ESCUELA POLITECNICA NACIONAL

FACULTAD DE INGENIERIA ELECTRICA

" IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO "

JOSE RAUL MEJIA RECALDE

TESIS PREVIA A LA OBTENCION DEL TITULO DE INGENIERO

EN ELECTRONICA Y CONTROL

QUITO, ABRIL DE 1991

A MIS PADRES,

A MIS HERMANOS,

A SANDRA

Certifico que el presente trabajo ha sido realizado en su totalidad por el señor:

José Raúl Mejía Recalde.

Ing. Marco Barragán

DIRECTOR DE TESIS

AGRADECIMIENTO

Agradezco en forma especial al Ing. Marco Barragán por su valiosa y acertada dirección en el desarrollo de esta tesis.

Agradezco a todas aquellas personas que de una u otra forma colaborarón desinteresadamente en la realización de este trabajo.

INDICE

CAPITULO IV EJEMPLOS DE APLICACION

CAPITULO V CONCLUSIONES

ANEXO 1.- MANUAL DE OPERACION

2.- DIAGRAMAS DE FLUJO

3.- LISTADOS DE LOS PROGRAMAS

CAPITULO I

INTRODUCCION

1.1 CONDICIONES GENERALES

Para la implementación de un sistema de control, es necesario conocer la dinámica de los sistemas físicos, industriales, etc, a través de las leyes de las ciencias, planteando modelos matemáticos, que intenten explicar las observaciones hechas de los mismos. Los procesos son tan complejos que resulta imposible describirlos o identificarlos en términos de ecuaciones matemáticas, a partir de las leyes de las ciencias como son la Física, Química, Economía, etc.

La descripción matemática de las características dinámicas de un sistema se denomina **modelo matemático**, por lo tanto el primer paso en el análisis de un sistema, es el poseer un modelo matemático.

Los modelos pueden tener muchas formas, dependiendo del sistema que se trate y del análisis a realizarse, así por ejemplo:

- En problemas de control óptimo, es ventajoso tener una representación en ecuaciones diferenciales de primer orden.

- Por otra parte para el análisis de respuesta transitoria o el análisis de respuesta de frecuencia, la representación mediante función de transferencia resulta la más conveniente.

Con el advenimiento de los computadores digitales, se ha desarrollado una serie de nuevos y poderosos procesos analíticos que permitan experimentar con modelos matemáticos que describan algún sistema en interés, naciendo así la importancia de los modelos matemáticos y su construcción como una parte integral de la investigación científica, especialmente en el campo del control de sistemas dinámicos.

Es posible definir al modelo científico, como una abstracción del sistema real, que tiene la posibilidad de utilizarse para propósitos de predicción y control, por lo tanto el objeto de un modelo matemático es dar a los investigadores una aproximación razonable del sistema real y debe incorporar la mayor parte de los aspectos importantes de éste.

1.2 PROPIEDADES DE LOS MODELOS

Los modelos matemáticos constan de cuatro elementos bien definidos: componentes, variables, parámetros y relaciones funcionales.

<u>Componentes</u>.- Los componentes de un sistema son los elementos constitutivos de éste, son propios de cada sistema.

En la mayoría de los casos se desconoce los componentes que conforman la estructura de la planta, y ante la dificultad de determinarlos a través de las ciencias básicas, se procede a una identificación de parámetros de algún modelo que representa al sistema.

<u>Variables</u>.- Las variables se emplean para relacionar un componente con otro y se clasifican en: exógenas, variables de estado y endógenas.

- Las varibles exógenas son las independientes o de entrada del modelo y se supone que han sido predeterminadas y proporcionadas independientemente del sistema que se modela, pueden considerarse que estas variables actúan sobre él, pero no reciben acción alguna de parte de él.

  Las variables exógenas se pueden clasificar en controlables y no controlables.
- Las variables de estado determinan el estado de un sistema o de uno de sus componentes.

- Las variables endógenas son las dependientes o de salida del sistema y son generadas por la interacción de las variables exógenas con las de estado.

Parámetros.- Los parámetros son los valores que identifican al sistema.

Relaciones funcionales.- Las relaciones funcionales son las que describen la interacción de las variables y componentes.

1.3 CLASIFICACION DE LOS MODELOS

Los modelos se clasifican en:

a.- MODELOS DETERMINISTICOS.

En este tipo de modelos, se suponen relaciones exactas para las características de operación, en lugar de funciones densidad de probabilidad.

b.- MODELOS ESTOCASTICOS.

Son aquellos modelos, en los que por lo menos una de las características de operación son del tipo aleatorio, constituyéndose los más cercanos a la realidad.

c.- MODELOS ESTATICOS.

Son aquellos en los que no se considera a la variable tiempo, las relaciones entre las características del sistema se dan cuando éste se encuentra en equilibrio.

d.- MODELOS DINAMICOS.

Son aquellos que tratan de las interacciones que varían con el tiempo.

1.4 FORMULACION DE LOS MODELOS

En la formulación de modelos matemáticos se debe considerar los siguientes aspectos:

1.- Número de variables del modelo.

La utilización de muy pocas variables exógenas (algunas de las cuales pueden ser estocásticas) que afectan a las variables endógenas darían modelos no válidos, en tanto que una abundancia de ellas haría imposible la simulación en un computador.

2.- Complejidad del modelo.

Interesa la formulación de modelos matemáticos sencillos que permitan la predioción, la descripoión y el diseño de sistemas de control razonablemente exaotos de un sistema dado, lográndose reducir el tiempo de computación.

3.- Validez del modelo.

En este punto es importante notar si el modelo describe adecuadamente al sistema de interés.

Formulado el(los) modelo(s) que describe(n) el comportamiento de un sistema, es necesario estimar los valores de los parámetros de dicho(s) modelo(s) y probar su significación estadística.

Finalmente, estimado el modelo, es necesario probarlo si describe adecuadamente al sistema de interés, por lo que es necesario aplicar pruebas de bondad de ajuste que determinen qué tan bien se ajusta, una distribución hipotética de probabilidad a los datos del mundo real.

El proceso de construcción del modelo de un sistema y la estimación de los mejores valores de los parámetros desconocidos a partir de datos experimentales, se conoce como " identificación de sistemas ".

En el desarrollo de técnicas de identificación de sistemas, se considera en forma básica el propósito para el cual el modelo va a ser usado; pudiendo ser: Control, Simulación, Predicción, etc.

El objetivo de esta tesis es el de implementar algorítmos de identificación de parámetros de un sistema representado en variables de estado.

1.5 RESTRICCIONES EN LA FORMULACION DE LOS MODELOS

En la identificación de un sistema se realiza algunas consideraciones iniciales que, para este caso son las siguientes:

- Sistemas de simple entrada y simple salida en el caso discreto.

- Sistemas lineales y fijos, por lo tanto la ecuación de salida y las ecuaciones de estado son, combinaciones lineales con coeficientes constantes de la señal de entrada y de las variables de estado.

- Sistemas de orden conocido hasta de cuarto orden, por cuanto el proceso de determinación del orden del modelo así como el utilizar modelos de mayor de cuarto orden no son útiles ni adecuados en la práctica.

- Sistema representado por variables de estado.

Para el caso discreto, el modelo correspondiente en el espacio de estado es:

$x(k+1) = A.x(k) + B.u(k)$ (Ecuación de Estado)

$y(k) = C.x(k) + D.u(k)$ (Ecuación de Salida)

Donde $u(k)$, $y(k)$ constituyen muestreos de la señal de entrada y de la señal de salida respectivamente, $x(k)$ es el vector de estado, donde el período de muestreo T se omite cuando se expresan las variables de estado al instante $t = TK$, $k = 0, 1, \ldots$.

Desarrollando las ecuaciones de estado y de salida, se tienen las siguientes expresiones:

$$x(k+1) = A.x(k) + B.u(k) \qquad (1.1)$$

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ \cdots \\ \cdots \\ xn(k+1) \end{vmatrix} = \begin{vmatrix} a11 & a12 & \ldots & \ldots & a1n \\ a21 & a22 & \ldots & \ldots & a2n \\ a31 & \ldots & & & \\ \cdots & & & & \\ an1 & an2 & \ldots & \ldots & ann \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ \cdots \\ \cdots \\ xn(k) \end{vmatrix} + \begin{vmatrix} b11 \\ b21 \\ \ldots \\ \ldots \\ bn1 \end{vmatrix} . u(k)$$

$$y(k) = C.x(k) + D.u(k) \tag{1.2}$$

$$y(k) = \begin{vmatrix} c11 & c12 & \ldots & \ldots & c1n \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ \cdots \\ \cdots \\ xn(k) \end{vmatrix} + D.u(k)$$

Los parámetros a identificar son los elementos componentes de las matrices A, B, C, D.

Se utiliza las formas canónicas controlable, observable y el modelo ARMA (Auto Regressive Moving Average) que son las más apropiadas para identificación, por disponer el menor número de parámetros por identificar si se dispone de los datos de entrada y de salida medidos en la planta.

Debido a la importancia del ruido (aleatorio) en el análisis de los sistemas, se introduce en los modelos elementos randómicos (ruido=variable exógena), para ello, se utiliza el modelo **ARMAX** (Auto Regressive Moving Average Exogeneous variable).

Las técnicas de identificación por desarrollarse son del tipo paramétricas, ya que se supone que el modelo que representa a la planta en estudio tiene un número finito de parámetros a diferencia de los sistemas no paramétricos que se describe por un número infinito de parámetros.

Se desarrolla la base teórica de los algoritmos de estimación de parámetros siguientes:

- Mínimos cuadrados ordinarios y recursivos para sistemas determinísticos y probabilísticos. (**Least-Squares Estimate [LS]**).

- Mejor estimador lineal insesgado. (**Best Linear Unbiased Estimate [BLUE]**).

- Probabilidad máxima o máximo de verosimilitud. (**Maximum Likelihood Estimate [MLE]**).

Los algoritmos de estimación de parámetros a implementarse en computador son los siguientes:

- Mínimos cuadrados ordinarios y recursivos para sistemas determinísticos y probabilísticos (Mejor estimador lineal insesgado) representados mediante variables de estado de las formas canónicas (controlable y ARMA).

Habiéndose seleccionado el modelo que describe la planta, y teniéndose las técnicas que permiten estimar los parámetros θ, los cuales son la mejor representación de los datos, se requiere tener una idea del error entre los valores estimados $\hat{\theta}$ y los valores verdaderos θ^o desconocidos, a partir de las señales de entrada $\{u(k)\}$ y de salida $\{y(k)\}$.

En esta tesis se enfocan tres criterios de error que son: Ecuación de error, Error de salida, Error de predicción de salida.

1.6 CONTENIDO DE LA TESIS

En el capítulo I, se desarrolla conceptos introductorios de los modelos, sus propiedades, clasificación, mecanismos de formulación y restricciones.

En el capítulo II, se aborda tres aspectos teóricos en lo referente a:

1.- Procesos estocáticos, estadísticas de un proceso y tipos de procesos estocásticos estacionarios y no-estacionarios. Conceptos sobre ruido blanco y criterios de estimadores justo o insesgado y de probabilidad máxima.

2.- Descripción de sistemas por varibles de estado, caso determinístico y probabilístico, analogía entre modelos representados por ecuación de diferencias, función de transferencia y variables de estado de las formas canónicas controlable, observable y ARMA para sistemas discretos.

3.- Tratamiento del error a través de la ecuación de error, error de salida y el error de predicción de salida.

En el capítulo III, se desarrolla la base teórica de los algoritmos de identificación siguientes:

- Mínimos cuadrados ordinarios, ponderados y recursivos.
- Estimador lineal Insesgado.
- Estimador de probabilidad máxima o de máxima verosimilitud.

En la parte final del capítulo III, se indican los algoritmos implementados en el computador para los casos de mínimos cuadrados ordinarios, recursivos (ponderados) y del mejor estimador lineal así como el de generación de las señales de entrada y de salida.

En el capítulo IV, se desarrollan ejercicios de aplicación para sistemas de primero, segundo y tercer orden con la totalidad de opciones que prestan los programas implementados.

En el capítulo V, se establece las conclusiones de los ejemplos planteados en el capítulo IV.

En el Anexo 1 se encuentran el manual de operación, diagramas de flujo y el listado de los programas implementados. Los programas son desarrollados en lenguaje QuickBasic versión 4.5, debido a la gran versatilidad y aceptación de este lenguaje a nivel de la Facultad de Ingeniería Eléctrica.

CAPITULO II

TEORIA

En el estudio de un sistema por medio de un modelo, se busca que éste sea una aproximación razonable del sistema real.

En procesos reales, especialmente industriales, la señal de entrada se encuentra afectada por agentes externos indeseables, por lo que es necesario para tener una mejor aproximación al sistema real, introducir en el modelo un elemento que comprenda la presencia de variables aleatorias, teniéndose el caso de modelos estocásticos.

No es objetivo de esta tesis el tratamiento de procesos estocásticos, ya que se va a realizar un estudio de las técnicas de identificación para el caso determinístico, por lo que, solo se realizará un estudio introductorio para el caso estocástico.

2.1. CONCEPTOS SOBRE PROCESOS ESTOCASTICOS.

Un sistema es determinístico, si la señal de entrada en un instante dado es la que determina la señal de salida de manera única.

Un sistema es estocástico, si la señal de salida, se ve afectada por la señal de entrada y por ruido aleatorío normalmente aditivo.

En teoría de probabilidades, se conoce que con cada experimento ϵ del tipo que se considere, se define al espacio **muestral** (S), como el conjunto de todos los resultados posibles de ϵ, pudiendo ser este finito o infinito, numerable o no numerable.

Una función x que asigna a cada uno de los elementos s de S, un número real xi, se denomina **variable aleatoria**. A cada uno de estos números se le asigna una probabilidad de que ocurra p(xi) = P(x=xi) (llamada la probabilidad de xi).

La función P se llama **función de probabilidad** de la variable aleatoria **x**. La colección de pares (xi,p(xi)) se conoce como **distribución de probabilidades de x**.

Por ejemplo:

Se define ϵ= lanzar dos monedas al aire, y si sale cara o sello, el resultado es c o s respectivamente

El espacio muestral S={cc,cs,sc,ss}={ϵ1,ϵ2,ϵ3,ϵ4}

donde ϵ1=cc, ϵ2=cs, ϵ3=sc, ϵ4=ss.

Si se define la variable aleatoria **x** como:

x ={número de caras obtenidas en los dos lanzamientos}

por lo tanto x(cc)=2, x(cs)=x(sc)=1, x(ss)=0

la probabilidad es:

P(X=1) = P(cs)+P(sc) = 1/4+1/4=1/2

P(X=2) = P(cc) = 1/4

P(X=0) = P(ss) = 1/4

Un proceso estocástico es una ampliación del concepto de variables aleatorias, en donde a cada resultado de un experimento se le asigna una función del tiempo x(t,ϵ) (funciones muestrales) de cualquier tipo.

Al conjunto de resultados posibles de un proceso estocástico se le conoce como **totalidad**.

Por ejemplo:

Si se toma el caso anterior de las dos monedas:

A cada uno de los resultados ϵ1, ϵ2, ϵ3, ϵ4 se asocia una función del tiempo x(t,ϵ1), x(t,ϵ2), x(t,ϵ3), x(t,ϵ4), por ejemplo puede ser x(t,ϵk) = seno kt.

La totalidad en este caso tiene 4 funciones muestrales a saber: seno t, seno 2t, seno 3t y seno 4t.

2.1.1 ESTADISTICAS DE UN PROCESO

Considérese una totalidad en donde, en cada instante de tiempo, un sistema puede estar afectado por una función de ruido, el conjunto de todas estas funciones son una totalidad (figura 2.1).

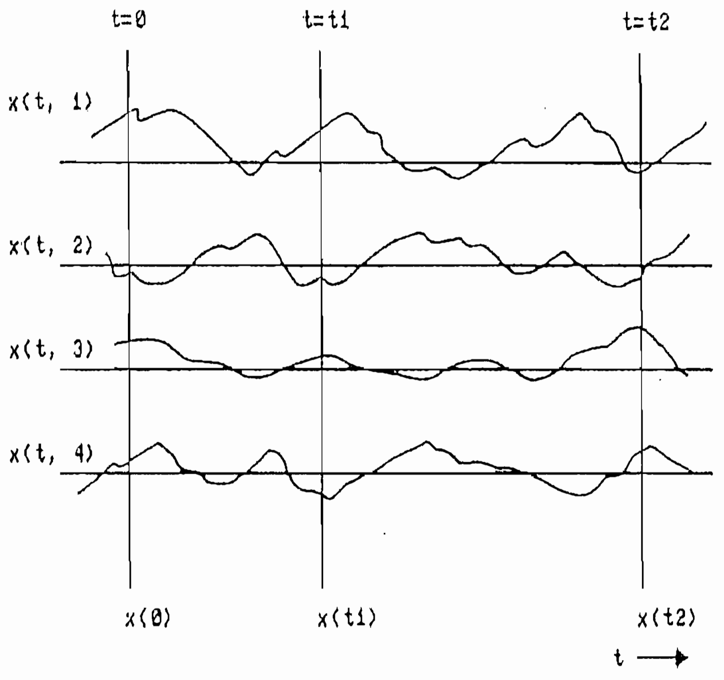


Fig 2.1 Conjunto de funciones de ruido, que representan una totalidad.

En algun instante de tiempo t=t1 la amplitud de las funciones muestrales $x(t,\epsilon)$, es un conjunto de valores $x(t1,\epsilon k)$ que tienen una distribución de probabilidad, donde lógicamente $x(t1,\epsilon k)$ es una variable aleatoria; similarmente para otros instantes t2, t3, ..., tn, definiéndose un conjunto de variables aleatorias sobre la totalidad.

Generalmente se omite ϵ, por lo que la expresión x(t), representa un proceso estocástico, y x(ti) una variable aleatoria.

Las estadísticas de cada una de estas variables individualmente, son llamadas estadísticas de primer orden de un proceso estocástico. Similarmente, las estadísticas de n variables conjuntamente, se conocen como estadísticas de orden n.

2.1.1.1 ESTADISTICAS DE PRIMER ORDEN

Las estadísticas de primer orden, especifican funciones densidad de probabilidad de las amplitudes de las funciones muestrales en un instante t.

A partir de la variable aleatoria x(t), también notada como x, cuya función densidad de probabilidad, es p(x;t), se puede tener el valor medio y el valor medio cuadrático por las siguientes ecuaciones:

$$\overline{x}(t) = \int_{-\infty}^{\infty} x * p(x;t) \, .dx \tag{2.1}$$

$$\overline{x^2}(t) = \int_{-\infty}^{\infty} x^2 * p(x;t) \, dx \tag{2.2}$$

En los procesos estocásticos, al igual que en teoría de probabilidades, se puede definir por analogía lo siguiente:

> Si hay un número total de funciones muestrales N, y se tiene m funciones muestrales; entonces la frecuencia relativa se puede escribir como sigue:

$$p(x;t) \, dx = \frac{m}{N} \tag{2.3}$$

$$p(x;t) = \frac{m}{Ndx} \tag{2.4}$$

2.1.1.2 ESTADISCAS DE SEGUNDO ORDEN O MAS

Las estadísticas de primer orden de un proceso no permanecen constantes con el tiempo, y no dañ todas las especificaciones de un proceso en un intervalo de tiempo (t1,t2), haciéndose necesario definir la relación entre los valores de la variable aleatoria en el instante t=t1 y los valores en el instante t=t2, notados x(t1) y x(t2) respectivamente (fig 2.2).

Para el análisis en dos instantes de tiempo t1, t2, se define la **función densidad de probabilidad conjunta** como p(x1,x2), también conocida como función densidad de probabilidad de segundo orden.

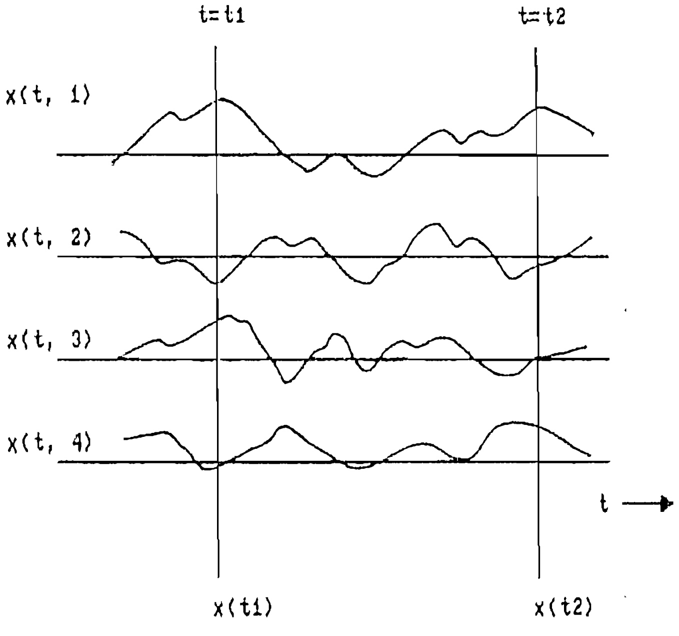


Fig 2.2 Funciones muestrales, para los instantes t=t1 y t=t2

En forma similar como se puede tener la frecuencia relativa para las estadísticas de primer orden, para las estadísticas de segundo orden, se tienen las siguientes expresiones:

$$p(x1,x2;t1,t2)\,dx1dx2 = \frac{m_{12}}{N} \qquad (2.5)$$

$$p(x1,x2;t1,t2) = \frac{m_{12}}{Ndx1dx2} \qquad (2.6)$$

m12 = es el número de funciones muestrales, que ocurren simultáneamente en la totalidad (N).

N = es la totalidad

El tratamiento de estadísticas de mayor orden no es necesario, por cuanto es suficiente con las estadísticas de primero y segundo orden para determinar la densidad del espectro de potencia.

A partir de las estadísticas de segundo orden, se definen dos conceptos importantes de las variables aleatorias en los instantes t1 y t2 que son: la Autocorrelación y la Autocovarianza.

AUTOCORRELACION

La autocorrelación, indica la relación entre las variables aleatorias x1 y x2 en el mismo proceso; en otras palabras, es la medida de la dependencia de las amplitudes de las funciones muestrales en t=t1, con las amplitudes de las mismas funciones en t=t2, de ahí el término "auto".

La función de autocorrelación en la totalidad es el valor esperado del producto entre x1 y x2.

$$R_{x1x2}=E\{x1.x2\} = \overline{x1.x2} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x1x2p(x1,x2;t1,t2)\,dx1dx2 \qquad (2.7)$$

<u>AUTOCOVARIANZA</u>

A la expresión $Cx1x2 = E\{(x1 - \overline{x1})(x2 - \overline{x2})\}$, se le llama covarianza de x1 y x2, notada por $\sigma x1x2$.

Aplicando las propiedades del valor esperado, se tiene el siguiente desarrollo:

$$Cx1x2 = E\{(x1 - \overline{x1})(x2 - \overline{x2})\} = E\{x1.x2 - x1.\overline{x2} - \overline{x1}x2 + \overline{x1}\overline{x2}\}$$

$$= E\{x1.x2\} - \overline{x1}.\overline{x2}$$

como $E\{x1\} = \overline{x1}$

en consecuencia $Cx1x2 = E\{x1.x2\} - E\{x1\}.E\{x2\}$ (2.8)

La autooovarianza se define como $Cx1x2 = E\{x1.x2\} - E\{x1\}.-E\{x2\}$, donde si alguno de los procesos tienen un valor esperado igual a cero:

entonces: Autocorrelación = Autocovarianza

$$x1 \text{ ó } x2 = 0 \; ; \; R_{x1x2} = C_{x1x2}$$

2.1.2 PROCESOS ESTOCASTICOS ESTACIONARIOS Y NO-ESTACIONARIOS

Un proceso es estacionario, cuando sus parámetros permanecen constantes en el tiempo; en caso contrario se dice que es no-estacionario.

Los sistemas estocásticos son **estrictamente estacionarios**, cuando ninguna de sus estadísticas dependen del tiempo. En este caso, la varianza, la correlación, la autocorrelación, la función densidad de probabilidad dependen únicamente de la diferencia de tiempos.

Para ilustrar la condición, de que los sistemas estocásticos, son estrictamente estacionarios se toma, por ejemplo las estadísticas de segundo orden; a partir de las cuales se puede realizar una extensión, para el caso de las estadísticas de orden n.

Sea la función densidad de probabilidad de segundo orden,

p(x1, x2; t1, t2) que está desplazada un intervalo T, se tiene entonces:

$$p(x1,x2;t1+T,t2+T) = p(x1,x2;t2-t1) = p(x1,x2;\tau)$$

Donde τ representa la diferencia de tiempos.

En conclusión, un sistema es estacionario cuando sus estadísticas de primer orden no dependen del tiempo, y para el caso de estadísticas de segundo orden o más, estas dependen de la diferencia de tiempo.

2.1.3 RUIDO BLANCO

El ruido blanco, es un proceso cuyo espectro de frecuencias tiene todos los componentes de frecuencias en igual proporción.

Para entender este concepto del ruido, es necesario abordar el teorema del límite central, el mismo que establece lo siguiente:

" Una variable aleatoria, constituida por la suma de un cierto número de variables aleatorias independientes, tiende a ser gaussiana, (de media 0 y varianza 1 notada como N(0,1)) cuando el número de variables aleatorias crece adquiriendo valores muy grandes."

Un proceso se dice que es gaussiano, si las variables aleatorias x(t1), x(t2),............, x(tn) son conjuntamente gaussianas para todo n y para todo grupo de (t1, t2, ...-..., tn). A los procesos de ruido blanco que son gaussianos se los conoce como ruido blanco gaussiano.

Un proceso gaussiano se identifica completamente por su valor medio y su función de autocorrelación. Para el caso de ruido blanco se tiene:

- La densidad espectral de frecuencia esta dado por P(w) = A/2
- La función de autocorrelación $R(\tau) = A/2\ \delta(\tau)$ (fig. 2.3), donde $\delta(\tau)$ es la función Delta de Dirac definida así:

$$\delta(\tau) = 0 \quad \text{si} \quad \tau <> 0 \qquad 1)$$

$$\int_{-\infty}^{\infty} \delta(\tau)\, dt = 1 \qquad 2)$$

De las propiedades de la función Delta y de la función de autocorrelación $R(\tau) = A/2\ \delta(\tau)$, se desprende que la autocorrelación es 0 para todo τ excepto para $\tau = 0$; lo que implica que las variables aleatorias $x(tk)$, $x(tj)$ son no correlacionadas si $tk <> tj$; por tanto dos muestras diferentes de ruido blanco gaussiano son no correlacionados.

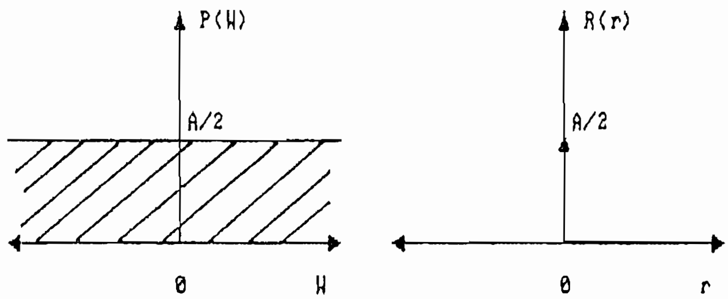


Fig 2.3 Señal de ruido blanco gaussiano a) Espectro de potencia b) Función de autocorrelación

Para el caso de dos señales independientes de ruido blanco gaussiano se tiene:

- Si $k<>j$ son no correlacionados y su valor esperado $Ev(k) = 0$ para algún instante de k; entonces el coeficiente de correlación $\rho = 0$.

- Si $k = j$ son correlacionados y su valor esperado $E[v(k).v(j)] - Ev(k).Ev(j) = \rho_{kj}.\sigma_k.\sigma_j$, se tiene:

$$E[v(k)^2] = \rho.\sigma 2; \quad si \quad \rho = 1;$$

$$E[v(k)^2] = \sigma^2$$

Resumiendo:

$$si\ k<>j\ ;\ E(v(k)v(j)) = 0 \tag{2.9}$$

$$si\ k=j\ ;\ E(v(k))E(v(j)) = E(v(k)^2) = \sigma^2\ ;\ \rho = 1$$

Cuando existe más de un componente de ruido, se tiene el caso de ruido gaussiano multidimensional, las componentes de ruido son vectores y las expresiones (2.9) son matrices, las mismas que se pueden escribir de la siguiente forma:

$$Si\ k<>;\quad E\ V.V^T = 0$$

$$Si\ k=j;\quad E\ V.V^T = \sigma^2.I$$

Donde V es la matriz de covarianza, e I es la matriz identidad.

Para algún valor de k, el valor esperado es igual a cero ($E(v(k)) = 0$), para dos instantes k y j el coeficiente de correlación[1] notado ρkj es:

$$\rho kj = E\{[v(k) - Ev(k)][v(j) - Ev(j)]\}/\sqrt{V(v(k)).V(v(j))}$$

$$\rho kj = E[v(k).v(j)] - Ev(k).Ev(j)/\sqrt{V(v(k)).V(v(j))}$$

Donde el numerador se llama covarianza de k y j y se nota σkj.

Si al denominador se lo separa en dos términos $\sqrt{V(v(k)).V(v(j))} = \sqrt{V(v(j))}.\sqrt{V(v(k))}$ notado como $\sigma_k.\sigma_j$, el coeficiente de correlación se puede expresar así:

$$\rho kj = \sigma_{kj} / \sigma_k.\sigma_j$$

$$\sigma_{kj} = \rho_{kj}.\sigma_k.\sigma_j$$

2.1.4 CRITERIOS DE ESTIMADORES

Sea X una variable aleatoria con una distribución de probabilidades que depende de un parámetro desconocido θ. Sea X1,

[1] Mide el grado de asociación entre k y j.

X2, ... ,Xn una muestra aleatoria de X y sean x1, x2,...., xn los valores muestrales correspondientes. Si g(X1,X2,...,Xn) es una función de la muestra que va a ser usada para estimar θ, se refiere a g como un estimador de θ. El valor que toma g, será mencionado como un estimador de θ, y notado como $\hat{\theta}$.

2.1.4.1 ESTIMADOR JUSTO O INSESGADO

Para el estimador justo o insesgado, conocido de publicaciones técnicas en inglés como **unbiased estimator**; se va a estimar θ de un experimento que se repite algunas veces, donde $\hat{\theta}$ es el estimador de θ. En cada experimento se tendrá un estimador de θ, por lo tanto, se desea que la media aritmética de los estimadores tiendan a θ.

El estimador $\hat{\theta}$ es un **estimador insesgado** de θ, si su valor esperado es igual a θ, para todo θ, caso contrario se dice que es un estimador con sesgo o vicio, y la diferencia $\theta - E(\hat{\theta}) = b$, es conocido como sesgo.

$$E(\hat{\theta}) = \theta \text{ ; para todo } \theta \qquad \textbf{INSESGADO} \qquad (2.10)$$

Se dice que $\hat{\theta}$ es el mejor estimador insesgado de θ si:

a) $E(\hat{\theta}) = \theta$

b)

$$\hat{\theta} = \sum_{i=1}^{n} aixi$$

es decir $\hat{\theta}$ es una función lineal de la muestra.

c) $V(\hat{\theta}) \leq V(\theta^{*})$, en donde θ^{*} es cualquier otro estimador de θ que cumplen con las condiciones a) b), donde V es la varianza de $\hat{\theta}$, es decir que entre todas las estimaciones insesgadas de θ; $\hat{\theta}$ tiene la varianza más pequeña.

Cuando el muestreo comienza a ser largo x1,......,xn, se dice que el estimador $\hat{\theta}$ es una estimación convergente de θ, si:

$$limit_{n \to \infty} Prob[|\hat{\theta}-\theta|>\epsilon] = 0 \quad \forall \epsilon > 0$$

Esta definición establece que una estimación es convergente, si al aumentar el tamaño de la muestra, el estimador de $\hat{\theta}$ converge a θ.

Para determinar la convergencia a partir de la definición, resulta complicado, por lo que se utiliza el teorema siguiente que resulta más útil.

Si **X** es discreta, $L(X1,X2,....,Xn;\theta)$ representa P[-X1=x1, X2=x2,....,Xn=xn], mientras que si es continua, $L(X1,X2,....,Xn;\theta)$ representa la función densidad de probabilidad conjunta de (X1,X2,....,Xn).

Si se ha obtenido la muestra L(X1,.........,Xn), los valores muestrales (x1,.......,xn) son conocidos, y θ es desconocido.

" El **estimador de máxima verosimilitud** de θ, llamado $\widehat{\theta}$, basado en una muestra aleatoria X1,X2,....,Xn es el valor de θ que maximiza a $L(X1,X2,....,Xn;\theta)$, considerando como una función de θ para una muestra dada X1,X2,....,Xn, en donde L está definida por la ecuación (2.11)".

A fin de establecer el estimador ML, se debe determinar el valor máximo de una función; puesto que la función logaritmo ln x, es una función creciente de x, es mucho màs conveniente encontrar el máximo del logaritmo de la función de probabilidad L, que de la función, la misma que tendrá su máximo para el mismo valor de θ. Bajo condiciones de que θ es un número real y que $L(X1,X2,....,Xn;\theta)$ es un función diferenciable de θ; entonces se puede obtener el estimador ML de θ al resolver lo que se conoce como la **ecuación de verosimilitud**.

$$(2.12) \qquad \frac{\partial}{\partial\theta}\ln L(X1,X2,\ldots.Xn;\theta) = 0$$

El estimador de máxima verosimilitud tiene la propiedad asintótica la misma que es más robusta, que la propiedad de convergencia para el caso de estimadores insesgados; especialmente si las estimaciones se hacen en una muestra grande.

" Si $\hat{\theta}$ es un estimador ML para el parámetro θ, definido sobre una muestra aleatoria x1,........,xn de una variable aleatoria X entonces para n suficientemente grande, la variable aleatoria θ tiende asintóticamente a la distribución normal N(θ,1/B), en donde

$$B = n.E[\frac{\partial}{\partial\theta}Lnf(X;\theta)]^2$$

2.2 DESCRIPCION DE SISTEMAS POR VARIABLES DE ESTADO, CASOS DETERMINISTICO Y PROBABILISTICO

Las variables de estado de un sistema dinámico, son el conjunto más pequeño de variables, que determinan el estado del sistema, tal que, el conocimiento de esas variables en t

= to, conjuntamente con las entradas para t >= to, determinan totalmente el comportamiento del sistema para cualquier tiempo t>= to. El estado de un sistema dinámico en el tiempo t queda determinado unívocamente por el estado en el tiempo to y la entrada para t>to. Al tratar con sistemas lineales invariantes en el tiempo, generalmente se elige el tiempo de referencia to igual a cero.

Las variables de estado de un sistema, no han de ser necesariamente magnitudes físicamente medibles u observables, sin embargo, en la práctica es conveniente optar por variables de estado, magnitudes fácilmente medibles.

Las ecuaciones de estado de un sistema lineal invariante en el tiempo para el caso discreto son:

$$\mathbf{x}(k+1) = \mathbf{A}.\mathbf{x}(k) + \mathbf{B}.\mathbf{u}(k) \quad \text{(Ecuación de Estado)} \quad (2.13)$$

$$\mathbf{y}(k) = \mathbf{C}.\mathbf{x}(k) + \mathbf{D}.\mathbf{u}(k) \quad \text{(Ecuación de Salida)} \quad (2.14)$$

En donde:

$\mathbf{u}(k)$ = Vector de entrada (mx1).

$\mathbf{y}(k)$ = Vector de salida (1x1).

$\mathbf{x}(k)$ = Vector de estado (nx1).

Las matrices **A** (nxn), B (nxm), C (1xn), D (1xm) están constituidas por los parámetros del sistema. Estos valores son constantes para el caso de sistemas invariantes en el tiempo.

En el caso de una señal de entrada y una salida, las matrices B y C son de orden nx1 y 1xn respectivamente, la matriz D es un escalar y en la mayoría de los casos es cero.

El período de muestreo T se omite cuando se expresan las variables al instante t = TK, k = 0, 1,.... .

Las ecuaciones de estado se encuentran representadas gráficamente por el diagrama de bloques de la fig 2.4.

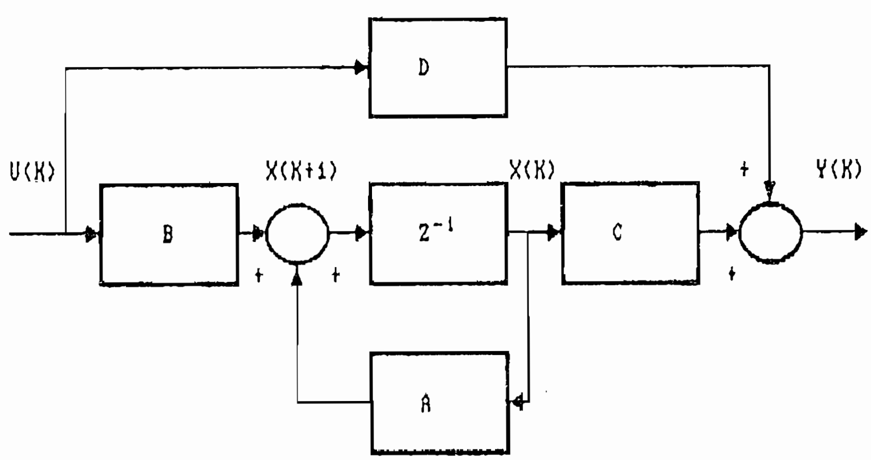


Fig. 2.4 Diagrama de bloques de las ecuaciones deestado

De las ecuaciones de estado (2.13) y (2.14), los parámetros a identificar son los elementos componentes de las matrices A, B, C, D, en donde la matriz D igual a cero.

Para un sistema de tercer orden, la representación en variables de estado son:

$$x(k+1) = A.x(k) + B.u(k)$$

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} b11 \\ b21 \\ b31 \end{vmatrix} . u(k)$$

$$y(k) = C.x(k) + D.u(k)$$

$$y(k) = \begin{vmatrix} c11 & c12 & c13 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix}$$

La matriz de parámetros desconocidos θ para el sistema de tercer orden está constituido por 15 elementos que son:

$$\theta = [a11\ a12\ a13\ a21\ a22\ a23\ a31\ a32\ a33\ b11\ b21\ b31\ c11\ c12\ c13]$$

Sacando la transformada Z de las ecuaciones de estado se puede obtener la matriz función de transferencia, de la siguiente manera:

De las ecuaciones de estado $\mathbf{x}(k+1) = \mathbf{A}.\mathbf{x}(k) + \mathbf{B}.\mathbf{u}(k)$ y $\mathbf{y}(k) = \mathbf{C}.\mathbf{x}(k) + \mathbf{D}.\mathbf{u}(k)$, se toma la transformada Z con condiciones iniciales igual a cero, entonces:

$Z.X(Z) = A.X(Z) + B.U(Z)$

$Y(Z) = C.X(Z) + D.U(Z)$; despejando $X(Z)$

$X(Z) = (Z.I - A)^{-1}.B.U(Z)$; reemplazando en $Y(Z)$

$Y(Z) = C.[(Z.I - A)^{-1}.B + D].U(Z)$

Si $G(Z) = Y(Z)/U(Z)$; entonces

$G(Z) = C.(ZI-A)^{-1}.B + D$ **Matriz Función de Transferencia** (2.15)

- $G(Z)$ **es propia** si,

$$limit_{Z\to\infty} G(Z) = cte$$

lo que implica que el orden del numerador es menor o igual que, el orden del denominador.

- $G(Z)$ **es estrictamente propia** si,

$$limit_{Z\to\infty}G(Z) = 0$$

Para que la Matriz Función de Transferencia sea físicamente realizable al menos debe ser propia; cuando es estrictamente propia la matriz D = 0.

Para pasar de G(Z) a A, B, C, y D, se debe hacer una realización controlable u observable, es decir una realización mínima.

En la expresión (2.15), los polos del sistema se obtiene del cálculo det(ZI-A) = 0; y una realización mínima si se elimina los factores repetidos del numerador y del denominador.

2.2.1. ANALOGIA PARAMETRICA ENTRE LOS DIFERENTES MODELOS

Como el propósito de la identificación es el determinar un modelo sobre el cual se va a aplicar técnicas de control, se hace necesario una analogía entre las diferentes formas de representar un modelo por ejemplo: si se tiene el modelo representado mediante ecuación de diferencias, determinar su equivalente por medio de función de transferencia, variables de estado, etc.

