentaremos las figuras 5.4-4, 5.4-5 y las figuras 5.4-6, 5.4-7 que muestran los resultados de las simulaciones del sistema retroalimentado en el que aparece la respuesta transitoria y la ley de control respectivamente para ambos casos, considerando condiciones iniciales  $x(0) = [5 \ 5 \ 5]^T$ .





*Figura 5.4-4 Trayectoria de Estado de  $(A + BK_1)$*



*Figura 5.4-5 Trayectoria de Estados de  $(A + BK_2)$*

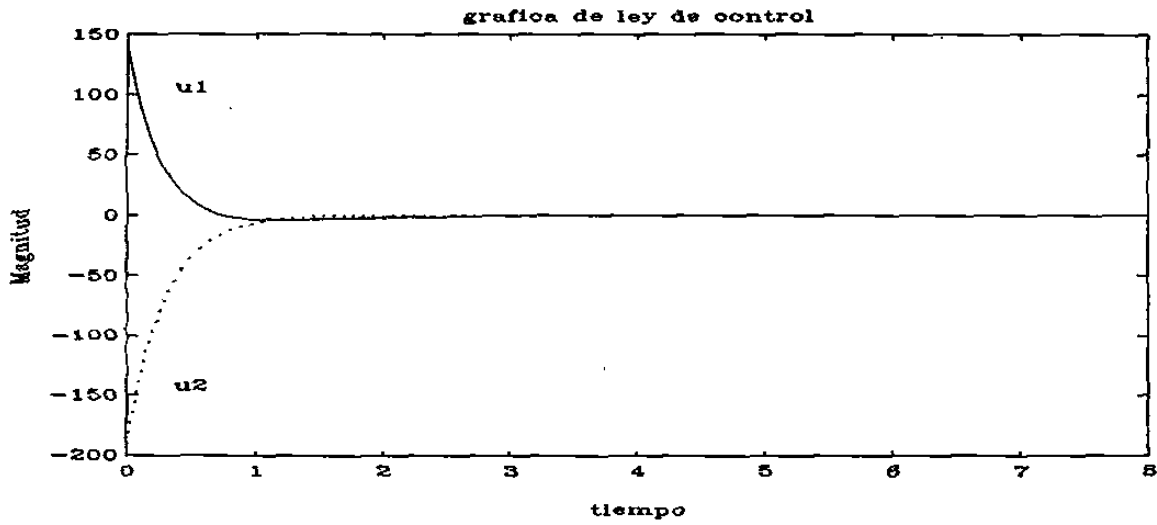


Figura 5.4-6 Gráfica de Ley de Control  $u = K_1 x$

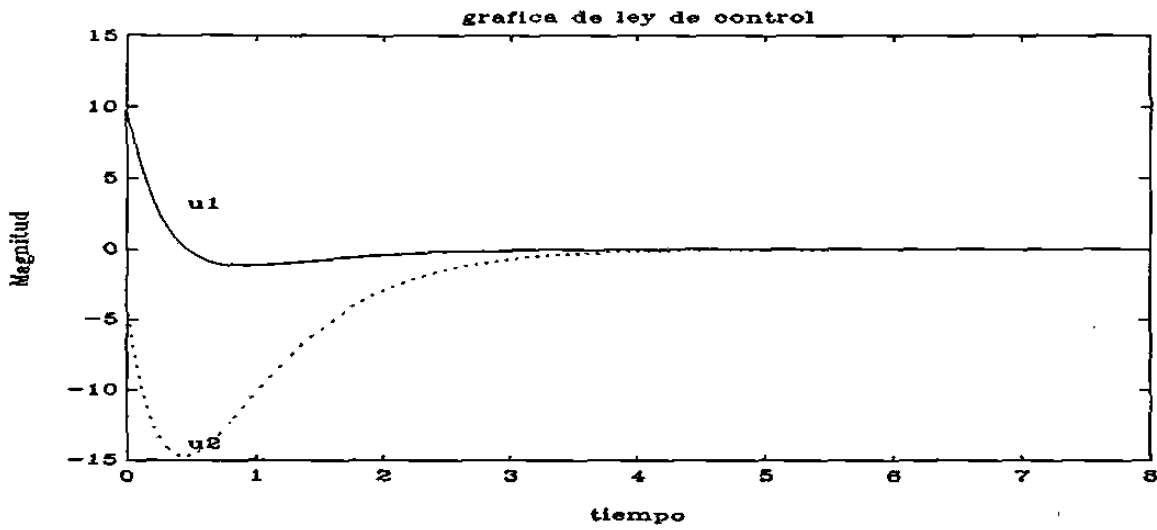


Figura 5.4-7 Gráfica de Ley de Control  $u = K_2 x$

Del resultado presentado en ésta sección se concluye:

- a) Las trayectorias de estado son más suaves (ver figuras 5.4-4, 5.4-5), cuando se minimiza la norma de Frobenious de la matriz de retroalimentación de estado, que con otra matriz.
- b) La magnitud de la ley de control también es más pequeña (ver figuras 5.4-6, 5.4-7), cuando se minimiza la norma de K. Esto se refleja en la aplicación de menor energía a la entrada para lograr la dinámica requerida.
- c) Ambos sistemas tienen la misma rapidez de respuesta debido a que tienen los mismos polos.
- d) El resultado anterior es válido únicamente para sistemas lineales multivariables debido a el grado de libertad que se presenta en el problema de asignación de polos.
- e) Es importante mencionar que como no se garantiza la convexidad de la función objetivo en el planteamiento del problema de optimización, no es posible garantizar la existencia de un mínimo único. Sin embargo, podemos observar que en algunos casos es bastante significativa la disminución tanto en la trayectoria de estado como en la ley de control como se presento en las simulaciones anteriores.

## Referencias de Cálculo Numérico

*El programa que realiza el cálculo de la matriz de retroalimentación de estado de norma mínima es una aportación personal y fue elaborado en lenguaje MATLAB por su facilidad de programación, este programa se encuentra en el apéndice C.1.*

## *Técnicas de Control Moderno*

### *6.1 Control Optimo*

Las técnicas conocidas como control clásico tienen una gran aplicación práctica, pero estas técnicas están limitadas para los sistemas lineales e invariantes en el tiempo. En este tipo de análisis el diseñador combina algunos métodos analíticos (como la Transformada de Laplace, Criterio de Routh, etc.) con métodos gráficos (Nyquist, Nichols, etc.) y sobre todo una buena parte es basado en conocimientos empíricos obtenidos gracias a la experiencia de cada diseñador.

El control óptimo es un componente importante de la teoría de control moderna el cual obtiene una solución analítica a ciertas especificaciones de diseño; además, de ser la mejor forma de controlar un sistema de control automático en el sentido de minimizar un criterio lineal cuadrático, como se verá posteriormente.

El control óptimo lineal supone que tanto la planta como el controlador son lineales. En esta sección se presentará la solución al problema del regulador óptimo lineal.

## 6.1.1 El Regulador Optimo

Supóngase que inicialmente los estados de la planta son diferentes de cero. El propósito del regulador óptimo consiste en aplicar una ley de control para llevar a la planta de un estado diferente de cero al estado cero, pero obteniendo la mejor trayectoria de estado posible y con la mínima energía aplicada por el controlador. El planteamiento del problema es el siguiente:

Considérese el siguiente sistema de control:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (6.1.1-1)$$

$$x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}$$

Ahora definamos el siguiente criterio cuadrático:

$$V(x(t_0), u(\cdot), t_0) = \int_{t_0}^{t_f} (u^T R u + x^T Q x) dt \quad (6.1.1-2)$$

donde  $x$  y  $u$  son el estado y la entrada de (6.1.1-1) respectivamente,  $R$  y  $Q$  son matrices positivas definidas. El criterio anterior es una medida de la cantidad de energía en el intervalo  $[t_0, t_f]$  de  $u$  y  $x$ , para el caso de que  $R$  y  $Q$  sean matrices identidad.

El problema del regulador óptimo consiste en encontrar una ley de control  $u(t)$ ,  $t \in [t_0, t_f]$ , de tal forma que se minimice el criterio  $V(x(t_0), u(\cdot), t_0)$ .

Resolviendo el problema de optimización planteado anteriormente obtenemos la ley de control óptima que minimiza la trayectoria de estado y la energía aplicada en la entrada del sistema, ver capítulo 2 de [1]. La solución se formula como:

$$\begin{aligned} u(t) &= -R^{-1}B^T P(t)x(t) \\ u(t) &= -K(t)x(t) \end{aligned} \quad (6.1.1-3)$$

$$\dot{P}(t) = Q - P(t)BR^{-1}B^TP(t) + P(t)A + A^TP(t) + P(t)A \quad (6.1.1-4)$$

donde la ecuación anterior es conocida como la ecuación de RICATTI, resolviendo esta ecuación obtenemos la solución al problema del regulador óptimo.