Para un sistema de tercer orden, se tiene el siguiente desarrollo:

$$y(k+3) + a1y(k+2) + a2y(k+1) + a3y(k) = b0u(k+3) + b1u(k+2) + b2u(k+1) + b3u(k) \qquad (2.17)$$

Sacando la transformada Z de la ecuación de diferencias, con condiciones iniciales iguales a cero:

$$Z^3.Y(Z) + a1.Z^2.Y(Z) + a2.Z.Y(Z) + a3.Y(Z) = b0.Z^3.U(Z) + b1.Z^2.U(Z) + b2.Z.U(Z) + b3.U(Z)$$

Dividiendo para Z^3 :

$$Y(Z) + a1.Z^{-1}.Y(Z) + a2.Z^{-2}.Y(Z) + a3.Z^{-3}.Y(Z) = b0.U(Z) + b1.Z^{-1}.U(Z) + b2.Z^{-2}.U(Z) + b3.Z^{-3}U(Z) \qquad (2.18)$$

Donde $Z^{-1}y(z)$, constituye un retardo unitario igual a la transformada Z de y(k-1).

Si se factora a la expresión (2.18) respecto a Y(Z) y U(Z), la función de transferencia es:

$$\frac{Y(Z)}{U(Z)} = \frac{b0+b1.Z^{-1}+b2.Z^{-2}+b3.Z^{-3}}{1+a1.Z^{-1}+a2.Z^{-2}+a3.Z^{-3}} \qquad (2.19)$$

La matriz de los parámetros para este caso es:

$$\theta = (a1\ a2\ a3\ b0\ b1\ b2\ b3)^T \qquad (2.20)$$

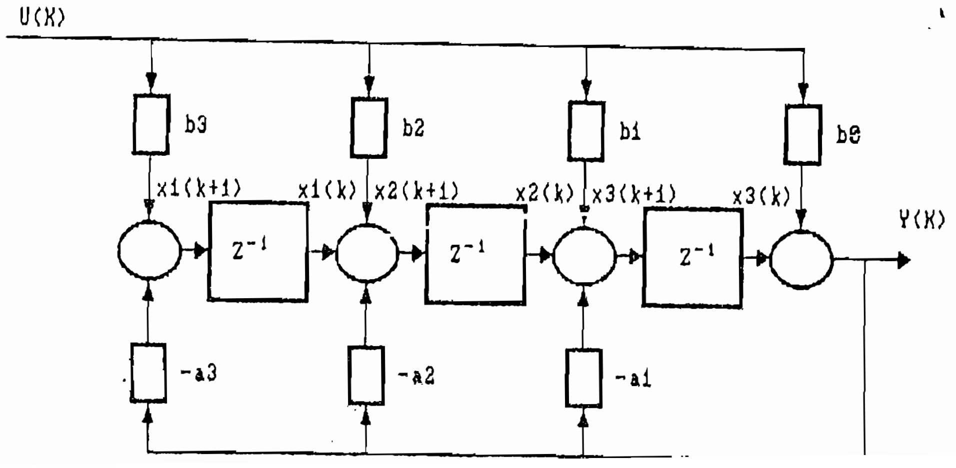
Si θ^o = matriz de parámetros verdaderos de la función de transferencia.

El problema de la identificación, es obtener la matriz θ de parámetros estimados como una buena aproximación de la matriz θ^o, a partir de las señales de entrada u(k) y de la señal de salida y(k).

Para obtener la representación en variables de estado, se parte de la ecuación (2.18), en la misma que se agrupa los términos de acuerdo a los retardos, y se llega a la expresión:

$$Y(Z) = Z^{-1}.(b1.U(Z) - a1.Y(Z)) + Z^{-2}.(b2.U(Z) - a2.Y(Z)) + Z^{-3}.(b3.U(Z) - a3.Y(Z)) + b0.U(Z)$$

En base a esta expresión se elaborá un diagrama de bloques representado por la fig (2.5).



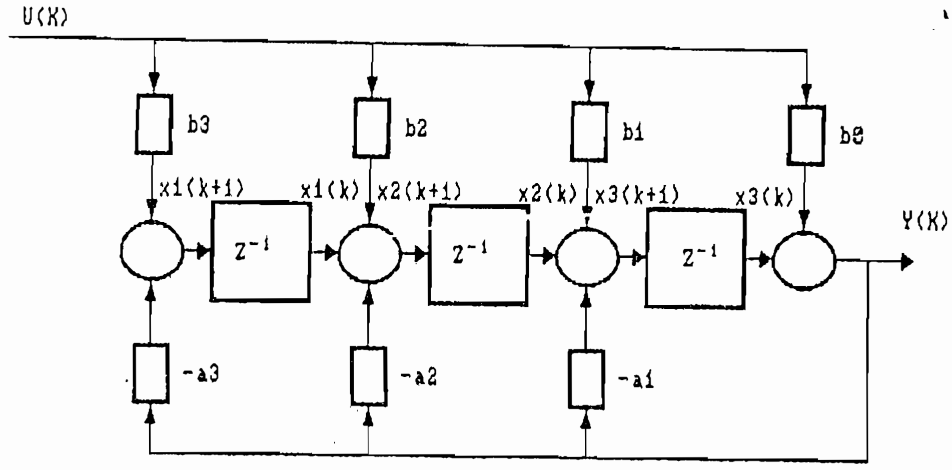


Fig 2.5 Diagrama de bloques de la ecuación (2.18) representada en variables de estado

De acuerdo a la Fig 2.5, las ecuaciones de estado son :

(2.21)

$$x1(k+1) = b3.u(k) - a3.y(k)$$
$$x2(k+1) = b2.u(k) - a2.y(k) + x1(k)$$
$$x3(k+1) = b1.u(k) - a1.y(k) + x2(k)$$
$$y(k) = b0u(k) + x3(k)$$

Reemplazando y(k) en cada una de las ecuaciones de estado y agrupando términos semejantes, se tiene:

$$x1(k+1) = b3.u(k) - a3.(b0u(k) + x3(k))$$
$$x1(k+1) = - a3.x3(k) + (b3 - a3.b0).u(k)$$
$$x2(k+1) = b2.u(k) - a2.y(k) + x1(k)$$
$$x2(k+1) = x1(k) - a2.x3(k) + (b2 - a2.b0).u(k)$$
$$x3(k+1) = b1.u(k) - a1.y(k) + x2(k)$$

$$x3(k+1) = x2(k) - a1.x3(k) + (b1 - a1.b0).u(k)$$

Las ecuaciones de estado anteriores, escritas en forma matricial:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 0 & -a3 \\ 1 & 0 & -a2 \\ 0 & 1 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} b3-a3.b0 \\ b2-a2.b0 \\ b1-a1.b0 \end{vmatrix} . u(k) \qquad (2.22)$$

$$y(k) = |0 \quad 0 \quad 1| . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + b0.u(k) \qquad (2.23)$$

Esta representación constituye la forma canónica observable; la misma que indica si es posible o no estimar los estados.

> " Se dice que un sistema es observable en el tiempo to, si, con el sistema en el estado x(to), es posible determinar este estado partiendo de la observación de la salida durante un intervalo de tiempo finito."

Aplicando el concepto observador, del dual controlador se obtiene la forma canónica controlable; la misma que indica si la señal de entrada afecta o cambia de alguna manera el valor de los estados o el valor de las salidas.

" Se dice que un sistema es controlable en el tiempo to, si es posible transferir un sistema por medio de un vector de control no restringido, desde cualquier estado inicial x(to) a cualquier otro estado en un intervalo de tiempo finito."

Para llegar al gráfico de la forma canónica controlable, a partir del gráfico de la forma canónica observable (fig 2.5), se intercambia lo siguiente:

- Las direcciones de las flechas.
- Los sumatorios por puntos.
- Las entradas por salidas.

Lo que permite llegar a la fig (2.6):

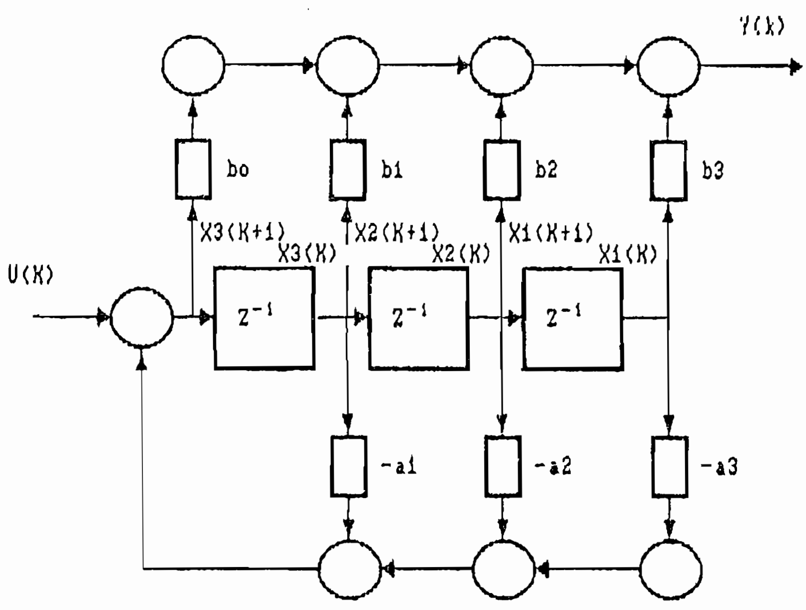


Fig 2.6 Diagrama de bloques de la forma canónica controlable

Las ecuaciones de estado son:

$$\begin{aligned} x1(k+1) &= x2(k) \\ x2(k+1) &= x3(k) \\ x3(k+1) &= -a3.x1(k) - a2.x2(k) - a1.x3(k) + u(k) \\ y(k) &= b3.x1(k) + b2.x2(k) + b1.x3(k) + b0(- \\ &\quad a3.x1(k) - a2.x2(k) - a1.x3(k) + u(k)) \end{aligned} \tag{2.24}$$

Agrupando términos semejantes:

$$\begin{aligned} y(k) &= (b3 - b0.a3)x1(k) + (b2 - b0.a2)x2(k) + \\ &\quad (b1 - b0.a1)x3(k) + b0.u(k)) \end{aligned}$$

Escribiendo las ecuaciones en forma matricial:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a3 & -a2 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} . u(k) \tag{2.25}$$

$$y(k) = \begin{vmatrix} b3-b0.a3 & b2-b0.a2 & b1-b0.a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + b0.u(k) \tag{2.26}$$

En la mayor parte de los casos el parámetro b0 es igual a cero, por los retardos inherentes a los sistemas reales, reduciéndose las ecuaciones (2.25) y (2.26) a lo siguiente:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a3 & -a2 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} . u(k) \tag{2.27}$$

$$y(k) = |b3 \;\; b2 \;\; b1| . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} \tag{2.28}$$

Donde D = 0

Un modelo de tercer orden representado por función de transferencia esta dado por la siguiente expresión:

$$H(Z,\theta) = \frac{b1 . Z^{-1} + b2 . Z^{-2} + b3 . Z^{-3}}{1 + a1 . Z^{-1} + a2 . Z^{-2} + a3 . Z^{-3}} \tag{2.29}$$

Todas estas representaciones son equivalentes desde el punto de vista que generan la misma señal de salida a partir de la misma señal de entrada.

2.2.1.2. Cuando la excitación es b.u(k)

La ecuación de diferencias es:

$$y(k+n) + a1y(k+n-1) + a2y(k+n-2) + \ldots\ldots + any(k) = b.u(k)$$

Para el caso de un sistema de tercer orden se tiene:

$$y(k+3) + a1y(k+2) + a2y(k+1) + a3y(k) = b.u(k)$$

Sacando la transformada Z de la ecuaciones de diferencias, con condiciones iniciales iguales a cero se tiene:

$$Z^3.Y(Z) + a1.Z^2.Y(Z) + a2.Z.Y(Z) + a3.Y(Z) = b.U(Z)$$

dividiendo para Z^3

$$Y(Z) + a1.Z^{-1}.Y(Z) + a2.Z^{-2}.Y(Z) + a3.Z^{-3}.Y(Z) = b.Z^{-3}.U(Z)$$

la función de transferencia es:

$$\frac{Y(Z)}{U(Z)} = \frac{b.Z^{-3}}{1+a1.Z^{-1}+a2.Z^{-2}+a3.Z^{-3}} \qquad (2.30)$$

Si se realiza un análisis igual al caso 2.2.1.1., las ecuaciones de estado para el caso observable son:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 0 & -a3 \\ 1 & 0 & -a2 \\ 0 & 1 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} b \\ 0 \\ 0 \end{vmatrix} . u(k) \qquad (2.31\ a)$$

$$y(k) = \begin{vmatrix} 0 & 0 & 1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} \qquad (2.31\ b)$$

para el caso controlable son:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a3 & -a2 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} . u(k) \qquad (2.32\ a)$$

$$y(k) = \begin{vmatrix} b & 0 & 0 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} \qquad (2.32\ b)$$

Del análisis anterior se concluye los siguientes aspectos:

- Las ecuaciones de estado controlable u observable pueden ser determinados en forma directa por la ecuación de diferencias o viceversa.

- Existe una relación directa entre los elementos de las matrices A, B, C de las ecuaciones de estado, los coeficientes de la ecuaciones de diferencias, y los coeficientes de la función de transferencia. Entonces, al disponer una de las representaciones, se puede obtener directamente cualesquiera de las otras representaciones, simplemente reemplazando los parámetros, de acuerdo a las expresiones determinadas.

Al adoptar un modelo matemático, se tiene un mayor o un menor número de parámetros que identifican al sistema; en un sistema de tercer orden representado por variables de estado, se tiene quince (15) parámetros, para el mismo sistema en el caso de ecuaciones de diferencias y función de transferencia se tiene solamente seis (6) parámetros. Por lo tanto, para un sistema de tercer orden, es suficiente para la determinación del modelo en forma completa identificar seis (6) parámetros, siendo los nueve (9) parámetros restantes redundantes.

En un sistema de segundo orden, será suficiente identificar 4 parámetros en lugar de los 8 existentes.

En esta Tesis se procederá a identificar sistemas discretos representados mediante variables de estado a partir de una representación canónica controlable, por cuanto tienen una representación mínima de parámetros.

Si se realiza una transformación de semejanza en la que **X** = T.Z, donde **T es la matriz nodal**[2] (cuadrada y no singular), y Z representa un nuevo conjunto de variables de estado se tiene:

si $x = T.Z$;

reemplazando en,

$x = A.x + B.U$; entonces:

$T.Z = A(T.Z) + B.U$; despejando Z,

$Z = T^{-1}.A.T.Z + T^{-1}.B.U$

$Y = C(T.Z) + D.U$

Reemplazando:

$$A = T^{-1}.A.T \;;\quad B = T^{-1}.B \;;\quad C = C.T \;;\quad D = D$$

$$Z = A.Z + B.U$$

$$Y = C.Z + D.U$$

[2] Matriz nodal, tiene por columna los valores propios de la matriz A.

Expresiones que constituyen la forma canónica de Jordan; en donde los elementos de la diagonal son los polos del sistema. Por cuestiones de explicación se considera de distinto valor, en consecuencia, los estados están desacoplados porque dependen solamente de si mismos y de la señal de entrada.

2.2.2 MODELO ARMA

Dentro del grupo de las formas canónicas, existe un modelo muy apropiado para identificación llamado **ARMA (Auto Regresive Moving Average)**, este modelo está constituido por los valores anteriores de la señal de entrada y de la señal de salida.

La ecuación de diferencias tiene la forma $y(k) = a1.y(k-1) + ... + an.y(k-n) + b0.u(k) + ... + bm.u(k-m)$, expresión que se diferencia de la ecuación (2.16), por depender de los n valores anteriores de la señal de salida y de los m valores anteriores de la señal de entrada. En forma similar al análisis realizado en el numeral 2.2.1.1, la matriz función de transferencia discreta, está dado por la expresión $G(Z) = (b0 + ... + bm.Z^{-m})/(1 - a1.Z^{-1} - .... - an.Z^{-n})$.

Para un sistema de tercer orden la función de transferencia es:

$G(Z) = (b0 + b1.Z^{-1} + b2.Z^{-2} + b3.Z^{-3})/(1 - a1.Z^{-1} - a2.Z^{-2} - a3.Z^{-3})$.

Se va a desarrollar la forma del modelo ARMA para el caso de variables de estado, así como se establecerá sus características. Resolviendo la función de transferencia, se tiene:

$Y(Z) - a1.Y(Z).Z^{-1} - a2.Y(Z).Z^{-2} - a3.Y(Z).Z^{-3} = b0.X(Z) + b1.X(Z).Z^{-1} + b2.X(Z).Z^{-2} + b3.X(Z).Z^{-3}$

Donde $Z^{-1}Y(Z)$ representa un retardo unitario de la señal de salida igual a la transformada Z de y(k-1). Realizando la transformada Z inversa, entonces:

$y(k) - a1y(k-1) - a2y(k-2) - a3y(k-3) = b1u(k-1) + b2u(k-2) + b3u(k-3)$

Expresión que constituye la ecuación de diferencias de tercer orden en la cual, si se despeja la salida en el instante K, se tiene:

$$y(k) = + a1y(k-1) + a2y(k-2) + a3y(k-3) + b1u(k-1) + b2u(k-2) + b3u(k-3) \quad (2.33)$$

En esta expresión, el valor actual de salida se obtiene a partir de los valores anteriores de la señal de entrada y de la señal de salida, característica que permite desarrollar técnicas de control en tiempo real.

El sistema de tercer orden se representa por el siguiente diagrama de bloques:

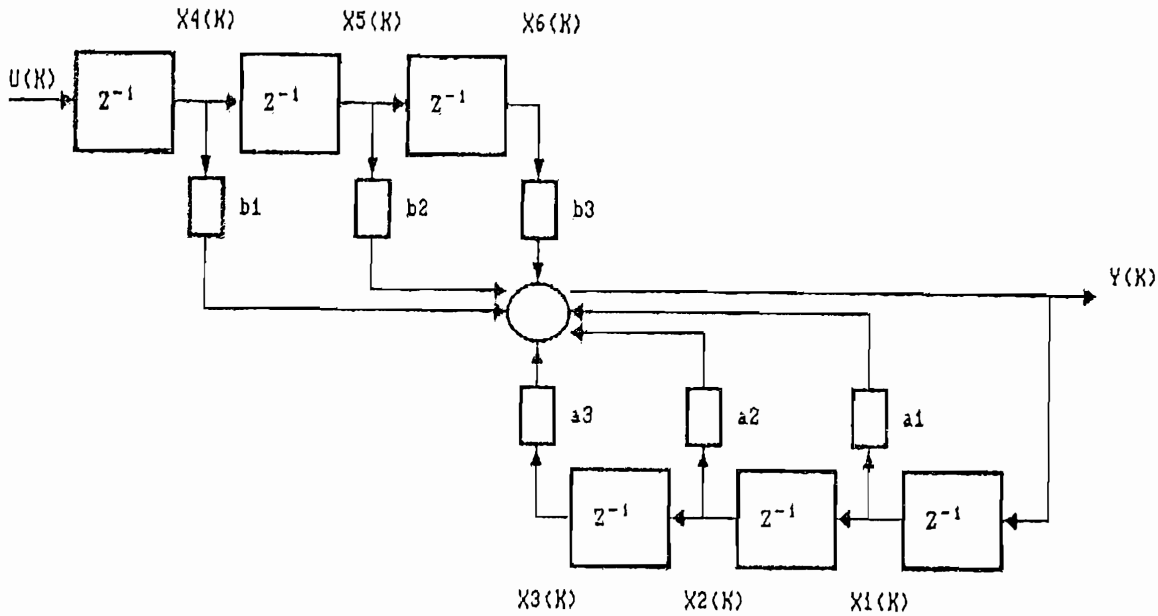


Fig 2.7 Diagrama de bloques del modelo ARMA de tercer orden

Las ecuaciones de estado del modelo ARMA, pueden ser fácilmente desarrolladas a partir de la figura 2.7; en donde se indica de que forma se deben expresar las variables de estado.

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \\ x4(k+1) \\ x5(k+1) \\ x6(k+1) \end{vmatrix} = \begin{vmatrix} a1 & a2 & a3 & b1 & b2 & b3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} \cdot \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \\ x4(k) \\ x5(k) \\ x6(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{vmatrix} . u(k)$$

(2.34 a)

$$y(k) = \begin{vmatrix} a1 & a2 & a3 & b1 & b2 & b3 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \\ x4(k) \\ x5(k) \\ x6(k) \end{vmatrix} \qquad (2.34\ b)$$

Las ecuaciones de estado del modelo ARMA, tienen los mismos seis parámetros por identificar, que los indicados en el numeral 2.2.1 y dispone de seis estados en las matrices.

Por lo tanto, para un sistema de tercer orden se trata de una representación no mínima. En efecto la matriz A, introduce tres polos en Z=0 y se trata de una representación no observable.

Los polos están dados por el polinomio característico:

$$y(k) = \det\begin{vmatrix} a1-\lambda & a2 & a3 \\ 1 & -\lambda & 0 \\ 0 & 1 & -\lambda \end{vmatrix} . \det\begin{vmatrix} -\lambda & 0 & 0 \\ 1 & -\lambda & 0 \\ 0 & 1 & -\lambda \end{vmatrix}$$

En consecuencia esta forma de representar a un modelo, (para el análisis de tercer orden), no sería la más apropiada, por considerarse una representación no mínima, pero tiene una propiedad muy importante para identificación, que se deriva de la comparación de la expresión (2.33) con la (2.34 b):

" **El vector de estado, está representado por los valores anteriores de la señal de salida y de la señal de entrada.**"

$x(k) = [y(k-1)\ y(k-2)\ y(k-3)\ u(k-1)\ u(k-2)\ u(k-3)]^T$

$$\begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \\ x4(k) \\ x5(k) \\ x6(k) \end{vmatrix} = \begin{vmatrix} y(k-1) \\ y(k-2) \\ y(k-3) \\ u(k-1) \\ u(k-2) \\ u(k-3) \end{vmatrix} \qquad (2.35)$$

La ecuación de salida en variables de estado, es la misma que la ecuación de diferencias.

A partir del modelo ARMA, se pueden obtener los modelos de media móvil MA (Moving Average), y autoregresivos AR (Auto regresive).

Para los modelos **MA** se toman los términos que contienen los parámetros ai igual a cero, teniéndose:

$$y(k) = b1u(k-1) + b2u(k-2) + b3u(k-3) \tag{2.36}$$

En la ecuación (2.36), la señal de salida está dado por los valores pasados de la señal de entrada.

Si el modelo es función de los valores pasados de la señal de salida para un número finito de períodos se tiene el caso de modelo auto-regresivo AR (Auto Regresive), dado por la expresión:

$$y(k) = a1y(k-1) + a2y(k-2) + a3y(k-3) \tag{2.37}$$

En la ecuación (2.37), la señal de salida es resultado de sus valores anteriores; pudiendo añadirse un término que es el ruido.

Si al modelo ARMA se añade una variable externa como es el ruido (ruido blanco), se obtiene el modelo ARMAX el mismo que, expresado en ecuación de diferencias tiene la siguiente forma:

$$y(k) - a1y(k-1) - a2y(k-2) - a3y(k-3) = b1u(k-1) + b2u(k-2) + b3u(k-3) + C.e(k) \quad (2.38)$$

En donde C, representa un conjunto de coeficientes diferentes de cero, y el término e(k), representa variables aleatorias de media cero y no correlacionados entre si.

Al igual que en procesos determinísticos, los cuales tienen su respectiva representación en variables de estado, para el caso de procesos estocásticos (probabilísticos), también existe la respectiva representación en variables de estado que son:

$$X(K+1) = A.X(K) + B.U(K) + V(K) \quad (2.39)$$

$$Y(K) = C.X(K) + e(K) \quad (2.40)$$

El término V incluye el error de modelamiento, donde los Vi tienen distribución gaussiana, con media 0 y son no correlacionados entre si. En la ecuación de salida aparece un término de error e(k), estos errores resultan del ruido externo o interno del proceso. La representación gráfica en diagrama de bloques es:

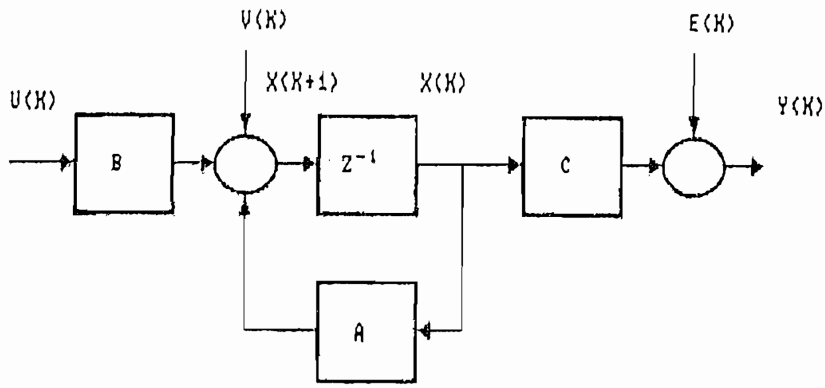


Fig. 2.8 Diagrama de bloques del modelo ARMAX

El estado es función de variables aleatorias, en consecuencia las variables de estado se tratan también de variables aleatorias. Para el caso estocástico, a más de la identifición de los parámetros, se debe realizar la estimación del estado de un sistema con un mínimo de error, análisis que le corresponde a la moderna teoría de la estimación lineal óptima.

2.3 EL PROBLEMA DE LA IDENTIFICACION.

Una vez que se selecciona el modelo que describe a la planta y se identifica sus parámetros, se trata de conocer la bondad de ajuste de los parámetros estimados $\hat{\theta}$ respecto de los parámetros verdaderos de la planta θ^{o}.

Dado que el vector θ^o, esta constituido por los parámetros reales de la planta, los mismos que son desconocidos, se determina el error en la identificación a partir de los datos de entrada U(k) y de salida Y(k), los cuales son los únicos que se pueden accesar en forma real. Por lo tanto, se aborda tres técnicas que permitan medir el error de la estimación, los mismos que son:

1.- Ecuación de error.

2.- Error de salida.

3.- Error de predicción de salida.

2.3.1 ECUACION DE ERROR

Para la técnica de la ecuación de error, se debe conocer las ecuaciones completas de la dinámica de la planta. Se toma el modelo en variables de estado para el análisis.

Un modelo en variables de estado para el caso continuo, está representado por la ecuación $\dot{X} = A.X + B.U$ en donde, la diferencial del estado es función del estado y de la excitación del sistema. Las matrices **A,B** están constituidos por los parámetros θ desconocidos de la planta ($\dot{X} = f\ (X,U,\theta)$).

Se asume lo siguiente:

- La forma del modelo, pero no sus parámetros θ, que describen la planta.
- La excitación, el estado y la derivada del estado.

En consecuencia, la ecuación de error de los valores actuales (medidos) $\dot{X}a$, Xa, Ua respecto a ciertos valores de $\hat{\theta}$ (estimados) es:

$$\dot{X}a - f(Xa, Ua, \theta) = e(t;\theta) \tag{2.41}$$

El error $e(t;\theta^o) = 0$, para los parámetros reales de la planta.

Se define el indice de funcionamiento como sigue:

$$J(\theta) = \int_0^T e^T(t;\theta) \, . \, e(t;\theta) \, . \, dt \tag{2.42}$$

Si $J(\hat{\theta})=0$, los valores estimados son iguales a los valores verdaderos ($\hat{\theta}=\theta^o$), es decir la ecuación de error es $e(t;\theta)=0$. En forma adicional el índice de funcionamiento de los valores estimados debe cumplir con la condición de que, $J(\hat{\theta}) \leq J(\theta)$ para cualquier otro conjunto de parámetros.

t;θ)=0. En forma adicional el índice de funcionamiento de los valores estimados debe cumplir con la condición de que, J(θ) <= J(θ) para cualquier otro conjunto de parámetros.

La técnica de la ecuación de error, para sistemas continuos no es apropiado, debido a que, las consideraciones iniciales asumidas como conocidas son muy fuertes así como irreales desde el punto de vista de instrumentación. Para sistemas discretos resulta apropiado, si se toma como base de análisis el modelo ARMA, por cuanto, el vector de estado esta formado por valores anteriores de la señal de entrada y de salida.

La ecuación de error para sistemas discretos representados mediante variables de estado es:

$$Xa(k+1) - A.Xa(k) - B.Ua(k) = e(k;\theta) \qquad (2.43)$$

Las matrices A y B contienen los parámetros desconocidos, para el modelo ARMA de tercer orden se tiene:

$$\begin{vmatrix} e1(k,\theta) \\ e2(k,\theta) \\ e3(k,\theta) \\ e4(k,\theta) \\ e5(k,\theta) \\ e6(k,\theta) \end{vmatrix} = \begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \\ x4(k+1) \\ x5(k+1) \\ x6(k+1) \end{vmatrix} - \begin{vmatrix} a1 & a2 & a3 & b1 & b2 & b3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \\ x4(k) \\ x5(k) \\ x6(k) \end{vmatrix} - \begin{vmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{vmatrix} . u(k)$$

(2.44)

Sustituyendo el vector de estado por (2.44) se tiene:

$$\begin{vmatrix} e1(k,\theta) \\ e2(k,\theta) \\ e3(k,\theta) \\ e4(k,\theta) \\ e5(k,\theta) \\ e6(k,\theta) \end{vmatrix} = \begin{vmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ u(k) \\ u(k-1) \\ u(k-2) \end{vmatrix} - \begin{vmatrix} a1 & a2 & a3 & b1 & b2 & b3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} . \begin{vmatrix} y(k-1) \\ y(k-2) \\ y(k-3) \\ u(k-1) \\ u(k-2) \\ u(k-3) \end{vmatrix} - \begin{vmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{vmatrix} . u(k)$$

(2.45)

desarrollando entonces:

$$e1(k;\theta) = y(k)-a1y(k-1)-a2y(k-2)-a3y(k-3)-b1u(k-1)-b2u(k-2)-b3u(k-3)$$

$$e2(k;\theta) = y(k-1)-y(k-1)$$

$$e3(k;\theta) = y(k-2)-y(k-2)$$

$$e4(k;\theta) = u(k)-u(k)$$

$$e5(k;\theta) = u(k-1)-u(k-1)$$

$$e6(k;\theta) = u(k-2)-u(k-2)$$

Los elementos de la ecuación de error son todos cero, excepto e1, que esta dado por la expresión:

$$e1(k;\theta) = x1(k+1)-a1x1(k)-a2x2(k)-a3x3(k)-b1x4(k)-b2x5(k)-b3x6(k) \qquad (2.46\ a)$$

ó

$$e1(k;\theta) = ya(k)-a1ya(k-1)-a2ya(k-2)-a3ya(k-3)-b1Ua(k-1)-b2Ua(k-2)-b3Ua(k-3) \qquad (2.46\ b)$$

Este error se mide a través de los valores anteriores de las señales de entrada y de salida, el mismo que es diferente a cero por cuanto los parámetros verdaderos para un valor actual es $\theta^{o} = [a^{o}1\ a^{o}2\ a^{o}3\ b^{o}1\ b^{o}2\ b^{o}3]^{T}$

El índice de funcionamiento para sistemas discretos está dado por la expresión:

$$J(\theta) = \sum_{k=0}^{N} e^{2}(k;\theta) \qquad (2.47)$$

Para el modelo ARMA:

$$J(\theta) = \sum_{k=1}^{N} e1^{2}(k;\theta)$$

La técnica de la ecuación de error tiene la desventaja de hacer fuertes suposiciones (estado, derivada del estado) para el caso continuo, y (valores anteriores de las señales de entrada y de salida) para el caso discreto, problemas que

La técnica de la ecuación de error tiene la desventaja de hacer fuertes suposiciones (estado, derivada del estado) para el caso continuo, y (valores anteriores de las señales de entrada y de salida) para el caso discreto, problemas que se obvian con el criterio de error de salida.

2.3.2 ERROR DE SALIDA

En esta técnica, se compara la salida actual de la planta, que es función de los parámetros verdaderos $\theta^o = [a^o1\ a^o2\ a^o3\ b^o1\ b^o2\ b^o3]^T$, con la salida actual del modelo que es función de los parámetros $\theta = [a1\ a2\ a3\ b1\ b2\ b3]^T$ figura 2.9

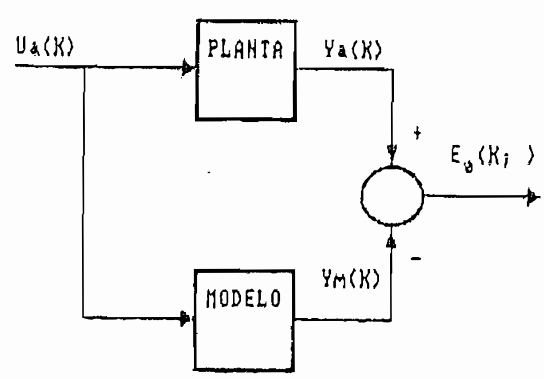


Fig 2.9 Diagrama de bloques del error de salida.

Los parámetros del modelo deben dar un índice de funcionamiento lo más pequeño posible.

El error de salida $e(k;\theta) = Ya(k) - Ym(k)$, en donde la salida del modelo $Ym(k) = a1ym(k-1)+a2ym(k-2)+a3ym(k-3)+b1Ua(k-1)+b2Ua(k-2)+b3Ua(k-3)$, reemplazando en la ecuación del error de salida, es:

$$e(k;\theta)= Ya(k)-a1ym(k-1)-a2ym(k-2)-a3ym(k-3)-b1Ua(k-1)-b2Ua(k-2)-b3Ua(k-3) \tag{2.48}$$

Comparando la ecuación (2.48) con la ecuación (2.46 b), se concluye que, la ecuación de error utiliza valores pasados de la planta, mientras que en el criterio del error de salida utiliza valores pasados del modelo.

2.3.3 ERROR DE PREDICCION DE SALIDA

La técnica del error de predicción de salida, a diferencia del criterio de error de salida, trabaja con la salida del modelo predictor, modelo mucho más complejo que el utilizado para la técnica del error de salida. En este caso la salida del predictor esta constituido por la señal de entrada comparada con la salida verdadera de la planta, lo que produce el mismo efecto que el de una realimentación (figura 2.10).

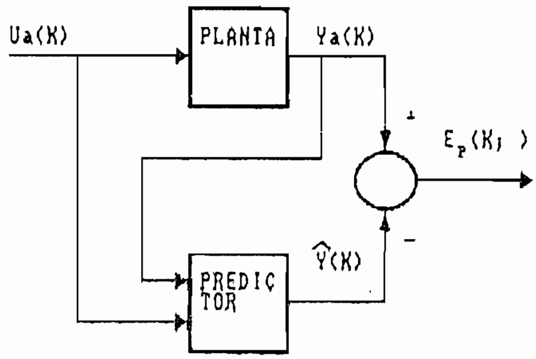


Fig 2.10 Diagrama de bloques del error de predicción de salida

El error de predicción ep(k;θ) = Ya(k) - $\hat{Y}$(k), en donde la salida del predictor $\hat{Y}$(k) depende del vector de parámetros θ, y es función de la entrada y salida actual de la planta, y una secuencia de variables aleatorias del sistema.

Conocido el error se puede definir el índice de funcionamiento como sigue:

$$J(\theta) = \sum_{k=1}^{N} e^2(k;\theta)$$

El escoger un criterio de error está supeditado a la dificultad de la implementación computacional para obtener el estimado $\hat{\theta}$ y de las propiedades del estimado.

CAPITULO III

METODOS Y ALGORITMOS DE IDENTIFICACION PARA EL MODELO EN VARIABLES DE ESTADO

Los algoritmos de identificación de parámetros que se van desarrollar, toman como base para el análisis al modelo ARMA y la técnica de la ecuación de error, asuntos tratados en el capítulo segundo.

La utilización del modelo ARMA representado mediante variables de estado en identificación permite tener las siguientes ventajas:

- El vector de estado está constituido por los valores anteriores de la señal de entrada y de la señal de salida.
- La ecuación de salida en variables de estado es similar a la ecuación de diferencias.

3.1 MINIMOS CUADRADOS

El algoritmo de mínimos cuadrados a desarrollarse se denomina ordinario, para diferenciarle del algoritmo de mínimos cuadrados recursivos que será desarrollado posteriormente.

3.1.1 MINIMOS CUADRADOS ORDINARIOS

El análisis de mínimos cuadrados ordinarios parte de la ecuación de salida del modelo ARMA dada por la expresión (2.34 b); que para sistemas de orden n, la ecuación de salida es:

$$y(k) = [a1\ a2\ ....\ an\ b1\ b2\ ....\ bn].[y(k-1)\ y(k-2)\ ....\ y(k-n)\ u(k-1)\ u(k-2)\ ....\ u(k-n)]^T \quad (3.1)$$

En consecuencia la ecuación de error es:

$$e(k;\theta) = y(k) - [a1\ a2\ ....\ an\ b1\ b2\ ....\ bn]\ [y(k-1))\ y(k-2)\ ....\ y(k-n)\ u(k-1)\ u(k-2)\ ....\ u(k-n)]^T \quad (3.2)$$

El análisis parte del conocimiento de los valores de las señales de entrada y de salida {u(0), u(1), u(2), ...-..u(N), y(0), y(1), ...y(N)} y, θ = [a1 a2......an b1 b2......bn]^T son los parámetros de el modelo por identificar.

Para los instantes de muestreo $k = n, n+1, \ldots\ldots, N$, se define el **vector de error de salida**, el primer elemento de este vector esta dado por la expresión (3.2).

Para K=n

$$e(n;\theta) = y(n) - [a1 \ldots\ldots an\ b1 \ldots\ldots bn]\ [y(n-1)) \ldots\ldots y(n-n)\ u(n-1) \ldots\ldots u(n-n)]^T$$

simplificando la notación matricial:

$$e(n;\theta) = y(n) - \psi^T(n).\theta$$

Donde $\psi(n) = [y(n-1)\ y(n-2)\ldots\ldots y(0)\ \ u(n-1)\ u(n-2)\ldots\ldots u(0)]^T$

Para K=n+1

$$e(n+1;\theta) = y(n+1) - [a1\ a2 \ldots\ldots an\ b1\ b2 \ldots\ldots bn]\ .\ [y(n))\ y(n-1) \ldots y(1)\ u(n)\ u(n-1) \ldots u(1)]^T$$

en forma simplificada:

$$e(n+1;\theta) = y(n+1) - \psi^T(n+1).\theta$$

. . . . . .

Para K=N

$$e(N;\theta) = y(N) - [a1 \;\; a2 \; .... \; an \; b1 \; b2 \; .... \; bn].[y(N-1) \; y(N-2) \; ... \; y(N-n) \; u(N-1) \; u(N-2) \; ... \; u(N-n)]^T$$

en forma simplificada:

$$e(N;\theta) = y(N) - \psi^T(N).\theta$$

Despejando el término de la señal de salida se tiene:

$$\begin{aligned} y(n) &= \psi^T(n).\theta + e(n;\theta) \qquad (3.3)\\ y(n+1) &= \psi^T(n+1).\theta + e(n+1;\theta)\\ &.......\\ y(N) &= \psi^T(N).\theta + e(N;\theta) \end{aligned}$$

Donde $\psi(k) = [y(k-1) \; y(k-2)....y(k-n) \; u(k-1) \; u(k-2)...u(k-n)]^T$

El conjunto de ecuaciones (3.3), expresadas con notación matricial viene dada por la siguiente expresión:

$$Y(N) = \Psi(N).\theta + \epsilon(N;\theta) \quad \text{Ecuación de error} \qquad (3.4)$$

En donde:

$$Y(N) = [y(n) \; y(n+1) \; y(n+2) \; .... \; y(N)]^T$$

Es una matriz columna de N-n+1 filas, que representa el vector de la señal de salida.

$$\Psi(N) = [\psi(n)\ \psi(n+1)\ \psi(n+2)\ \ldots\ldots\ldots\ldots\ \psi(N)]^T$$

Es una matriz de 2n columnas y N-n+1 filas, que está constituida por los valores anteriores de las señales de entrada y de salida.

$$\epsilon(N;\theta) = [e(n;\theta)\ e(n+1;\theta)\ e(n+2;\theta)\ \ldots\ldots\ e(N;\theta)]^T$$

Es una matriz columna de N-n+1 filas, que representa el error de modelación.

$$\theta = [a1\ a2 \ldots\ldots\ldots an\ b1\ b2 \ldots\ldots\ldots\ bn]^T$$

Es una matriz columna de 2n filas, que esta constituida por los parámetros a identificar.

La identificación paramétrica mediante el algoritmo mínimos cuadrados ordinarios, busca obtener valores de θ tales que, la suma de los cuadrados del error e(k) sea lo más pequeño posible respecto de los parámetros verdaderos de la planta θ^o. El índice de funcionamiento es:

$$J(\theta) = \sum_{k=n}^{N} e^2(k;\theta) \tag{3.5a}$$

en forma matricial:

$$J(\theta) = \epsilon^T(N;\theta)\epsilon(N;\theta) \tag{3.5b}$$

La matriz $J(\theta)$ es una función cuadrática de 2n parámetros, notada para el caso del algoritmo de mínimos cuadrados como $\widehat{\theta}_{LS}$, en donde $J(\widehat{\theta}_{LS}) \leq J(\theta)$; es decir, el índice de funcionamiento para los valores estimados por mínimos cuadrados, deben ser menor o máximo igual al índice de funcionamiento para cualquier otro conjunto de valores de parámetros.

Para minimizar la función de costo, se obtine la derivada parcial respecto θ y se iguala esta expresión a cero.

$$\frac{\partial J(\theta)}{\partial \theta} = 0$$

Resolviendo:

$$\begin{aligned} J(\theta) &= (Y - \Psi.\theta)^T.(Y - \Psi.\theta) \\ &= Y^T.Y - Y^T.\Psi.\theta - \theta^T.\Psi^T.Y + \theta^T.\Psi^T.\Psi.\theta \end{aligned}$$

La derivada parcial de matrices respecto a θ, utiliza la propiedad dada por la siguiente expresión

$$\frac{\partial \mathbf{a}^T.Q.\mathbf{a}}{\partial \mathbf{a}} = 2\mathbf{a}^T.Q$$

teniéndose lo siguiente:

$$\frac{\partial J(\theta)}{\partial \theta} =$$

$$= -2.Y^T.\Psi + 2.\theta^T.\Psi^T.\Psi$$

Reemplazando θ por $\hat{\theta}_{LS}$ y simplificando la expresión:

$$\Psi^T.\Psi.\hat{\theta}_{LS} = \Psi^T.Y \qquad (3.6)$$

A esta expresión se la conoce como **Ecuación normal** y da los valores $\hat{\theta}_{LS}$ estimados. Para que tenga una única solución, se debe realizar las siguientes consideraciones:

1.- Número de parámetros θ mínima.

Por ejemplo, un sistema de tercer orden representado mediante variables de estado, tiene quince parámetros para tener una descripción completa del sistema. El mismo sistema expresado por alguna de las formas canónicas presenta únicamente seis parámetros.

Si se mantiene la descripción de quince parámetros la ecuación normal podría no tener una única solución, en consecuencia es necesario para obtener una única solución, una representación con el menor número de parámetros.

2.- Señal de entrada persistentemente excitable.

Por ejemplo, en un sistema de tercer orden, donde se aplica una señal de entrada u(k) igual a una constante c, la ecuación de salida es:

$y(3) = a1y(2) + a2y(1) + a3y(0) + b1.c + b2.c + b3.c$

.

.

$y(N) = a1y(N-1) + a2y(N-2) + a3y(N-3) + b1.c + b2.c + b3.c$

En donde los parámetros b1,b2,b3, siempre aparecen como la suma b1+b2+b3, (linealmente dependientes), lo que permite concluir que, con una señal de entrada constante no se logra excitar a toda la dinámica de la planta, es decir, la señal de entrada debe variar lo suficiente para evitar que los parámetros no sean linealmente dependientes.

Si se considera la señal de entrada persistentemente excitable, la matriz $\Psi^T.\Psi$ es una matriz no-singular, por lo tanto tiene inversa, teniéndose:

$$\hat{\theta}_{LS} = (\Psi^T.\Psi)^{-1}.\ \Psi^T.Y \tag{3.7}$$

En ciertos análisis, los datos obtenidos al final del experimento son más importantes que los del comienzo del mismo, haciéndose necesario de un criterio de peso de los errores o ponderación de los errores.

3.1.1.1 MINIMOS CUADRADOS PONDERADOS

La ecuación del índice de funcionamiento para el caso de ponderación de los errores esta dado por la siguiente expresión:

$$J(\theta) = \sum_{K=1}^{N} w(k)\, e^2(k;\theta)$$

$$= \epsilon^T(N;\theta).W.\epsilon(N;\theta) \tag{3.8}$$

En donde:

$w(k)$ = función de ponderación.

Si $\mathbf{W} = I$ se tiene el caso de mínimos cuadrados ordinarios.

Si $w(k) = (1-\gamma)\gamma^{N-k}$, se tiene el caso de mínimos cuadrados ponderados, en la cual:

- Para k cercano a N, se dará peso a valores recientes.
- Para k cercano a un n, se dará peso a valores pasados.
- El valor de gamma: $0 < \gamma < 1$

$$\hat{\theta}_{WLS} = (\Psi^T.W.\Psi)^{-1}.\ \Psi^T.W.Y \qquad (3.9)$$

La ecuación (3.9) corresponde a un filtro de primer orden, operando sobre el error cuadrático, donde el factor $(1 - \gamma)$ constituye la ganancia del filtro. Para γ cercanos a 1 el filtro tiene memoria y los efectos de ruido son disminuidos; para valores pequeños se dice que no tiene memoria.

3.1.2 MINIMOS CUADRADOS RECURSIVOS

En el análisis de mínimos cuadrados ordinarios, la estimación de los parámetros se fundamenta en la suposición de que se disponen de todos los elementos de las matrices **Y**, **Ψ** y **W**, para un conjunto de datos de longitud N, quedando la ecuación normal resuelta.

Si los datos se obtienen en forma secuencial, y no en forma de lotes (mínimos cuadrados ordinarios), se desea analizar la naturaleza de la identificación de parámetros con nuevos datos incluidos en el análisis, para este caso se utiliza el algoritmo de **mínimos cuadrados recursivos**.

El análisis parte del estudio de mínimos cuadrados ponderados, a cuyas matrices se añade un dato (N+1), por lo tanto, las expresiones matriciales $\Psi^T.W.\Psi$ y $\Psi^T.W.Y$ deben estar estructuradas de tal manera que, consideren la presencia de un dato adicional (N+1).

Se asumirá que la ponderación es, $w = a.\gamma^{N-K}$, en donde:

- Si $a = 1$ y $\gamma = 1$, se tiene el caso de mínimos cuadrados recursivos ordinarios.

- Si $a = 1 - \gamma$, se tiene el caso de mínimos cuadrados recursivos ponderados exponencialmente.

Las matrices son:

$$Y(N+1) = [y(n)\ y(n+1)\ y(n+2)\ \ldots\ldots\ldots\ldots\ y(N)\ y(N+1)]^T$$

Es una matriz columna de N-n+2 filas.

$\Psi(N+1) \quad = [\psi(n) \ \psi(n+1) \ \psi(n+2) \ \ldots\ldots\ldots\ldots \ \psi(N) \ \psi(N+1)]^T$

Es una matriz de 2n columnas y N-n+2 filas.

$\epsilon(N+1;\theta) = \ [e(n;\theta) \ e(n+1;\theta) \ \ e(n+2;\theta) \ \ldots\ldots \ e(N;\theta) \ \ e(N+1-;\theta)]^T$

Es una matriz columna de N-n+2 filas.

$\theta \qquad = [a1 \ a2 \ldots\ldots\ldots an \ b1 \ b2 \ldots\ldots\ldots \ bn]^T$

Es una matriz columna de 2n filas.

A la expresión $\Psi^T(N).W(N).\Psi(N)$ añadiéndole un dato se tiene:

$$\Psi^T(N+1)\, W(N+1)\, \Psi(N+1) = \sum_{k=n}^{N+1} \psi(k) \,.\, w(k) \,.\, \psi^T(k) = \sum_{k=n}^{N+1} \psi(k)\, a\gamma^{N+1-k} \psi^T(k)$$

separándole en dos términos:

$$\Psi^T.\,W.\,\Psi = \sum_{k=n}^{N} \psi(k) \,.\, a \,.\, \gamma \,.\, \gamma^{N-k} .\, \psi(k) + \psi(N+1) \,.\, a \,.\, \psi^T(N+1) \qquad (3.10)$$

Como se requiere obtener la matriz inversa de la expresión (3.10); se define la matriz P $_{(2n \times 2n)}$ **(MATRIZ DE COVARIANZA)** como sigue:

$$P(N+1) = [\Psi^{T}(N+1).W.\Psi(N+1)]^{-1} \qquad (3.11)$$

Reemplazando en la expresión (3.10) se tiene:

$$P(N+1) = [\gamma P^{-1}(N) + \psi(N+1)a\psi^{T}(N+1)]^{-1} \qquad (3.12)$$

Para resolver la ecuación (3.12), es necesario conocer la inversa de la suma de dos matrices, por lo que se aplicará el Lema de Inversión de Matrices (Householder 1964).

<u>Lema de Inversión de Matrices</u>

Sean A,B,C y D matrices de dimensiones compatibles; y , A una matriz no singular, se cumple que:

$$(A + B.C.D)^{-1} = A^{-1} - A^{-1}.B.(C^{-1} + D.A^{-1}.B)^{-1}.D.A^{-1} \qquad (3.13)$$

A fin de demostrar la igualdad (3.13) se probará que:

$$(A + B.C.D) .[A^{-1} - A^{-1}.B.(C^{-1} + D.A^{-1}.B)^{-1}.D.A^{-1}] = I,$$

donde I es la matriz identidad.

Desarrollando el producto término a término se tiene:

$AA^{-1} + BCDA^{-1} - AA^{-1}B[C^{-1} + DA^{-1}B]^{-1}DA^{-1} - BCDA^{-1}B[C^{-1} + DA^{-1}B]^{-1}DA^{-1} = I$

$I + BCDA^{-1} - B[C^{-1} + DA^{-1}B]^{-1}DA^{-1} - BCDA^{-1}B[C^{-1} + DA^{-1}B]^{-1}DA^{-1} = I$

$I + BCDA^{-1} - BCC^{-1}[C^{-1} + DA^{-1}B]^{-1}DA^{-1} - BCDA^{-1}B[C^{-1} + DA^{-1}B]^{-1}DA^{-1} = I$

agrupando términos

$I + BCDA^{-1} - BC\{C^{-1}[C^{-1} + DA^{-1}B]^{-1} + DA^{-1}B[C^{-1} + DA^{-1}B]^{-1}\}DA^{-1} = I$

$I + BCDA^{-1} - BC\{[C^{-1} + DA^{-1}B][C^{-1} + DA^{-1}B]^{-1}\}DA^{-1} = I$

en consecuencia:

$I + BCDA^{-1} - BCDA^{-1} = I$

$I = I$

Aplicando el lema de inversión de matrices a la ecuación (3.12), y considerando que:

$A = \gamma P^{-1}(N)$

$B = \psi(N+1)$

$C = a$

$D = \psi^{T}(N+1)$

Se determina la siguiente expresión:

$$P(N+1)=\frac{P(N)}{\gamma}-\frac{P(N)}{\gamma}\psi\left(\frac{1}{a}+\psi^{T}\frac{P(N)}{\gamma}\psi\right)^{-1}\psi^{T}\frac{P(N)}{\gamma}$$

(3.14)

La expresión (3.14) constituye el término $(\Psi^T.W.\Psi)^{-1}$, de la ecuación (3.9).

Para completar el análisis, falta por determinar la expresión $\Psi^T.W.Y$ de tal manera que, considere la precesencia de un dato adicional, teniéndose:

$$\Psi^T W Y=\left|\psi(n)\ \ \ldots\ldots\ \ \psi(N)\ \ \psi(N+1)\right|.\begin{vmatrix} a\gamma^{N+1-n} & 0 & \ldots & 0 \\ 0 & a\gamma^{N-n} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & a\gamma & 0 \\ 0 & \ldots & \ldots & a \end{vmatrix}.\begin{vmatrix} y(n) \\ \ldots. \\ \ldots. \\ \ldots \\ y(N) \\ y(N+1) \end{vmatrix}$$

(3.15)

separando en dos términos:

$$\Psi^T.W.Y(N+1) = \gamma.\Psi^T.W.Y(N) + \psi(N+1).a.y(N+1) \quad (3.16)$$

Reemplazando en la expresión (3.9), la ecuación (3.14) para el término $(\Psi^T.W.\Psi)^{-1}$ y, la ecuación (3.16) para el término $\Psi^T.W.Y$. En forma adicional, para facilitar la notación se reemplaza, $P(N) = P$; $\psi(N+1) = \psi$; $y(N+1) = y$; teniéndose:

$$\hat{\theta}_{WLS}(N+1) = [\frac{P}{\gamma} - \frac{P}{\gamma}\psi(\frac{1}{a} + \psi^T\frac{P}{\gamma}\psi)^{-1}\psi^T\frac{P}{\gamma}] . [\gamma\Psi^T W Y(N) + \psi a y]$$

(3.17)

Multiplicando los factores:

$$\hat{\theta}_{WLS}(N+1) = P\Psi^T W Y(N) + P/\gamma\psi a y - [P\psi(1/a + \psi^T P/\gamma\psi)^{-1}\psi^T P/\gamma] . [\Psi^T W Y(N)] - [P/\gamma.\psi(1/a + \psi^T P/\gamma.\psi)^{-1}.\psi^T P/\gamma].\psi a y$$

si $P.\Psi^T.W.Y(N) = \hat{\theta}_{WLS}(N)$ la expresión se reduce a:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + P/\gamma.\psi a y - [P.\psi(1/a + \psi^T P/\gamma.\psi)^{-1}. \psi^T P/\gamma].[\Psi^T.W.Y(N)] - [P/\gamma.\psi(1/a + \psi^T P/\gamma.\psi)^{-1}. \psi^T P/\gamma.\psi a y$$

(3.18)

Se inserta la matriz identidad $I = [1/a + \psi^T P/\gamma.\psi]^{-1}.[1/a + \psi^T P/\gamma.\psi]^{-1}$; se tiene:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + P/\gamma.\psi I a y - [P\psi(1/a + \psi^T P/\gamma.\psi)^{-1} . \psi^T P/\gamma].[\Psi^T W Y(N)] - [P/\gamma.\psi(1/a + \psi^T P/\gamma.\psi)^{-1}.\psi^T P/\gamma]\psi a y$$

Agrupando términos semejantes y simplificando:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + P/\gamma. \psi.[1/a + \psi^T P/\gamma.\psi]^{-1}.\{Y(N+1) - \psi^T.\hat{\theta}_{WLS}(N)\}$$

Se define la Ganancia de Kalman L(N+1) como sigue:

$$L(N+1)=\frac{P}{\gamma}\psi\left[\frac{1}{a}+\psi^{T}\frac{P}{\gamma}\psi\right]^{-1} \qquad (3.19)$$

Por tanto, la ecuación se reduce a:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + L(N+1).\{Y(N+1) - \psi^{T}\hat{\theta}_{WLS}(N)\} \qquad (3.20\ a)$$

Si se considera $\epsilon = \{Y(N+1) - \psi^{T}\hat{\theta}_{WLS}(N)\}$, la ecuación (3.20 a) se puede escribir como sigue:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + L(N+1).\epsilon \qquad (3.20\ b)$$

Donde ϵ constituye el error de predicción. Esta expresión permite afirmar que, el valor siguiente (N+1) de los parámetros θ es igual al valor anterior (N), más el error de un dato muestreado.

3.2 BLUE (MEJOR ESTIMADOR LINEAL INSESGADO)

El tratamiento de mínimos cuadrados del numeral anterior es del tipo determinístico. En los casos reales a diferencia de los casos ideales, los datos están sujetos a perturbaciones aleatorias (ruido), siendo importante considerar este efecto en un proceso. Para este caso el desarrollo corresponde a un tratamiento del tipo estocástico.

Se parte de un modelo determinístico con error aleatorio en los datos (ruido blanco 2.1.3), siendo la ecuación de salida:

$$Y = \Psi . \theta^{o} + V \tag{3.21}$$

Ecuación de tratamiento similar a la ecuación de error (3.4) del algoritmo de mínimos cuadrados ordinarios; donde **V** es una matriz de variables aleatorias (ruido blanco Gaussiano) con media cero y son no correlacionadas; por lo tanto **Y** es una variable aleatoria.

Se asume que los datos actuales son generados de la expresión (3.21) con $\theta = \theta^{o}$ y :

$$E[v(k)] = 0 \quad \text{para algún } k$$

$$E[v(k).v(j)] = \rho_{kj}.\sigma^2 = 0 \text{ si } k<>j$$

$$E[v(k).v(j)] = \rho_{kj}.\sigma^2 = \sigma^2 \text{ si } k = j$$

Para el caso de que V es un vector: $E\ V.V^T = \sigma^2.I$

Al igual que en el caso de mínimos cuadrados determinísticos, se trata de encontrar θ de tal manera que se minimice la función de costo $J(\theta)$, que es:

$$J(\theta) = \sum_{k=n}^{N} e^2(k;\theta), \quad e(k;\theta) = v(k) \tag{3.22}$$

en forma matricial:

$$J(\theta) = V^T(N;\theta)V(N;\theta) = (Y - \Psi.\theta)^T.(Y - \Psi.\theta)$$

Expresión similar a la determinada para el caso determinístico, (3.5 b), diferenciándose en que el error a más de depender de los parámetros seleccionados, depende de la señal de ruido aleatorio. Por lo tanto los parámetros θ se estiman a partir de la misma expresión (3.7) $\hat{\theta}_{LS} = (\Psi^T.\Psi)^{-1}.\Psi^T.Y$.

No se puede esperar una buena estimación de los parámetros si el sistema se encuentra afectado por ruido aleatorio, por lo que es necesario conocer que tan satisfactorio es el proceso de identificación.

Para el caso estocástico existen tres características que permiten determinar si la estimación es buena, las mismas que son:

1.- CONSISTENCIA
2.- SESGO
3.- MEJOR ESTIMADOR LINEAL

3.2.1 CONSISTENCIA

Un estimador $\hat{\theta}$ de θ^o, se dice que es consistente, si para un número N finito de muestras, el vector paramétrico estimado $\hat{\theta}(N)$ converge al valor de θ^o; es decir la diferencia entre $\hat{\theta}$ y θ^o se hace despreciable.

Para medir la diferencia entre $\hat{\theta}(N)$ y θ^o, se utiliza el criterio de la suma del error medio cuadrático de $\hat{\theta} - \theta^o$, teniéndose:

$$limit_{N\to\infty} E(\hat{\theta}(N) - \theta^o)^T(\hat{\theta}(N) - \theta^o) = 0 \qquad (3.23)$$

$$limit_{N\to\infty} tr\; E(\hat{\theta}(N) - \theta^o)(\hat{\theta}(N) - \theta^o)^T = 0$$

Expresión que dice que, $\hat{\theta}(N)$ converge a θ^o cuando N tiende a infinito.

A partir de las ecuaciones (3,7) y (3.21) se tiene:

$$\begin{aligned}\hat{\theta}_{LS} - \theta^o &= (\Psi^T.\Psi)^{-1}.\ \Psi^T.Y - \theta^o \\ &= (\Psi^T.\Psi)^{-1}.\ \Psi^T.(\Psi.\theta^o + V) - \theta^o \\ &= (\Psi^T.\Psi)^{-1}.\ \Psi^T.\Psi.\theta^o + (\Psi^T.\Psi)^{-1}.\ \Psi^T.V - \theta^o\end{aligned}$$

$$= \theta^o + (\Psi^T.\Psi)^{-1}.\ \Psi^T.V - \theta^o$$

$$= (\Psi^T.\Psi)^{-1}.\ \Psi^T.V$$

reemplazando en la ecuación (3.23):

$$E(\widehat{\theta}_{LS} - \theta^o).(\widehat{\theta}_{LS} - \theta^o)^T = E\ \{(\Psi^T.\Psi)^{-1}.\Psi^T.V.V^T.\Psi.(\Psi^T.\Psi)^{-1}\}$$

$$E(\widehat{\theta}_{LS} - \theta^o).(\widehat{\theta}_{LS} - \theta^o)^T = (\Psi^T.\Psi)^{-1}.\Psi^T.EV.V^T.\Psi.(\Psi^T.\Psi)^{-1}$$

$$E(\widehat{\theta}_{LS} - \theta^o).(\widehat{\theta}_{LS} - \theta^o)^T = \sigma^2.(\Psi^T.\Psi)^{-1}$$

La estimación de mínimos cuadrados es consistente si cumple con la expresión siguiente:

$$limit_{N\to\infty} tr\ \sigma^2(\Psi^T\Psi)^{-1} = 0 \qquad (3.24)$$

3.2.2 SESGO (BIAS)

Si $\widehat{\theta}$ es un estimador de θ^o , se dice que es un estimador sesgado, si la diferencia entre el valor esperado del estimador $\widehat{\theta}$, con el valor verdadero de θ^o, para todo $\theta(N)$ es diferente de cero.

$$E\widehat{\theta}(N) - \theta^o = b \qquad (3.25)$$

Si b es igual a cero (b = 0) para todo $\hat{\theta}(N)$; se tiene el caso de un estimador insesgado.

Se va a comprobar que el estimador de mínimos cuadrados dada por la expresión (3.7) ($\hat{\theta}_{LS} = (\Psi^T.\Psi)^{-1}.\ \Psi^T.Y$) se trata de un estimador insesgado.

Partiendo de la expresión $\hat{\theta}_{LS} - \theta^o = (\Psi^T.\Psi)^{-1}.\ \Psi^T.V$, y aplicando la definición de sesgo se tiene el siguiente desarrollo:

$$E\hat{\theta}_{LS} - \theta^o = E(\Psi^T.\Psi)^{-1}.\ \Psi^T.V$$

$$E\hat{\theta}_{LS} - \theta^o = (\Psi^T.\Psi)^{-1}.\ \Psi^T.EV$$

Como se trata de ruido blanco con media cero, EV = 0, se tiene:

$$E\hat{\theta}_{LS} - \theta^o = 0$$

En consecuencia un estimador de mínimos cuadrados, es un estimador insesgado.

3.2.3 MEJOR ESTIMADOR LINEAL

Esta propiedad tiene que ver con el error del estimador $\hat{\theta}(N)$ para un número finito de muestras N, respecto al valor real de los parámetros θ^o.

Ante la dificultad de tener un estimador de $\hat{\theta}$ tal que, el cuadrado del error de $\hat{\theta} - \theta^o$ sea el mínimo, se busca que, el estimador sea el mejor estimador lineal de Y, y que no tenga sesgo, entonces:

$$\hat{\theta} = L.Y \tag{3.26}$$

$E(\hat{\theta}) = \theta^o$; Para el caso de estimador insesgado.

Reemplazando en la ecuación (3.26), la ecuación de error $Y = \Psi.\theta^o + V$, es:

$$\begin{aligned}
\hat{\theta} &= L.(\Psi.\theta^o + V) \\
E(\hat{\theta}) &= E[L.(\Psi.\theta^o + V)] \\
\theta^o &= EL.(\Psi.\theta^o + V) \\
\theta^o &= E[L.\Psi.\theta^o + L.V]
\end{aligned}$$

en consecuencia

$$\theta^o = L.\Psi.\theta^o \quad ó \quad L.\Psi = I \tag{3.27}$$

Se debe definir la matriz L, de tal manera que, el error medio cuadrático sea lo más pequeño posible.

El índice de funcionamiento esta dado por:

$$J(L) = tr\ E(\hat{\theta} - \theta^o).(\hat{\theta} - \theta^o)^T \qquad (3.28)$$

reemplazando la ecuación (3.26) en la ecuación (3.28), se tiene:

$$J(L) = tr\ E\{(L.Y - \theta^o).(L.Y - \theta^o)^T\}$$

$$J(L) = tr\ E[L.(\Psi.\theta^o + V) - \theta^o].[L.(\Psi.\theta^o + V) - \theta^o]^T$$

$$J(L) = tr\ E[L.\Psi.\theta^o + L.V - \theta^o].[L.\Psi.\theta^o + L.V - \theta^o]^T$$

$$J(L) = tr\ E[\theta^o + L.V - \theta^o].[\theta^o + L.V - \theta^o]^T$$

$$J(L) = tr\ E[L.V].[L.V]^T$$

$$J(L) = tr\ EL.V.V^T.L^T$$

$$J(L) = tr\ L.E(V.V^T).L^T \qquad (3.29)$$

$$J(L) = tr\ L.R.L^T$$

donde $E(V.V^T) = R$ representa la covarianza del ruido.

Si se considera que θ es un escalar, L es una matriz fila, por lo tanto no existe la traza de esta matriz. L debe ser

lo más pequeña posible.

Para calcular el mínimo se utiliza los **multiplicadores de Lagrange** (λ = lambda).

$$J(L) = L.R.L^T + \lambda.(\Psi^T.L^T - I) \qquad (3.30)$$

Para encontrar un mínimo se debe cumplir que, tal que, la ecuación del índice de funcionamiento (3.29) sea lo más pequeña posible.

Para calcular el mínimo se utiliza los **multiplicadores de Lagrange** (λ = lambda).

$$J(L) = L.R.L^T + \lambda.(\Psi^T.L^T - I) \qquad (3.30)$$

Para encontrar un mínimo se debe cumplir que,

$$\frac{\partial J}{\partial L^T}\Big|_{L=\hat{L}} = 0$$

entonces, resolviendo:

$$J(L) = L.R.L^T + \lambda.\Psi^T.L^T - \lambda.I$$

$$\frac{\partial J}{\partial L^T}\Big|_{L=\hat{L}} = 2\hat{L}R + \lambda\Psi^T = 0$$

$$2.L.R + \lambda.\Psi^T = 0$$

donde L es:

$$2.L.R = -\lambda.\Psi^T$$

$$L = -\tfrac{1}{2}\lambda.\Psi^T.R^{-1} \qquad (3.31)$$

Reemplazando (3.31) en la ecuación (3.27) se tiene:

$$-\tfrac{1}{2}\,\lambda.\Psi^{T}.R^{-1}.\Psi = I$$

En donde, los multiplicadores de Lagrange λ son:

$$\lambda = -2.(\Psi^{T}.R^{-1}.\Psi)^{-1} \qquad (3.32)$$

reemplazando λ en la expresión (3.31) se tiene:

$$L = (\Psi^{T}.R^{-1}.\Psi)^{-1}.\Psi^{T}.R^{-1} \qquad (3.33)$$

Por lo tanto el estimador BLUE es:

$$\hat{\theta}_{B} = L.Y = (\Psi^{T}.R^{-1}.\Psi)^{-1}.\Psi^{T}.R^{-1}.Y \qquad (3.34)$$

Esta ecuación es similar a la determinada, para el algoritmo de mínimos cuadrados ordinarios ponderados. Por tanto, por analogía se tiene que, la ponderación para el caso del estimador BLUE esta dado por $W = R^{-1}$, donde $R = \sigma^{2}.I$.

Entonces la varianza de $\hat{\theta}_{LS}$, la cual es definida como $E(\hat{\theta}_{LS} - \theta^{o}).(\hat{\theta}_{LS} - \theta^{o})^{T}$ es:

$$\mathrm{var}(\hat{\theta}_{LS}) = \sigma^{2}.(\Psi^{T}.R^{-1}.\Psi)^{-1} \qquad (3.35)$$

La ecuación (3.34) permite encontrar un estimador que debe ser el mejor y el único.

Para demostrar la afirmación anterior, se parte del hecho que existe otro estimador que sea lineal e insesgado $\bar{\theta}$.

$$\bar{\theta} = L.Y = (\hat{L} + \bar{L}).Y$$

donde L esta dado por la expresión (3.33) y ademas cumple que $L.\Psi = I$, por lo tanto:

$$(\hat{L} + \bar{L}).\Psi = I$$
$$\hat{L}.\Psi + \bar{L}.\Psi = I$$
$$I + \bar{L}.\Psi = I$$
$$\bar{L}.\Psi = 0 \qquad (3.36)$$

Finalmente falta por demostrar que la función de costo del estimador es la mejor así, $J(\bar{\theta}) = tr\ E(\bar{\theta} - \theta^o).(\bar{\theta} - \theta^o)^T$, y se puede escribir de la siguiente manera:

$$J(\bar{\theta}) = tr\ E(\bar{\theta} - \hat{\theta} + \hat{\theta} - \theta^o).(\bar{\theta} - \hat{\theta} + \hat{\theta} - \theta^o)^T$$
$$J(\bar{\theta}) = tr\ \{E(\bar{\theta} - \hat{\theta})(\bar{\theta} - \hat{\theta})^T + 2E.(\bar{\theta} - \hat{\theta})(\bar{\theta} - \theta^o)^T + E(\hat{\theta} - \theta^o)(\hat{\theta} - \theta^o)^T\} \qquad (3.37)$$
$$J(\bar{\theta}) = tr\ \{E(\bar{\theta} - \hat{\theta})(\bar{\theta} - \hat{\theta})^T\} + 2tr\{E.(\bar{\theta} - \hat{\theta})(\bar{\theta} - \theta^o)^T\} + tr\ E(\hat{\theta} - \theta^o)(\hat{\theta} - \theta^o)^T\}$$

Por lo tanto la ecuación (3.38) se reduce a:

$$J(\overline{\theta}) = tr\ \{E(\overline{\theta} - \widehat{\theta})(\overline{\theta} - \widehat{\theta})^T\} + J(\widehat{\theta})$$

Expresión que permite afirmar que, para cualquier θ, $J(\widehat{\theta}) \leq J(\overline{\theta})$, si $\widehat{\theta}$ es el estimador BLUE.

3.3 MAXIMO DE VEROSIMILITUD (MAXIMUM LIKELIHOOD)

El estimador de máximo de verosimilitud o de probabilidad máxima, consiste en maximizar una función densidad de probabilidad para las variables aleatorias envueltas. Se toma para el análisis como base la distribución normal o gaussiana, pero el método no tiene ninguna restricción de la forma de la función densidad de probabilidad.

Para una variable aleatoria x (escalar), la distribución normal esta dado por la expresión:

$$f_x(\xi) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left[\frac{-1(\xi-\mu)^2}{2\sigma^2}\right]} \qquad (3.39)$$

notado como $N(u,\sigma)$, en donde:

σ^2 = la varianza de x notada var(x); σ = desviación standar.
u = valor medio.

Para el caso de tener n variables aleatorias, la función densidad de probabilidad, es una función densidad de probabilidad conjunta normal, el valor medio es un vector de valores medios u y la covarianza es una matriz de covariancias R no singular.

$$E(\mathbf{x}) = \mathbf{u}, \qquad E(\mathbf{x} - \mathbf{u}).(\mathbf{x} - \mathbf{u})^T = R$$

Por tanto la función densidad de probabilidad se puede escribir de la siguiente manera:

$$f_x(\xi) = \frac{1}{\sqrt{(2\pi)^n \det R}} e^{\frac{-1}{2}(\xi-\mu)^T R^{-1}(\xi-\mu)} \qquad (3.40)$$

notado N(u,R)

Si los elementos del vector **x** son no correlacionados, y tienen idéntico valor medio u, y varianza σ^2, R = σ^2.I, det R = $(\sigma^2)^n$ entonces la ecuación (3.40) puede escribirse de la siguiente manera:

$$f_x(\xi) = \frac{1}{\sqrt{(2\pi\sigma^2)^n}} e^{\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(\xi_i-\mu)^2} \qquad (3.41)$$

Para el estimador de probabilidad máxima se asume la función densidad de probabilidad de las observaciones, y como base del análisis el modelo ARMAX. A partir de la ecuación (2.-40), donde $C_e(k) = v(k)$ representa variables aleatorias de media cero y son no correlacionados entre si (ruido blanco) se tiene el siguiente desarrollo:

$$y(k) = a1y(k-1)) + a2y(k-2) + \ldots + any(k-n) + b1u(k-1) + b2u(k-2) + \ldots + bnu(k-n) + v(k) \qquad (3.42)$$

Se parte como datos el conjunto de entradas y salidas

$$\{u(0),\ u(1),\ u(2),\ldots\ldots u(N), y(0), y(1),\ldots\ldots y(N)\}$$

Despejando el error, se tiene la siguiente expresión:

$$v(k;\theta) = y(k) - a1y(k-1)) - a2y(k-2) - \ldots - any(k-n) - b1u(k-1) - b2u(k-2) - \ldots - bnu(k-n)$$

El sistema esta representado por los parámetros

$$\theta = [a1\ a2\ldots\ldots an\ b1\ b2\ldots\ldots bn]^T$$

Como y(k) depende de los valores anteriores de las señales de entrada y de salida se define el **vector de error de salida V** para n instantes de muestreo k = n, n+1, ..., N, donde el primer elemento de este vector esta dado por la expresión.

Para K=n

Como y(k) depende de los valores anteriores de las señales de entrada y de salida se define el **vector de error de salida V** para n instantes de muestreo k = n, n+1, ..., N, donde el primer elemento de este vector esta dado por la expresión.

Para K=n

$$v(n;\theta) = y(n) - a1y(n-1)) - a2y(n-2) - .... - any(0) - b1u(n-1) + b2u(n-2) +.... + bnu(0)$$

utilizando notación matricial:

$$v(n;\theta) = y(n) - \psi^T(n).\theta$$

Para K=n+1

$$v(n+1;\theta) = y(n+1) - a1y(n)) - a2y(n-1) - .... - any(1) - b1u(n) - b2u(n-1) -.... - bnu(1)$$

en forma matricial:

$$v(n+1;\theta) = y(n+1) - \psi^T(n+1).\theta$$

......

$$Y(N) = [y(n)\ y(n+1)\ y(n+2)\ ....\ y(N)]^T$$

Matriz columna de N-n+1 filas

$\Psi(N) = [\psi(n)\ \psi(n+1)\ \psi(n+2)\ \ldots\ldots\ldots\ldots\ \psi(N)]^T$

Matriz de 2n columnas y N-n+1 filas

$V(N) = [v(n)\ v(n+1)\ v(n+2)\ \ldots\ldots\ v(N)]^T$

Matriz columna de N-n+1 filas

Se asume que v(k) tiene una distribución normal de media cero, varianza σ^2 y la covarianza entre v(k) y v(j) es cero para k <> j (V es $N(0, \sigma^2.I)$).

$\theta^o = [a1\ a2\ldots\ldots\ldots an\ b1\ b2\ldots\ldots\ldots\ bn]^T$

Matriz columna de 2n filas

Si se asume que u, y, ai, bi son conocidos, se puede calcular v(k), de los datos u e y. Para determinar el valor de V a partir de Y, es necesario determinar el modelo inverso de la planta:

$$V(N) = Y - \Psi.\theta^o \qquad (3.44)$$

Se utiliza la función densidad de probabilidad $f(Y|\theta^o)$, para el desarrollo del estimador de máximo de verosimilitud; entonces:

$$f(Y|\theta^o) = \frac{1}{\sqrt{(2\pi\sigma^2)^m}} e^{\frac{-1}{2\sigma^2}(Y-\Psi\theta^o)^T(Y-\Psi\theta^o)} \qquad (3.45)$$

donde $m = N - n + 1$, es el número de muestras de Y.

La función de probabilidad $f(Y|\theta^o)$ queda determinada en forma general, si se reemplaza θ^o por θ.

Tomando el logaritmo natural de la función, de tal forma de poder encontrar el máximo de la misma se tiene:

$$Ln\{f(Y|\theta)\} = Ln(2.\pi.\sigma^2)^{-m/2} + Ln\{\ exp[-\tfrac{1}{2}\ .(Y - \Psi.\theta)^T(Y - \Psi.\theta)]\}$$

$$-Ln\{f(Y|\theta)\} = m/2\ Ln(2.\pi) + m/2\ Ln(\sigma^2) + \tfrac{1}{2}\ .(Y - \Psi.\theta)^T(Y - \Psi.\theta) \qquad (3.46)$$

Los estimadores $\hat{\theta}_{ML}$ y σ^2_{ML} son los valores de θ y σ^2 que permiten que la función $Ln\{f(Y|\theta)\}$ se haga máxima; por lo tanto se encuentra las derivadas parciales respecto a θ y σ^2 , e igualando a cero:

De las ecuaciones (3.47 a,b) se concluye que, el estimador

- Derivando respecto a σ^2_{ML}:

$$\partial(-Ln\{f(Y|\theta)\})/\partial\sigma^2_{ML}$$

$$= \partial[m/2\ Ln(2.\pi) + m/2\ Ln(\sigma^2) + \tfrac{1}{2}(Y - \Psi.\theta)^T(Y - \Psi.\theta)]/\partial\sigma^2_{ML}$$

$$= m/2\ .1/\sigma^2_{ML} - \tfrac{1}{2}\ .(Y - \Psi.\theta)^T(Y - \Psi.\theta)/\sigma^4_{ML}$$

igualando a cero y despejando σ^2_{ML}

$$m/2\ .1/\sigma^2_{ML} = \tfrac{1}{2}\ .(Y - \Psi.\theta)^T(Y - \Psi.\theta)/\sigma^4_{ML}$$

$$\sigma^2_{ML} = (Y - \Psi.\theta)^T(Y - \Psi.\theta)/m \qquad (3.47a)$$

- Derivando respecto a θ_{ML}:

$$= \partial(-Ln\{f(Y|\theta)\})/\partial\theta_{ML}$$

$$= \partial[m/2\ Ln(2.\pi) + m/2\ Ln(\sigma^2) + \tfrac{1}{2}\ (Y - \Psi.\theta)^T(Y - \Psi.\theta)]/\partial\theta_{ML}$$

$$= \tfrac{1}{2}\ .(Y - \Psi.\theta)^T(-\Psi) + (Y - \Psi.\theta)(-\Psi^T)$$

igualando a cero y despejando θ_{ML}.

$$(Y - \Psi.\theta)^T(-\Psi) + (Y - \Psi.\theta)(-\Psi^T)/2\sigma^2 = 0$$

$$(\Psi^T.\Psi.\theta_{ML} - \Psi^T.Y)/\sigma^2 = 0 \qquad (3.47b)$$

De las ecuaciones (3.47 a,b) se concluye que, el estimador de máximo de verosimilitud, da los mismos valores estimados que los encontrados por el método de mínimos cuadrados.

En igual forma, si se toma al modelo ARMAX en forma general, en donde se considera valores pasados del ruido y ponderados por la magnitud de ci, se tiene el siguiente desarrollo:

La ecuación de diferencias es:

$$y(k) = a1y(k-1)) + a2y(k-2) + \ldots + any(k-n) + b1u(k-1) + \ldots + bnu(k-n) + c1v(k-1) + \ldots + cnv(k-n) + v(k) \qquad (3.48)$$

Por facilidad de análisis se tiene que c1 no es igual a cero y los restantes ci = 0, por lo tanto la expresión del error es:

$$v(k) + c1v(k-1) = y(k) - a1y(k-1)) - a2y(k-2) - \ldots - any(k-n) - b1u(k-1) - \ldots - bnu(k-n) \qquad (3.49)$$

se define z(k) como la suma de dos variables aleatorias v(k) y v(k-1), (v(k) + c1v(k-1) = z(k)) entonces:

$$z(k) = y(k) - a1y(k-1)) - a2y(k-2) - \ldots - any(k-n) - b1u(k-1) - \ldots - bnu(k-n)$$

en forma matricial:

$$Z(N) = Y - \Psi.\theta^o \tag{3.50}$$

El valor medio y la covarianza es:

$$E\{z(k)\} = E\{v(k) + c1v(k-1)\} = 0 \text{ , para todo } k$$

$$\begin{aligned} E\{z(k).z(j)\} &= E\{v(k) + c1v(k-1)\}\{v(j) + c1v(j-1)\} \\ &= \sigma^2 (1 + c1^2) \quad k = j \\ &= \sigma^2.c1 \quad k = j - 1 \\ &= \sigma^2.c1 \quad k = j + 1 \\ &= 0 \quad \text{para otro valor} \end{aligned}$$

La forma de la covarianza de Z(N) es:

$$R = E\{Z(N).Z(N)^T\}$$

$$\begin{vmatrix} 1+c1^2 & c1 & 0 & \ldots & \ldots & 0 \\ c1 & 1+c1^2 & c1 & 0 & \ldots & 0 \\ 0 & c1 & 1+c1^2 & c1 & \ldots & 0 \end{vmatrix}$$

Con el valor medio y la covarianza definidas, la función densidad de probabilidad normal, es:

$$g(Z(N)|\theta^o) = ((2.\pi)^m \det R)^{-½} \exp(-½Z^T.R^{-1}.Z) \tag{3.51}$$

3.4.1 ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

El algoritmo de mínimos cuadrados ordinarios consta de los siguientes pasos:

1.- Definición de las variables compartidas entre los diferentes programas y dimensionamiento de todas las variables a utilizarse en el algoritmo.

2.- Recuperar los valores del número de datos generados, n, m, señal de entrada y señal de salida, magnitud del escalón, varianza y magnitud del ruido.

3.- Seleccionar el tipo de proceso

Se puede seleccionar las tres opciones siguientes:

3.1 Mínimos Cuadrados Ordinarios Determinísticos.- Para este caso, la señal de salida no se encuentra afectada por ruido aleatorio aditivo.