Para el caso de sistemas de control digital las ecuaciones quedan:

$$x(k+1) = Ax(k) + Bu(k) \quad (6.1.1-5)$$

$$x(k) \in \mathbb{R}^n, u(k) \in \mathbb{R}^m, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}$$

$$L(k) = [R + B^TP(k+1)B]^{-1}B^TP(k+1)A \quad (6.1.1-7)$$

$$P(k) = Q + A^TP(k+1)A - L^T(k) [(B^TP(k+1)B + R)] L(k) \quad (6.1.1-8)$$

$$u(k) = -L(k)x(k) \quad (2.1.1-9)$$

Cuando  $t_f \rightarrow \infty$  en el criterio (6.1.1-2) el problema del regulador óptimo se simplifica bastante ya  $\dot{P}(t) \rightarrow 0$  y la ecuación (6.1.1-4) se reduce a una ecuación algebraica. Este problema es conocido como el regulador óptimo asintótico.

## Referencias de cálculo numérico

*El cálculo numérico se realiza con el programa CONOP.EXE que calcula la matriz de retroalimentación de estado aplicada en la ley de control óptima para el caso del regulador asintótico. El programa se apoya en la subrutina RICTNWT de ORACLS [2]. El formato de los datos de entrada es el siguiente:*

### Datos de Entrada

$n$   $n \in \mathbb{N}$  número de filas de la matriz  $A$  de (6.1.1-1)

$A$   $A \in \mathbb{R}^{n \times n}$ , matriz  $A$  de (6.1.1-1) ordenada por columnas

$m \quad m \in \mathbb{N}$  número de columnas de la matriz  $B$  de (6.1.1-1)

$B \quad B \in \mathbb{R}^{n \times m}$ , matriz  $B$  de (6.1.1-1) ordenada por columnas

Datos de Salida

$P \quad P \in \mathbb{R}^{n \times n}$ , solución de la ecuación de RICATTI

$K \quad K \in \mathbb{R}^{m \times n}$ , matriz de retroalimentación de estado óptima

*Ejemplo:* Calcular la ley de control óptimo del siguiente sistema de control:

$$\dot{x} = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 1 & 0 \\ 2 & 1 & 100 \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} u$$

$$Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad R = 1$$

Efectuando el cálculo mediante el programa mencionado en las referencias numéricas obtenemos la siguiente solución:

$$P = \begin{pmatrix} 16.43 & -5.18 & -5.78 \\ -5.18 & 5.05 & 11.75 \\ -5.78 & 1.76 & 222.89 \end{pmatrix}$$

$$k = (0.5996 \quad -6.7067 \quad -211.1406)$$

Aplicando la ley de control al sistema original obtenemos las siguientes simulaciones

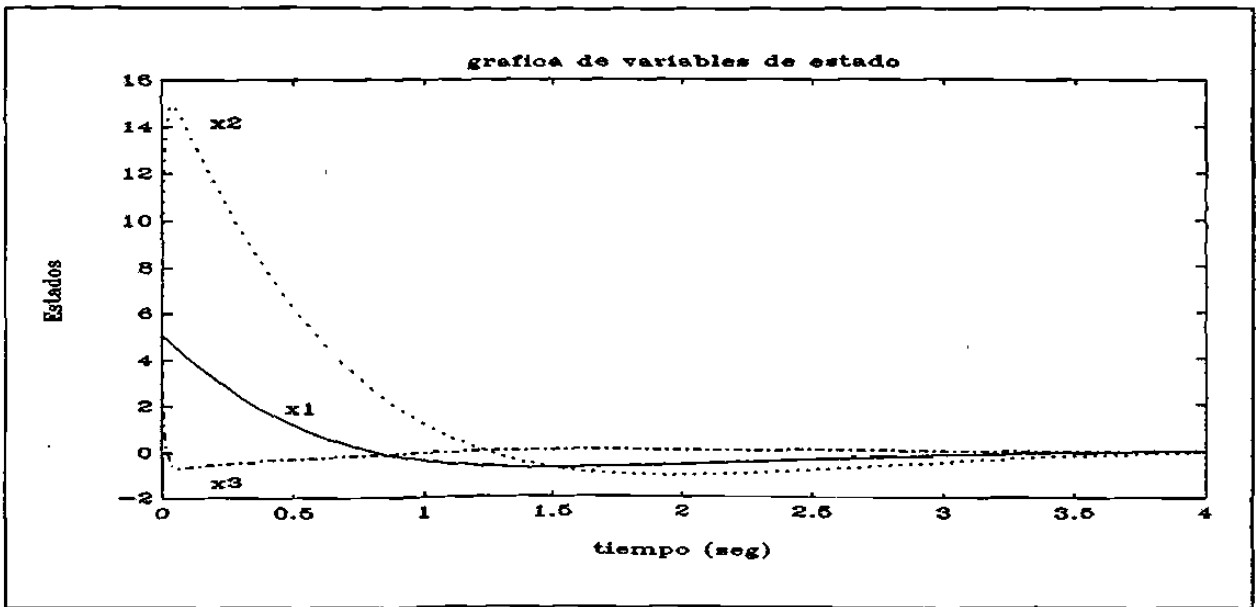


Figura 6.1.1 Gráfica de Trayectoria de Estado Optima

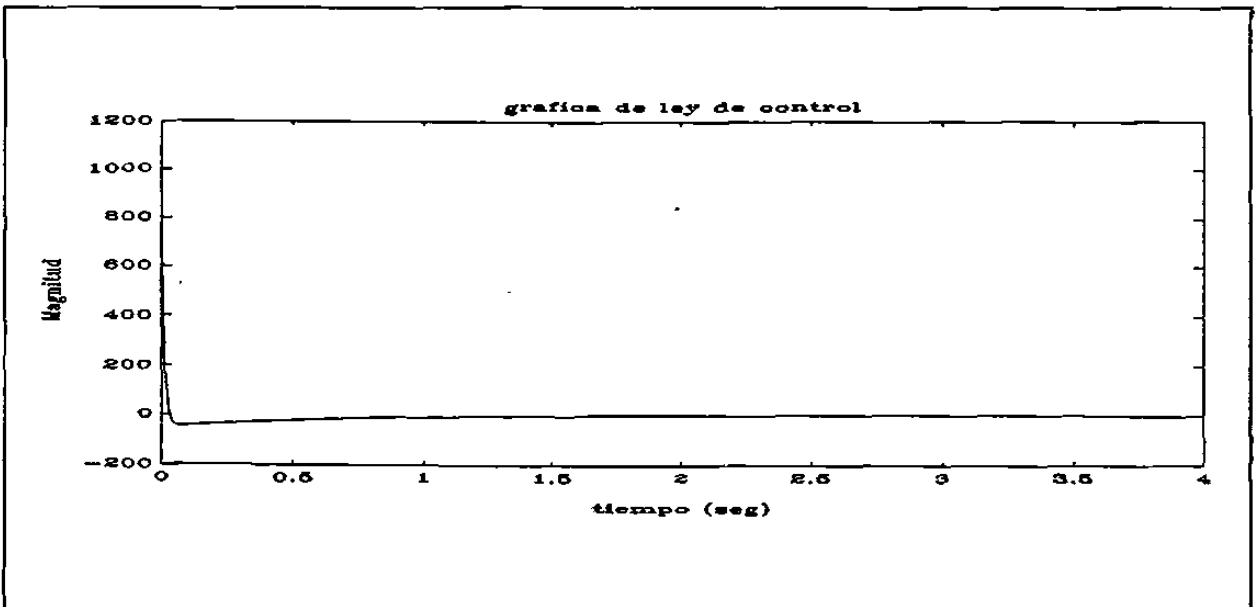


Figura 6.1.2 Gráfica de Ley de Control Optima



### 6.1.2 Estimadores de Estado y Filtro de Kalman

En el problema del regulador óptimo suponemos que el estado del sistema está disponible para su medición, ya que la ley de control resultante está en función del estado del sistema. En la mayoría de los casos, los estados del sistema no están disponibles, por lo que es necesario estimar dichos estados mediante un estimador u observador de estado.

*Estimador de estado:* Sea el siguiente sistema de control

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (6.1.2-1)$$

$$y = Cx(t) \quad (6.1.2-2)$$

$$A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{p \times n}$$

derivando la ecuación (6.1.2-2) y sustituyendo la ecuación (6.1.2-1) obtenemos:

$$\dot{y} - CBu(t) = CAx(t) \quad (6.1.2-3)$$

nuevamente derivando la ecuación (6.1.2-3) obtenemos:

$$\ddot{y} - CB\dot{u}(t) - CABu(t) = CA^2 x(t) \quad (6.1.2-4)$$

continuando de la misma forma hasta n-1 veces, podemos obtener:

$$z = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} x(t) = Vx(t) \quad (6.1.2-5)$$

Como podemos observar claramente la ecuación (6.1.2-5) es un sistema de ecuaciones lineales que tiene solución única si y sólo si,  $V$  tiene rango completo (ver capítulo 3). De lo anterior tenemos que si la matriz  $V$  tiene rango  $n$  entonces podemos expresar el estado del sistema como combinaciones lineales de la entrada y la salida así como sus derivadas. Esta es una forma de estimar el estado de un sistema, pero prácticamente es que es poco usual ya que es necesario tener las derivadas de las salidas y entradas, esto representa

amplificación de ruido y posible error en la estimación del estado.