3.2 Mínimos Cuadrados Ordinarios Probabilísticos.- Para este caso, se añade a la señal de salida un 10% de ruido.

3.3 Salir.- Opción que enlaza al programa de Mínimos Cuadrados Ordinarios, con el programa del Menú Principal.

4.- Verificación y/o cambio del orden del modelo n, y determinación del número de muestras.

- El orden del modelo debe ser mayor que cero y menor que cuatro; valores dentro de los cuales se encuentran los modelos reales.
- El número de muestras es mayor que el doble del modelo, y menor que 50 - 2*n.

El orden del modelo, al igual que el número de muestras a seleccionarse tienen valores límites, los mismos que se justifican en la medida de las limitaciones de la memoria del computador. Sin embargo son rangos suficientes para un correcto análisis del algoritmo.

5.- Formar la matriz $Y(N) = [y(n)\ y(n+1) \ldots\ldots y(N)]^T$.

6.- Formar la matriz $\Psi(N) = [\psi(n)\ \psi(n+1)\ \psi(n+2) \ldots\ldots\ldots\ldots\ldots \psi(N)]^T$.

Donde $\psi(n) = [y(n-1)\ y(n-2) \ldots y(0)\ u(n-1)\ u(n-2) \ldots u(0)]^T$

7.- Formar la matriz $\Psi(N)^T = [\psi(n)\ \psi(n+1)\ \psi(n+2)\ \ldots\ldots\ldots\ \psi(N)]$.

8.- Multiplicar las matrices $\Psi(N)^T.\ \Psi(N)$.

9.- Multiplicar las matrices $\Psi(N)^T.\ Y(N)$.

10.- Invertir la multiplicación de matrices $(\Psi(N)^T.\ \Psi(N))^{-1}$.

Se utiliza el algoritmo de inversión de matrices de Gauss con pivotaje completo, comparando el valor absoluto de cada uno de los términos de las matrices con un valor de epsilon igual a 0.0001. Si el valor del determinante es menor a 0.000001 trunca el proceso de inversión de matrices, indicándose que la señal de entrada no excita a la dinámica de la planta.

11.- Determinar el vector de parámetros $\hat{\theta}_{LS} = (\Psi(N)^T.\ \Psi(N))^{-1}.\ (\Psi(N)^T.\ Y(N))$.

12.- Calcular la señal de salida con los valores de los parámetros $\hat{\theta}$ obtenidos en el punto anterior.

13.- Almacenar el número de datos, n, m, muestras, salida generada, salida calculada y la matriz θ de los parámetros calculados.

14.- Presentación de los resultados en pantalla e indicar si se quiere imprimir los resultados.

3.4.2 ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

El algoritmo de mínimos cuadrados recursivos consta de los siguientes pasos:

1.- Definición de las variables compartidas entre los diferentes programas y dimensionamiento de todas las variables a utilizarse en el algoritmo.

2.- Recuperar los valores del número de datos generados, n, m, señal de entrada y señal de salida, magnitud del escalón, varianza y magnitud del ruido.

3.- Seleccionar el tipo de proceso

Se puede seleccionar las tres opciones siguientes:

3.1 Mínimos Cuadrados Recursivos Determinísticos.- Para este caso, la señal de salida no se encuentra afectada por ruido aleatorio aditivo.

3.2 Mínimos Cuadrados Recursivos Probabilísticos.- Para este caso, se añade a la señal de salida un 10% de ruido.

3.3 Salir.- Opción que enlaza al programa de Mínimos Cuadrados Recursivos, con el programa del Menú Principal.

4.- Verificación y/o cambio del orden del modelo n, y determinación del número de muestras.

- El orden del modelo debe ser mayor que cero y menor o igual que cuatro.
- El número de muestras es mayor que el doble del modelo, y menor que 50 - 2*n.

5.- Generación de los datos de la matriz de ponderación $w = a.\gamma^{N-K}$, se puede escoger de dos opciones:

5.1 Ponderación $a = 1$; $\gamma = 1$.

Si $a = 1$ y $\gamma = 1$,se tiene el caso de mínimos cuadrados recursivos ordinarios.

5.2 Ponderación Exponencial.

En la cual, a = 1 - γ (0 < gamma < 1), se tiene el caso de mínimos cuadrados recursivos ponderados exponencialmente.

Una vez que, se ha seleccionado el tipo de ponderación se forma la matriz de ponderación que tiene la siguiente forma:

$$W=\begin{vmatrix} a\gamma^{N+1-n} & 0 & \ldots & 0 \\ 0 & a\gamma^{N-n} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & a\gamma & 0 \\ 0 & \ldots & \ldots & a \end{vmatrix}$$

6.- Determinar las condiciones iniciales de las matrices $\theta(N)$ y Covarianza $P(N)$ de la siguiente manera:

- Para el caso de la matriz $\theta(N) = 0$.

  $\theta = [0\ 0 \ldots\ldots 0\ 0 \ldots\ldots 0]^T$

- Para el caso de la matriz de Covarianza $P(N) = \alpha.I$, en donde α es un escalar, e I es la matriz identidad. El valor de α, se selecciona este entre el rango de valores 1.000 a 1´000.000, a lo que se añade la sumatoria del cuadrado de la señal de salida, teniéndose la siguiente expresión:

$$\alpha = \frac{(10.000)}{N+1} \sum_{i=0}^{N} . y^2(i)$$

7.- Determinar el número de iteraciones (1 < iteraciones < 50 -2*n).

8.- Formar la matriz $\psi(k+1)$ = [y(k) y(k-1)....y(k-n+1) u(k) u(k-1)....u(k-n+1)]^T.

9.- Formar la matriz transpuesta $\psi^T(k+1)$ = [y(k) y(k-1)..-..y(k-n+1) u(k) u(k-1)....u(k-n+1)]

10.- Determinar la matriz de Kalman, $L(N+1) = P/\gamma.\ \psi.[1/a + \psi^T P/\gamma.\psi]^{-1}$.

11.- Determinar la matriz de parámetros theta, $\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + L(N+1)\cdot\{Y(N+1) - \psi^T.\hat{\theta}_{WLS}(N)\}$

12.- Determinar la matriz P(N+1), $P(N+1) = P(N)/\gamma\ [I - L(N+1)\ \psi^T(N+1)]$

$\psi(k+2)$ = [y(k+1) y(k)....y(k-n+2) u(k+1) u(k)....u(k-n+2)]^T

14.- Regresar al paso 7 de acuerdo al número de iteraciones especificadas.

15.- Almacenar el número de iteraciones, los parámetros iniciales a(n,1) y c(1,1) y los calculados por el algoritmo de mínimos cuadrados recursivos.

13.- Formar el siguiente término de la matriz de parámetros $\psi(k+2)$ = [y(k+1) y(k)....y(k-n+2) u(k+1) u(k)....u(k-n+2)]^T

14.- Regresar al paso 7 de acuerdo al número de iteraciones especificadas.

15.- Almacenar el número de iteraciones, los parámetros iniciales a(n,1) y c(1,1) y los calculados por el algoritmo de mínimos cuadrados recursivos.

16.- Presentación de los resultados en pantalla e indicar si se quiere imprimir los resultados.

3.4.3 ALGORITMO DE GENERACION DE DATOS

El algoritmo de generación de datos consta de los siguientes pasos:

1.- Definición de las variables compartidas entre los diferentes programas y dimensionamiento de todas las variables a utilizarse en el algoritmo.

2.- Presentar el modelo mediante variables de estado de la forma canónica controlable y verificar la existencia del archivo de datos.

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a3 & -a2 & -a1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} . u(k)$$

$$y(k) = \begin{vmatrix} b3 & b2 & b1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix}$$

Donde D = 0

3.- Introducir el orden del modelo ($0 < n \leq 4$), este rango de definición del orden del modelo se establece, en la medida de que para trabajar en forma adecuada con modelos reales, es suficiente disponer de modelos de hasta cuarto orden.

4.- Introducir el valor de m ($0 < m \leq n$), este rango se establece en la medida de que en análisis de control clásico, el número de ceros es menor o igual que el número de polos.

5.- Ingresar los elementos an,j y c1,j de las matrices A, B, C para un sistema representado mediante variables de estado de la forma canónica controlable. En donde, los $a_{n,j}$ y $c_{1,j}$ son ≥ -4 y ≤ 4, debido a limitaciones de memoria del computador.

6.- Formar las matrices A, B, C y presentar en pantalla.

7.- Generar la señal de entrada, la misma que se puede seleccionar entre las siguientes opciones:

7.1 Entrada escalón.- Se ingresa la amplitud de la señal escalón, la misma que se iguala con la variable que representa la señal de entrada.

7.2 Ruido Blanco.- Para determinar la señal de ruido blanco, se realiza una interpretación matemática al teorema del Límite Central (2.1.3 Ruido Blanco) a fin de simular una distribución normal con media μ y varianza σ^2.

Si una variable aleatoria con distribución normal dada por la expresión siguiente:

$$f(x) = \frac{1}{\sigma . \sqrt{2 . \pi}} . e^{-\frac{1}{2} . [\frac{(x-\mu)}{\sigma}]^2}$$

tiene los valores de μ y σ, 0 y 1 respectivamente, la función recibe el nombre de distribución normal o estándar.

Cualquier distribución normal puede convertirse en estándar haciendo y = (x - μ)/σ, entonces se tiene:

$$f(x) = \frac{1}{\sigma.\sqrt{2.\pi}}.e^{-\frac{1}{2}.y^2} \qquad (3.53)$$

Si x1, x2, ... , xn representan variables aleatorias independientes, con la misma distribución de probabilidades (μ,σ²), entonces:

$$limit_{n\to\infty} P.[a<\frac{\sum_{i=1}^{n}.xi-n.\mu}{\sqrt{n.\sigma}}<b] = \frac{1}{\sqrt{2.\pi}}.\int_a^b e^{[-\frac{y^2}{2}]}.dy$$

en donde:

$$\sum_{i=1}^{n}\mu_i = n.\mu; \quad \sum_{i=1}^{n}\sigma_i^2 = n.\sigma^2$$

El procedimiento para simular valores normales, requiere la suma de K valores de variables aleatorias (x1, x2, ..., xk) distribuidas uniformemente en el intervalo $0 < xi < 1$, donde $\mu = (a+b)/2$ y $\sigma = (b - a)/\sqrt{(12)}$.

$$Y = \frac{\sum_{i=1}^{k} ri - \frac{k}{2}}{\sqrt{\frac{k}{12}}}$$

Como Y es una variable aleatoria con distribución normal estándar, entonces se puede escribir la igualdad siguiente:

$$\frac{x-\mu}{\sigma} = \frac{\sum_{i=1}^{k} ri - \frac{k}{2}}{\sqrt{\frac{k}{12}}}$$

siendo

$$x = \sigma \, . \, \frac{\sum_{i=1}^{k} ri - \frac{k}{2}}{\sqrt{\frac{k}{12}}} + \mu$$

Tomando el valor de k = 12, para evitar realizar operaciones con raíz cuadrada y considerando que el ruido blanco gaussiano tiene media cero, entonces:

$$x = \sigma . \sum_{i=1}^{12} ri-6 \qquad (3.54)$$

Expresión que permite obtener ruido blanco Gaussiano.

7.3 Ruido.- Se ingresa la amplitud de la señal de ruido que se multiplica por el número aleatorio generado por el computador, valor que se iguala con la variable que representa la señal de entrada.

7.4 Salir.- Opción que enlaza al programa de Generación de Datos, con el programa del Menú Principal.

8.- Generación de la señal de salida a partir del modelo ARMA representado mediante variables de estado.

9.- Almacenar el número de datos generados, los valores de n, m los valores de las señales de entrada y salida generados y los valores de las matrices A, C ingresados, la magnitud de la señal escalón, la varianza y la magnitud del ruido.

10.- Presentación de los resultados en pantalla e indicar si se quiere imprimir los resultados.

Los diagramas de flujo de los algoritmos de Mínimos Cuadrados Ordinarios, Mínimos Cuadrados Recursivos y de Generación de los Datos se encuentran en el Anexo 1, incluyéndose el listado de los programas en lenguaje BASIC.

CAPITULO IV

EJEMPLOS DE APLICACION

4.1 INTRODUCCION

Con la finalidad de determinar la validez de los programas implementados para la identificación de parámetros de un modelo mediante variables de estado, en este capítulo se desarrollan varios ejemplos, los cuales cubrirán todas las posibilidades que prestan los programas. Las conclusiones correspondientes se establecerán en el capítulo V.

Se ejecutan los ejemplos con cada una de las señales de entrada y sus respectivas señales de salida implementadas en el programa de generación de datos, para los algoritmos de Mínimos Cuadrados Ordinarios (Detérministicos y Probabilísticos) y Mínimos Cuadrados Recursivos (Detérministicos y Probabilísticos).

La identificación paramétrica de sistemas estocásticos, es abordada en los ejemplos en los que se añade un 10% de ruido a la señal de salida. Por cuanto, quedó demostrado en el desarrollo teórico del numeral 3.2 del capítulo III que sus ecuaciones corresponden exactamente a las de los sistemas determinísticos para el algoritmo de Mínimos Cuadrados Ordinarios.

4.2 EJEMPLOS DE APLICACION

Para el análisis, se utiliza sistemas de primero, segundo y tercer orden, representados mediante variables de estado en la forma canónica controlable. Estos ejemplos, se desarrollan de cualquiera de las formas de representar a un modelo para sistemas discretos, que pueden ser: Ecuación de Diferencias, Función de Transferencia o directamente Variables de Estado. Pero para la ejecución de los programas, necesariamente debe ser introducido sus datos de acuerdo a la forma canónica controlable en variables de estado.

La analogía paramétrica entre los diferentes modelos y la base teórica de esta forma canónica se encuentran desarrollados en el numeral 2.2.1 del capítulo II.

Cada ejemplo incluye su desarrollo respectivo, y la impresión de los resultados y de los gráficos de la ejecución de los programas.

4.2.1 EJEMPLO 1.- SISTEMA DE PRIMER ORDEN

Para un sistema de primer orden, se parte de un circuito serie RC (ver figura 4.1), cuya ecuación de diferencial es de la forma:

$$\frac{dy(t)}{dt} = -\frac{1}{RC}y(t) + \frac{1}{RC}u(t) \quad para\ kT \leq t < (k+1)T \qquad (4.1)$$

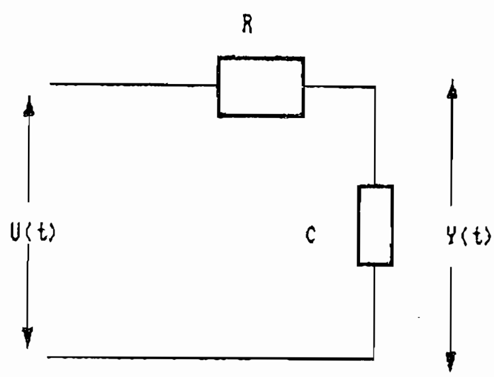


Fig. 4.1 Sistema de primer orden (Circuito Serie RC)

Se determina la ecuación de diferencias, considerando que la señal de entrada aplicada es segmentariamente constante $u(t) = u(kT)$ para $kT \leq t < (k+1)T$.

Sacando la Transformada de Laplace de la ecuación (4.1) y considerando $t = kT$ como el tiempo inicial, se tiene:

$$RC.[SY(S) - y(kT)] + Y(S) = U(kT)/S$$

$$[x1(k+1)] = [0.37].[x1(k)] + [1].u(k) \quad (4.3)$$

$$y(k) = [0.63].u(k) \quad (4.4)$$

$$\theta = [0.37 \;\; 0.63] \quad (4.5)$$

4.2.1.1 DESARROLLO (MINIMOS CUADRADOS ORDINARIOS)

- Señal de entrada ruido blanco, varianza del ruido = 1, (Cuadro 1, Figura 1 a).

  Resultados: Valor de los parámetros a11, c11, para el caso determinístico con muestras = 20 (Cuadro 2, Figura 1 b); y para el caso probabilístico con muestras = 10, 40 (Cuadros 3,4, Figura 2 a,b). Gráficos de las señales de salida generada y calculada.

- Señal de entrada ruido aleatorio u(k) = 5, (Cuadro 5, Figura 3 a).

  Resultados: Valor de los parámetros a11, c11 para los casos determinístico (Cuadro 6, Figura 3 b) y probabilístico con muestras = 20 (Cuadro 7, Figura 4). Gráficos de las señales de salida generada y calculada.

4.2.1.2 DESARROLLO (MINIMOS CUADRADOS RECURSIVOS)

- Señal de entrada ruido blanco, varianza del ruido =1.

  Resultados: Valor de los parámetros a11, c11 para el caso determinístico con gamma= 1, 0.9; alfa = 1.000, 10.000; muestras = 10, 40; iteraciones = 10, 40; (Cuadros 8,9, Figuras 5 a,b); y para el caso probabilístico con gamma= 0.9; alfa = 10.000; muestras = 40; iteraciones = 40 (Cuadro 10, Figura 6 a). Gráficos de los parámetros a(n,1) y c(1,n) generados, calculados.

- Señal de entrada ruido aleatorio u(k) = 5.

  Resultados: Valor de los parámetros a11, c11, para los casos determinístico (Cuadro 11, Figura 6 b) y probabilístico (Cuadros 12, Figura 7 a) con gamma = 0.9; alfa = 10.000; muestras = 40; iteraciones = 40. Gráficos de los parámetros a(n,1) y c(1,n) generados y calculados.

4.2.2 EJEMPLO 2.- SISTEMA DE SEGUNDO ORDEN

Para un sistema de segundo orden, se parte de la ecuación de la dinámica rotacional de un cuerpo rígido para el control de la altitud de un satélite con respecto a un solo eje (ver figura (4.2), cuya ecuación de diferencial es de la forma:

$$(4.6) \quad I.\frac{d^2\theta(t)}{dt^2} = M_c + M_p$$

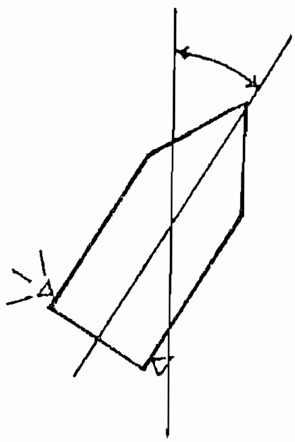


Fig. 4.2 Sistema de segundo orden (Control de un satélite)

Donde:

I = Momento de Inercia del satélite respecto al centro de masa.

Mc = Torque de control.

Mp = Torque de perturbación.

θ = Angulo del eje del satélite respecto a una referencia inarcial.

Normalizando la ecuación (4.6), respecto al momento de inercia, donde u = Mc/I y wd = Mp/I se tiene:

$$\frac{d^2\theta(t)}{dt^2} = u(t) + wd(t)$$

Considerando que no existe perturbaciones, la descripción del modelo en variables de estado para el caso continuo, si θ = x1; x2 = dx1/dt; u(t) = dx2/dt es:

$$\begin{vmatrix} \dfrac{dx1(t)}{dt} \\ \dfrac{dx2(t)}{dt} \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} x1(t) \\ x2(t) \end{vmatrix} + \begin{vmatrix} 0 \\ 1 \end{vmatrix} \cdot u(t) \qquad (4.7)$$

$$\theta(t) = |1 \quad 0| \cdot \begin{vmatrix} x1(t) \\ x2(t) \end{vmatrix} \qquad (4.8)$$

Para discretizar las ecuaciones de estado continuas llegando a expresiones de la forma X(k+1) = **A**.**X**(k) + B.U(k); y(k) = C.X(k) se tiene:

$$A = \mathcal{L}^{-1}.(SI-A)^{-1};\ B = (\int_0^T \mathcal{L}^{-1}.(SI-A)^{-1}).B;\ C = C$$

resolviendo:

$$A = \begin{vmatrix} 1 & T \\ 0 & 1 \end{vmatrix};\ B = \begin{vmatrix} \dfrac{T^2}{2} \\ T \end{vmatrix}$$

considerando T= 1 (s), las ecuaciones de estado para el caso discreto son:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \end{vmatrix} = \begin{vmatrix} 1 & 1 \\ 0 & 1 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \end{vmatrix} + \begin{vmatrix} 0.5 \\ 1 \end{vmatrix} . u(k) \tag{4.9}$$

$$y(k) = |1 \quad 0| . \begin{vmatrix} x1(k) \\ x2(k) \end{vmatrix} \tag{4.10}$$

Otra forma de análisis es sacando la tranformada de Laplace de la ecuación (4.6) y normalizando la misma, entonces:

$$\theta(S) = 1/S^2.U(S);\ \text{donde } G(S) = \theta(S)/u(S)$$

Incluyendo un ZOH (Zero Orden Hold), dispositivo de retención de orden cero, donde su transformada Z es $G_H(S) = Z-1/Z$, y si $G1(Z) = Z\{G_H(S).G(S)\}$ entonces:

$$G1(Z) = \frac{Z-1}{Z} \cdot \mathbf{Z}\left[\frac{G(S)}{S}\right]$$

Reemplazando $G(S) = 1/S^2$,

$$G1(Z) = \frac{Z-1}{Z} \cdot \mathbf{Z}\left[\frac{1}{S^3}\right] \cdot \frac{2}{2}$$

Sacando la transformada Z y simplificando G1(Z) es:

$$G1(Z) = \frac{T^2}{2} \cdot \frac{Z+1}{(Z-1)^2} \qquad (4.11)$$

De la expresión (4.11) se puede ver que el sistema tiene dos polos que se ubican en el punto 1 del plano Z, es decir que el sistema es **inestable**.

Considerando que T = 1 (s), la descripción en variables de estado, a partir de la función de transferencia, se puede escribir en forma directa, de acuerdo al análisis desarrollado en el numeral 2.2.1 del capítulo II, el modelo en variables de estado de la forma canónica controlable es:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ -1 & 2 \end{vmatrix} \cdot \begin{vmatrix} x1(k) \\ x2(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 1 \end{vmatrix} \cdot u(k) \qquad (4.12)$$

$$y(k) = |0.5 \quad 0.5| \cdot \begin{vmatrix} x1(k) \\ x2(k) \end{vmatrix} \qquad (4.13)$$

$$\theta = [2 \quad -1 \quad 0.5 \quad 0.5] \qquad (4.14)$$

4.2.2.1 DESARROLLO (MINIMOS CUADRADOS ORDINARIOS)

- Señal de entrada escalón u(k) = 5.

  Resultados: Para los casos determinístico y probabilístico, no es posible generar resultados.

- Señal de entrada ruido blanco, varianza del ruido = 1.0 (Cuadro 13, Figura 7 b).

  Resultados: Valor de los parámetros, a21, a22, c11, c12 para el caso determinístico con muestras = 5, 15 (Cuadros 14, 15, Figura 8 a, b). Gráficos de las señales de salida generadas y calculadas.

- Señal de entrada ruido aleatorio u(k) = 5, (Cuadro 16, Figura 9 a).

  Resultados: Valor de los parámetros a21, a22, c11, c12 para el caso probabilístico con muestras = 5 (Cuadro 17, Figura 9 b). Gráficos de las señales de salida generada y calculada.

4.2.2.2 DESARROLLO (MINIMOS CUADRADOS RECURSIVOS)

- Señal de entrada ruido aleatorio u(k) = 5.

  Resultados: Valor de los parámetros, a21, a22, c11, c12 para el caso determinístico con gamma = 0.9; alfa = 1.000; muestras = 5; iteraciones = 20 (Cuadro 18, Figura 10 a). Gráficos de los parámetros a(n,1) y c(1,n) generados y calculados.

4.2.3 EJEMPLO 3.- SISTEMA DE TERCER ORDEN

A diferencia de los ejemplos de los sistemas de primero y segundo orden, en los cuales se parte de un sistema real para la obtención de las respectivas ecuaciones de estado discretas; para el ejemplo de un sistema de tercer orden se presentan directamente las ecuaciones de estado, las mismas que son:

$$\begin{vmatrix} x1(k+1) \\ x2(k+1) \\ x3(k+1) \end{vmatrix} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.3 & -0.5 & -0.8 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} . u(k) \tag{4.15}$$

$$y(k) = \begin{vmatrix} 0 & 0.8 & 2 \end{vmatrix} . \begin{vmatrix} x1(k) \\ x2(k) \\ x3(k) \end{vmatrix} \tag{4.16}$$

$$\theta = [-0.8 \quad -0.5 \quad 0.3 \quad 2 \quad 0.8 \quad 0] \tag{4.17}$$

4.2.3.1 DESARROLLO (MINIMOS CUADRADOS ORDINARIOS)

- Señal de entrada escalón u(k) = 5, (Cuadro 19, Figura 10 b).

  Resultados: Para el caso determinístico y probabilístico con muestras = 40, no existe resultados.

- Señal de entrada ruido blanco, varianza del ruido = 1.0, (Cuadro 20, Figura 11 a).

  Resultados: Valor de los parámetros, a31, a32, a33, c11, c12, c13 para el caso determinístico con muestras = 40 (Cuadro 21, Figura 11 b), y para el caso probabilístico con muestras = 40 (Cuadro 22, Figura 12 a). Gráficos de las señales de salida generada y calculada.

En esta parte, se disminuye el orden del modelo, siendo n = 2; m = 2.

  Resultados: Valor de los parámetros, a21, a22, c11, c12 para el caso determinístico con muestras = 40 (Cuadro 23, Figura 12 b). Gráficos de las señales de salida generada y calculada.

- Señal de entrada ruido aleatorio u(k) = 5 (Cuadro 24, Figura 13 a).

  Resultados: Valor de los parámetros a31, a32, a33, c11, c12, c13, para el caso probabilístico con muestras = 40, (Cuadro 25, Figura 13 b). Gráficos de las señales de salida generada y calculada.

En esta parte, se disminuye el orden del modelo, siendo n = 2; m = 2.

Resultados: Valor de los parámetros, a21, a22, c11, c12 para el caso probabilístico con muestras = 40, (Cuadro 26, Figura 14 a). Gráficos de las señales de salida generada y calculada.

4.2.3.2 DESARROLLO (MINIMOS CUADRADOS RECURSIVOS)

- Señal de entrada escalón u(k) = 5.

  Resultados: Valores de los parámetros a31, a32, a33, c11, c12, c13 para el caso determinístico con gamma = 0.95; alfa = 10.000; muestras = 40; iteraciones = 40, (Cuadro 27, Figura 14 b). Gráficos de los parámetros a(n,1) y c(1,1) generados y calculados.

- Señal de entrada ruido blanco, varianza del ruido = 1.0.

  Resultados: Valores de los parámetros a31, a32, a33, c11, c12, c13 para los casos determinístico, (Cuadro 28, Figura 15 a) y probabilístico, (Cuadro 29, Figura 15 b) con gamma= 0.9; alfa = 10.000; muestras = 40; iteraciones = 40.

Gráficos de los parámetros a(n,1) y c(1,n) generados, calculados.

- Señal de entrada ruido aleatorio, u(k) = 5.

  Resultados: Valores de los parámetros a31, a32, a33, c11, c12, c13 para el caso determinístico con gamma= 0.9; alfa = 10.000; muestras = 40; iteraciones = 40, (Cuadro 30, Figura 16). Gráficos de los parámetros a(n,1) y c(1,n) generados y calculados.

En esta parte, se disminuye el orden del modelo, siendo n = 2; m = 2.

Resultados: Valor de los parámetros, a21, a22, c11, c12 para el caso determinístico con gamma= 0.9; alfa = 10.000; muestras = 40; iteraciones = 40, (Cuadro 31).

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:BLAN10.DIN

**** D a t o s   i n i c i a l e s ***

```
Orden del modelo          =  1
Valor de m                =  1
Magnitud Escalón          =  0
Varianza (Ruido Blanco)=  1
Magnitud Ruido            =  0
```

Matriz A

****
+0.370

Matriz B

****
+1.000

Matriz C

****
+0.630

Cuadro 1 Entrada = Ruido Blanco, Varianza = 1

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : c:BLAN1ODE.DMO

**** Parámetros Calculados ***

Orden del modelo = 1
Valor de m = 1
Número de muestras = 20

Matriz A

****
+0.370000

Matriz B

****
+1.000000

Matriz C

****
+0.630000

** Tiempo de procesamiento : 0 min. 3 seg. **

Cuadro 2 Valor de los parámetros a11, c11 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN11PR.DMO

**** Parámetros Calculados ***

```
Orden del modelo   =  1
Valor de m         =  1
Número de muestras =  10
```

Matriz A

****
+0.392086

Matriz B

****
+1.000000

Matriz C

****
+0.656203

** Tiempo de procesamiento : 0 min. 3 seg. **

Cuadro 3 Valor de los parámetros a11, c11 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN12PR.DMO

**** Parámetros Calculados ***

Orden del modelo = 1
Valor de m = 1
Número de muestras = 40

Matriz A

****
+0.389330

Matriz B

****
+1.000000

Matriz C

****
+0.626460

** Tiempo de procesamiento : 0 min. 4 seg. **

Cuadro 4 Valor de los parámetros a11, c11 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:RUI10.DIN

**** D a t o s   i n i c i a l e s ***

Orden del modelo	=	1
Valor de m	=	1
Magnitud Escalón	=	0
Varianza (Ruido Blanco)	=	0
Magnitud Ruido	=	5

Matriz A

****
+0.370

Matriz B

****
+1.000

Matriz C

****
+0.630

Cuadro 5 Entrada = Ruido Aleatorio U(k) = 5

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:RUI1ODE.DMO

**** Parámetros Calculados ***

Orden del modelo = 1
Valor de m = 1
Número de muestras = 20

Matriz A

****
+0.370000

Matriz B

****
+1.000000

Matriz C

****
+0.630000

** Tiempo de procesamiento : 0 min. 3 seg. **

Cuadro 6 Valor de los parámetros a11, c11 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:RUI11PR.DMO

**** Parámetros Calculados ***

Orden del modelo = 1
Valor de m = 1
Número de muestras = 20

Matriz A

****
+0.384598

Matriz B

****
+1.000000

Matriz C

****
+0.629043

** Tiempo de procesamiento : 0 min. 3 seg. **

Cuadro 7 Valor de los parámetros a11, c11 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN13DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	1
Valor de m	=	1
Número de muestras	=	10
Número de iteraciones	=	10
Valor de Gamma	=	1
Valor de alfa	=	1000

Matriz A

****

+0.369252

Matriz B

****

+1.000000

Matriz C

****

+0.629177

** Tiempo de procesamiento : 0 min. 3 seg. **

Cuadro 8 Valor de los parámetros a11, c11 (determístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN14DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	1
Valor de m	=	1
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****
+0.369989

Matriz B

****
+1.000000

Matriz C

****
+0.629990

** Tiempo de procesamiento : 0 min. 13 seg. **

Cuadro 9 Valor de los parámetros a11, c11 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN15PR.DMR

**** Parámetros Calculados ***

```
Orden del modelo        =  1
Valor de m              =  1
Número de muestras      =  40
Número de iteraciones   =  40
Valor de Gamma          =  .9
Valor de alfa           =  10000
```

Matriz A

****
+0.392772

Matriz B

****
+1.000000

Matriz C

****
+0.642795

** Tiempo de procesamiento : 0 min. 13 seg. **

Cuadro 10 Valor de los parámetros a11, c11 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:RUI12DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	1
Valor de m	=	1
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****
+0.370000

Matriz B

****
+1.000000

Matriz C

****
+0.630000

** Tiempo de procesamiento : 0 min. .13 seg. **

Cuadro 11 Valor de los parámetros a11, c11 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:RUI13PR.DMR

**** Parámetros Calculados ***

Orden del modelo	=	1
Valor de m	=	1
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****

+0.373634

Matriz B

****

+1.000000

Matriz C

****

+0.633643

** Tiempo de procesamiento : 0 min. 13 seg. **

Cuadro 12 Valor de los parámetros a11, c11 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:BLAN20.DIN

**** D a t o s   i n i c i a l e s ***

```
Orden del modelo         =  2
Valor de m               =  2
Magnitud Escalón         =  0
Varianza (Ruido Blanco)= 1
Magnitud Ruido           =  0
```

Matriz A

****

```
+0.000     +1.000
-1.000     +2.000
```

Matriz B

****

```
+0.000
+1.000
```

Matriz C

****

```
+0.500     +0.500
```

Cuadro 13 Entrada = Ruido Blanco, Varianza = 1

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN2ODE.DMO

**** Parámetros Calculados ***

```
Orden del modelo    =  2
Valor de m          =  2
Número de muestras  =  5
```

Matriz A

****

```
        +0.000000   +1.000000
        -1.000000   +2.000000
```

Matriz B

****

```
        +0.000000
        +1.000000
```

Matriz C

****

```
        +0.500000   +0.500000
```

** Tiempo de procesamiento : 0 min. 5 seg. **

Cuadro 14   Valor de los parámetros a21, a22, c11, c12 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN21DE.DMO

**** Parámetros Calculados ***

Orden del modelo = 2
Valor de m = 2
Número de muestras = 15

Matriz A

****

+0.000000	+1.000000
-1.000164	+1.999397

Matriz B

****

+0.000000
+1.000000

Matriz C

****

+0.499947	+0.499924

** Tiempo de procesamiento : 0 min. 6 seg. **

Cuadro 15 Valor de los parámetros a21, a22, c11, c12 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:RUI20.DIN

**** D a t o s   i n i c i a l e s ***

```
Orden del modelo        =  2
Valor de m              =  2
Magnitud Escalón        =  0
Varianza (Ruido Blanco)=  0
Magnitud Ruido          =  5
```

Matriz A

****

```
+0.000      +1.000
-1.000      +2.000
```

Matriz B

****

```
+0.000
+1.000
```

Matriz C

****

```
+0.500      +0.500
```

Cuadro 16 Entrada = Ruido Aleatorio U(k) = 5

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:RUI2OPR.DMO

**** Parámetros Calculados ***

Orden del modelo = 2
Valor de m = 2
Número de muestras = 5

Matriz A

****

+0.000000	+1.000000
-1.176302	+2.087319

Matriz B

****

+0.000000
+1.000000

Matriz C

****

+0.591089	+0.495923

** Tiempo de procesamiento : 0 min. 5 seg. **

Cuadro 17 Valor de los parámetros a21, a22, c11, c12 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:RUI21DE.DMR

**** Parámetros Calculados ***

```
Orden del modelo        =  2
Valor de m              =  2
Número de muestras      =  5
Número de iteraciones   =  20
Valor de Gamma          =  .9
Valor de alfa           =  1000
```

```
Matriz  A

*********
        +0.000000   +1.000000
        -0.999921   +2.000159

Matriz B

********
        +0.000000
        +1.000000

Matriz C

********
        +0.502135   +0.499090
```

** Tiempo de procesamiento : 0 min. 22 seg. **

Cuadro 18   Valor de los parámetros a21, a22, c11, c12 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:ESC30.DIN

**** D a t o s   i n i c i a l e s ***

Orden del modelo	=	3
Valor de m	=	3
Magnitud Escalón	=	5
Varianza (Ruido Blanco)	=	0
Magnitud Ruido	=	0

Matriz A

****

+0.000	+1.000	+0.000
+0.000	+0.000	+1.000
+0.300	-0.500	-0.800

Matriz B

****

+0.000
+0.000
+1.000

Matriz C

****

+0.000	+0.800	+2.000

Cuadro 19 Entrada = Escalón U(k) = 5

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

RUTINA DE GENERACION DE DATOS

Archivo de datos : C:BLAN30.DIN

**** D a t o s   i n i c i a l e s ***

```
Orden del modelo        =  3
Valor de m              =  3
Magnitud Escalón        =  0
Varianza (Ruido Blanco)=  1
Magnitud Ruido          =  0
```

Matriz A

```
*********
        +0.000      +1.000      +0.000
        +0.000      +0.000      +1.000
        +0.300      -0.500      -0.800
```

Matriz B

```
********
        +0.000
        +0.000
        +1.000
```

Matriz C

```
********
        +0.000      +0.800      +2.000
```

Cuadro 20 Entrada = Ruido Blanco, Varianza = 1

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN3ODE.DMO

**** Parámetros Calculados ***

Orden del modelo = 3
Valor de m = 3
Número de muestras = 40

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.299985	-0.500029	-0.800028

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

-0.000014	+0.800002	+1.999987

** Tiempo de procesamiento : 0 min. 22 seg. **

Cuadro 21 Valor de los parámetros a31, a32, a33, c11, c12, c13 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN31PR.DMO

**** Parámetros Calculados ***

Orden del modelo = 3
Valor de m = 3
Número de muestras = 40

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.299516	-0.498175	-0.802151

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

-0.038894	+0.764191	+2.001463

** Tiempo de procesamiento : 0 min. 22 seg. **

Cuadro 22 Valor de los parámetros a31, a32, a33, c11, c12, c13 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:BLAN32DE.DMO

**** Parámetros Calculados ***

```
Orden del modelo    =  2
Valor de m          =  2
Número de muestras  =  40

Matriz  A

*********
        +0.000000   +1.000000
        +0.987382   -1.164856

Matriz B

********
        +0.000000
        +1.000000

Matriz C

********
        +0.041817   +0.637155
```

** Tiempo de procesamiento : 0 min. 11 seg. **

Cuadro 23 Valor de los parámetros a21, a22, c11, c12, (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:RUI30PR.DMO

**** Parámetros Calculados ***

Orden del modelo = 3
Valor de m = 3
Número de muestras = 40

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.301173	-0.501312	-0.798535

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

+0.015925	+0.810806	+2.011221

** Tiempo de procesamiento : 0 min. 22 seg. **

Cuadro 25 Valor de los parámetros a31, a32, a33, c11, c12, c13 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS ORDINARIOS

Archivo de datos : C:RUI31PR.DMO

**** Parámetros Calculados ***

Orden del modelo = 2
Valor de m = 2
Número de muestras = 40

Matriz A

****

+0.000000	+1.000000
+0.972898	-1.153392

Matriz B

****

+0.000000
+1.000000

Matriz C

****

+0.693540 +0.860676

** Tiempo de procesamiento : 0 min. 11 seg. **

Cuadro 26 Valor de los parámetros a21, a22, c11, c12, (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:ESC30DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	3
Valor de m	=	3
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.95
Valor de alfa	=	10000

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.311918	-0.511715	-0.786040

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

+0.926726	+0.926725	+0.926726

** Tiempo de procesamiento : 1 min. 49 seg. **

Cuadro 27 Valor de los parámetros a31, a32, a33, c11, c12, c13 (determínistico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN33DE.DMR

**** Parámetros Calculados ***

Orden del modelo = 3

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN33DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	3
Valor de m	=	3
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.300025	-0.500026	-0.799971

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

+0.000022	+0.799967	+1.999886

** Tiempo de procesamiento : 1 min. 49 seg. **

Cuadro 28 Valor de los parámetros a31, a32, a33, c11, c12, c13 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:BLAN34PR.DMR

**** Parámetros Calculados ***

Orden del modelo	=	3
Valor de m	=	3
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****

+0.000000	+1.000000	+0.000000
+0.000000	+0.000000	+1.000000
+0.297970	-0.498169	-0.801672

Matriz B

****

+0.000000
+0.000000
+1.000000

Matriz C

****

-0.083337	+0.767165	+1.951044

** Tiempo de procesamiento : 1 min. 49 seg. **

Cuadro 29 Valor de los parámetros a31, a32, a33, c11, c12, c13 (probabilístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:RUI32DE.DMR

**** Parámetros Calculados ***

```
Orden del modelo         =  3
Valor de m               =  3
Número de muestras       =  40
Número de iteraciones    =  40
Valor de Gamma           =  .9
Valor de alfa            =  10000
```

Matriz A

```
*********
        +0.000000   +1.000000   +0.000000
        +0.000000   +0.000000   +1.000000
        +0.300004   -0.500004   -0.799996
```

Matriz B

```
********
        +0.000000
        +0.000000
        +1.000000
```

Matriz C

```
********
        -0.000002   +0.800007   +1.999992
```

** Tiempo de procesamiento : 1 min. 49 seg. **

Cuadro 30 Valor de los parámetros a31, a32, a33, c11, c12, c13 (determinístico)

E S C U E L A   P O L I T E C N I C A   N A C I O N A L

F A C U L T A D   D E   I N G E N I E R I A   E L E C T R I C A

TESIS DE GRADO

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

JOSE RAUL MEJIA RECALDE

ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS

Archivo de datos : C:RUI33DE.DMR

**** Parámetros Calculados ***

Orden del modelo	=	2
Valor de m	=	2
Número de muestras	=	40
Número de iteraciones	=	40
Valor de Gamma	=	.9
Valor de alfa	=	10000

Matriz A

****

+0.000000	+1.000000
+0.967308	-1.153240

Matriz B

****

+0.000000
+1.000000

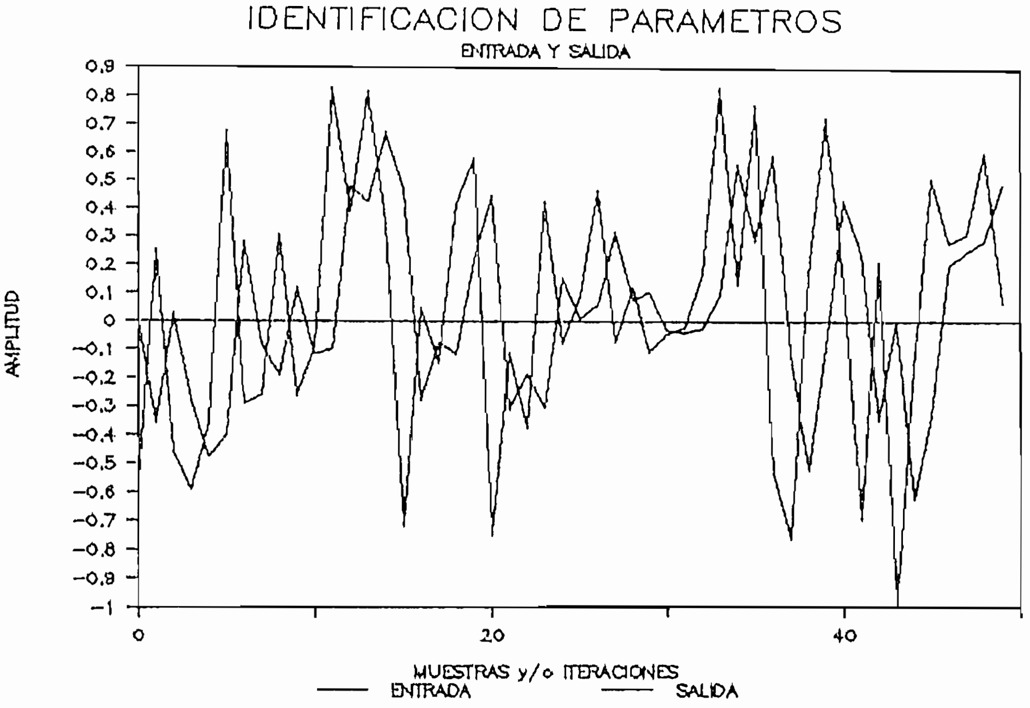
Matriz C

****

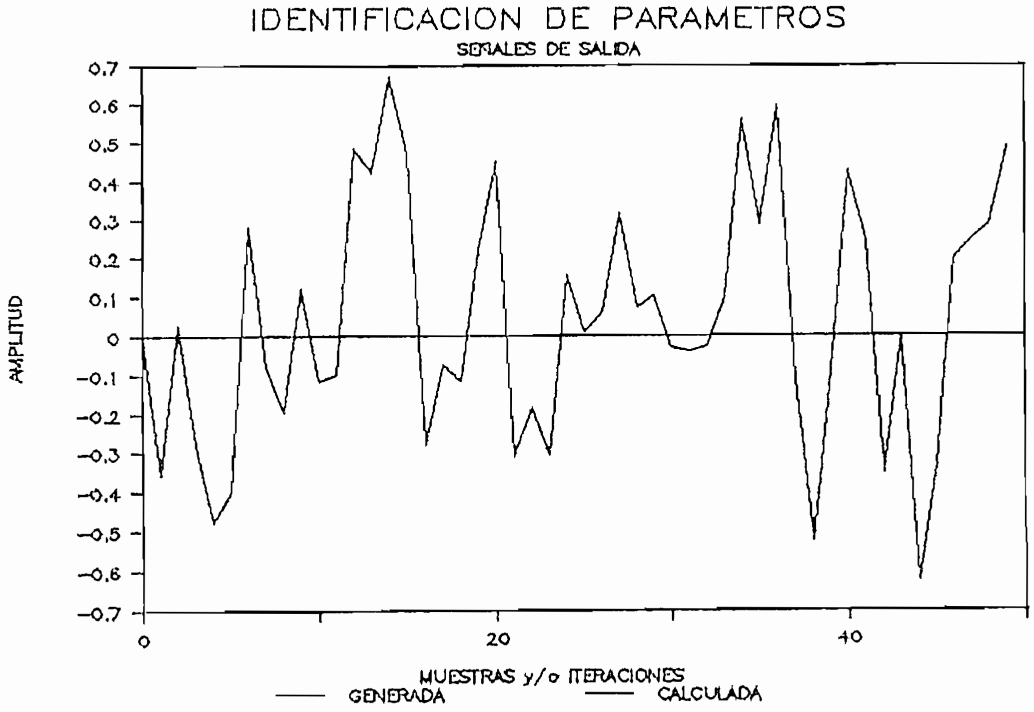
+0.143193	+1.210595

** Tiempo de procesamiento : 0 min. 44 seg. **

Cuadro 31 Valor de los parámetros a21, a22, c11, c12, (determinístico)

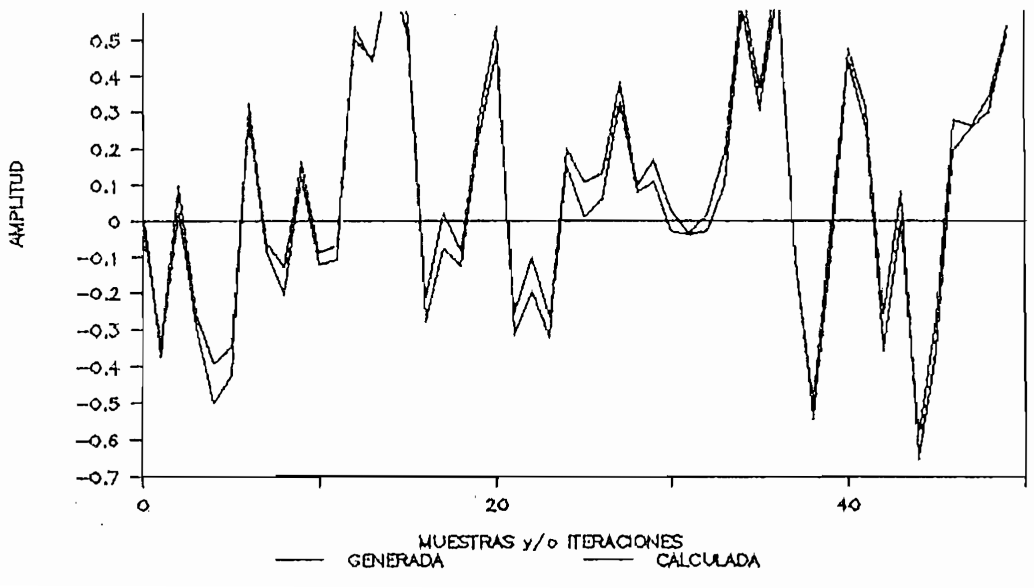


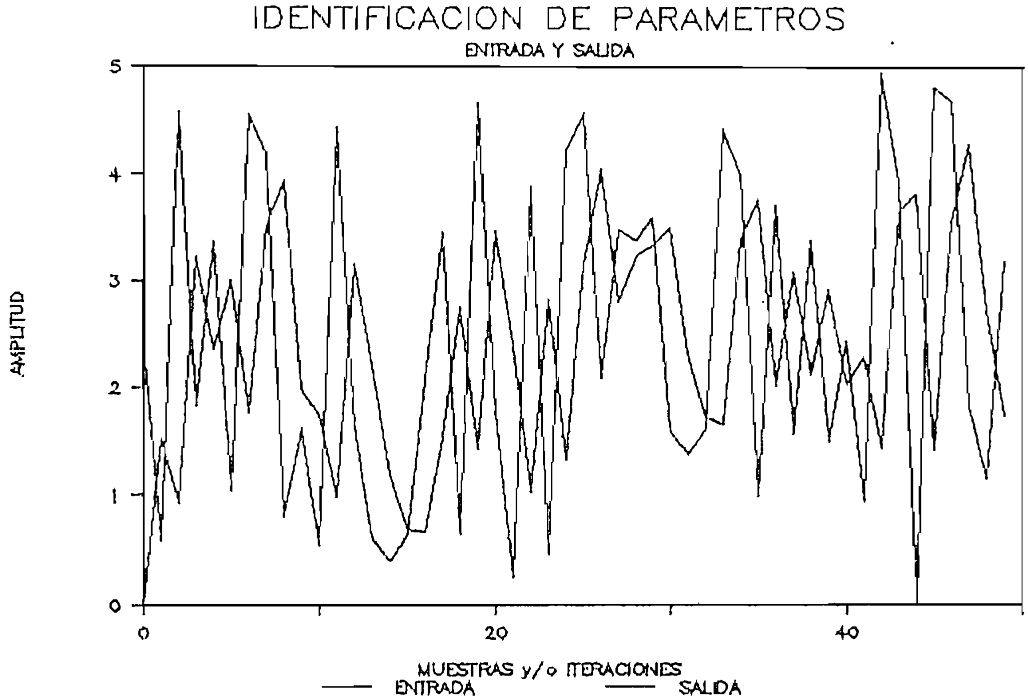
a)



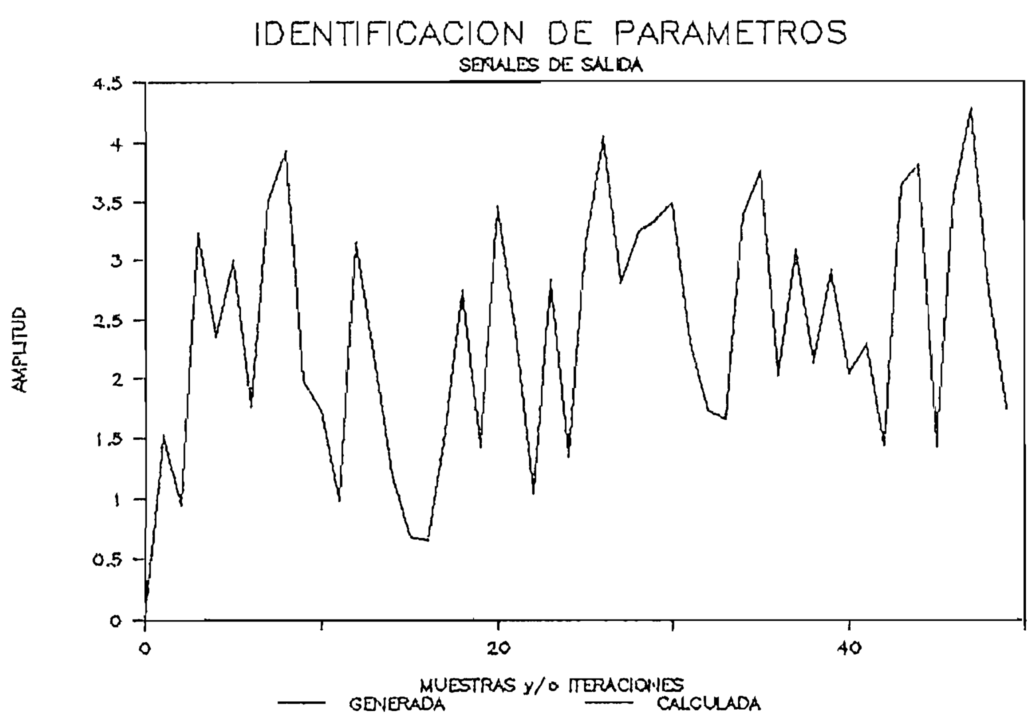
b)

Figura 1 a) Cuadro 1; b) Cuadro 2





a)



b)

Figura 3 a) Cuadro 5; b) Cuadro 6

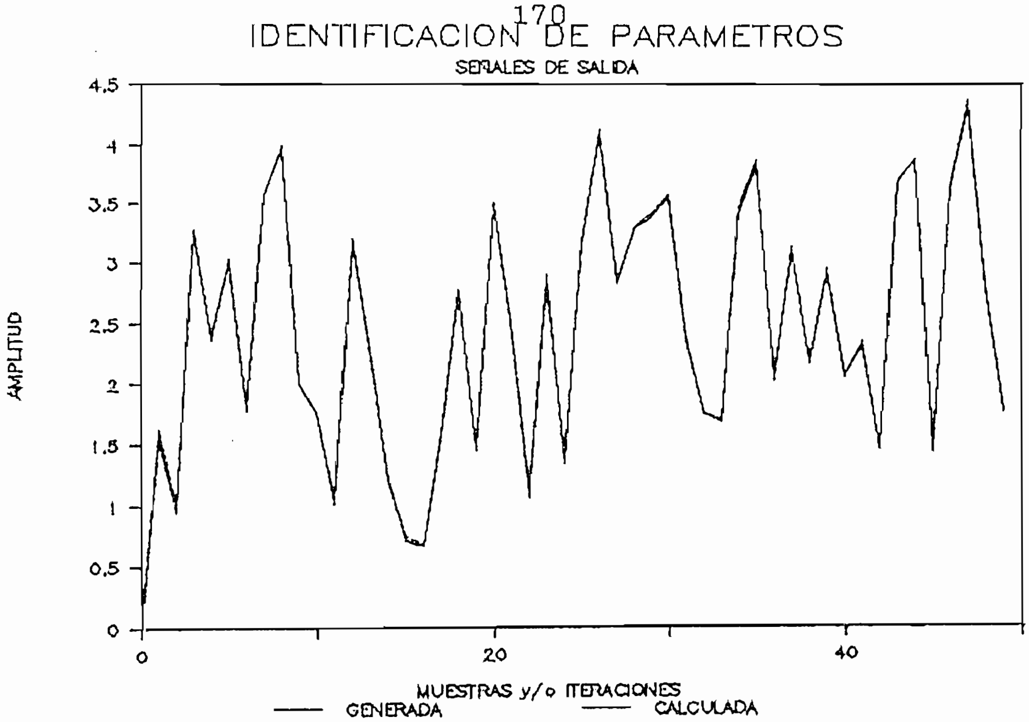
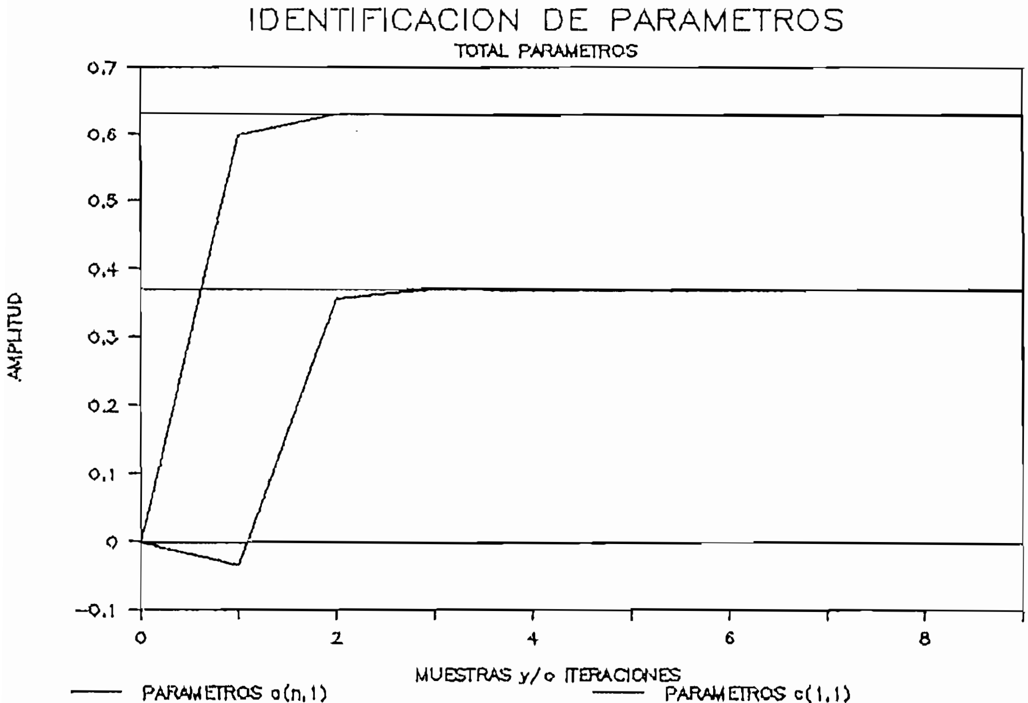
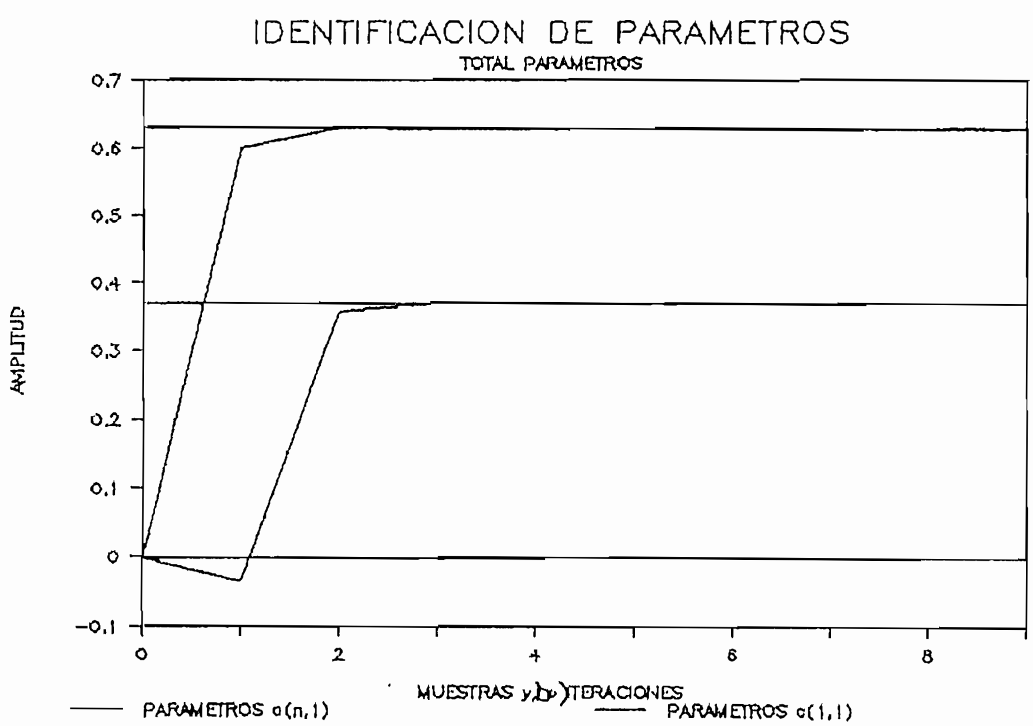


Figura 4 Cuadro 7

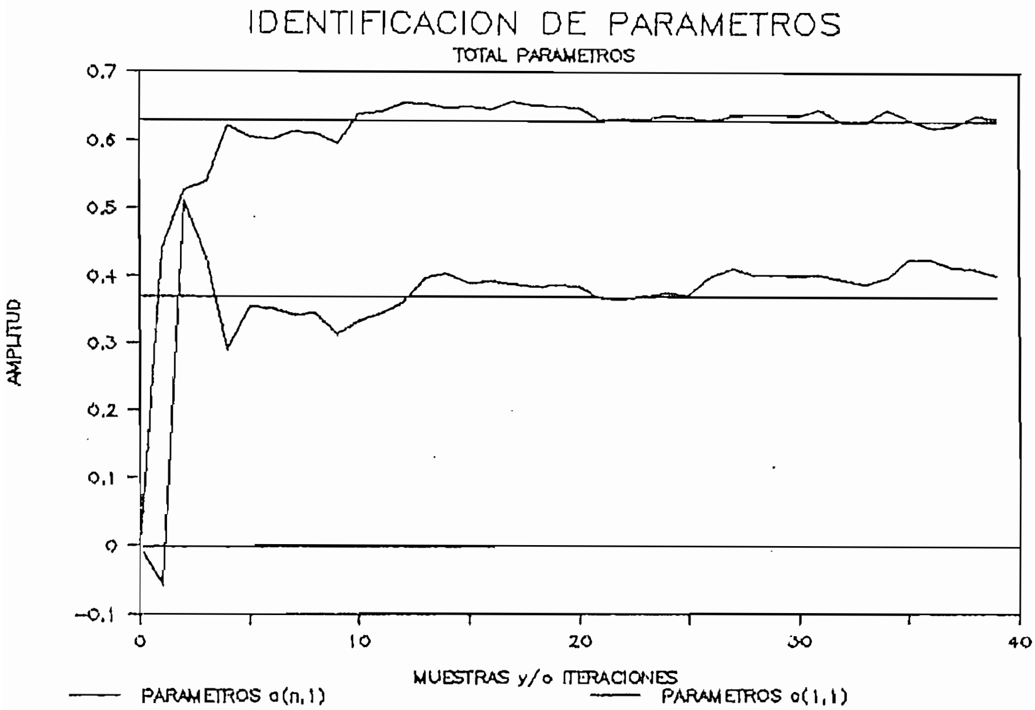


a)

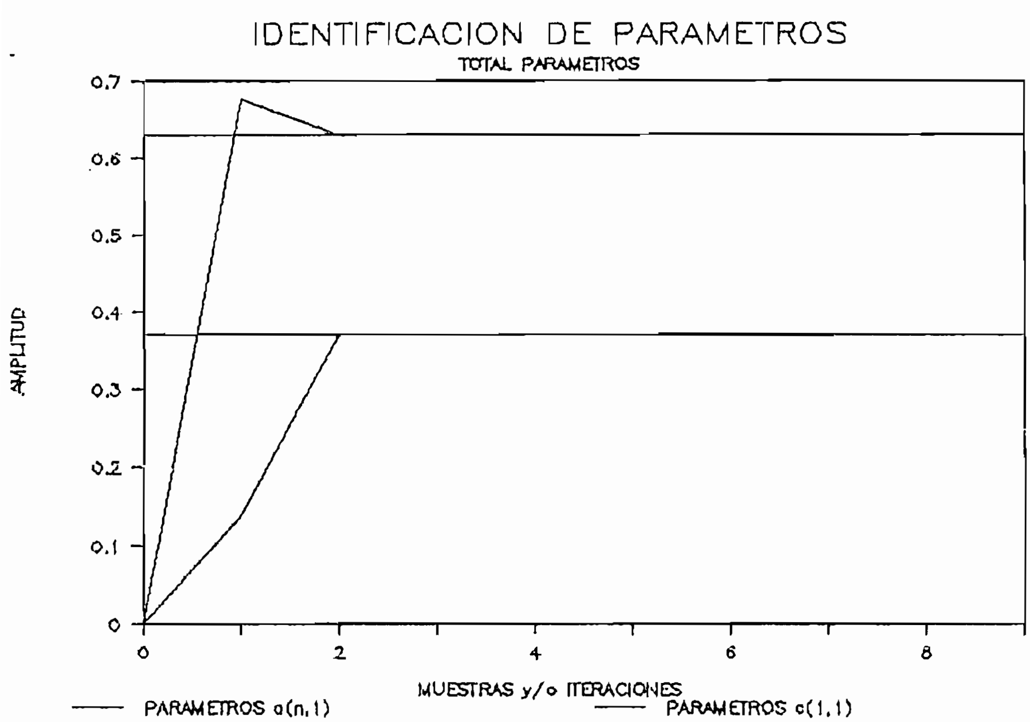


b)

Figura 5 a) Cuadro 8; b) Cuadro 9

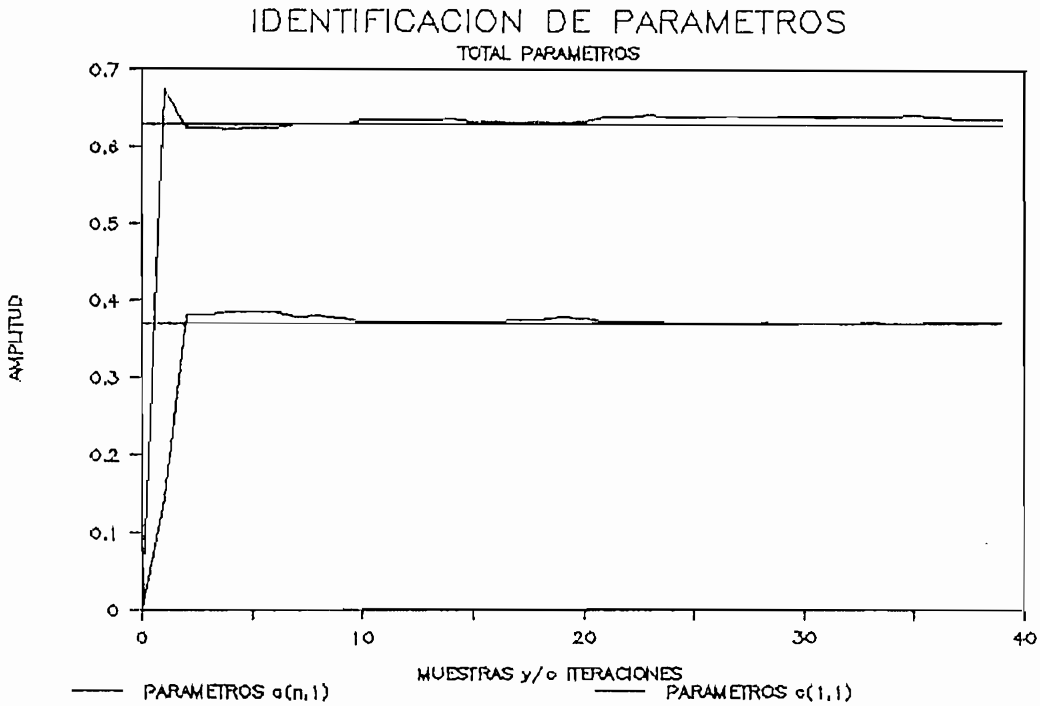


a)

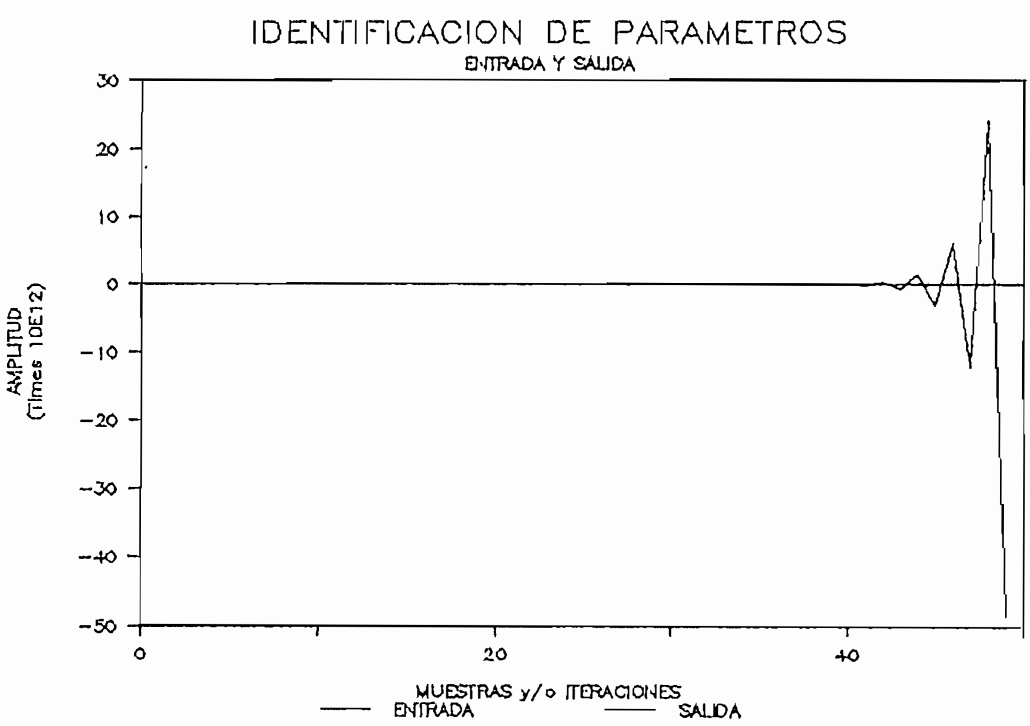


b)

Figura 6 a) Cuadro 10; b) Cuadro 11

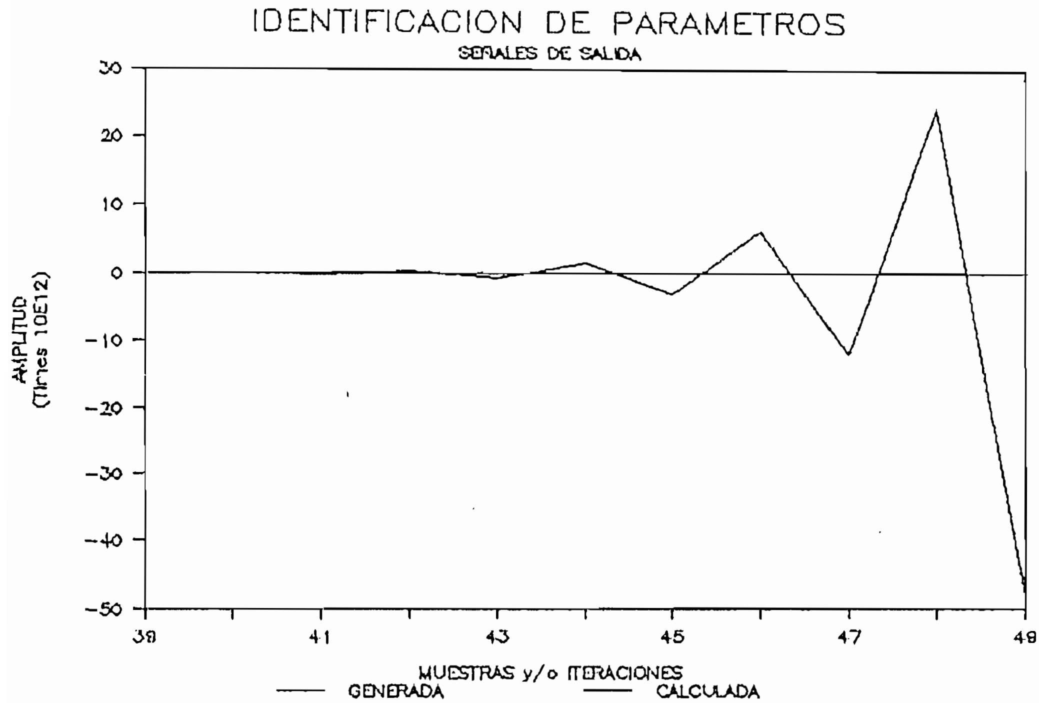


a)

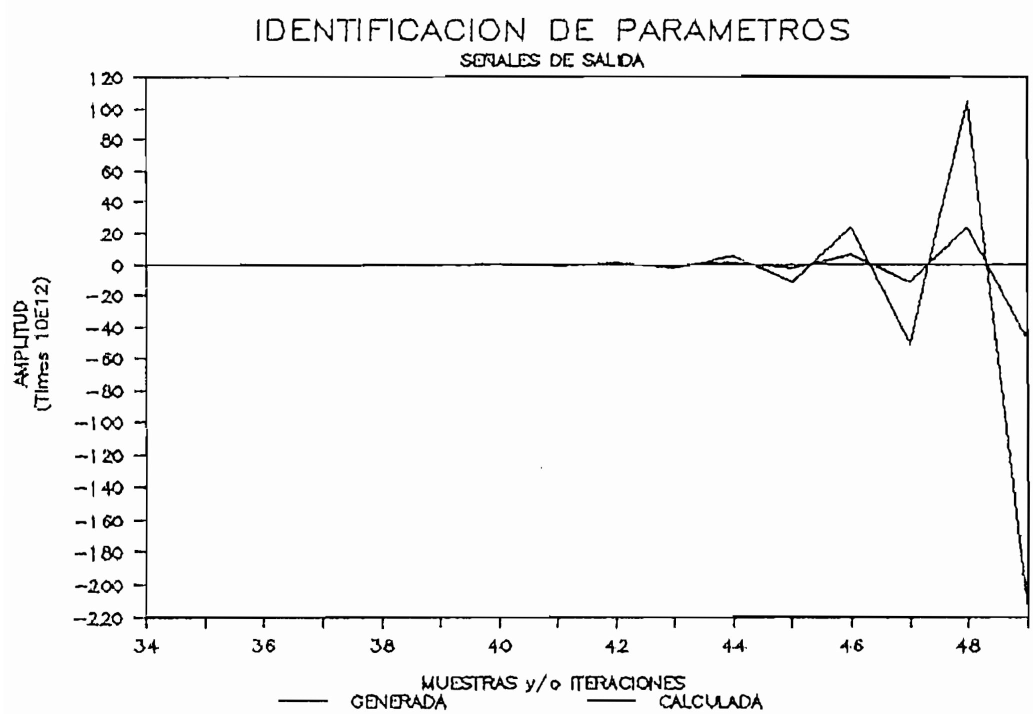


b)

Figura 7 a) Cuadro 12; b) Cuadro 13

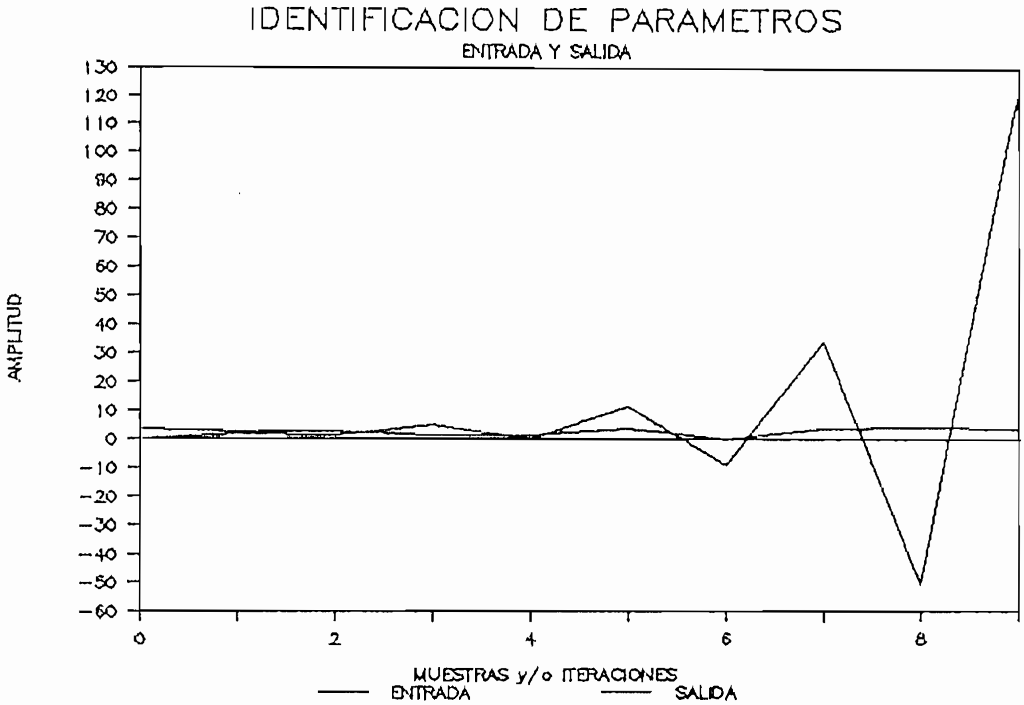


a)

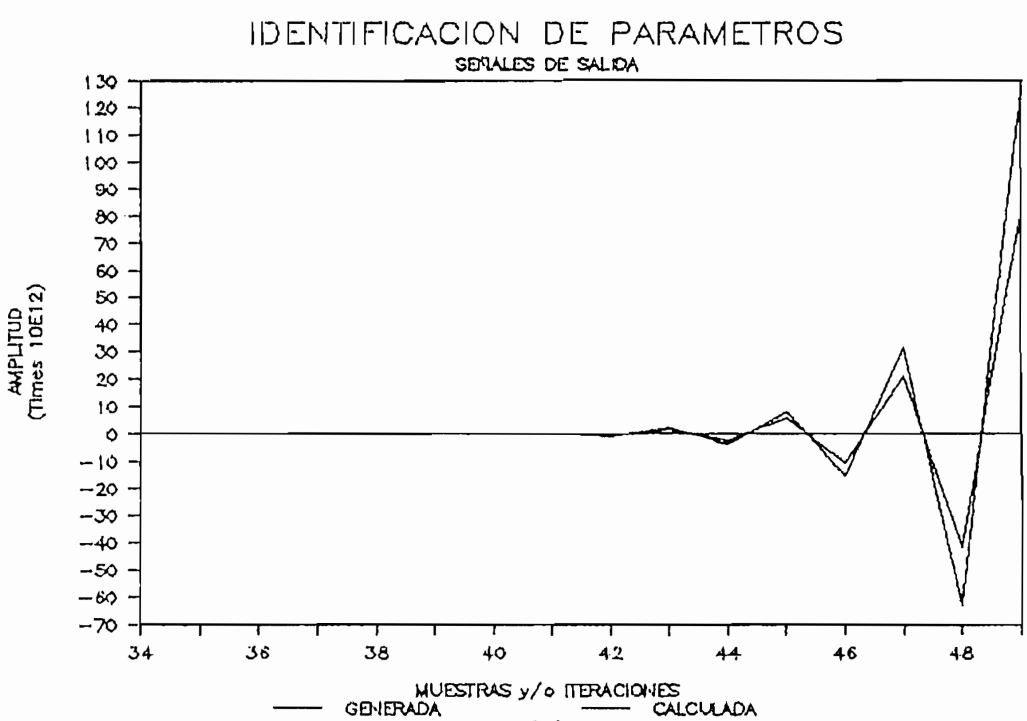


b)

Figura 8 a) Cuadro 14; b) Cuadro 15

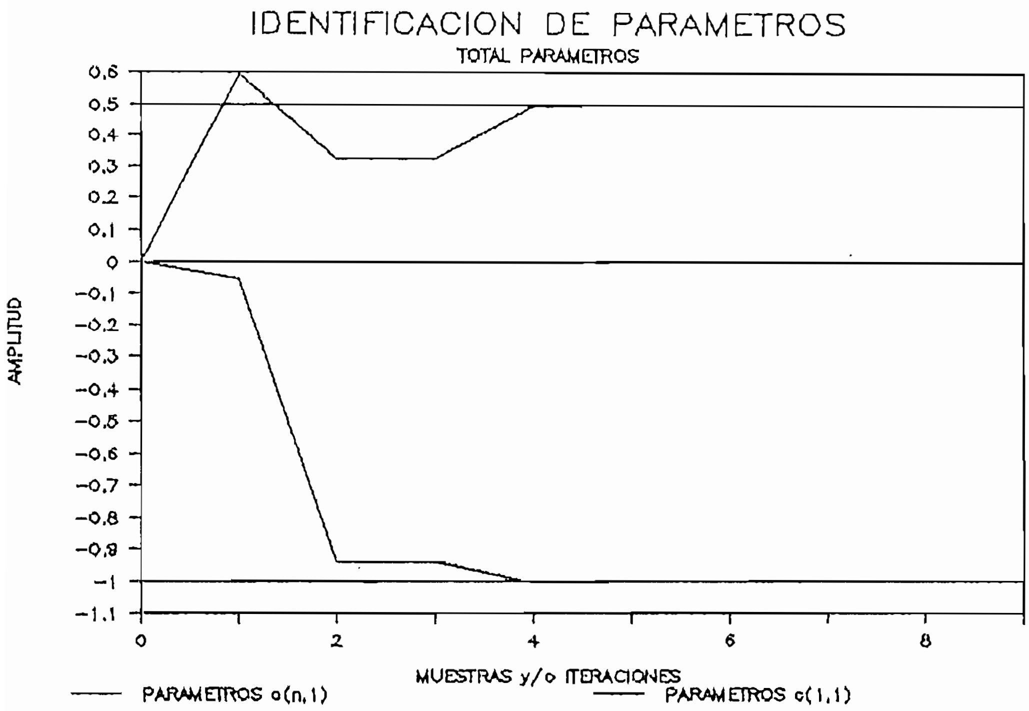


a)