*Método asintótico de estimación de estado:* Considerese las ecuaciones (6.1.2-1) y (6.1.2-2) y supóngase que el sistema dinámico estimado siguiente.

$$\dot{x}_e(t) = Ax_e(t) + Bu(t) \quad (6.1.2-6)$$

$$y_e(t) = Cx_e(t) \quad (6.1.2-7)$$

Entonces, para que  $x_e(t)$  sea igual a  $x(t)$  es necesario que el error entre ellos sea igual a cero, por lo tanto de la ecuación (6.1.2-1) y (6.1.2-6) obtenemos:

$$\frac{d}{dt}(x(t) - x_e(t)) = A(x(t) - x_e(t)) \quad (6.1.2-8)$$

Ahora considérese el sistema de ecuaciones dinámicas (6.1.2-1) y (6.1.2-2) y supóngase que las variables de estado no son accesibles, aunque las matrices A,B,C son conocidas. El problema de estimación de estado consiste en reconstruir el estado de (6.1.2-1), haciendo uso solamente de sus entradas y las salidas.

El estimador de estado asintótico consiste en comparar la salida  $y(t)$  con  $y_e(t)$  y la diferencia entre las dos servirá como un término correctivo para hacer cada vez más pequeño el error entre  $x(t)$  y  $x_e(t)$ .

Haciendo una manipulación adecuada de las ecuaciones dinámicas (6.1.2-1) y (6.1.2-6), obtenemos el siguiente sistema de ecuaciones dinámicas, ver capítulo 7 de [5]:

$$\dot{e}(t) = (A - LC)e(t) \quad (6.1.2-9)$$

Donde  $e(t)$  denota el error entre el estado estimado y el estado real del sistema. Si los valores propios de  $(A - LC)$  pueden ser arbitrariamente asignados (sistema de ecuaciones (6.1.2-1) observable) podemos hacer que el error tienda a cero en un tiempo determinado.

De esta forma podemos estimar los estados de un sistema ya que el error entre el estado real y el estimado tenderá a cero; esto indica que decir que el estado estimado será igual al estado real.

*Filtro de Kalman-Bucy:* Generalmente no se tiene disponible el estado de un sistema, pero bajo algunas restricciones se puede estimar. El problema en la práctica es la existencia de ruido tanto en la medición de la salida como en la planta, las cuales usamos para estimar el estado, afectando por lo tanto los resultados. En un esfuerzo por tratar de reconstruir el estado lo más fielmente posible, es decir eliminar el ruido existente, se creó el *Filtro de Kalman-Bucy* [14]. Este consiste en minimizar el ruido de medición en la salida y en el proceso, por tal razón puede ser considerado como un observador óptimo en el sentido de minimizar el ruido que existe en la medición y el proceso. El problema del filtrado óptimo se plantea a continuación.

Considérese el sistema de control siguiente:

$$\dot{x}(t) = Ax(t) + Bu(t) + v(t) \quad (6.1.2-10)$$

$$y(t) = Cx(t) + w(t) \quad (6.1.2-11)$$

Donde  $v(t)$  y  $w(t)$  representan los términos de ruido en el proceso y en la salida respectivamente. La restricción es que el tipo de ruido debe ser blanco, gaussiano y de media igual a cero, ver capítulo 6 de [1]. El problema del estimador óptimo consiste en reconstruir un estado estimado  $x_e(t)$  definido sobre  $(t_0, t]$  de tal forma que el siguiente criterio sea minimizado:

$$J = \text{Cov} \{ (x(t) - x_e(t)) W (x(t) - x_e(t))^T \} \quad (6.1.2-12)$$

donde:  $W$  es una matriz positiva definida  
 $\text{Cov}$  es la covarianza

Si el parámetro  $\lambda$  es detectable, la solución al problema del observador óptimo existe y está dado por:

$$A x_e(t) + L(t)[y(t) - C x_e(t)] \quad (6.1.2-13)$$

donde:

$$L(t) = P(t)C^TR^{-1} \quad (6.1.2-14)$$

P debe satisfacer:

$$\dot{P}(t) = AP(t) + P(t)A^T + BQB^T - P(t)C^TR^{-1}CP(t) \quad (6.1.2-15)$$

Se puede demostrar mediante el teorema de dualidad que el problema de filtrado óptimo es equivalente a calcular la ley de control para el regulador óptimo sustituyendo únicamente B por  $C^T$  en la ecuación de RICATTI para el problema del regulador óptimo, ver capítulo 7 de [1].

## Referencias de cálculo numérico

*El cálculo numérico del filtro de Kalman-Bucy es realizado mediante el programa que resuelve el problema del regulador óptimo aplicando el teorema de dualidad, ver capítulo 7 de [1].*

### 6.2 Estabilidad Robusta

Una de las partes más importantes en un sistema de control. La mayoría de las veces es complicado obtener un modelo fiel de un determinado proceso, debido a errores en la medición, además de los errores paramétricos; por lo tanto se desea garantizar estabilidad en dicho sistema a pesar de los errores mencionados anteriormente. Generalmente dichos errores se reflejan como incertidumbres en los coeficientes del polinomio característico en el dominio de la frecuencia ó como incertidumbres en los elementos de una matriz para el caso del análisis en el dominio del tiempo. Existen varios resultados para el caso de matrices, otros para el caso de análisis en el dominio de la frecuencia [3], [28]. En esta sección presentaremos un método gráfico para garantizar estabilidad de un polinomio con incertidumbre en sus coeficientes. Este análisis se realiza en el dominio de la frecuencia.

Antes de continuar es necesario presentar el siguiente criterio, conocido como el criterio de Mikhailov.

Considérese el siguiente polinomio:

$$P(s) = s^n + a_{n-1}s^{n-1} + \dots + a_0 \quad (6.2.0-1)$$

sustituyendo  $s=j\omega$  obtenemos:

$$P(j\omega) = (j\omega)^n + a_{n-1}(j\omega)^{n-1} + \dots + a_0 \quad (6.2.0-2)$$

Haciendo la siguiente definición:

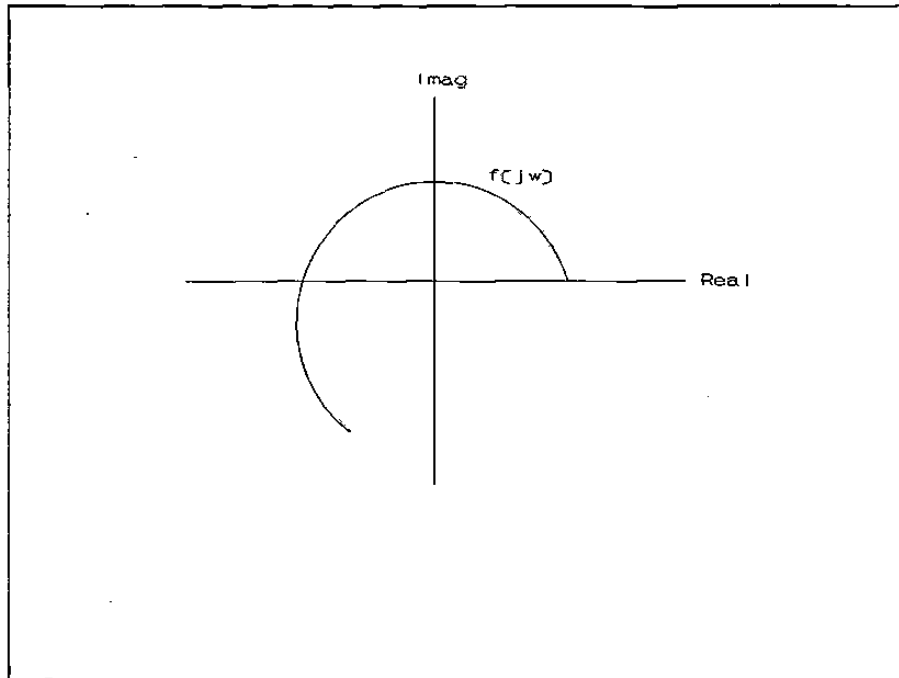
$$U(\omega) = a_0 - a_2\omega^2 + a_4\omega^4 - \dots \quad (6.2.0-3)$$

$$V(\omega) = a_1\omega - a_3\omega^3 + a_5\omega^5 - \dots \quad (3.2.0-4)$$

Entonces:

$$P(j\omega) = U(\omega) + jV(\omega) \quad (6.2.0-5)$$

variando la frecuencia en el intervalo  $[0, \infty)$  podemos obtener una curva como la que se muestra en la figura 6.2-1.



*Figura 6.2-1 Curva de  $P(j\omega)$*

**Criterio de Mikhailov:** El polinomio  $P(s)$  es estable *si y sólo si*:

- a)  $P(j\omega)$  para cambios de  $\omega$  entre  $0$  y  $+\infty$ , recorre  $n$  cuadrantes en sentido inverso a las manecillas del reloj.
- b) La curva  $P(s)$  al recorrer los  $n$  cuadrantes no pasa por el origen.

El criterio anterior es un método gráfico para determinar la estabilidad de un sistema de control, ya que el polinomio  $P(s)$  podría ser el polinomio característico de un determinado proceso.

### **6.2.1 Margen de Robustez**

Como mencionamos anteriormente es necesario garantizar estabilidad cuando el

sistema tiene incertidumbre, pero uno de los problemas más interesantes dentro de estabilidad robusta es poder determinar la máxima variación que puede tener dicha incertidumbre para garantizar estabilidad en el sistema; a este problema se le conoce como *margen de robustez*. En esta sección presentaremos un método mediante el cual podemos determinar el margen de robustez para un polinomio con incertidumbres independientes. Antes de presentar el método mencionaremos un teorema en el cual se basa el resultado anterior.

Sea la familia de polinomios de grado  $n$  con incertidumbres independientes en los coeficientes:

$$P(s, q) = s^n + \sum_{i=0}^{n-1} q_i s^i \quad (6.2.1-1)$$

donde la incertidumbre del coeficiente  $q_i$  se representa como:

$$q_i^- \leq q_i \leq q_i^+$$

y sean los siguientes polinomios  $P_i(s)$ ,  $i=1,2,3,4$ .

$$\begin{aligned} P_1(s) &= q_0^- + q_1^- s + q_2^+ s^2 + q_3^+ s^3 + q_4^- s^4 + \dots \\ P_2(s) &= q_0^+ + q_1^+ s + q_2^- s^2 + q_3^- s^3 + q_4^+ s^4 + \dots \\ P_3(s) &= q_0^+ + q_1^- s + q_2^- s^2 + q_3^+ s^3 + q_4^+ s^4 + \dots \\ P_4(s) &= q_0^- + q_1^+ s + q_2^+ s^2 + q_3^- s^3 + q_4^- s^4 + \dots \end{aligned}$$

**Teorema (Kharitonov):** La familia de polinomios  $P(s, q)$  es estable si y sólo si,  $P_1(s)$ ,  $P_2(s)$ ,  $P_3(s)$ ,  $P_4(s)$  son estables, ver [3].

Ahora supóngase que:

$$\begin{aligned} q_i^- &= a_i - r \\ q_i^+ &= a_i + r \end{aligned}$$

El polinomio nominal de la familia (6.2.1-1) está dado por:

$$P_0(s) = a_0 s^n + a_1 s^{n-1} + \dots + a_n \quad (6.2.1-2)$$

El problema de margen de robustez consiste en determinar el máximo valor de  $r$  ( $r_{\max}$ ) para garantizar estabilidad de todos los polinomios de la familia  $P(s,q)$  (estabilidad robusta).

Sean los polinomios:

$$P_1(s) = s^{n-1} + s^{n-2} - s^{n-3} - s^{n-4} + \dots \quad (6.2.1-3)$$

$$P_2(s) = s^{n-1} - s^{n-2} - s^{n-3} + s^{n-4} + \dots \quad (6.2.1-4)$$

$$P_3(s) = -s^{n-1} - s^{n-2} + s^{n-3} + s^{n-4} + \dots \quad (6.2.1-5)$$

$$P_4(s) = -s^{n-1} + s^{n-2} + s^{n-3} - s^{n-4} + \dots \quad (6.2.1-6)$$

Ahora se formarán las siguientes matrices  $H_i$  con los coeficientes de los polinomios  $P_i(s)$ , donde  $i=0,1,2,3,4$ , donde  $P_0(s)$  es el de la ecuación (6.2.1-2):

$$H_0 = \begin{pmatrix} a_1 & a_3 & a_5 & \dots & \dots \\ a_0 & a_2 & a_4 & \dots & \dots \\ 0 & a_1 & a_3 & a_5 & \dots \\ 0 & a_0 & a_2 & a_4 & \dots \\ & & & & \ddots \\ & & & & & a_n \end{pmatrix}$$

Igualmente  $H_1, H_2, H_3, H_4$  se forman con los coeficientes de  $P_1(s), P_2(s), P_3(s), P_4(s)$ . A las matrices con estructura como  $H_0$  se les llama matrices de Hurwitz asociadas a un polinomio. Un hecho importante es que  $P_i$  es estable si y sólo si,  $\lambda(H_i) \in \mathbb{C}$ ,  $i=0,1,2,3,4$ .

Definiendo  $\lambda_{\max}^+(M)$  como el máximo valor propio real de la matriz  $M$  y  $\lambda_{\min}^+(M) =$



0+ si M no tiene valores propios reales positivos.

Entonces  $r_{\max}$  puede ser determinado mediante la siguiente formula, ver [3].

$$r_{\max} = \min_{i=1,2,3,4} \frac{1}{\lambda_{\max}^* (-H_0^{-1} H_i)} \quad (6.2.1-7)$$

Nótese que al determinar  $r_{\max}$  conocemos el máximo rango de variación en los coeficientes del polinomio característico de tal forma que preserve estabilidad. Ahora presentaremos un ejemplo donde se determinará el valor  $r_{\max}$ .

*Ejemplo:* Sea el siguiente polinomio característico:

$$P_0(s) = s^4 + 5s^3 + 8s^2 + 8s + 3$$

Ahora supóngase que el polinomio tiene la siguiente incertidumbre en sus coeficientes.

$$P(s, q) = s^4 + (5 + q_3)s^3 + (8 + q_2)s^2 + (8 + q_1)s + (3 + q_0)$$

con:

$$-r < q_i < r \quad \text{para } i=0,1,2,3$$

Se desea encontrar el máximo valor de r que garantice estabilidad robusta.

$$H_0 = \begin{pmatrix} 5 & 8 & 0 & 0 \\ 1 & 8 & 3 & 0 \\ 0 & 5 & 8 & 0 \\ 0 & 1 & 8 & 3 \end{pmatrix} \quad H_1 = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$H_2 = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix} \quad H_3 = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$H_4 = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

Aplicando la formula (6.2.1-7) obtenemos:

$$r_{\max} = 1.81$$

## Referencias de Cálculo Numérico

*El cálculo en computadora es realizado mediante un programa desarrollado en lenguaje MATLAB que calcula el valor de  $r_{\max}$  a partir de un polinomio nominal dado. Este programa se presenta en el apéndice C.2.*

### 6.2.2 Estabilidad $l_p$ -Robusta

En esta sección presentaremos una extensión del criterio de Mikhailov para el caso en que los coeficientes del polinomio característico varían dentro de una región determinada.

Considérese el polinomio característico de un sistema lineal invariante en el tiempo:

$$G(s) = a_0 + a_1 s + \dots + a_n s^n \quad (6.3.2-1)$$

donde los coeficientes están sujetos a las siguientes variaciones:

$$\left( \sum_{k=0}^n \left| \frac{a_k - a_k^0}{\alpha_k} \right|^p \right)^{\frac{1}{p}} \leq \gamma^2 \quad (6.2.2-2)$$

$$\gamma \geq 0, \quad a_0^0 > 0, \quad \alpha_k > 0, \quad 1 \leq p \leq \infty$$

donde  $a_k^0$  - El coeficiente nominal o sin perturbar .

Considérese además que:

$$\frac{1}{p} + \frac{1}{q} = 1 \quad (6.2.2-3)$$

$$S_p(\omega) = \left( \alpha_0^q + (\alpha_2 \omega^2)^q + (\alpha_4 \omega^4)^q + \dots \right)^{\frac{1}{q}} \quad (6.2.2-4)$$

$$T_p = \left( \alpha_1^q + (\alpha_3 \omega^2)^q + (\alpha_5 \omega^4)^q + \dots \right)^{\frac{1}{q}} \quad (6.2.2-5)$$

Definiendo:

$$Z(\omega) = X(\omega) + j\omega Y(\omega) \quad (6.2.2-6)$$

donde:

$$X(\omega) = \frac{U(\omega)}{S_p(\omega)}, \quad Y(\omega) = \frac{V(\omega)}{T_p(\omega)} \quad (6.2.2-7)$$

Ahora denotaremos la norma  $\ell_p$  como:

$$\Gamma_\gamma^p = \{z = x + y : (|x|^p + |y|^p)^{\frac{1}{p}} \leq \gamma\} \quad (6.2.2-8)$$

Con estas consideraciones básicas enunciaremos el teorema presentado por Tsypkin y Polyak en [28], que establece condiciones necesarias y suficientes para la estabilidad de un polinomio cuando los coeficientes varían dentro de la región definida por (6.2.2-2).