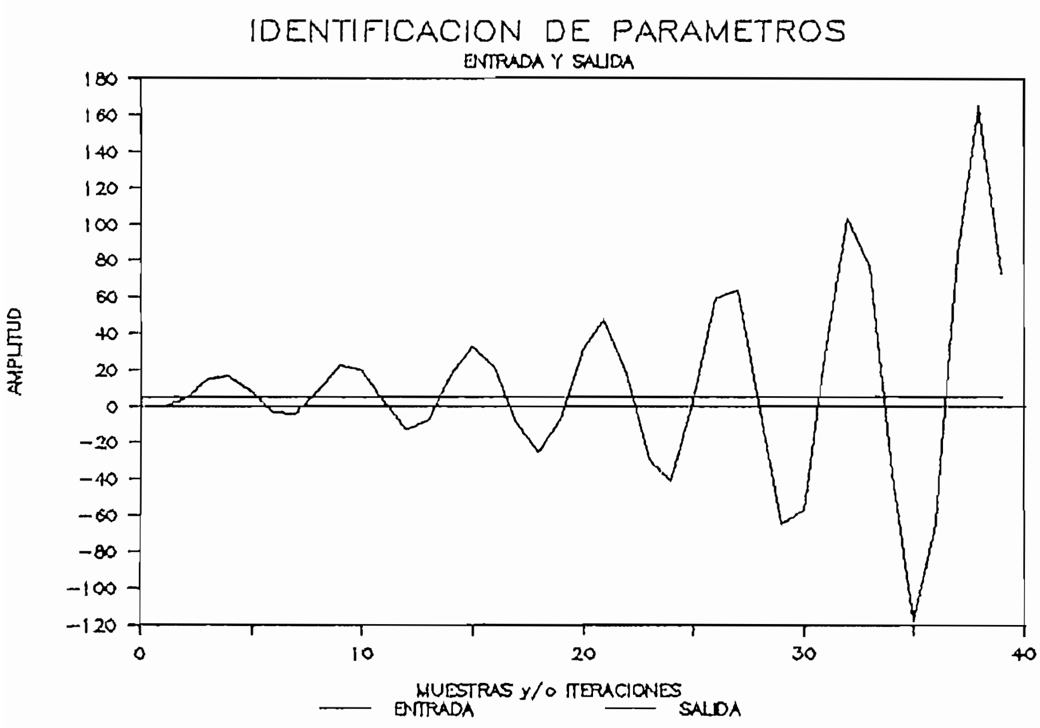


b)

Figura 9 a) Cuadro 16; b) Cuadro 17

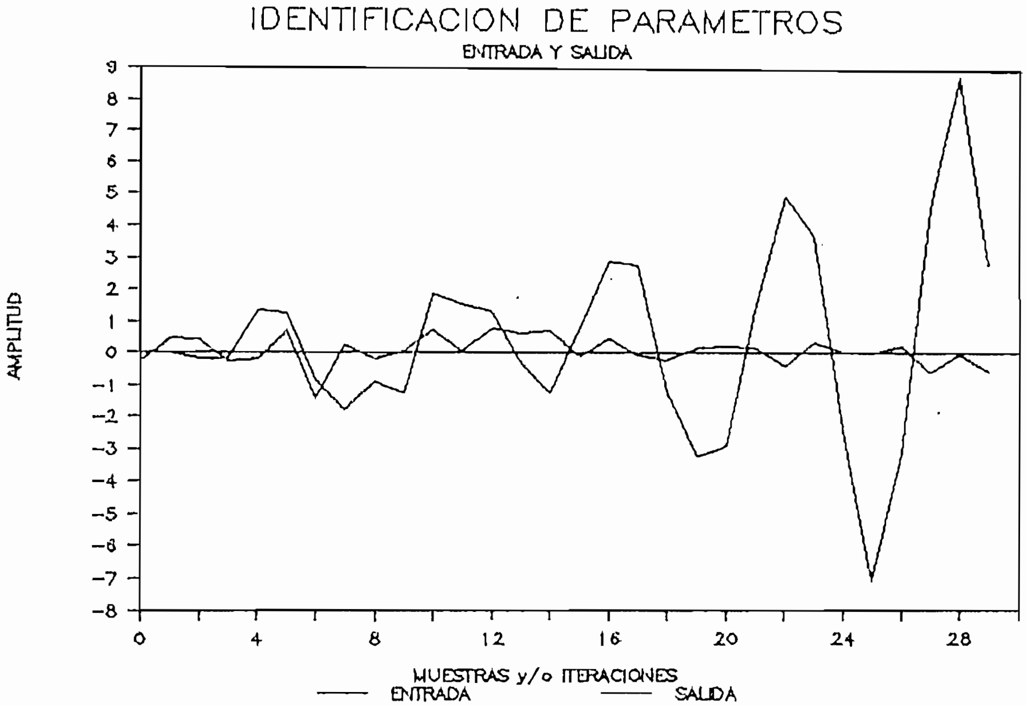


a)

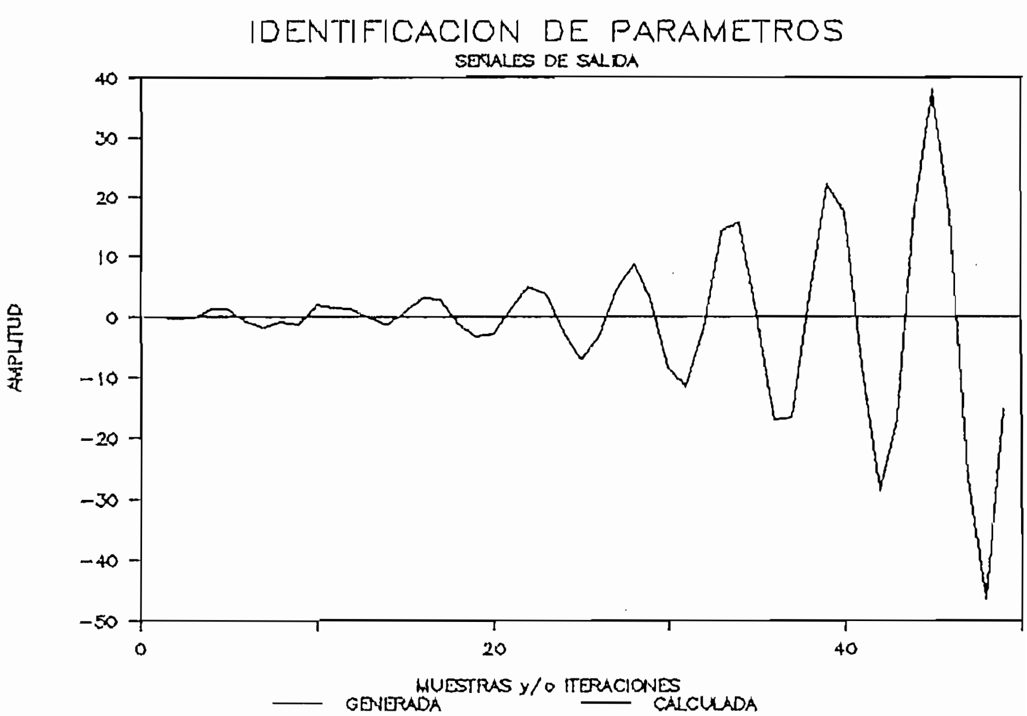


b)

Figura 10 a) Cuadro 18; b) Cuadro 19

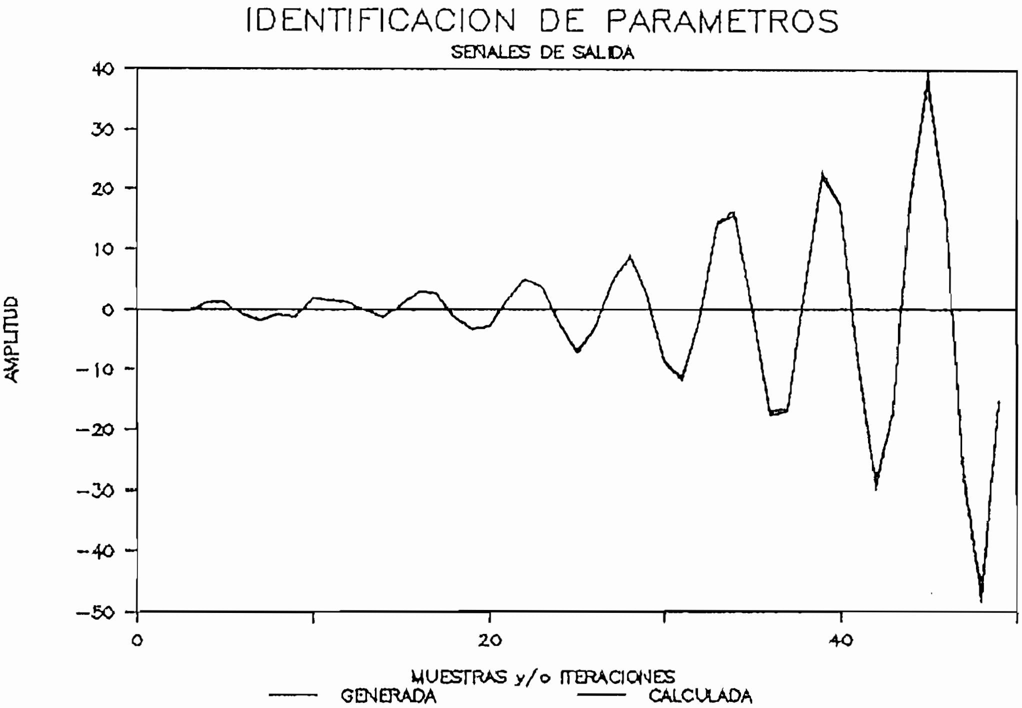


a)

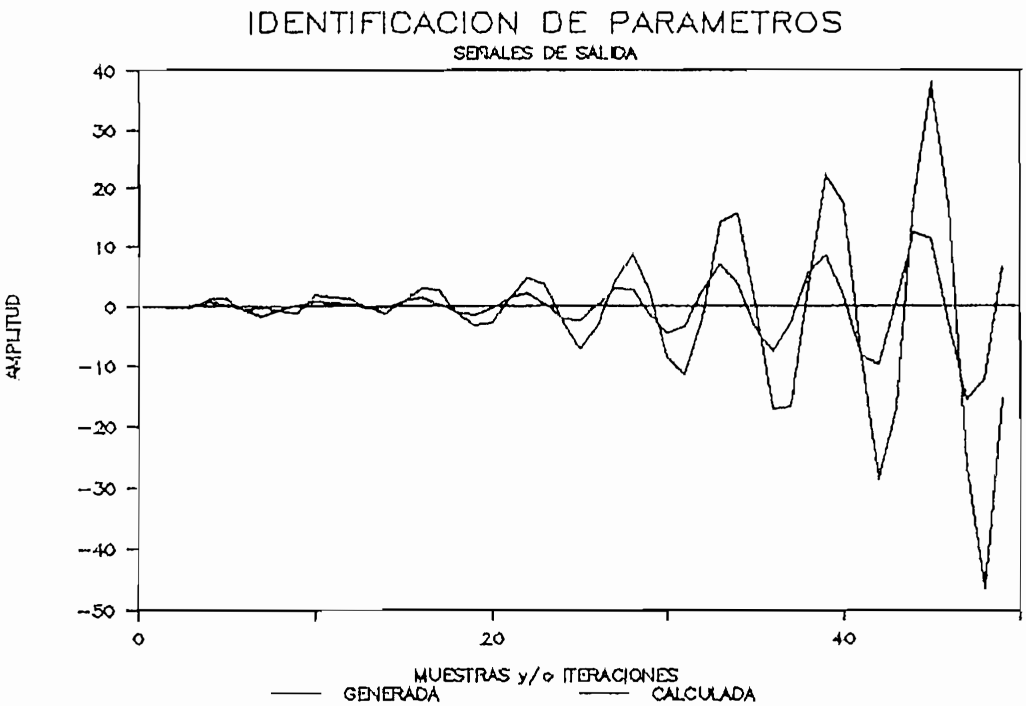


b)

Figura 11 a) Cuadro 20; b) Cuadro 21

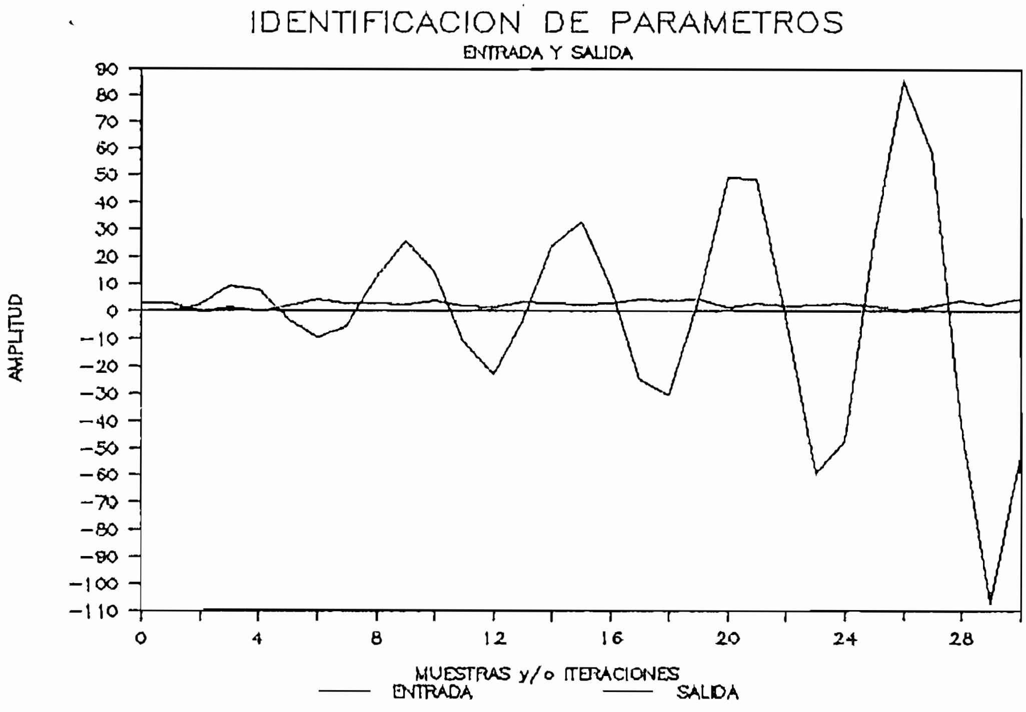


a)

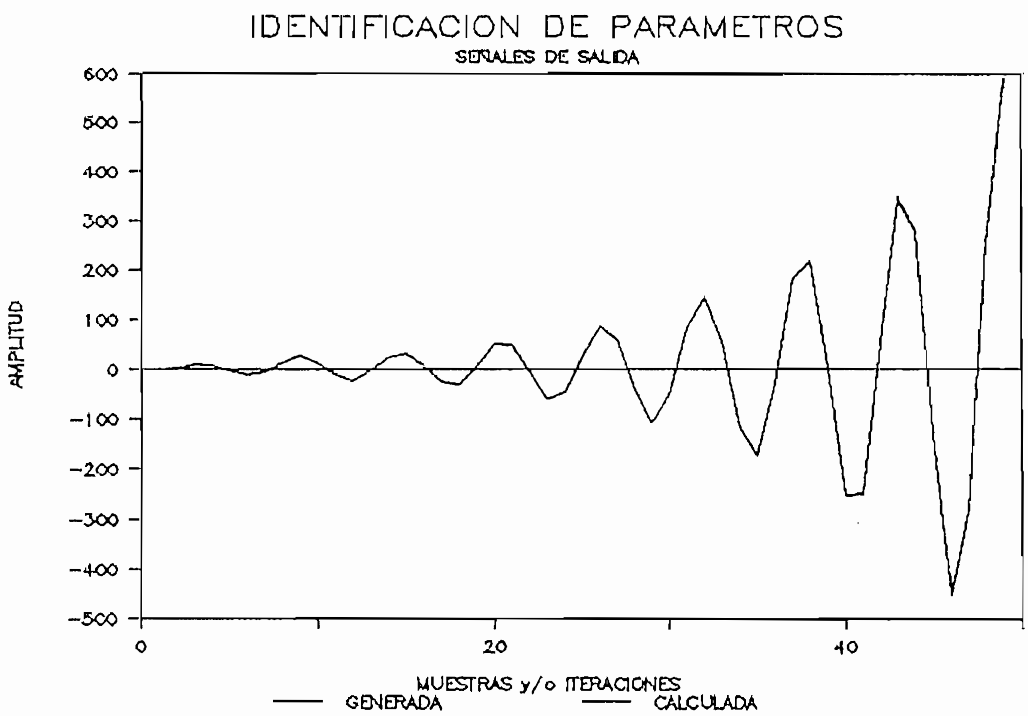


b)

Figura 12 a) Cuadro 22; b) Cuadro 23

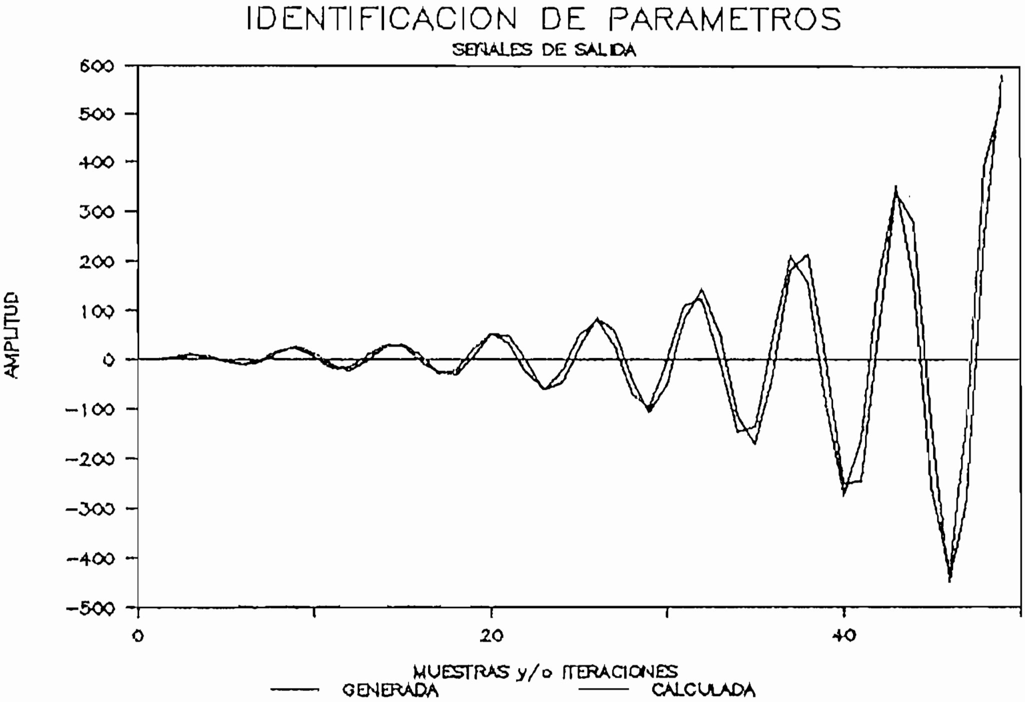


a)

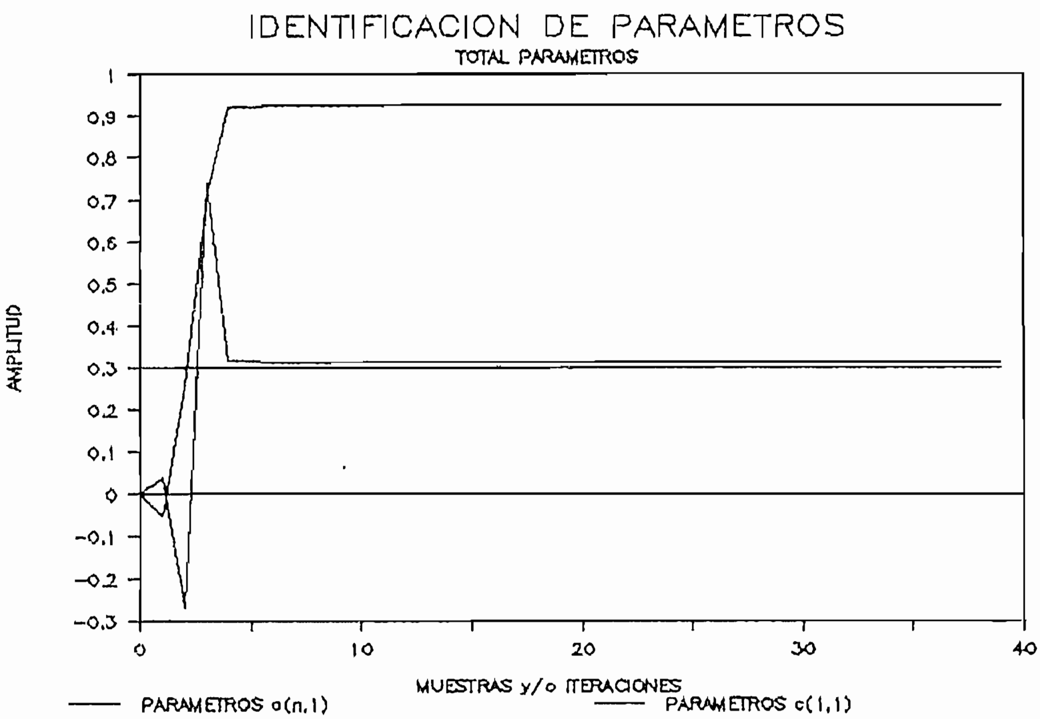


b)

Figura 13 a) Cuadro 24; b) Cuadro 25

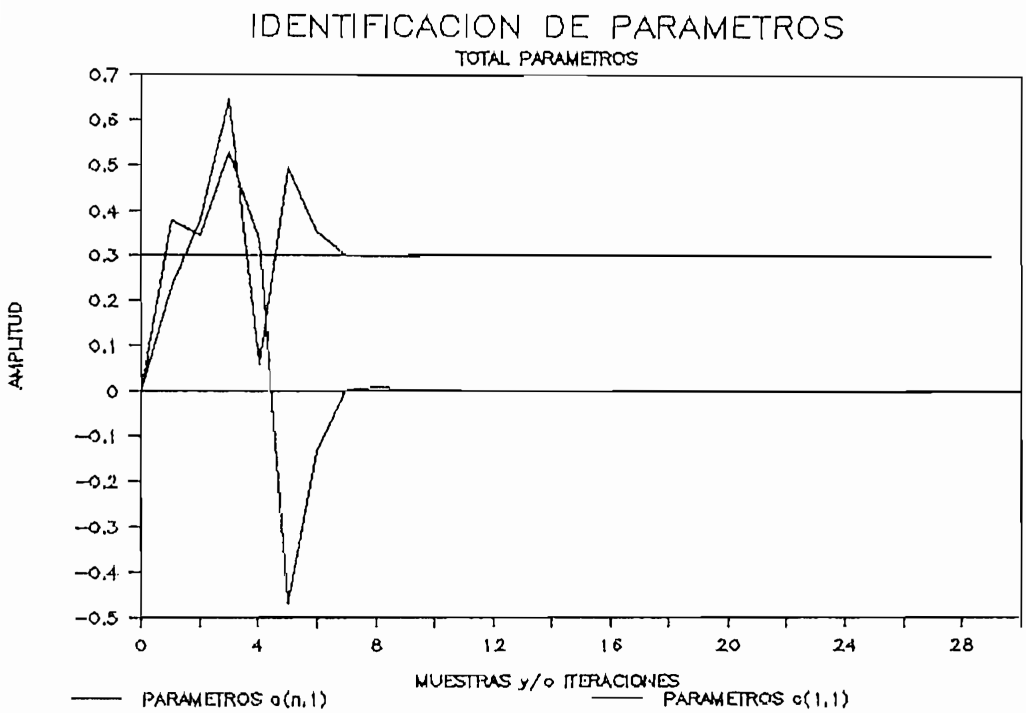


a)

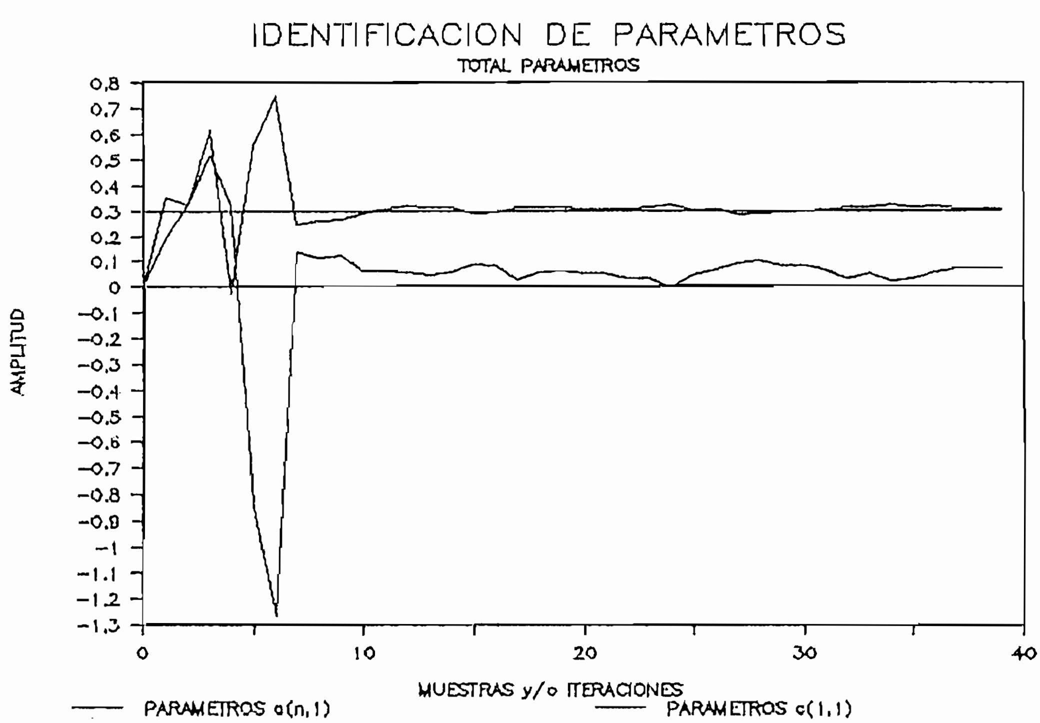


b)

Figura 14 a) Cuadro 26; b) Cuadro 27



a)



b)

Figura 15 a) Cuadro 28; b) Cuadro 29

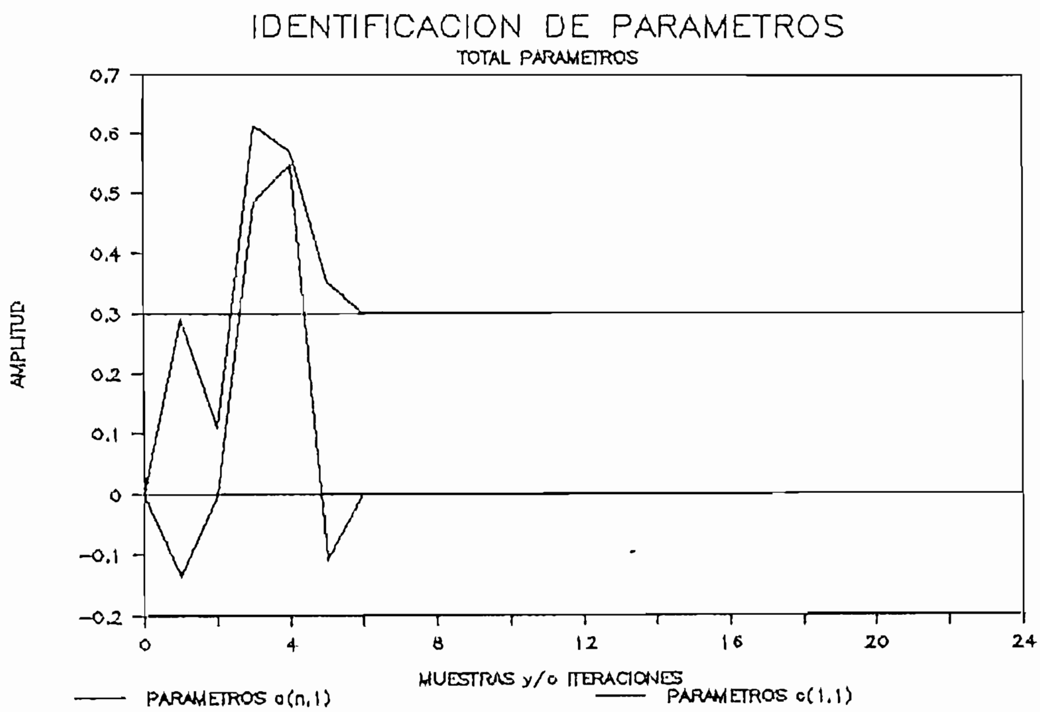


Figura 16 Cuadro 30

CAPITULO V

CONCLUSIONES

5.1 INTRODUCCION

En este capítulo, se establece el análisis y las conclusiones de los ejemplos desarrollados en el capítulo IV; de acuerdo al orden del sistema, al tipo de señal y al tipo de algoritmo utilizado.

5.2 SISTEMA DE PRIMER ORDEN

Para el sistema de primer orden se utiliza como señales de entrada ruido blanco con varianza = 1 y ruido aleatorio de magnitud = 5, (cuadro 1, 5; figura 1a, 3a gráficos de las señales de entrada y de salida) para los algoritmos de mínimos cuadrados ordinarios y recursivos, con o sin ruido aditivo a la señal de salida. Los resultados numéricos y gráficos, se encuentran indicados desde el cuadro 1, hasta el 12 y desde la figura 1 a) hasta la 7 a).

De los resultados numéricos y gráficos, se puede concluir los siguientes aspectos:

- Para el algoritmo de mínimos cuadrados ordinarios, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (no tienen sesgo) en forma rápida (cuadros 2, 6; figuras 1b, 3b) para las dos señales de entrada indicados. En los gráficos se observa que, las señales de salida generadas y calculadas se superponen.

- Para el algoritmo de mínimos cuadrados ordinarios, caso probabilístico, los parámetros estimados convergen a sus valores verdaderos, en forma lenta respecto al caso anterior; a mayor número de muestras mejor es la estimación (cuadros 3, 4, 7; figuras 2a, 2b, 4) para las dos señales de entrada indicadas. En los gráficos se observa que, las señales de salida generada y calculada tienden a superponerse.

- Para el algoritmo de mínimos cuadrados recursivos, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (no tienen sesgo) en forma rápida (cuadros 8, 9, 11; figuras 5a, 5b, 6b) para las dos señales de entrada indicadas. El algoritmo converge más rápidamente si el valor de la ponderación es 0.9 y si el valor de la condición inicial de la matriz de covarianza es 10.000.

En los gráficos se observa que, las valores de los parámetros generados y calculados son justos aproximadamente en la tercera iteración.

- Para el algoritmo de mínimos cuadrados recursivos, caso probabilístico, los parámetros estimados convergen a sus valores verdaderos, en forma lenta respecto al caso anterior (cuadros 10, 12; figuras 6a, 7a) para las dos señales de entrada indicadas; a pesar de presentar las mejores condiciones de convergencia, en lo referente a la condición inicial de la matriz de covarianza, y al valor de la ponderación.

  En los gráficos se observa que, las valores de los parámetros estimados tienden a los generados en un número alto de iteraciones.

5.3 SISTEMA DE SEGUNDO ORDEN

Para el sistema de segundo orden se utiliza como señales de entrada escalón de magnitud = 5, ruido blanco con varianza = 1 y una señal aleatoria de magnitud = 5, (cuadro 13, 16; figura 7b, 9a gráficos de las señales de entrada y de salida) para los algoritmos de mínimos cuadrados ordinarios y recursivos, con y sin ruido aditivo a la señal de salida. Los resultados numéricos y gráficos, se encuentran indicados desde el cuadro 13, hasta el 18 y desde la figura 7 b) hasta la 10 a).

De los resultados numéricos y gráficos, se puede concluir los siguientes aspectos:

- Para el algoritmo de mínimos cuadrados ordinarios, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (no tienen sesgo) en forma rápida (cuadros 14, 15; figuras 8a, 8b) con la señal de ruido blanco; a pesar de tratarse de un sistema inestable. La estimación se realiza, para un número pequeño de datos ya que al ser el sistema inestable, tomar más datos implica tener valores muy altos, lo que produciría la divergencia del algoritmo; en sistemas físicos se llegaría a un estado de saturación de los elementos.

  En el gráfico 8a) se observa que, la señal de salida generada y calculada se superponen; para la figura 8b), la señal de salida calculada tiende a la generada.

- Para el algoritmo de mínimos cuadrados ordinarios, caso probabilístico, los parámetros estimados convergen a sus valores verdaderos, en forma lenta respecto al caso anterior; con un número de muestras mínimo (cuadro 17; figura 9b), para la señal de entrada ruido aleatorio, por ser un sistema inestable. En el gráfico se observa que, la señal de salida generada, y calculada tienden a superponerse.

- Para el algoritmo de mínimos cuadrados recursivos, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (cuadro 18; figura 10a), para la señal de entrada aleatoria; con el menor número de muestras y con los mejores valores, en lo referente a la condición inicial de la matriz de covarianza, y al valor de la ponderación.

  En los gráficos se observa que, las valores de los parámetros estimados tienden a los generados en aproximadamente la sexta iteración.

5.4 SISTEMA DE TERCER ORDEN

Para el sistema de tercer orden se utilizó como señales de entrada: escalón de magnitud = 5, ruido blanco con varianza = 1 y señal aleatoria de magnitud = 5, (cuadro 19, 20, 24; figura 10b, 11a, 13a gráficos de las señales de entrada y de salida) para los algoritmos de mínimos cuadrados ordinarios y recursivos, con y sin ruido aditivo a la señal de salida. En forma adicional se presenta ejemplos equivalentes de menor orden.

Los resultados numéricos y gráficos, se encuentran indicados desde el cuadro 19, hasta el 31 y desde la figura 10 b) hasta la 16).

De los resultados numéricos y gráficos, se puede concluir los siguientes aspectos:

- Para el algoritmo de mínimos cuadrados ordinarios, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (no tienen sesgo) en forma rápida (cuadro 21; figura 11b) para la señal de entrada ruido blanco. En el gráfico se observa que, la señal de salida generada y calculada se superponen.

- Para el algoritmo de mínimos cuadrados ordinarios, caso probabilístico, los parámetros estimados convergen a sus valores verdaderos, en forma lenta respecto al caso anterior; con un número de muestras mínimo (cuadro 22, 25; figura 12a, 13b), para las señales de entrada ruido blanco y ruido aleatorio. En el gráfico se observa que, la señal de salida generada y calculada tienden a superponerse.

- Para el sistema de tercer orden y el algoritmo de mínimos cuadrados ordinarios casos determinístico y probabilístico, se cambia a segundo orden el sistema (cuadro 23, 26; figura 12b, 14a), y se obtiene ciertos parámetros estimados, los cuales son equivalentes a los de tercer orden, por cuanto la señal de salida calculada tiende a la señal de salida generada.

- Para el algoritmo de mínimos cuadrados recursivos, caso determinístico, los parámetros estimados convergen a sus valores verdaderos (cuadros 27, 28, 30, 31; figura 14b 15a, 16), de la siguiente manera:

  - Para la señal de entrada escalón, el algoritmo converge únicamente para los términos autoregresivos, diverge para los términos de media móvil.

  - Para las señales de entrada ruido blanco y ruido aleatorio; los valores estimados convergen rápidamente a los valores verdaderos, si se considera los mejores valores, en lo referente a la ponderación y al valor de la condición inicial de la matriz de covarianza.

En los gráficos se observa que, las valores de los parámetros estimados tienden a los generados en aproximadamente la décima iteración.

- Para el algoritmo de mínimos cuadrados recursivos, caso probabilístico, los parámetros estimados convergen a sus valores verdaderos (cuadro 29; figura 15b) en forma lenta.

- Para el sistema de tercer orden y el algoritmo de mínimos cuadrados recursivos se cambia a segundo orden el sistema (cuadro 31), y se obtiene ciertos parámetros estimados, los cuales dan una buena aproximación al sistema de tercer orden.

Del análisis y conclusiones particulares anteriormente señaladas, se puede concluir en forma general lo siguiente:

- Para el algoritmo de mínimos cuadrados ordinarios, los parámetros estimados son insesgados y convergen a los parámetros verdaderos en forma rápida, para cualquiera sea el orden del sistema, sea este estable o inestable, y tenga o no presencia de ruido aditivo en la señal de salida.

- Para el algoritmo de mínimos cuadrados recursivos, los parámetros estimados son insesgados y convergen a los parámetros verdaderos en forma más lenta que para el algoritmo de mínimos cuadrados ordinarios, para cualquiera sea el orden del sistema, sea estable o inestable, y tenga o no precensia de ruido aditivo en la señal de salida. La convergencia depende de la ponderación y de las condiciones iniciales adoptadas.

Concluyéndose que los mejores valores en la estimación de los parámetros es tener una ponderación menor que uno, dándose mayor importancia a los datos muestreados al final de un proceso y un valor de alfa alto lo que implića una mejor condición inicial.

- Para la señal de entrada escalón, el algoritmo de mínimos cuadrados ordinarios en los sistemas de segundo y tercer orden no converge, por cuanto las ecuaciones en sus términos de media móvil son linealmente dependientes; lo que permite concluir que con una señal de entrada constante, no se logra excitar a toda la dinámica de la planta, requiriéndose una señal de entrada persistentemente excitable.Para el caso del algoritmo de mínimos cuadrados recursivos únicamente se puede obtener una solución adecuada para los términos autoregresivos.

- De un modelo real, se puede obtener con una buena aproximación su equivalente de menor orden con los algoritmos de mínimos cuadrados ordinarios y recursivos.

ANEXO 1

1.- MANUAL DE OPERACION

En esta parte se establece una serie de lineamientos necesarios para la adecuada utilización de los programas implementados en esta tesis.

A.1 OBJETIVO

El programa de identificación de parámetros en variables de estado está constituido por cinco módulos que son:

1.- Programa Maestro (IPVE.EXE).

2.- Programa de Generación de los Datos (GENDATOS.EXE).

3.- Programa del Algoritmo de Mínimos Cuadrados Ordinarios (MICUAOR.EXE).

4.- Programa del Algoritmo de Mínimos Cuadrados Recursivos (MICUARE.EXE).

5.- Programa de Gráficos (GRAFICO.BAT).

Estos programas en conjunto, tienen la finalidad de calcular los parámetros de las matrices A, B, C de un sistema representado por variables de estado de la forma canónica controlable, a través de los algoritmos de mínimos cuadradados ordinarios y recursivos en sistemas determinísticos y probabilísticos. Los datos con los que se ejecutan estos algoritmos son generados por las señales de entrada escalón, ruido blanco y ruido aleatorio y los datos para la señal de salida con el modelo ARMA en variables de estado. Estas señales y parámetros son almacenados en el disco de datos que se seleccione, para su posterior graficación.

A.2 REQUERIMIENTOS DE HARDWARE Y SOFTWARE

Este programa fue desarrollado en un computador XT/compatible con 640 Kb de memoria RAM, una unidad de diskette de 5 1/4" de 360 Kb, una unidad de disco duro de 30 Mb y un monitor monocromático de alta resolución, conectado a una impresora de 80 columnas, 180 CPS (Caracteres Por Segundo), matriz de impresión de 9 pines con capacidad NLQ (Near Letter Quality).

Los requerimientos mínimos de hardware son:

- 512 Kb de memoria RAM.
- Dos unidades de diskette de 5 1/4" de 360 Kb.
- Monitor monocromático con opción de gráficos.
- Impresora matricial de 80 columnas.

Nota: El programa permite seleccionar entre monitor monocromático y color.

El programa se encuentra implementado en dos diskettes de 5 1/4" doble lado, doble densidad (360 Kb). Requiere para su ejecución de varios programas adicionales de interfase en el manejo de pantallas y gráficos, que son:

- KEY-FAKE.COM

  Programa utilitario de la PC Magazine, que permite almacenar en memoria la pulsación de una secuencia de teclas.

- BATH-ENHANCER.EXE

  Programa utilitario del NORTON UTILITY V4.5, que permite manejar comando ampliados del DOS.

- 123.EXE

  Programa del paquete LOTUS V2.01, que permite manejar macros de gráficos de datos.

- BRUN45.EXE

  Programa utilitario del lenguaje QBASIC V4.5, que permite correr el programa compilado y el manejo de errores.

El diskette No. 1 (PROGRAMAS) contiene los programas fuentes *.BAS, y ejecutables *.EXE desarrollados en lenguaje BASIC de la MICROSOFT y compilados con el mismo software.

Los archivos que debe disponer este diskette son los siguientes:

- KEY-FAKE.COM
- BE.EXE
- BRUN45.EXE
- IPVE.EXE, GENDATOS.EXE, MICUAOR.EXE, MICUARE.EXE (Programas ejecutables).
- IPVE.BAS, GENDATOS.BAS, MICUAOR.BAS, MICUARE.BAS (Programas fuentes).
- ID.BAT, GRAFICO.BAT
- ID.DAT, MENU.DAT

El diskette No. 2 (GRAFICOS) contiene los programas desarrollados en lenguaje de MACROS del LOTUS y el programa LOTUS V2.01, necesario para la ejecución del programa de gráficos.

Los archivos que debe disponer este diskette son los siguientes:

- KEY-FAKE.COM

- BE.EXE
- 123.EXE, 123.SET, 123.CMP, 123.CNF, 123COLOR.SET, 123MONO.SET
- GRAFICO.BAT
- GRAFICO.WK1
- MENU.DAT

A.3 DESCRIPCION DE LOS PROGRAMAS

ID.BAT

Es el programa inicial de arranque, tiene la finalidad de reservar un espacio de memoria RAM del computador para el utilitario KEY-FAKE.COM, y ejecutar el programa BE.EXE para utilizar los comandos ampliados del DOS. Se despliega en pantalla por vez única la pantalla de presentación de la tesis y se transfiere el control al programa IPVE.EXE.

ID.DAT

Este programa permite desplegar las ventanas de la presentación de la tesis, a través de comandos ampliados del DOS del programa BE.EXE.

IPVE.EXE

Programa maestro que despliega el menú principal y permite el enlace entre los diferentes programas.

GENDATOS.EXE

Programa de generación de los datos de las señales de entrada y de salida. La señal de entrada a generarse se puede seleccionar a partir de un menú y la señal de salida se genera a partir del modelo ARMA representado mediante variables de estado.

MICUAOR.EXE

Programa para calcular los parámetros de un modelo representado en variables de estado de la forma canónica controlable por el algoritmo de mínimos cuadrados ordinarios.

MICUARE.EXE

Programa para calcular los parámetros de un modelo representado en variables de estado de la forma canónica controlable por el algoritmo de mínimos cuadrados recursivos.

GRAFICO.BAT

Este programa permite enlazar a los programas ejecutables *.EXE con los programas gráficos desarrollados mediante macros de LOTUS.

Primero verifica si se encuentra en el drive A el diskette No. 2 de Gráficos y despliega la ventana de selección del tipo de monitor monocromático o color y si se desea graficar los resultados; a continuación se ejecuta el programa LOTUS cargando la hoja GRAFICO.WK1 que dispone de las macros. Este proceso es transparente al usuario, por cuanto el programa KEY-FAKE almacena una secuencia de teclas necesarias para realizar este proceso automáticamente.

MENU.DAT

Este programa permite desplegar la ventana de selección del tipo de monitor monocromático o color y la selección entre graficar o regresa al menú principal, a través de comandos ampliados del DOS del programa BE.EXE.

A.4 EJECUCION DEL PROGRAMA

Para la ejecución del programa seguir los siguientes pasos:

- Encender el computador y la impresora.

OPCION 1 MENU PRINCIPAL (DETERMINAR UNIDAD DE DATOS)

- Al correr por primera vez al programa, se debe ejecutar obligatoriamente la opción 1 del menú principal, lo que permite seleccionar la unidad del disco (A, B, C), en la que se va a almacenar los resultados.

En caso de seleccionar las opciones 2, 3, 4 antes de la opción 1, el programa despliega un mensaje " **No se conoce la unidad del disco de datos** " y regresa al menú principal.

De continuar en forma repetida la ejecución del programa, este asume la unidad previamente seleccionada.

<u>**OPCION 2 MENU PRINCIPAL (GENERACION DE DATOS)**</u>

- Ingresar el nombre del archivo de los datos, máximo pueden ser ocho caracteres; caso contrario se pide ingresar nuevamente el nombre del archivo. El programa automáticamente pone la extensión **.DIN** al archivo de datos generados.

- Si no existe el archivo de datos indicado, se procede a enlazar el programa del menú principal IPVE, con el programa de generación de datos GENDATOS.

  Si existe el archivo se puede seleccionar entre las dos opciones siguientes:

  1.- RECUPERAR LOS DATOS DESDE EL ARCHIVO.

  2.- REEMPLAZAR LOS DATOS.

Con la opción 1 se recupera los datos y se regresa al menú principal. Con la opción 2 se enlaza al programa de generación de datos GENDATOS.EXE.

- Ingresar el orden del modelo n, (0<n<=4), el valor de m (0<m<=n), los elementos de las matrices A, C (-4<=aij, cij<=4) de acuerdo a la forma canónica controlable y se generan los elementos 0 y 1 de las matrices A, B, C.

- Seleccionar el tipo de señal de entrada, del menú siguiente:

    1.- ENTRADA ESCALON
    2.- RUIDO BLANCO.
    3.- RUIDO.
    4.- SALIR.

    - Para la señal escalón, se debe ingresar la amplitud del escalón.
    - Para la señal de ruido blanco, se debe ingresa la varianza del ruido V>0.
    - Para la señal de ruido aleatorio se debe ingresar la amplitud del ruido.
    - La opción 4, regresa al programa del menú principal.

- Se ejecuta la generación de la señal de entrada y de la señal de salida, almacenando los resultados en el archivo *.DIN.

- Se despliega en pantalla los valores introducidos de n, m, valor de la señal de entrada (magnitud del escalón, varianza del ruido, magnitud del ruido) y las matrices A, B, C de la forma canónica controlable, indicándose si se desea imprimir los resultados antes de regresar al programa del menú principal.

OPCION 3 MENU PRINCIPAL (MINIMOS CUADRADOS ORDINARIOS)

- El programa de mínimos cuadrados ordinarios MICUAOR.EXE (ver diagrama de flujo Programa de Mínimos Cuadrados Ordinarios), verifica si se ha generado el archivo de datos, y llama al archivo de resultados de este algoritmo, con igual nombre de los archivos de datos generados cambiando únicamente la extensión con .DMO, teniéndose la opción de poder cambiar el nombre; pero la extensión será la misma.

- Seleccionar el tipo de proceso.

  1.- MINIMOS CUADRADOS ORDINARIOS DETERMINISTICOS.

2.- MINIMOS CUADRADOS ORDINARIOS PROBABILISTICOS.

3.- SALIR.

- Con la opción 1, el programa MICUAOR, ejecuta directamente el algoritmo.
- Con la opción 2, el programa MICUAOR, añade a la señal de salida un 10% de una señal de ruido.
- Con la opción 3, se regresa al programa del menú principal.

- Ingresar el nuevo valor de n, si se desea cambiar el orden del modelo.

- Ingresar el número de muestras, que es mayor que 2*N y menor de 50-2*N.

- Ejecutar el programa de mínimos cuadrados ordinarios y se almacena los resultados en el archivo *.DMO.

- Se despliega en pantalla los valores introducidos de n, m, número de muestras y las valores calculados de las matrices A, B, C, indicándose si se desea imprimir los resultados antes de pasar al programa del menú principal.

OPCION 4 MENU PRINCIPAL (MINIMOS CUADRADOS RECURSIVOS)

- El programa de mínimos cuadrados recursivos MICUARE.EXE (ver diagrama de flujo Programa de Mínimos Cuadrados Recursivos), verifica si se ha generado el archivo de datos, y llama al archivo de resultados de este algoritmo, con igual nombre de los archivos de datos generados cambiando únicamente la extensión con .DMR, teniéndose la opción de poder cambiar el nombre; pero la extensión será la misma.

- Seleccionar el tipo de proceso.

  1.- MINIMOS CUADRADOS RECURSIVOS DETERMINISTICOS.
  2.- MINIMOS CUADRADOS RECURSIVOS PROBABILISTICOS.
  3.- SALIR.

  - Con la opción 1, el programa MICUARE, ejecuta directamente el algoritmo.
  - Con la opción 2, el programa MICUARE, añade a la señal de salida un 10% de una señal de ruido.
  - Con la opción 3, se regresa al programa del menú principal.

- Ingresar el nuevo valor de n; si se desea cambiar el orden del modelo.

- Ingresar el número de muestras, que es mayor que 2*N y menor de 50-2*N.

- Ingresar la forma de la matriz de ponderación W.

  1.- **PONDERACION A=1; GAMMA=1.**
  2.- **PONDERACION EXPONENCIAL.**

  - Para la ponderación A=1, GAMMA= 1, se forma directamente la matriz de ponderación.
  - Para la ponderación exponencial se debe introducir el valor de gamma; 0<gamma<1.

- Fijar las condiciones iniciales de la matriz de covarianza P(N) y de la matriz de parámetros Theta.

  - Se considera la matriz de parámetros Theta igual a cero.
  - Para generar la condición inicial de la matriz de covarianza, ingresar el valor de alfa; 1000 <= alfa <=1´000.000.

- Definir el número de Iteraciones de ejecución del algoritmo; este valor tiene las misma restricciones del número de muestras seleccionadas.

- Ejecutar el programa de mínimos cuadrados recursivos y se almacena los resultados en al archivo *.DMR.

- Se despliega en pantalla los valores introducidos de n, m, número de muestras, número de iteraciones, valor de gamma, valor de alfa y las valores calculados de las matrices A, B, C, indicándose si se desea imprimir los resultados antes de pasar al programa del menú principal.

OPCION 5 MENU PRINCIPAL (GRAFICAR RESULTADOS)

- Al seleccionar esta opción, se sale momentáneamente de los programas desarrollados en Qbasic 4.5 al DOS, ejecutándose el archivo de lotes GRAFICO.BAT (ver diagrama de flujo GRAFICO.BAT).

- Se despliega en pantalla "Inserte el disco #2 (Gráficos)"; proceder a insertar el disco 2 y presionar cualquier tecla para continuar.

- Luego de verificarse si se trata del disco correcto, se despliega en pantalla el siguiente menú.

SELECCIONAR TIPO DE MONITOR

- MONOCROMATICO.
- COLOR.

- Al utilizar por primera vez el programa, seleccionar obligatoriamente el tipo de monitor que se dispone. De continuar en forma repetida la ejecución del programa, este asume el tipo de monitor previamente seleccionado.

SELECCIONAR TIPO DE PROCESO

- GRAFICAR.
- REGRESAR.

- Al seleccionar la opción Graficar, se carga el programa 123.EXE y a continuación arranca la macro de graficación de los resultados, la misma que se encuentra escrita en la hoja GRAFICO.WK1 (Ver diagrama de flujo programa ejecución gráficos).

- Seleccionar el tipo de proceso:

  - UNIDAD
  - SEÑALES GENERADAS
  - GRAFICOS ALGORITMO MCO
  - GRAFICOS ALGORITMO MCR
  - SALIR

- Para seleccionar una de las opciones del menú principal de Gráficos presionar las teclas de desplazamiento izquierda-derecha del cursor hasta localizarse sobre la opción requerida y presionar **ENTER**.

  - Al ejecutar por primera vez el programa de gráficos, seleccionar obligatoriamente la unidad del disco de datos, el programa tiene por definición inicial la unidad B:.
    En caso de necesitar seleccionar otra unidad pulsar la tecla **ESC** (**escape**), escribir [unidad]:\[camino], y presionar la tecla **ENTER**.
    Cada vez que se ingrese al programa de gráficos, se debe seleccionar la unidad y el directorio en el que se encuentran los datos.

  - Al seleccionar la opción **SEÑALES GENERADAS** se presentan en pantalla los archivos de datos

***.DIN** del directorio seleccionado. Posicionarse con el cursor en el archivo de datos requerido y presionar **ENTER**, desplegándose el siguiente submenú:

- **ENTRADA**

  Grafica la señal de entrada generada.
- **SALIDA**

  Grafica la señal de salida generada.
- **TOTAL**

  Grafica conjuntamente la señal de entrada y la señal de salida.
- **MENU**

  Permite regresar al menú anterior.

Despues de seleccionarse alguna de estas opciones, se presenta en pantalla el submenú siguiente:

- **TOTAL**

  Presenta en pantalla el gráfico completo
- **SEGMENTO**

  Presenta en pantalla un segmento del gráfico de acuerdo a la selección de un límite inferior y superior.

Para dejar de visualizar el gráfico en pantalla, presionar cualquier tecla, teniéndose el siguiente submenú:

- **CONTINUAR**

  Regresa al menú anterior sin grabar el gráfico.

- **ALMACENAR**

  Graba el gráfico de acuerdo al directorio y al nombre que seleccione como archivo de LOTUS *.PIC, el mismo que posteriormente puede ser impreso con el programa Print Graph.

- Al seleccionar la opción **GRAFICOS ALGORITMO MCO** se presentan en pantalla los archivos de datos ***.DMO** del directorio seleccionado; posicionarse con el cursor en el archivo de datos requerido y presionar **ENTER**, desplegándose el siguiente submenú:

  - **SALIDA GENERADA**

    Grafica la señal de salida generada, mediante el programa GENDATOS.EXE.

  - **SALIDA CALCULADA**

Grafica la señal de salida calculada mediante el algoritmo de Mínimos Cuadrados Ordinarios.

- **TOTAL**

  Grafica conjuntamente las señales de salida generada y calculada.

- **MENU**

  Permite regresar al menú anterior.

Despues de seleccionarse alguna de estas opciones, se presenta en pantalla el submenú siguiente:

- **TOTAL**

  Presenta en pantalla el gráfico completo

- **SEGMENTO**

  Presenta en pantalla un segmento del gráfico de acuerdo a la selección de un límite inferior y superior.

Para dejar de visualizar el gráfico en pantalla, presionar cualquier tecla, teniéndose el siguiente submenú:

- **CONTINUAR**

Regresa al menú anterior sin grabar el gráfico.

- **ALMACENAR**

  Graba el gráfico de acuerdo al directorio y al nombre que seleccione como archivo de LOTUS *.PIC, el mismo que posteriormente puede ser impreso con el programa Print Graph.

- Al seleccionar la opción **GRAFICOS ALGORITMO MCR** se presentan en pantalla los archivos de datos ***.DMR** del directorio seleccionado, posicionarse con el cursor en el archivo de datos requerido y presionar **ENTER**, desplegándose el siguiente submenú:

  - **PARAMETROS GENERADOS**

    Grafica el valor de los parámetros a(n,1) y c(1,1) introducidos en el programa GENDATOS.EXE.

  - **PARAMETROS CALCULADOS**

    Grafica el valor de los parámetros a(n,1) y c(1,1) calculados mediante el algoritmo de Mínimos Cuadrados Recursivos.

  - **TOTAL**

Grafica conjuntamente las señales de salida generada y calculada.

- **MENU**

  Permite regresar al menú anterior.

Despues de seleccionarse alguna de estas opciones, se presenta en pantalla el submenú siguiente:

- **TOTAL**

  Presenta en pantalla el gráfico completo

- **SEGMENTO**

  Presenta en pantalla un segmento del gráfico de acuerdo a la selección de un límite inferior y superior.

Para dejar de visualizar el gráfico en pantalla, presionar cualquier tecla, teniéndose el siguiente submenú:

- **CONTINUAR**

  Regresa al menú anterior sin grabar el gráfico.

- **ALMACENAR**

  Graba el gráfico de acuerdo al directorio y al nombre que seleccione como archivo de LOTUS *.PIC, el mismo que posteriormente puede ser impreso con el programa Print Graph.

- Al seleccionar la opción **SALIR** se regresa al programa GRAFICO.BAT.

- Seleccionar la opción **REGRESAR** y se retorna al programa del menú principal.

De presentárse algún error en la graficación, existe macros de manejo de errores que permite reestablecer el proceso; de persistir el error, presionar la tecla ESC hasta que desaparesca la línea de menús y presionar las teclas **ALT A**.

<u>OPCION 6 MENU PRINCIPAL (SALIR AL SISTEMA OPERATIVO)</u>

Termina la ejecución del programa y retorna al sistema operativo.

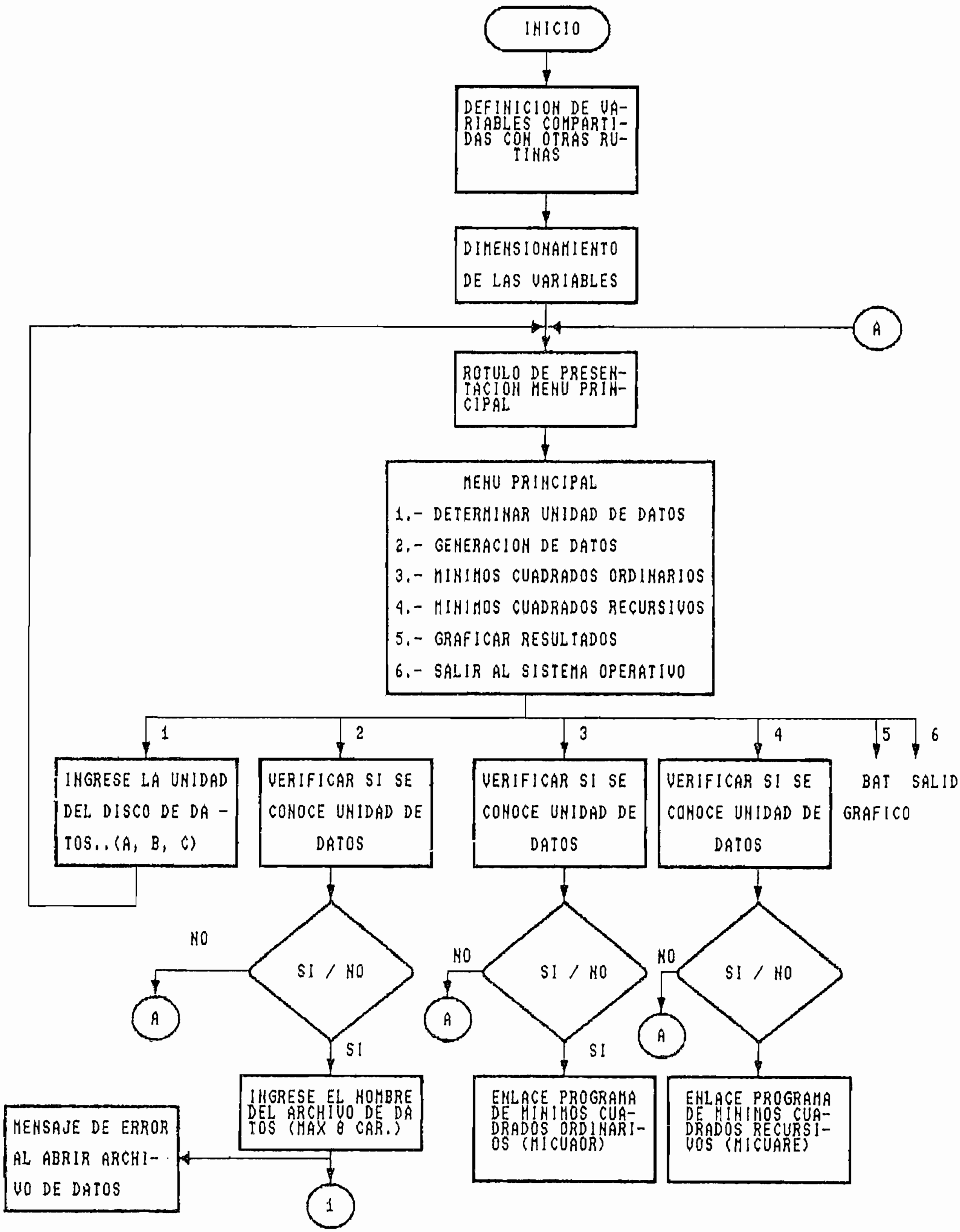
En cada uno de los programas desarrollados se han incluido rutinas de manejo de errores, los mismos que despliegan un mensaje de error explicativo y el código del error, el mismo que se puede consultar en el manual **BASIC LANGUAGE REFERENCE** (Microsoft Corporation 1987, p. 476).

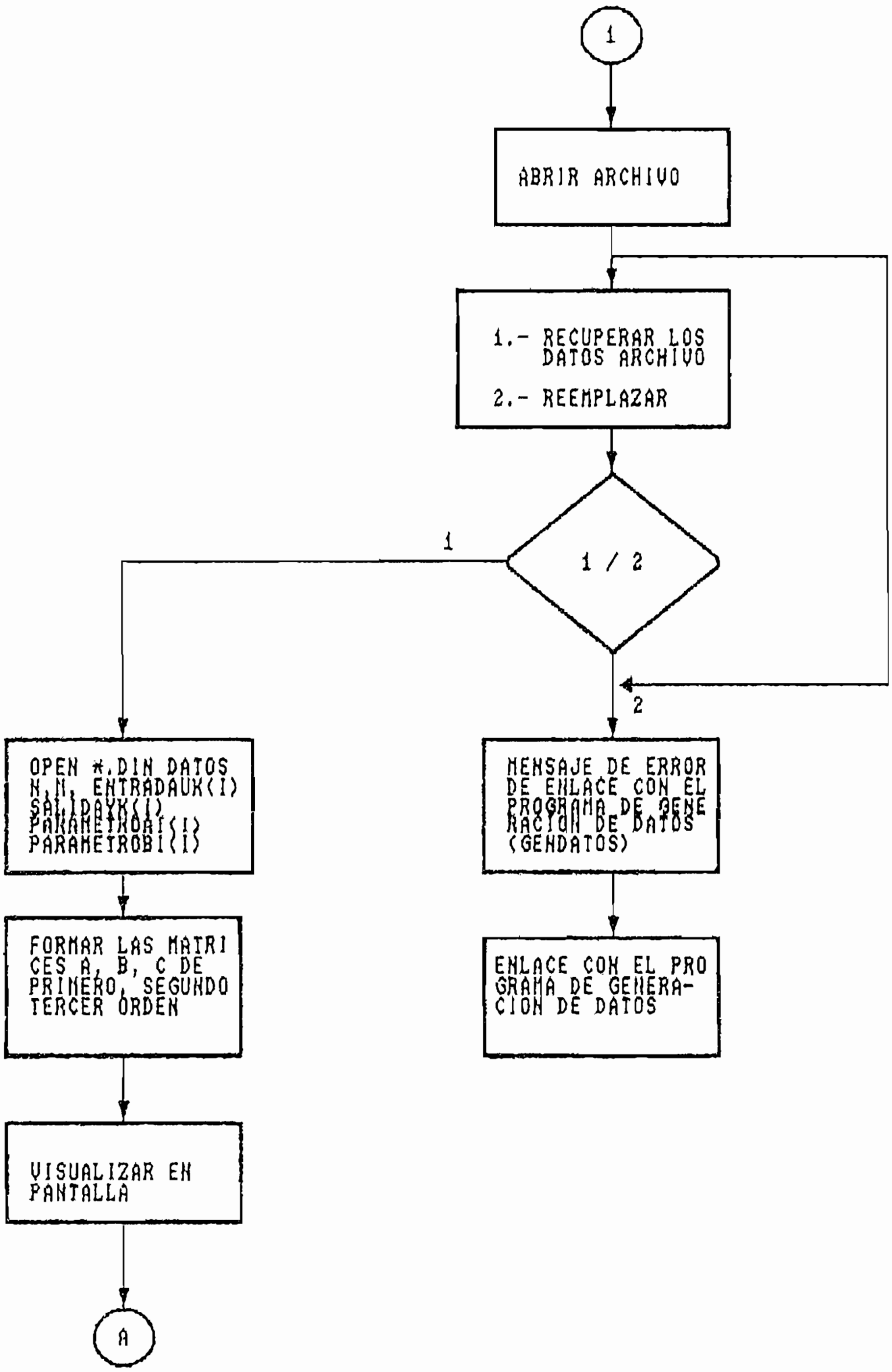
DIAGRAMAS DE FLUJO

2.-

IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO

DIAGRAMA DE FLUJO PROGRAMA MAESTRO




1
ABRIR ARCHIVO
1.- RECUPERAR LOS DATOS ARCHIVO
2.- REEMPLAZAR
1 / 2
1
2
OPEN *.DIN DATOS N,M, ENTRADAUK(I) SALIDAYK(I) PARAMETROAI(I) PARAMETROBI(I)
MENSAJE DE ERROR DE ENLACE CON EL PROGRAMA DE GENERACION DE DATOS (GENDATOS)
FORMAR LAS MATRICES A, B, C DE PRIMERO, SEGUNDO TERCER ORDEN
ENLACE CON EL PROGRAMA DE GENERACION DE DATOS
VISUALIZAR EN PANTALLA
A

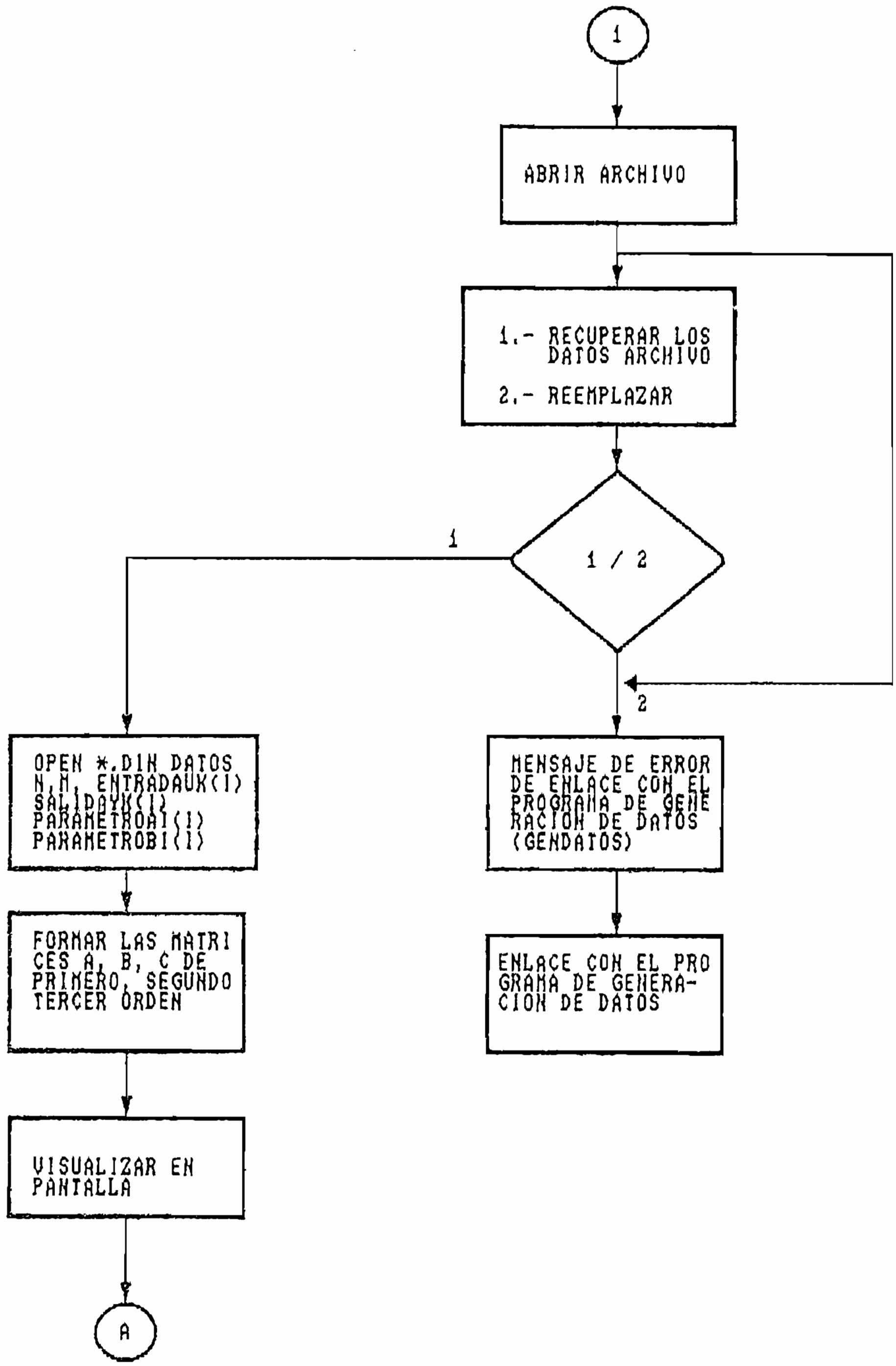
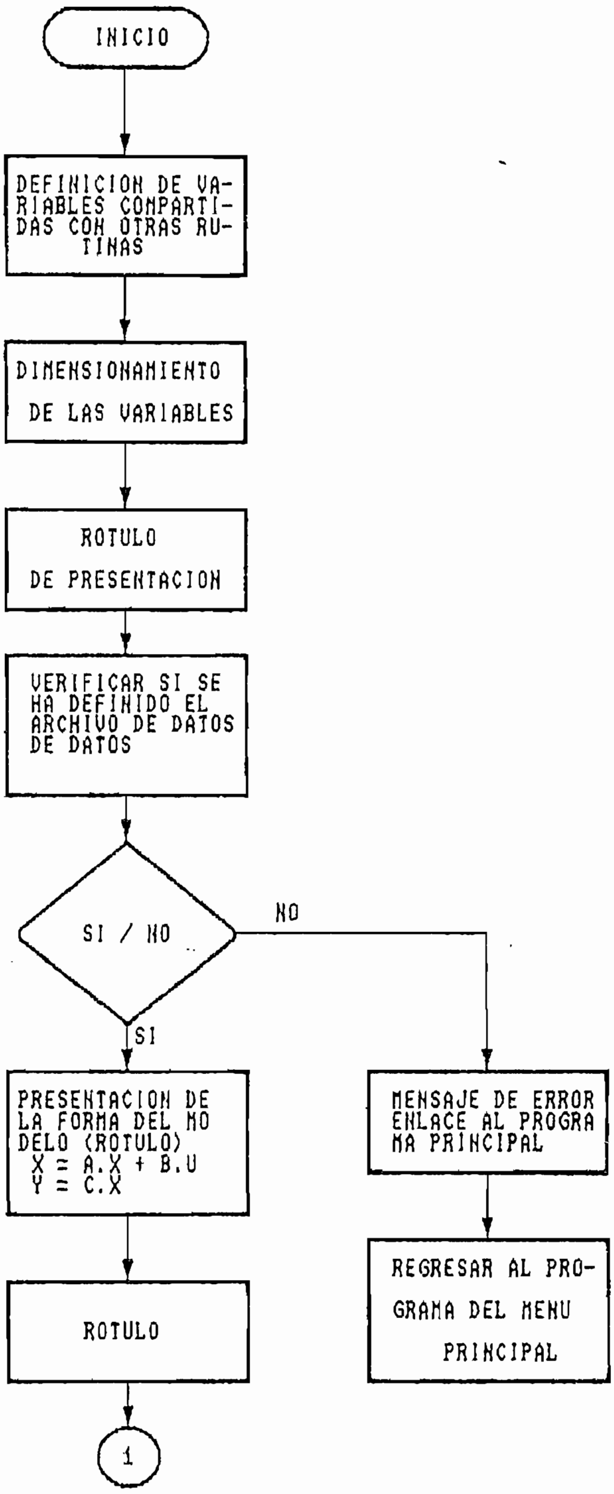
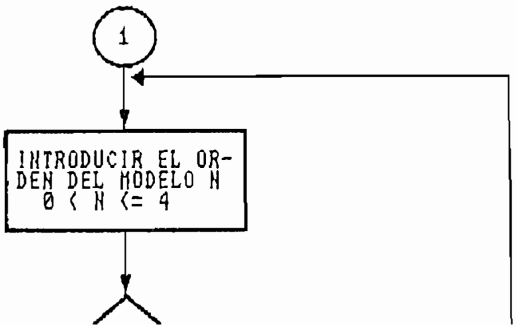
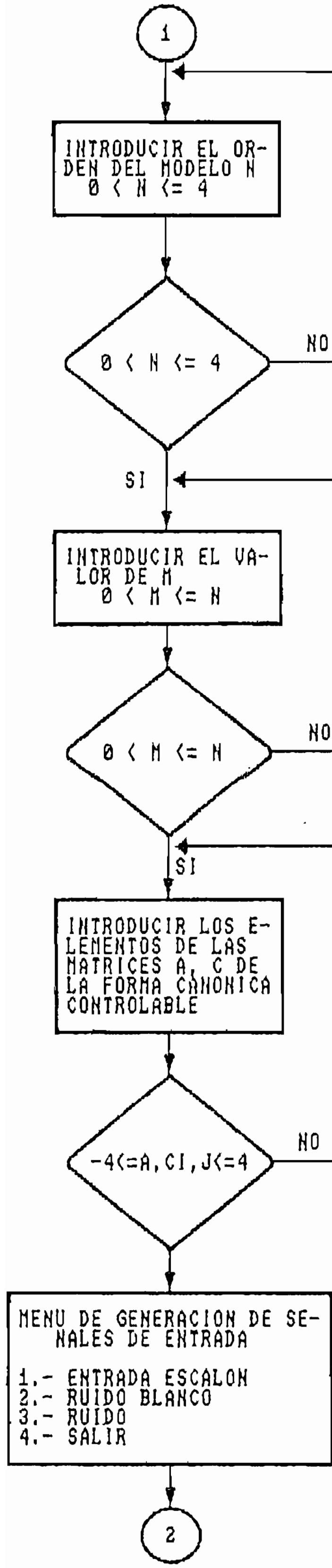
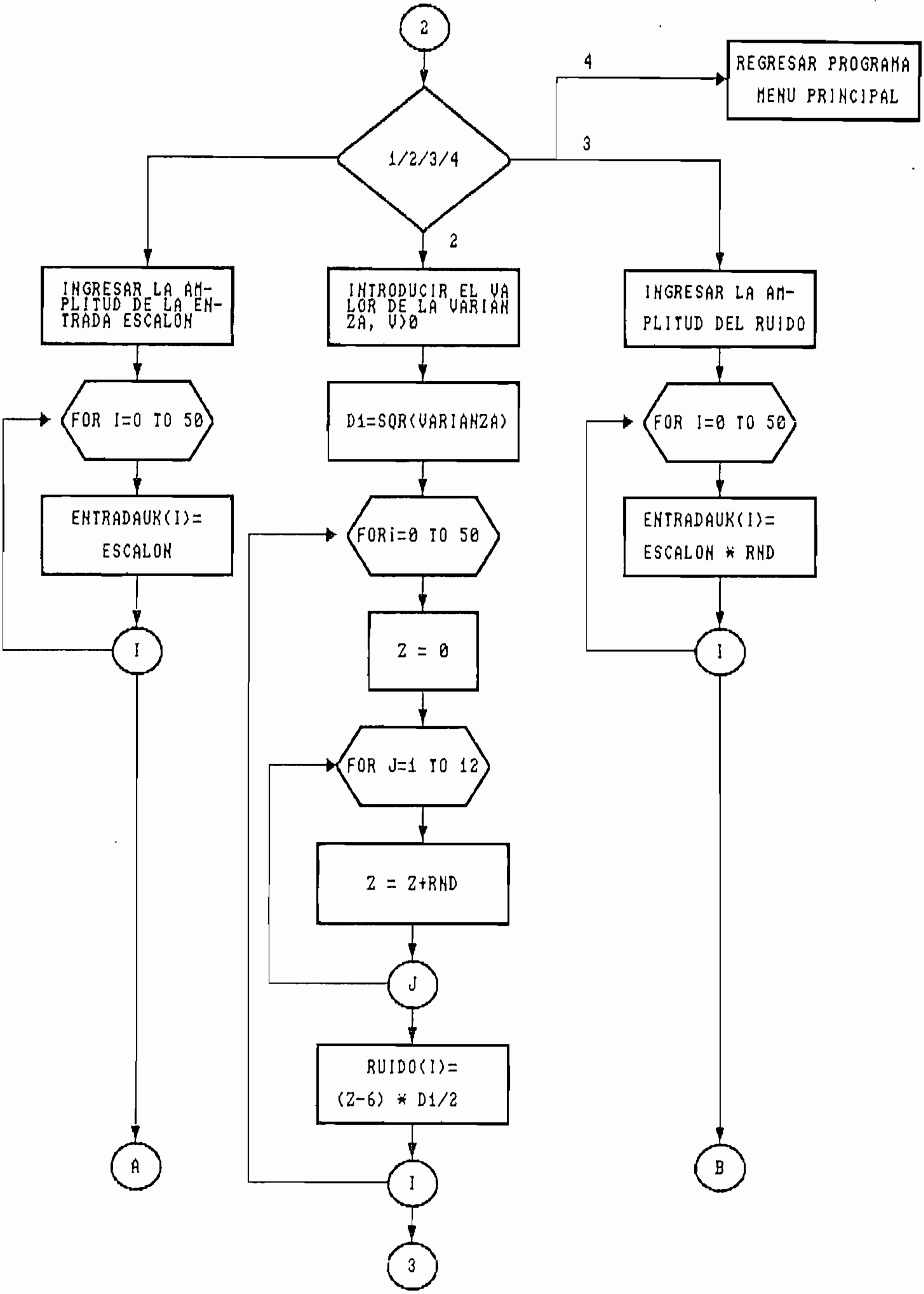

1
ABRIR ARCHIVO
1.- RECUPERAR LOS DATOS ARCHIVO
2.- REEMPLAZAR
1 / 2
1
2
OPEN *.DIN DATOS N,M, ENTRADAUK(I) SALIDAYK(I) PARAMETROAI(I) PARAMETROBI(I)
FORMAR LAS MATRICES A, B, C DE PRIMERO, SEGUNDO TERCER ORDEN
VISUALIZAR EN PANTALLA
A
MENSAJE DE ERROR DE ENLACE CON EL PROGRAMA DE GENERACION DE DATOS (GENDATOS)
ENLACE CON EL PROGRAMA DE GENERACION DE DATOS

DIAGRAMA DE FLUJO PROGRAMA DE GENERACION DE DATOS

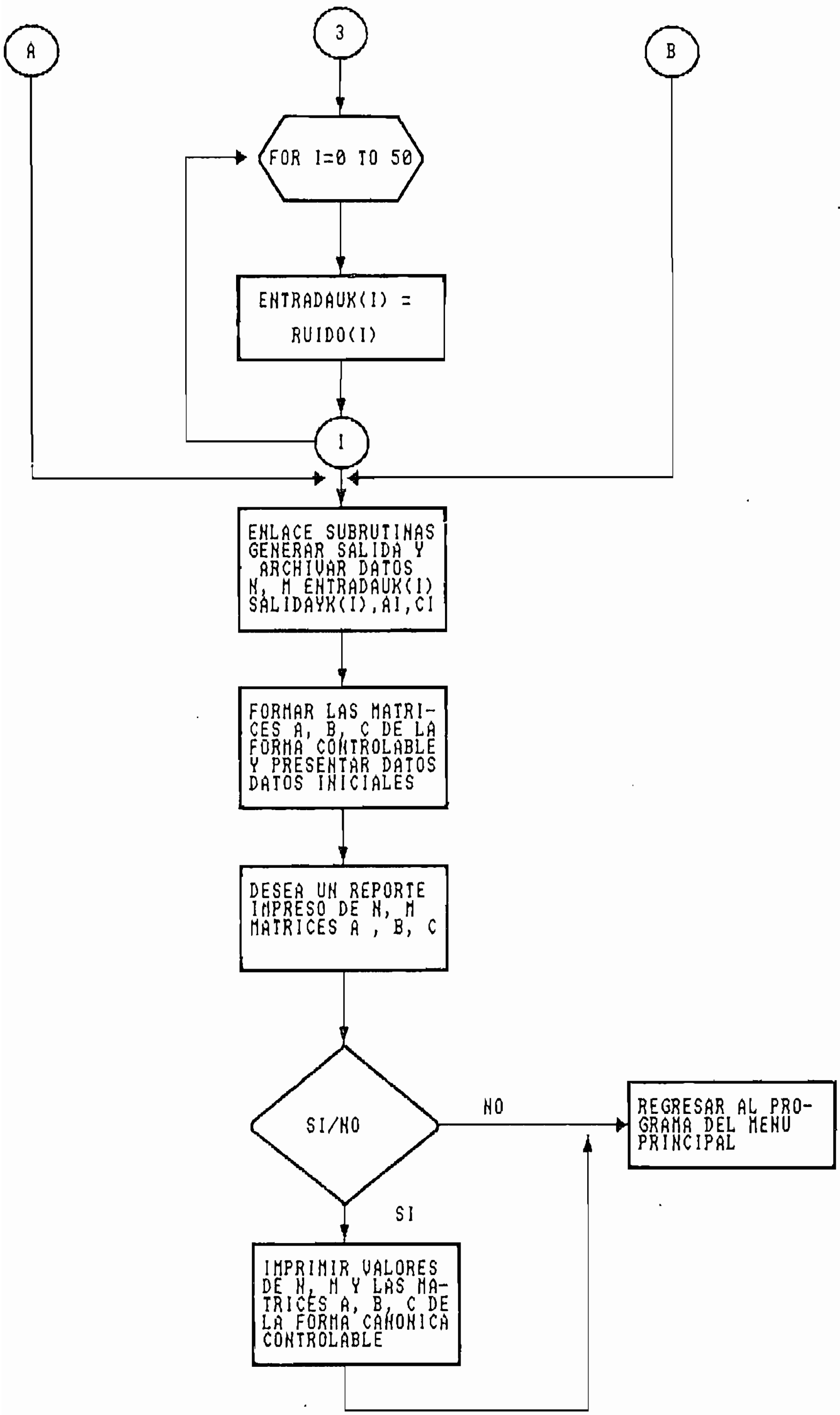




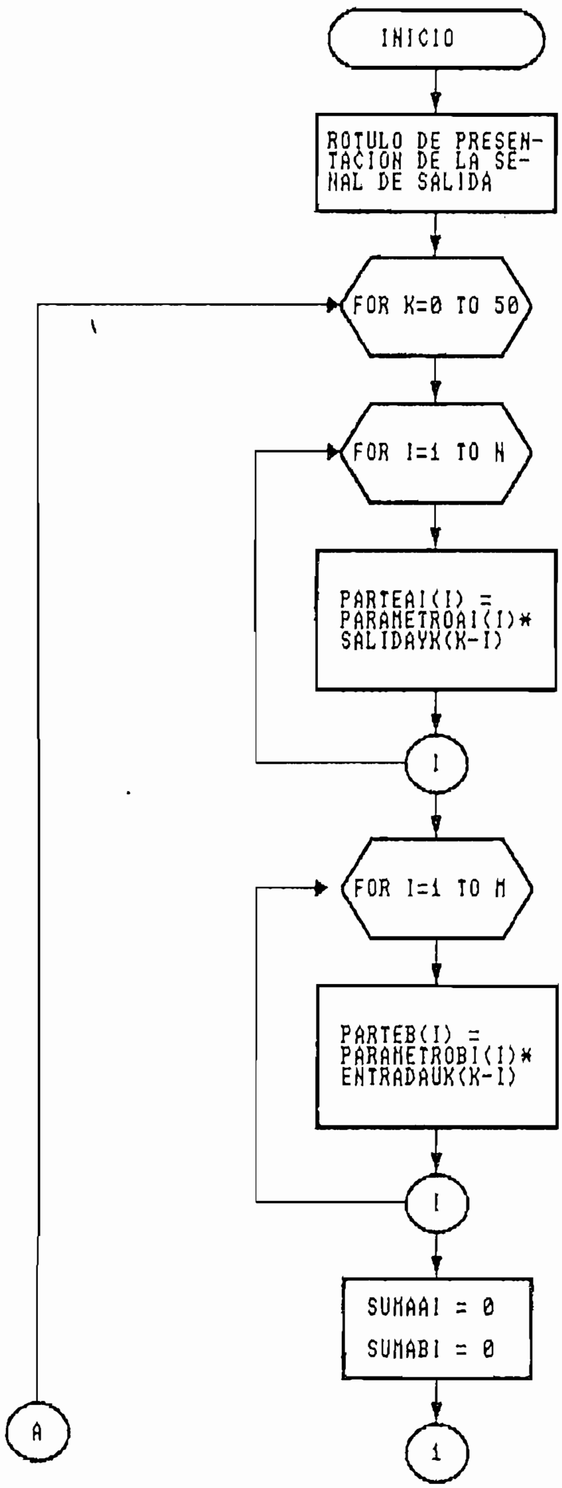

1
INTRODUCIR EL ORDEN DEL MODELO N
0 < N <= 4
0 < N <= 4
NO
NO SE PUEDE GENERAR EL PROCESO
SI
INTRODUCIR EL VALOR DE M
0 < M <= N
0 < M <= N
NO
REPETIR EL VALOR DE M
SI
INTRODUCIR LOS ELEMENTOS DE LAS MATRICES A, C DE LA FORMA CANONICA CONTROLABLE
-4<=A,CI,J<=4
NO
MENU DE GENERACION DE SEÑALES DE ENTRADA
1.- ENTRADA ESCALON
2.- RUIDO BLANCO
3.- RUIDO
4.- SALIR
2