**Teorema (Tsypkin and Polyak):** El sistema (6.2.2-1) sujeto a las variaciones (6.2.2-2) es estable (también conocido como  $\ell_p$ -robustamente estable) si y sólo si:

- a) La curva  $Z(\omega)$  recorre  $n$  cuadrantes
- b) La curva  $Z(\omega)$  al recorrer  $n$  cuadrantes no interseca  $\Gamma_\gamma^p$
- c) Los puntos frontera  $Z(0)$ ,  $Z(\infty)$  tienen coordenadas con valores absolutos mayores que  $\gamma$ , es decir:

$$|X(0)| > \gamma, \quad |Y(0)| > \gamma, \quad |X(\infty)| > \gamma, \quad |Y(\infty)| > \gamma \quad (6.2.2-9)$$

El método anterior es un método gráfico mediante el cual podemos garantizar estabilidad de un sistema, cuando los coeficientes del polinomio característico están sujetos a cierto tipo de variaciones. El resultado anterior también puede plantearse como encontrar el máximo valor de  $\gamma$  en (6.2.2-2) que garantice estabilidad del sistema (6.2.2-1). En seguida presentaremos un ejemplo donde se calcula el valor admisible máximo de  $\gamma$  que garantice estabilidad en el sistema.

*Ejemplo:*

Sea el siguiente polinomio característico de un proceso determinado:

$$G_0(s) = a_0^0 + a_1^0 s + \dots + a_6^0 s^6$$

donde:

$\alpha_0 = 43.35$	$a_0^0 = 433.5$
$\alpha_1 = 33.36$	$a_1^0 = 667.25$
$\alpha_2 = 25.137$	$a_2^0 = 502.72$
$\alpha_3 = 15.075$	$a_3^0 = 251.25$
$\alpha_4 = 5.6175$	$a_4^0 = 80.25$
$\alpha_5 = 1.4$	$a_5^0 = 14.0$
$\alpha_6 = 0.1$	$a_6^0 = 1.0$

Los coeficientes de  $G_0(s)$  varían de tal forma que el polinomio resultante es:

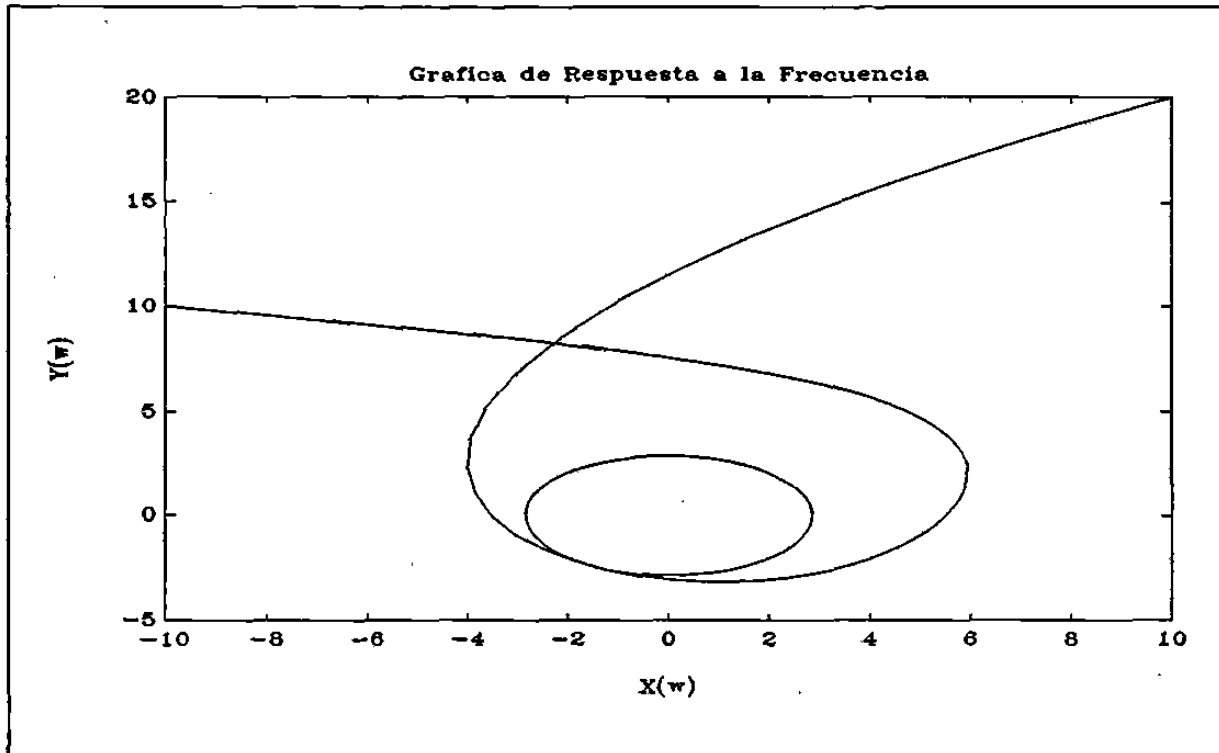
$$G(s) = a_0 + a_1 s + \dots + a_n s^n$$

donde:

$$\left( \sum_{k=0}^n \left| \frac{a_k - a_k^0}{\alpha_k} \right|^2 \right) \leq \gamma^2$$

Encontrar el máximo valor de  $\gamma$  de tal forma que el sistema anterior sea estable.

Utilizando el método anterior podemos obtener la gráfica de  $Z(\omega)$  variando  $\omega$  entre 0 y 100:



Aplicando el teorema anterior, podemos encontrar de la gráfica que el círculo más grande inscrito en la curva de respuesta a la frecuencia tiene un valor de  $\gamma < 2.8434$ . Entonces la solución el problema anterior es:

$$\sum_{k=0}^n \left( \frac{a_k - a_k^0}{\alpha_k} \right)^2 < (2.8434)^2$$

## Referencias de cálculo numérico

*El cálculo numérico es realizado con un programa codificado en lenguaje MATLAB que calcula  $\gamma_{max}$  a partir de  $a_i^0$  y  $\alpha_i$  dados, ver apéndice C.3.*

## *Conclusiones*

La motivación de realizar el trabajo anterior fué principalmente el formar una programoteca que sirviera para resolver algunos de los problemas presentados en sistemas de control lineales, con el objeto de que posteriormente pudieran desarrollarse nuevos programas que resuelvan problemas de control aún no desarrollados. Aún más, la realización del presente trabajo fué de gran apoyo tanto en la formación académica personal para la continuación de los estudios en el area de control automático.

El objetivo principal del trabajo presentado fué crear un conjunto de programas que sirven para hacer análisis y síntesis de sistemas de control lineales, apoyandonos en las más confiables utilerías de software disponibles, como LINPACK, EISPACK tomando como base los programas de ORACLS; además, de algunos programas desarrollados en MATLAB. Es muy importante hacer notar que existen una gran cantidad de algoritmos numéricos que resuelven hoy en día la mayoría de los problemas presentados en la práctica, pero es más conveniente usar utilerías disponibles a programar dichos algoritmos, puesto que de esta forma podemos ahorrarnos tiempo y lo que es más importante la mayoría de las veces los resultados obtenidos son mucho mejores.

Como se pudo observar claramente la mayoría de los programas se encuentran disponibles como programas ejecutables los cuales tienen la ventaja de que pueden ser fácilmente accesados desde otros lenguajes como BASIC, PASCAL, C. A diferencia de los

paquetes existentes como MATLAB, CC entre otros se tiene la ventaja de que son compatibles con cualquier otro lenguaje como se menciono anteriormente, esto puede ser de gran utilidad en el caso de que deseen desarrollar sistemas que requieran efectuar algun cálculo disponible en este trabajo. Por otro lado algunos programas fueron desarrollados en lenguaje MATLAB por la gran facilidad que este representa para su programación.



# Apéndice A

## Transformaciones Importantes

### A.1 Transformaciones Elementales

Considerese  $A \in \mathbb{C}^{n \times n}$ . Las siguientes operaciones son llamadas operaciones o transformaciones elementales con matrices.

- A1.1 Multiplicar una fila o columna por un número real o complejo diferente de cero.
- A1.2 Intercambiar dos filas o dos columnas.
- A1.3 Agregar al producto de un número por una fila o columna a otra fila ó columna.

Estas transformaciones se pueden conseguir usando las siguientes matrices elementales. Considerese el caso cuando  $n=5$  y  $c, d \in \mathbb{C}$ .

$$E_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad E_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$E_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & d & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

La premultiplicación de  $E_1$  por cualquier matriz opera sobre las filas de dicha matriz y la posmultiplicación opera sobre las columnas. La operación A1.1 se realiza con la premultiplicación y posmultiplicación de  $E_1$  por la matriz de interés. La operación A1.2 con la premultiplicación y posmultiplicación de  $E_2$  por la matriz de interés. La operación A1.3 con la premultiplicación y posmultiplicación de  $E_3$  por la matriz de interés.

## A.2 Transformación de Similitud

Sea  $A \in \mathbb{R}^{n \times n}$ ,  $Q \in \mathbb{R}^{n \times n}$ ;  $A$  es similar a  $\tilde{A}$  si existe una matriz  $Q$  no singular tal que:

$$A = Q\tilde{A}Q^{-1}$$

Esta transformación es llamada transformación de similitud y tiene entre otras propiedades la de que  $\lambda(A) = \lambda(\tilde{A})$ .

*prueba:*

$$\lambda(A) = |\lambda I - A| = |\lambda I - Q\tilde{A}Q^{-1}|$$

$$= |\lambda QQ^{-1} - Q\tilde{A}Q^{-1}|$$

$$= |(\lambda Q - Q\tilde{A})Q^{-1}|$$

$$= |Q(\lambda I - \tilde{A})Q^{-1}|$$

$$\lambda(A) = |\lambda I - \tilde{A}| = \lambda(\tilde{A})$$

### A.3 Transformación de Householder

Una transformación de Householder es una matriz cuadrada de la forma:

$$H = I - 2 \frac{xx^T}{x^T x}$$

$$H \in \mathbb{R}^{n \times n}, \quad x \in \mathbb{R}^n$$

**Propiedades:**

A3.1 H es simétrica

$$H^T = \left( I - 2 \frac{xx^T}{x^T x} \right)^T$$

$$H^T = I - 2 \frac{xx^T}{x^T x}$$

$$H^T = H$$

A3.2 H es ortogonal

$$\begin{aligned}
H^T H &= \left( I - 2 \frac{xx^T}{x^T x} \right)^T \left( I - 2 \frac{xx^T}{x^T x} \right) \\
H^T H &= \left( I - 2 \frac{xx^T}{x^T x} \right) \left( I - 2 \frac{xx^T}{x^T x} \right) \\
&= I - 2 \frac{xx^T}{x^T x} - 2 \frac{xx^T}{x^T x} + 4 \frac{xx^T xx^T}{(x^T x)^2} \\
H^T H &= I
\end{aligned}$$

#### A.4 Factorización QR

Sea  $A \in \mathbb{R}^{p \times q}$ , entonces se puede calcular una sucesión  $H_1, \dots, H_q$  de cuando más  $q$  matrices de Householder de tal manera que:

$$H_q H_{q-1} \dots H_1 A = R$$

donde  $R \in \mathbb{R}^{p \times q}$  es triangular superior y tiene elementos no negativos en la diagonal principal; igualmente:

$$A = QR$$

en donde  $Q = H_1 H_2 \dots H_q$ ,  $Q \in \mathbb{R}^{p \times p}$ , ortogonal.

Este tipo de factorización es muy importante debido a que las matrices triangulares tienen propiedades especiales para el cálculo de valores propios, ver [26].

#### A.5 Factorización LU

Sea  $A \in \mathbb{R}^{n \times n}$  entonces,  $A$  se puede factorizar mediante la eliminación de Gauss como:

$$A = LU$$

Si y sólo si, A es no singular. Donde L es una matriz triangular inferior y U es una matriz triangular superior, ver sección 3.7 de [21].

Esta factorización tiene gran aplicación en la solución de sistemas de ecuaciones lineales cuando tienen solución única.

### *A.6 Descomposición en Valores Singulares*

En 1980 Klema (ver [6]) enfocó su atención en la descomposición de valores singulares (SVD), Esta descomposición constituye el más importante desarrollo del algebra lineal moderna ya que es el más elegante algoritmo en algebra numérica y proporciona información cuantitativa a cerca de la estructura de un sistema de ecuaciones lineales; es además el único método confiable para determinar el rango de una matriz. La descomposición en valores singulares es utilizando generalmente para matrices cuadradas de rango completo, pero su mayor aplicación se obtiene cuando se quiere solucionar sistemas de ecuaciones lineales con más incógnitas que ecuaciones. En esta sección se presentará la definición de SVD así como algunas propiedades importantes.

Sea  $A \in \mathbb{C}^{p \times q}$  entonces:

- a) Existe una matriz unitaria  $U \in \mathbb{C}^{p \times p}$  (ortogonal si  $U \in \mathbb{R}^{p \times p}$ ), una matriz unitaria  $V \in \mathbb{C}^{q \times q}$  (ortogonal si  $V \in \mathbb{R}^{q \times q}$ ) y una matriz  $\Sigma \in \mathbb{R}^{p \times q}$  "diagonal" con elementos  $\sigma_i \geq 0$ ; siendo  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_s$ , donde  $s = \text{rango} = \min\{p, q\}$ , tal que la descomposición en valores singulares.

$$A = U \Sigma V^H$$

es válida

- b) Los números reales  $\sigma_i^2$  conforman los valores propios de  $A^H A$  y los vectores propios

asociados son las columnas  $v_i$  de  $V$ ; igualmente, los  $\sigma_i^2$  conforman los valores propios de  $AA^H$ , y los vectores propios asociados son las columnas  $u_i$  de  $U$ . Los  $\sigma_i$  se llaman valores singulares de  $A$ , los  $u_i$  se llaman los vectores singulares izquierdos de  $A$ ; y los  $v_i$  son los vectores singulares derechos de  $A$ , que se relacionan por  $Av_i = \sigma_i u_i$  para  $1 \leq i \leq s$ , ver sección 8.4 de [21].

### Propiedades:

A6.1 Sea  $A \in \mathbb{C}^{m \times n}$

$$A^H A = (U \Sigma V^H)^H (U \Sigma V^H)$$

$$A^H A = V (\Sigma^H \Sigma) V^H$$

$$D = \Sigma^H \Sigma$$

$$D = V^H (A^H A) V$$

Esto significa que los vectores propios de  $A^H A$  forman a  $V$ , con los valores propios (reales no negativos) asociados en la diagonal de  $D = \Sigma^H \Sigma$ .

A6.2 De modo parecido

$$A A^H = U (\Sigma \Sigma^H) U^H$$

$$D^T = U^H (A A^H) U$$

Lo cual significa que los vectores propios de  $A A^H$  forman a  $U$  asociados en la diagonal de  $D^T = \Sigma \Sigma^H$

A6.3 El rango de  $A$  es igual al número de valores singulares diferentes de cero

La descomposición en valores singulares despierta el interés en el caso más general de la inversa de una matriz (pseudoinversa) debido a la facilidad y precisión en su cálculo. La pseudoinversa en el sentido de Moore-Penrose de una matriz se define a continuación.

Sea  $A \in \mathbb{C}^{m \times n}$  y sea la descomposición en valores singulares de  $A$ .

$$A = U\Sigma V^H$$

Entonces la pseudoinversa de A se define como:

$$A^\dagger = V\Sigma^\dagger U^H$$

donde:

$$\Sigma^\dagger = [\Sigma^\dagger \ 0], \text{ si } p < q$$

$$\Sigma^\dagger = [\Sigma^\dagger \ 0], \text{ si } p > q$$

$$\Sigma^\dagger = [\Sigma^\dagger], \text{ si } p = q$$

$$\Sigma^\dagger = \text{diag}\{1/\sigma_1, 1/\sigma_2, \dots, 1/\sigma_r, 0, \dots, 0\}$$

donde r es el rango de la matriz A.

La pseudoinversa en el sentido de Moore-Penrose de una matriz  $A \in \mathbb{C}^{p \times q}$  debe satisfacer las siguientes identidades.

$$AA^\dagger A = A$$

$$A^\dagger AA^\dagger = A^\dagger$$

$$AA^\dagger = (AA^\dagger)^T$$

$$A^\dagger A = (A^\dagger A)^T$$

# Apéndice B

## Programas de Soporte

### B.1 Matriz de Hessenberg

Toda matriz cuadrada puede ser transformada en una forma canónica de Jordan por transformaciones de similitud.

Sea  $A$  una matriz cuadrada de dimension  $n \times n$  y considérese lo siguiente:

$$P_1 A \triangleq \left( \begin{array}{c|cccc} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & & & & & \\ 0 & & & P_{11} & & \\ \vdots & & & & & \\ 0 & & & & & \end{array} \right) = \begin{pmatrix} x & x & x & \dots & x \\ x & x & x & \dots & x \\ 0 & x & x & \dots & x \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & x & x & \dots & x \end{pmatrix}$$

La matriz  $P_{11}$  se elige de tal forma que la primera columna de  $A$ , excepto los primeros dos valores sean todos cero. La inversa de  $P_1$  tiene la misma forma que  $P_1$ , por lo tanto la posmultiplicación de  $P_1 A$  por  $P_1^{-1}$  no opera sobre la primera columna de  $P_1 A$ ; por lo tanto el patron de ceros en la primer columna de  $P_1 A$  es preservado en  $P_1 A P_1^{-1}$ . Después



encontraremos  $P_2$  tal que:

$$\left( \begin{array}{cc|ccc} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & & & & \\ 0 & 0 & & & & \\ \vdots & \vdots & & & & \\ 0 & 0 & & & & \end{array} \right) P_{22} P_1 A P_1^{-1} = \left( \begin{array}{cccccc} x & x & x & x & \dots & x \\ x & x & x & x & \dots & x \\ 0 & x & x & x & \dots & x \\ 0 & 0 & x & x & \dots & x \\ \vdots & \vdots & \vdots & \vdots & \dots & x \\ 0 & 0 & x & x & \dots & x \end{array} \right)$$

De nuevo la inversa de  $P_2$  tiene la misma forma que  $P_2$  y la posmultiplicación de  $P_2 P_1 A P_1^{-1}$  por  $P_2^{-1}$  no opera sobre las primeras dos columnas. Por lo tanto el patron de ceros en las primeras dos columnas se preserva en  $P_2 P_1 A P_1^{-1} P_2^{-1}$ . Procediendo similarmente podemos obtener la matriz transformada siguiente:

$$\left( \begin{array}{cccccc} x & x & x & \dots & x & x & x \\ x & x & x & \dots & x & x & x \\ 0 & x & x & \dots & x & x & x \\ 0 & 0 & x & \dots & x & x & x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & x & x & x \\ 0 & 0 & 0 & \dots & 0 & x & x \end{array} \right)$$

Esta matriz es llamada la forma superior de Hessenberg.

### Referencias de cálculo numérico

*El cálculo numérico de la matriz de Hessenberg es realizado mediante la subrutina ORTHES de EISPACK [10].*

## B.2 Solución de $AX + XB = C$

Sea la ecuación:

$$AX + XB = C \quad (3.1)$$

Suponiendo que tiene solución; se puede obtener numéricamente como sigue:

La matriz A es reducida a una forma Schur inferior  $A^*$  por una transformación de similitud (Ver sección 7.4 de [11]); entonces A es reducida a la siguiente forma:

$$A^* = U^T A U = \begin{pmatrix} A_{11}^* & 0 & \dots & 0 \\ A_{21}^* & A_{22}^* & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_{p1}^* & A_{p2}^* & \dots & A_{pp}^* \end{pmatrix} \quad (3.2)$$

donde toda matriz  $A_{ii}^*$  es de orden al menos dos.

Similarmente B es reducida a la forma Schur superior  $B^*$  por una transformación de similitud

$$B^* = V^T B V = \begin{pmatrix} B_{11}^* & B_{12}^* & \dots & B_{1q}^* \\ 0 & B_{22}^* & \dots & B_{2q}^* \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \dots & B_{qq}^* \end{pmatrix} \quad (3.3)$$

nuevamente  $B_{ii}^*$  es de orden al menos dos. Entonces definiendo

$$C^* = U^T C V = \begin{pmatrix} C_{11}^* & \dots & C_{1q}^* \\ \vdots & \vdots & \vdots \\ C_{p1}^* & \dots & C_{pq}^* \end{pmatrix} \quad (3.4)$$

$$X^* = U^T X V = \begin{pmatrix} X_{11}^* & \dots & X_{1q}^* \\ \vdots & \vdots & \vdots \\ C_{p1}^* & \dots & C_{pq}^* \end{pmatrix} \quad (3.5)$$

y aplicando la propiedad de ortogonalidad  $U^T U = I$ ,  $V^T V = I$ ; la ecuación (3.1) es equivalente a

$$A^* X^* + X^* B^* = C^* \quad (3.6)$$

y si las particiones  $A^*$ ,  $B^*$ ,  $C^*$  y  $X^*$  son conformes, entonces

$$A_{kk}^* X_{kl}^* + X_{kk}^* B_{ll}^* = C_{kl}^* - \sum_{j=1}^{k-1} A_{kj}^* X_{jl}^* - \sum_{i=1}^{l-1} X_{ki}^* B_{il}^* \quad (3.7)$$

Las ecuaciones anteriores pueden resolverse sucesivamente para  $X_{11}^*$ ,  $X_{21}^*$ , ...,  $X_{p1}^*$ ,  $X_{22}^*$ , ... con la solución de (3.1) dada por:

$$X = U X^* V^T \quad (3.8)$$

## Referencias de cálculo numérico

*El cálculo anterior es obtenido mediante la subrutina AXPXB codificado en lenguaje Fortran presentado en ORACLS [2]. Este programa fue obtenido del algoritmo presentado en [4].*

# Apéndice C

## Programas en MATLAB

### C.1 Programa de retroalimentación de estado de norma mínima

#### Programa Principal

```
%  
% K = retromul(A,B,G,ATILDE)  
%  
% Esta función obtiene la matriz de retroalimentación de estado  
% de norma menor a un valor previamente establecido aplicando  
% el algoritmo presentado en [3].  
%  
% A - Matriz A del sistema (1)  
% B - Matriz B del sistema (1)  
% G - Matriz arbitraria  
% ATILDE - Matriz cuyos valores propios son los deseados  
%
```

```
function K = retromul(A,B,G,ATILDE)  
[n,m] = size(B);  
Gk = G;  
lim = 1e-3
```

```
Epsilon = lim + 1;
```

```
Fk = zeros(m,n);
```

```
% ejecuta loop mientras Epsilon sea mayor que lim
```

```
while (Epsilon > lim),
```

```
% obtiene el valor de  $X_k$  de la ecuación  $A \cdot X_k + X_k \cdot ATILDE = -B \cdot G_k$ 
```

```
n0 = norm(Fk,'fro');
```

```
Xk = lyap(A,-ATILDE,B*Gk);
```

```
Xki = inv(Xk);
```

```
% obtiene el valor de  $F_k$  y  $H_k$ 
```

```
Fk = Gk*Xki;
```

```
Xkit = Xki';
```

```
Hk = Fk*Xkit;
```

```
% obtiene el valor de  $U_k$  de la ecuación  $ATILDE \cdot U_k + U_k \cdot A = H_k \cdot F_k$ 
```

```
Uk = lyap(ATILDE,-A,-Hk'*Fk);
```

```
% calcula el valor del Gradiente  $\delta I / \delta G$ 
```

```
GRAD = 2*(Hk - B'*Uk');
```

```
% calcula la longitud del paso alfa y obtiene un nuevo valor de
```

```
%  $G_{k+1}$ 
```

```
alfa = fmin('fobj',0,10);
```

```
Gk = Gk - alfa*GRAD;
```

```
% calcula el valor de epsilon
```

```
n1 = norm(Fk,'fro');
```

```
Epsilon = n1 - n0;
```

```
end
```

```
% obtiene el valor de K
```

```
K = Fk;  
end
```

## Función Objetivo

```
function f = fobj(x)  
    Ginc = Gk*Xki + x*GRAD*Xki;  
    f = trace(Ginc'*Ginc);  
end
```

**NOTA:** *Antes de ejecutar el programa es necesario declarar como globales las siguientes variables.*

Gk  
GRAD  
Xki

## C.2 Programa que efectúa el cálculo del margen de robustez en los coeficientes de los polinomios

### Programa Principal

```
%  
% RMAX = MAXRAD(PC)  
%  
% obtiene el máximo valor de r para que el polinomio  
% característico sea robustamente estable.  
%  
% el polinomio nominal es:  
%
```

```

%  $P(s) = sn + PC(2) sn-1 + \dots + PC(n)$ 
%
% el polinomio perturbado es:
%
%  $P(s,q) = sn + [PC(2) + qn-1] sn-1 + \dots + [PC(n) + q0]$ 
%
%  $-r < qi < r$ 
%
```

```

function rmax = maxrad(pc)
    n = max(size(pc));
```

```

% Detección de estabilidad del polinomio nominal
```

```

    ra = roots(pc);
    for i=1:n-1,
        if (real(ra(i)) >= 0),
            apu=1;
        end
    end
    if (apu == 1),
        disp(' ')
        disp('El polinomio es inestable')
        break
    end
```

```

% Construcción de las matrices Hi
```

```

    h0 = zeros(n-1);
    h1 = h0;
    h2 = h0;
    h3 = h0;
    h4 = h0;
    for i=1:4,
        p(i,1) = 0;
    end
    for k=0:1,
        sw = k;
```

```

signo = 1;
for i=2:n,
    sw = sw + 1;
    p(k+1,i) = signo;
    if (sw == 2),
        signo = -1*signo;
        sw = 0;
    end
end
end
end
p(3,:) = -p(1,:);
p(4,:) = -p(2,:);
h0 = forma(h0,pc,[1,2]);
h0 = forma(h0,pc,[2,1]);
h1 = forma(h1,p(1,:),[1,2]);
h1 = forma(h1,p(1,:),[2,1]);
h2 = forma(h2,p(2,:),[1,2]);
h2 = forma(h2,p(2,:),[2,1]);
h3 = forma(h3,p(3,:),[1,2]);
h3 = forma(h3,p(3,:),[2,1]);
h4 = forma(h4,p(4,:),[1,2]);
h4 = forma(h4,p(4,:),[2,1]);

```

*% Cálculo de la inversa de H0*

```
ih0 = inv(h0);
```

*% Aplicación de la formula de Barmish*

```

l(1) = max(eig(-ih0*h1));
l(2) = max(eig(-ih0*h2));
l(3) = max(eig(-ih0*h3));
l(4) = max(eig(-ih0*h4));
rmax = 1/max(l);

```

```
end
```



## Función auxiliar FORMA.M

```
%  
% A = FORMA(A,PC,IDX)  
%  
% función que forma la matriz de Hurwitz a partir  
% de un polinomio característico  
%  
% A - Matriz Hurwitz  
% PC - Coeficientes del polinomio  
% IDX - clave para la formación de las matrices Hi  
% puede tomar los valores de [1 2] y [2 1]  
%
```

```
function a = forma(a,pc,idx)  
    [n,m] = size(a);  
    p = 0;
```

```
% construcción de las filas de Hi
```

```
    for i = idx(1):2:n,  
        p = p+1;  
        k = p;  
        for j = idx(2):2:n+1,  
            a(i,k) = pc(j);  
            k = k+1;  
        end  
    end
```

```
end
```

### C.3 Programa de Estabilidad $l_p$ -robusta

#### Programa Principal

```
% g = LPROST(A,ALFA,N,W)
%
%      obtiene la gráfica de estabilidad robusta
%      Lp.
%
%      A - Vector que contiene los valores nominales de los
%      coeficientes del polinomio característico.
%
%      ALFA - los valores topes de los coeficientes perturbados
%
%      N - orden del polinomio
%      W - barrido de frecuencia
%
% Copyright (c) 27/DICI/1991.
% Gerardo Romero Galvan F.I.M.E. U.A.N.L.
```

```
function gamma = lprost(a,alfa,p,w)
    dima = size(a);
    dimalfa = size(alfa);
    n = max(dima);
    m = max(dimalfa);
    a(n+1) = 0;
    alfa(m+1) = 0;
    if n ~ = m
        disp('Error en dimension de vectores');
        break
    end
    xnum = a(1);
    xden = alfa(1)^p;
    ynum = a(2);
    yden = alfa(2)^p;
    k = 0;
```

```

yden = alfa(2) ^ p;
k = 0;
for i=3:2:n,
    k = k + 1;
    signo = (-1) ^ k;
    xnum = xnum + signo*a(i)*w.^(i-1);
    ynum = ynum + signo*a(i+1)*w.^(i-1);
    xden = xden + (alfa(i)*w.^(i-1)).^ p;
    yden = yden + (alfa(i+1)*w.^(i-1)).^ p;
end
xde = xden.^(1/p);
yde = yden.^(1/p);
x = xnum./xde;
y = ynum./yde;
mag = (x.^2 + y.^2).^ .5;
magmin = min(mag);
if p >= 4,
    magmin = (magmin + 3*magmin/p ^ 2)/sqrt(2);
end
plot(x,y,'g')
pause
hold on
graflp(magmin,p)
pause
hold off
gamma = magmin;
end

```

## Función GRAFLP

```

%
% [x,y] = GRAFLP(r,p,n)   Grafica la norma lp para cualquier radio r
%                          dado y cualquier p dado.
%
%
```

```

%
%
function [x1,y1] = graf1p(r,p,n)
    if p < 1,
        disp(' ')
        disp(' ')
        disp('***** Error en el coeficiente p de Lp *****');
        disp(' ')
        disp(' ')
        break
    end
    inc = 4*r/n;
    x = -r:inc:r;
    y = (r.^p - (abs(x)).^p).^(1/p);
    xd = fliplr(x);
    yd = fliplr(y);
    xx = [x xd];
    yy = [y -yd];
    plot(xx,yy,'g')
    pause
    x1 = xx;
    y1 = yy;
end

```

# Referencias

- [1] Anderson B.D.O. and Moore J.B., *Optimal Control*, Prentice-Hall Inc, 1990.
- [2] Armstrong E.S., *ORACLS, A Design System for Linear Multivariable Control*, Marcel Dekker Inc, 1980.
- [3] Barmish B.R., "*New Tools for Robust Control Systems*", curso corto, Monterrey N.L., México, Enero de 1992.
- [4] Bartels R.H. and Stewart G.W., "*Solution of Matrix Equation  $AX + XB = C$  [F4]*", Algorithm 432, Collected Algorithms from CACM.
- [5] Chen C.T., *Linear System Theory and Design*, CBS College Publishing, 1984.
- [6] Cuthbert T.R. Jr, *Optimization Using Personal Computer with Application to Electrical Networks*, John Wiley & Sons Inc, 1977.
- [7] D'azzo J.J. and Houpis C.H., *Linear Control System Analysis and Design Conventional and Modern*, Third Edition, McGraw-Hill, 1988.
- [9] De Souza E. and Bhattacharyya S.P., "*Controllability, Observability and the Solution of  $AX - XB = C$* ", Linear Algebra and Its Applications 39, 1981.
- [8] Dongarra J.J., Moler C.B., Bunch J.R., Stewart G.W., *LINPACK User's Guide*, SIAM, Philadelphia 1979.
- [10] Garbow B.S., Boyle S.M., Dongarra J.J., Moler C.B., *Matrix Eigensystem Routines EISPACK*, Lecture Notes in Computer Science, Volume 51, Springer-Verlag, 1977.
- [11] Golub G.H. and Van Loan C.F., *Matrix Computations*, John Hopkins University Press, 1983.

- [12] Gourishankar V., "*Further Comments on the Flexibility Offered by State Feedback in Multivariable System Beyond Closed Loop Eigenvalue Assignment*", IEEE Trans Aut Cont, Vol AC-22, N.5, pp. 888-889, Octubre 1977.
- [13] Householder A.S., *The Theory of Matrices in Numerical Analysis*, Dover, 1964.
- [14] Kalman R.E. and Bucy R.S., "*New Results in Linear Filtering and Prediction Theory*" IEEE Press, Kalman Filtering: Theory and Application, pp. 34-47, 1985.
- [15] Kautsky J. and Nichols N., "*Robust Multivariable Eigenvalue Assignment by State Feedback in Linear System*", Contemporary Mathematics, 1985.
- [16] Keel L.H., Fleming J.A., Bhattacharyya S.P., "*Minimum Norm Pole Assignment via Sylvester Equation*", Contemporary Mathematics, 1985.
- [17] Kuo B.C., *Sistemas Automáticos de Control*, CECSA, 1987.
- [18] Lancaster P. and Tismenetsky M., *The Theory of Matrices with Applications*, Second Edition, Academic Press, 1985.
- [19] Luenberger D.G., *Optimization by Vector Space Methods*, John Wiley & Sons, 1969.
- [20] Moore B.C., "*On the Flexibility Offered by State Feedback in Multivariable System Beyond Closed Loop Eigenvalue Assignment*", IEEE Trans Aut Cont, Volume AC-21, pp. 689-692, Octubre 1976.
- [21] Noble B.N. and Daniel J.W., *Algebra Lineal Aplicada*, Prentice-Hall, 1989.
- [22] Ogata K., *Ingenieria de Control Moderna*, Prentice-Hall, 1980.
- [23] Pierre D.A., *Optimization Theory with Applications*, Dover, New York, 1986.
- [24] Porter B. and D'azzo J.J., "*Comments on the Flexibility Offered by State Feedback in Multivariable Systems Beyond Closed Loop Eigenvalue Assignment*", IEEE Trans Aut

Cont, Vol AC-22, N.5, pp 888-889, Octubre 1977.

- [25] Prawda J., *Métodos y Modelos de Investigación de Operaciones*, LIMUSA, 1990.
- [26] Schied F. and Di-Constanzo, *Métodos Numéricos*, McGraw-Hill, Segunda Edición, 1991.
- [27] Slotine E. J-J. and Weiping L., *Applied Nonlinear Control*, Prentice-Hall, 1991.
- [28] Tsytkin Ya. Z. and Polyak B.T., "*Frequency Domain Criteria for  $l_p$ -Robust Stability of Continuous Linear System*", IEEE Trans Aut Cont, vol AC-36, N-12, pp 1464-1469, diciembre 1991.