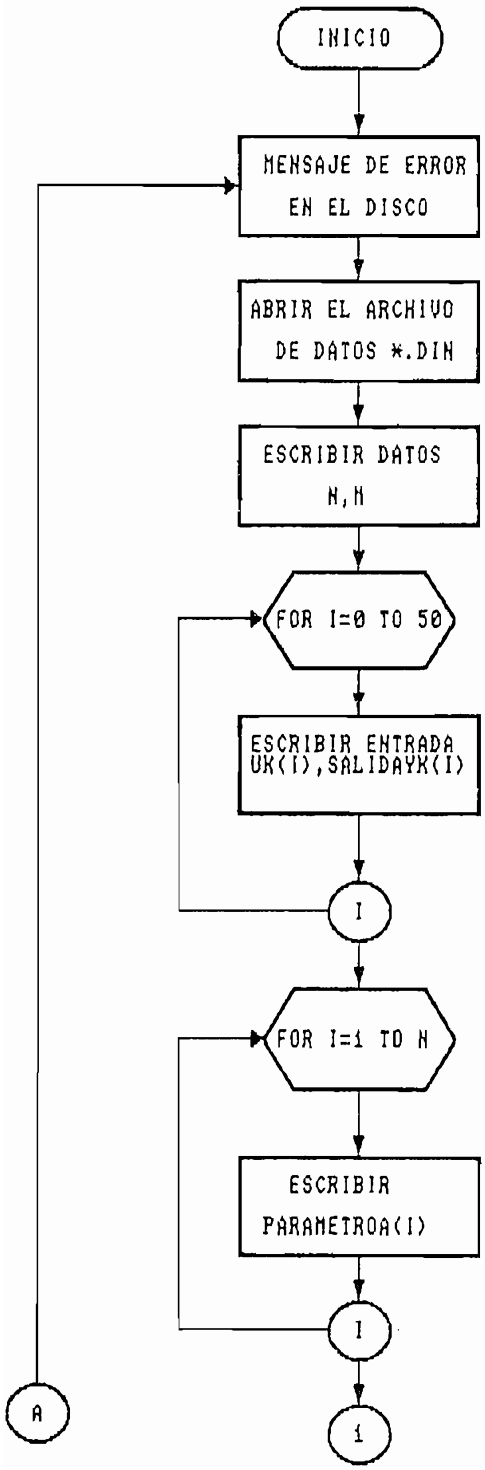
2
1/2/3/4
4
REGRESAR PROGRAMA MENU PRINCIPAL
3
2
INGRESAR LA AMPLITUD DE LA ENTRADA ESCALON
INTRODUCIR EL VALOR DE LA VARIANZA, V)0
INGRESAR LA AMPLITUD DEL RUIDO
FOR I=0 TO 50
D1=SQR(VARIANZA)
FOR I=0 TO 50
ENTRADAUK(I)= ESCALON
FORi=0 TO 50
ENTRADAUK(I)= ESCALON * RND
Z = 0
I
I
FOR J=1 TO 12
Z = Z+RND
J
RUIDO(I)= (Z-6) * D1/2
A
I
B
3


A
3
B
FOR I=0 TO 50
ENTRADAUK(I) =
RUIDO(I)
I
ENLACE SUBRUTINAS GENERAR SALIDA Y ARCHIVAR DATOS N, M ENTRADAUK(I) SALIDAYK(I),AI,CI
FORMAR LAS MATRICES A, B, C DE LA FORMA CONTROLABLE Y PRESENTAR DATOS DATOS INICIALES
DESEA UN REPORTE IMPRESO DE N, M MATRICES A , B, C
SI/NO
NO
REGRESAR AL PROGRAMA DEL MENU PRINCIPAL
SI
IMPRIMIR VALORES DE N, M Y LAS MATRICES A, B, C DE LA FORMA CANONICA CONTROLABLE

SUBRUTINA GENERACION DE LA SENAL DE SALIDA



SUBRUTINA PARA LA GENERACION DEL ARCHIVO DE DATOS



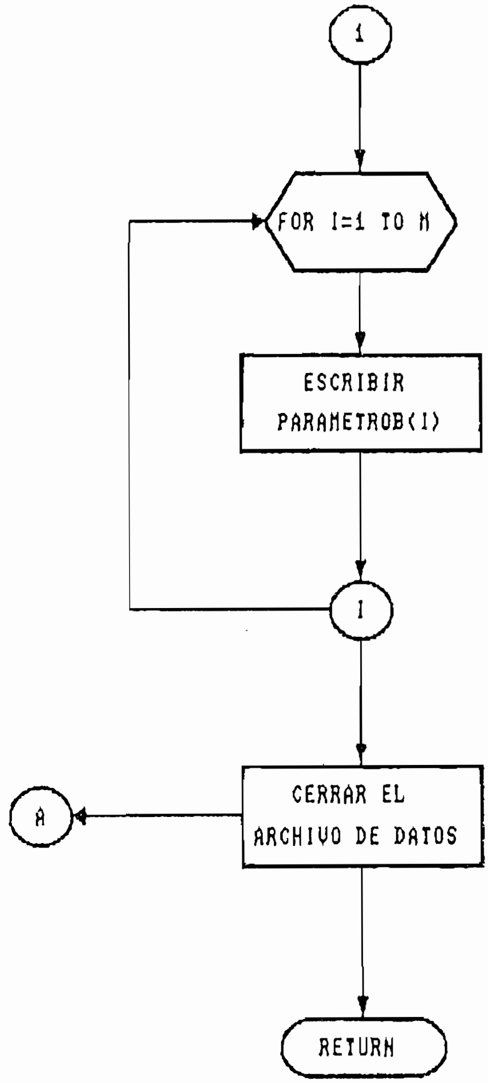
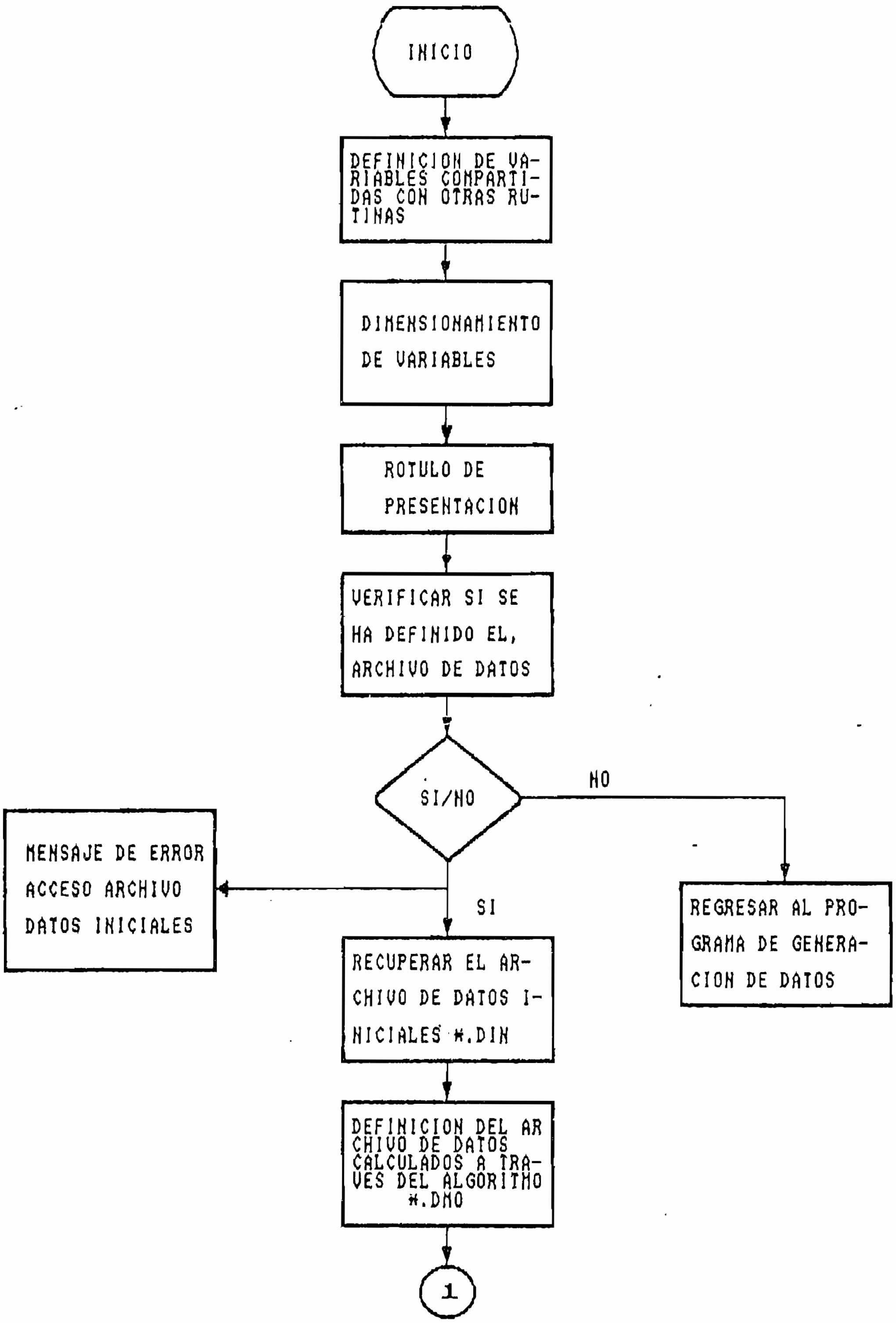
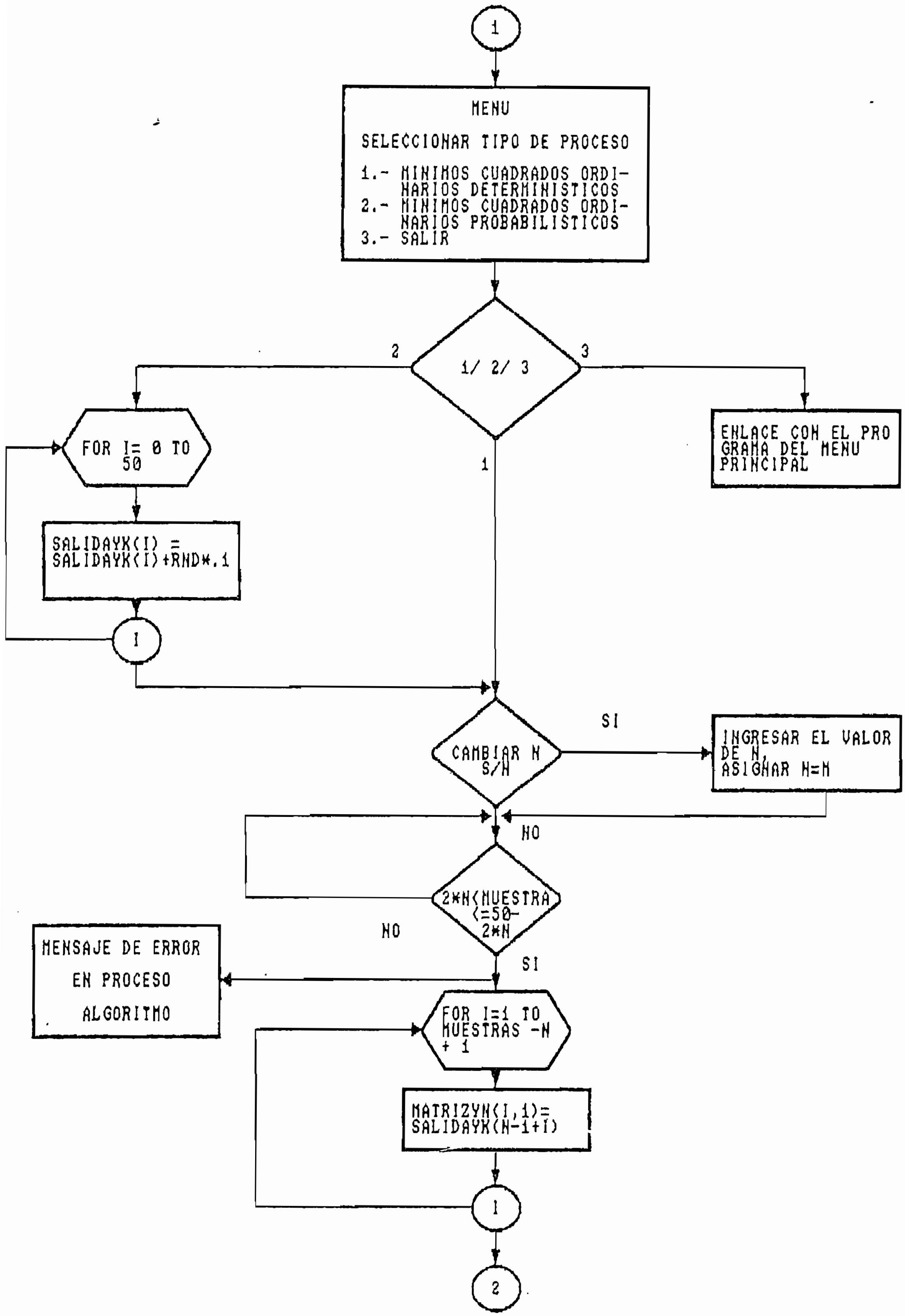
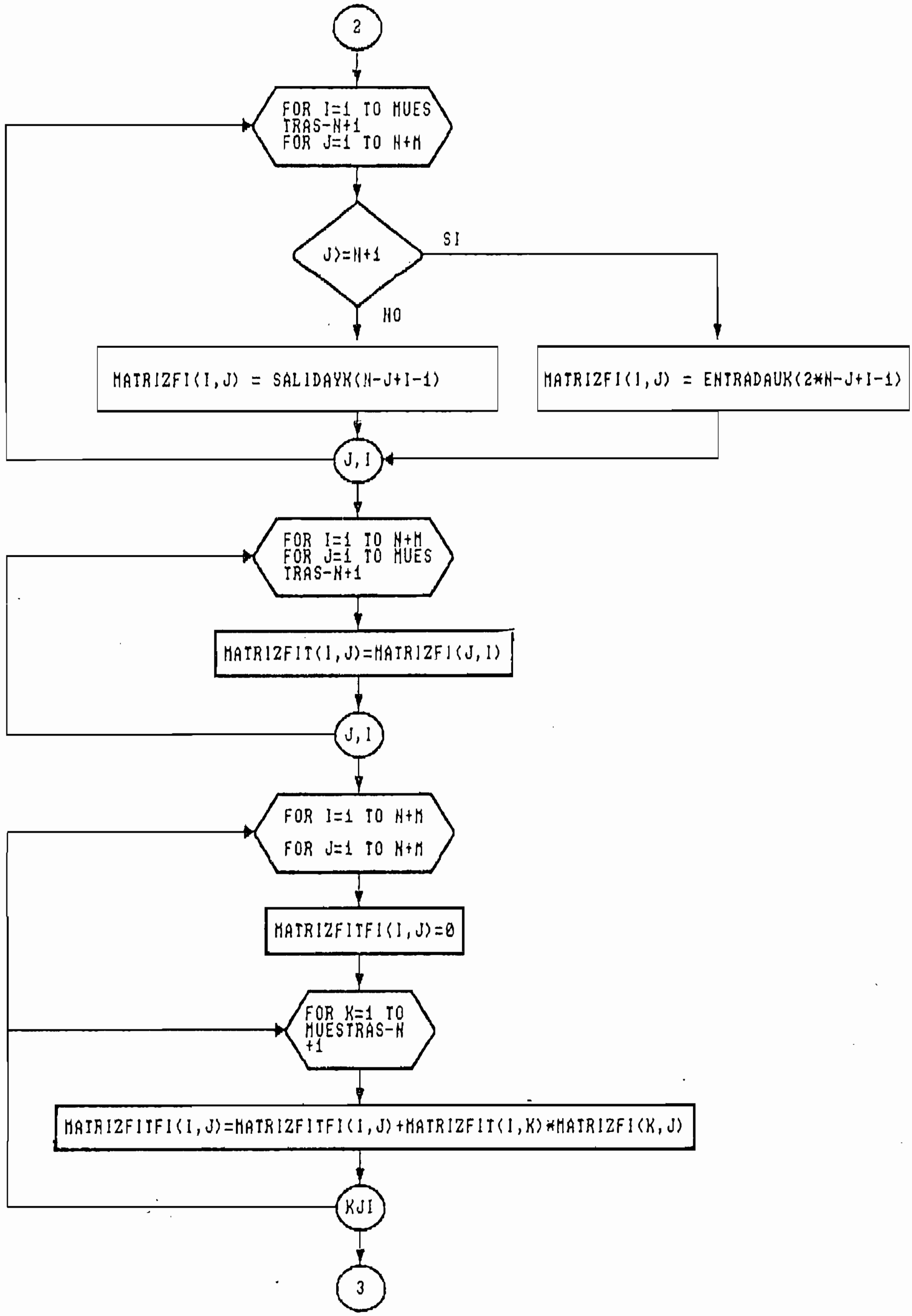
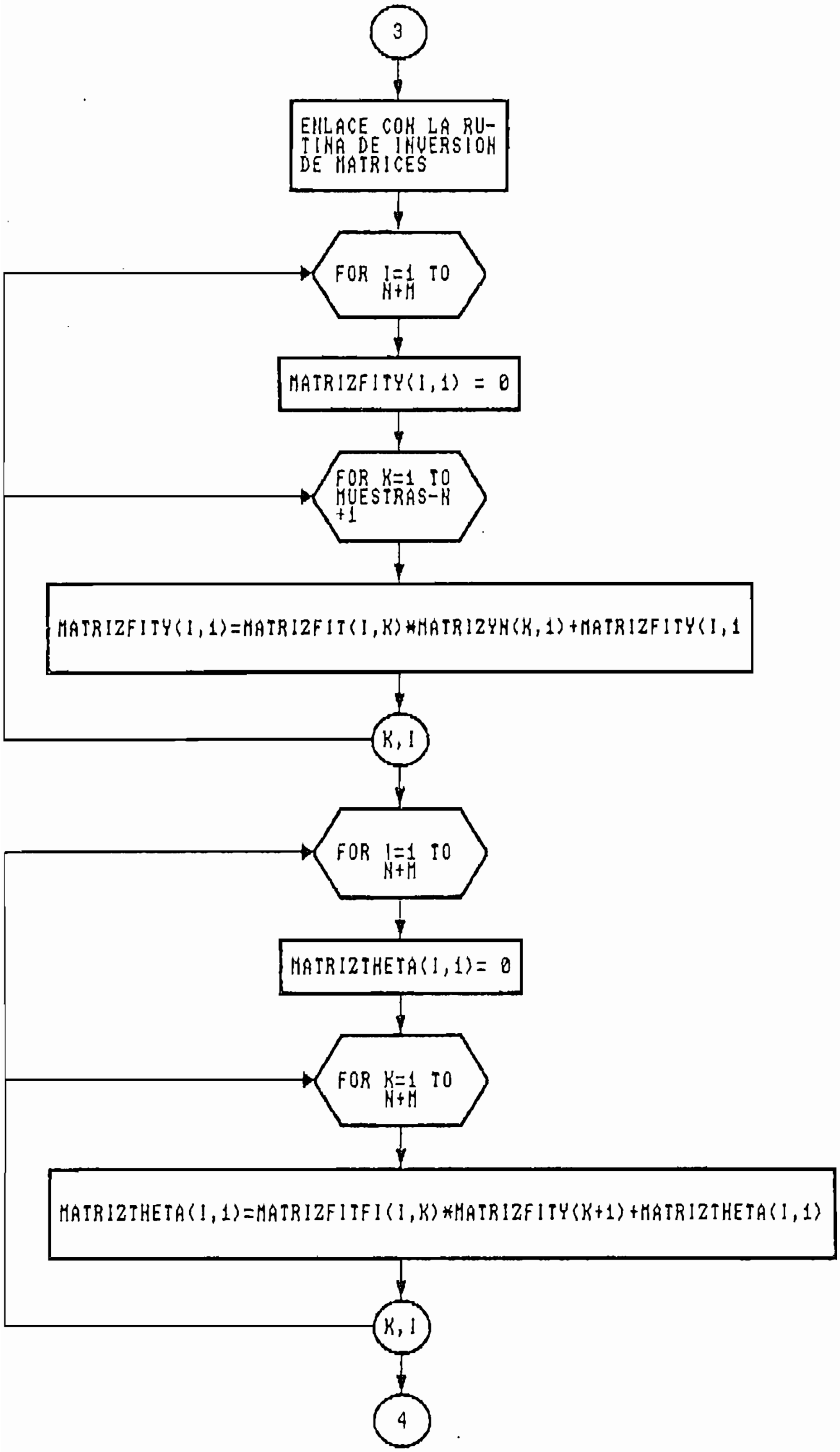

1
FOR I=1 TO M
ESCRIBIR
PARAMETROB(I)
I
CERRAR EL
ARCHIVO DE DATOS
A
RETURN

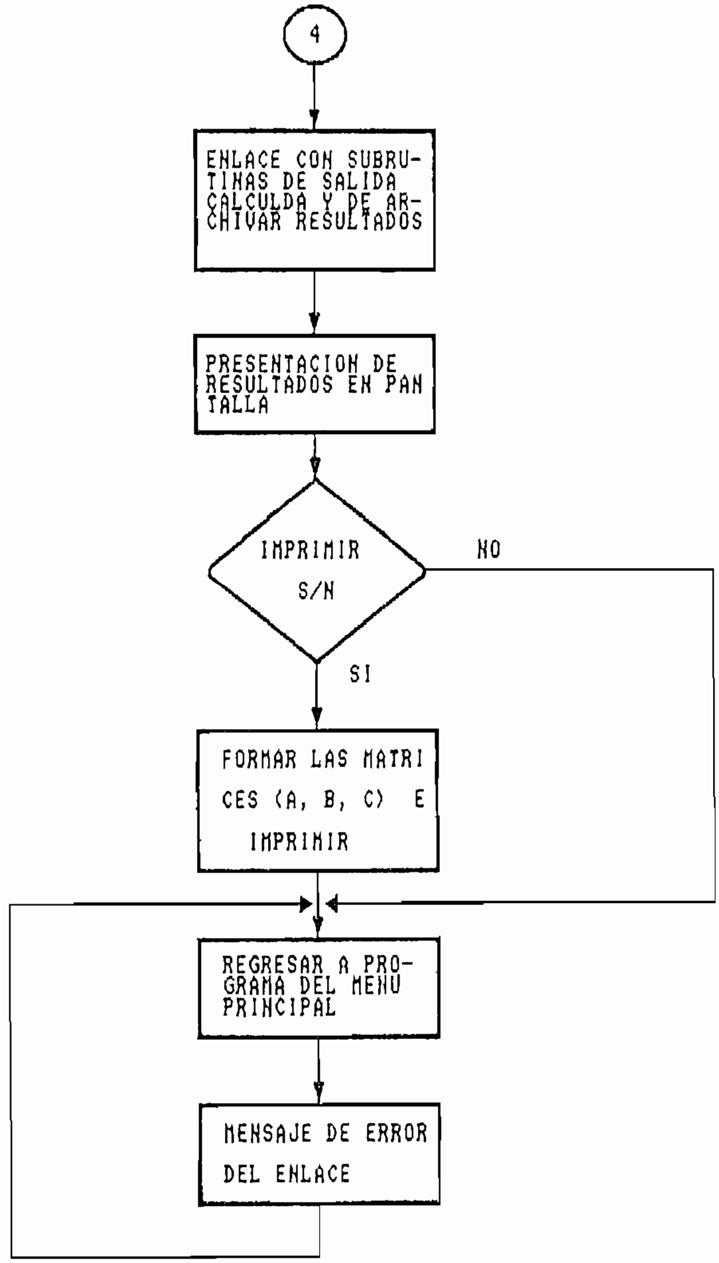
DIAGRAMA DE FLUJO PROGRAMA DE MINIMOS CUADRADOS ORDINARIOS



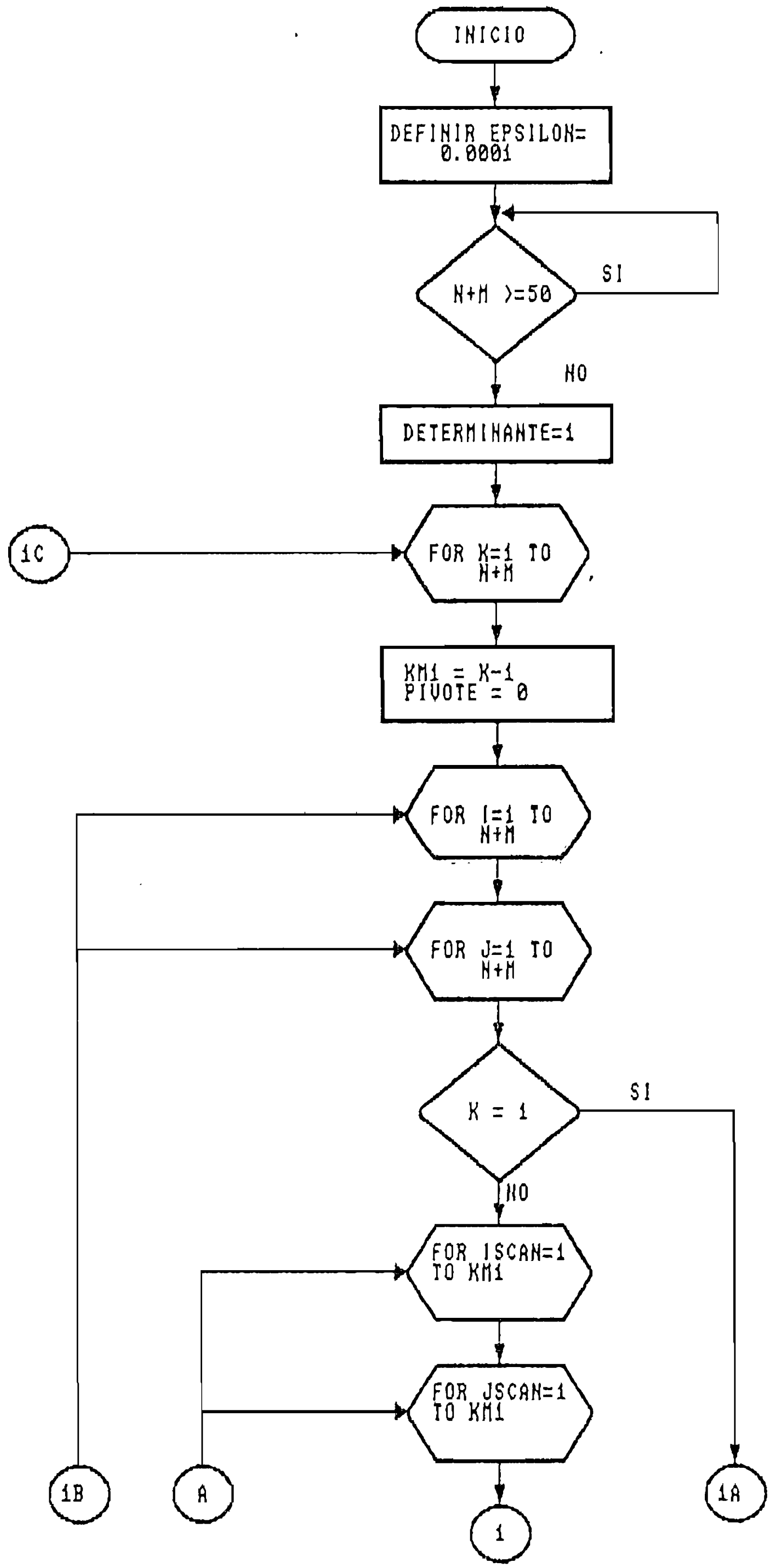

1
MENU
SELECCIONAR TIPO DE PROCESO
1.- MINIMOS CUADRADOS ORDINARIOS DETERMINISTICOS
2.- MINIMOS CUADRADOS ORDINARIOS PROBABILISTICOS
3.- SALIR
1/ 2/ 3
2
3
1
FOR I= 0 TO 50
SALIDAYK(I) = SALIDAYK(I)+RND*.1
I
ENLACE CON EL PROGRAMA DEL MENU PRINCIPAL
CAMBIAR N S/N
SI
INGRESAR EL VALOR DE N, ASIGNAR N=N
NO
2*N<MUESTRA <=50-2*N
NO
SI
MENSAJE DE ERROR EN PROCESO ALGORITMO
FOR I=1 TO MUESTRAS -N + 1
MATRIZYN(I,1)= SALIDAYK(N-1+I)
I
2

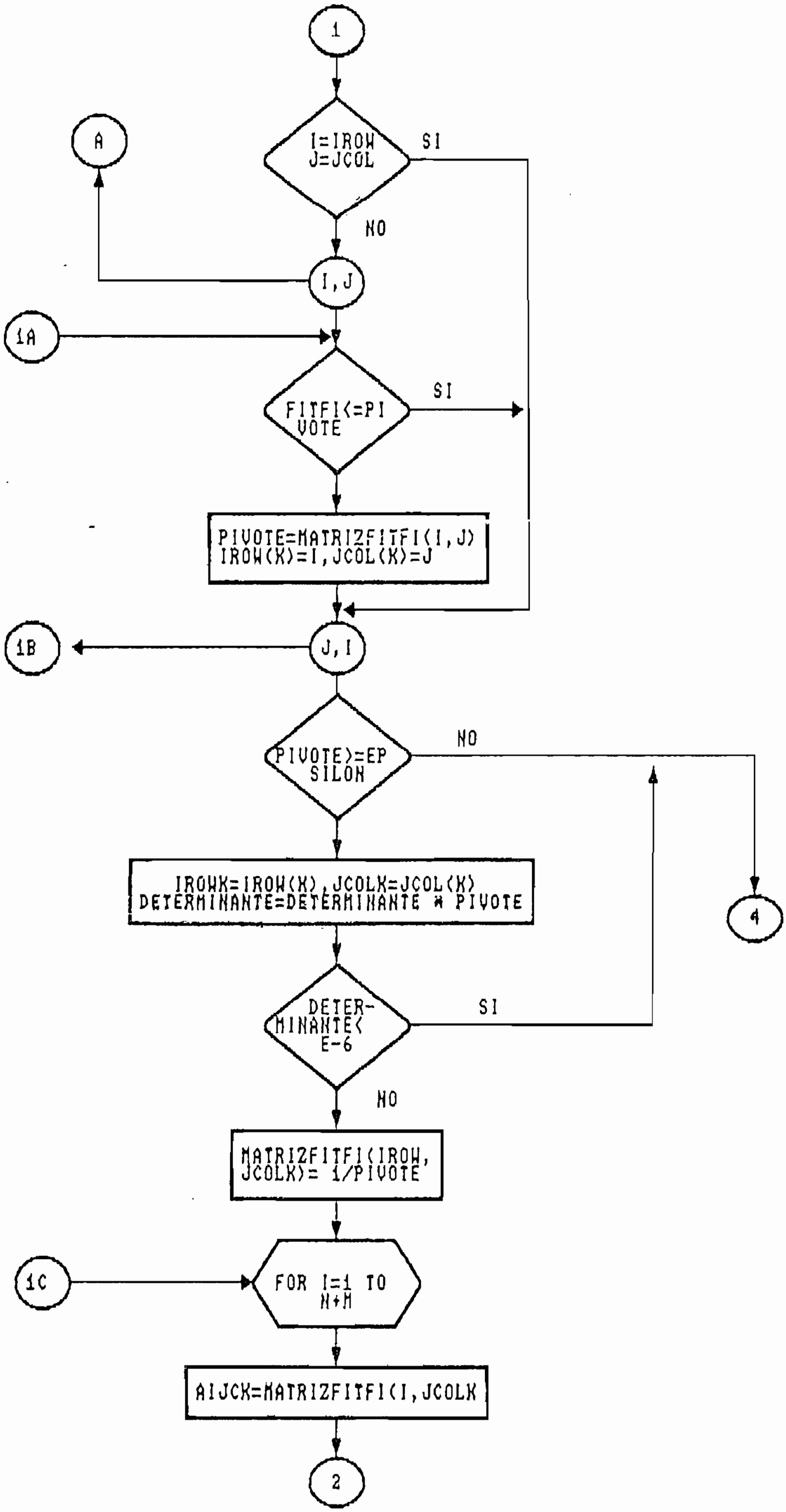

2
FOR I=1 TO MUES TRAS-N+1
FOR J=1 TO N+M
J>=N+1
SI
NO
MATRIZFI(I,J) = SALIDAYK(N-J+I-1)
MATRIZFI(I,J) = ENTRADAUK(2*N-J+I-1)
J,I
FOR I=1 TO N+M
FOR J=1 TO MUES TRAS-N+1
MATRIZFIT(I,J)=MATRIZFI(J,I)
J,I
FOR I=1 TO N+M
FOR J=1 TO N+M
MATRIZFITFI(I,J)=0
FOR K=1 TO MUESTRAS-N +1
MATRIZFITFI(I,J)=MATRIZFITFI(I,J)+MATRIZFIT(I,K)*MATRIZFI(K,J)
KJI
3

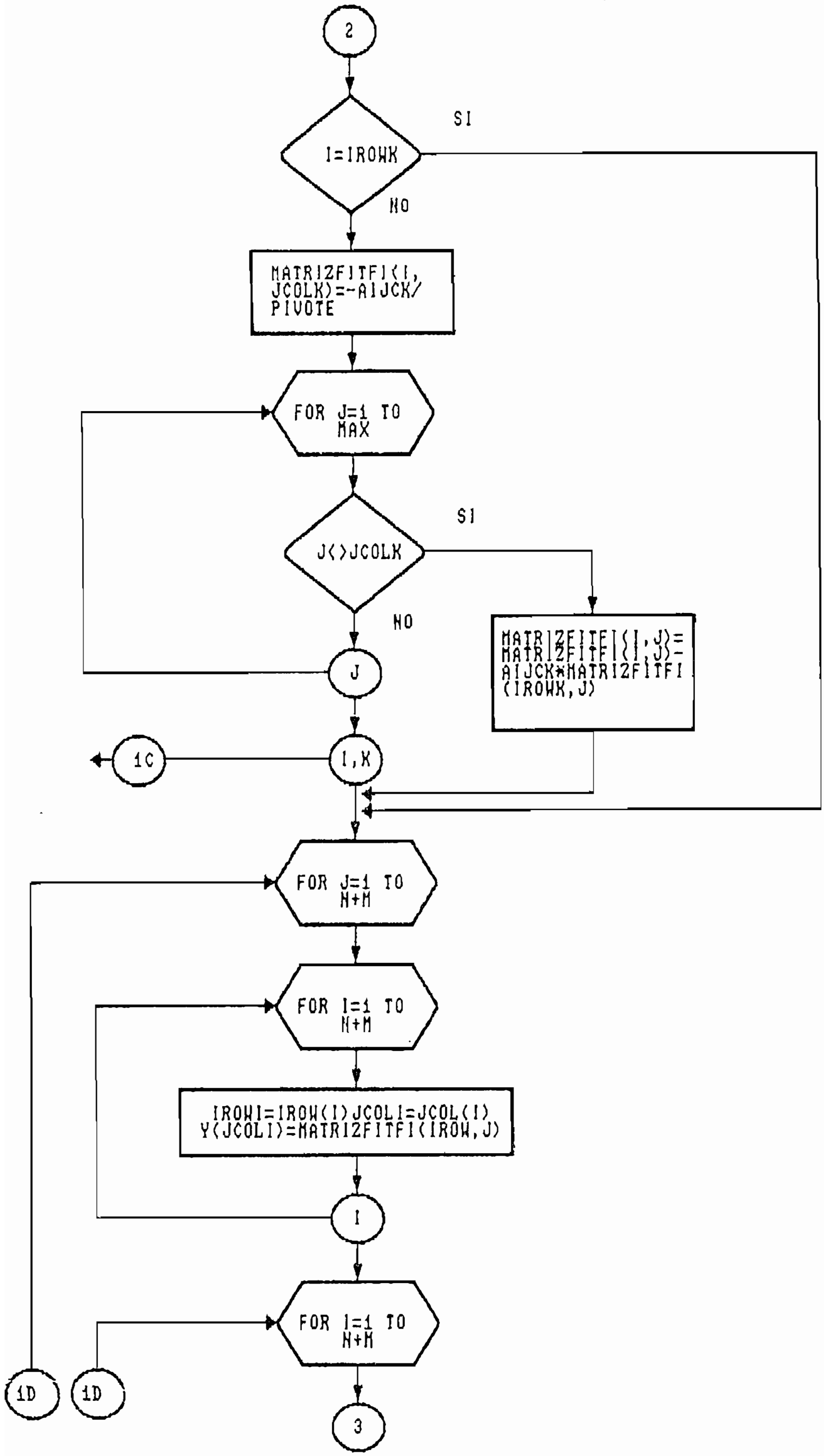

3
ENLACE CON LA RUTINA DE INVERSION DE MATRICES
FOR I=1 TO N+M
MATRIZFITY(I,1) = 0
FOR K=1 TO MUESTRAS-N +1
MATRIZFITY(I,1)=MATRIZFIT(I,K)*MATRIZYN(K,1)+MATRIZFITY(I,1
K,I
FOR I=1 TO N+M
MATRIZTHETA(I,1)= 0
FOR K=1 TO N+M
MATRIZTHETA(I,1)=MATRIZFITFI(I,K)*MATRIZFITY(K+1)+MATRIZTHETA(I,1)
K,I
4

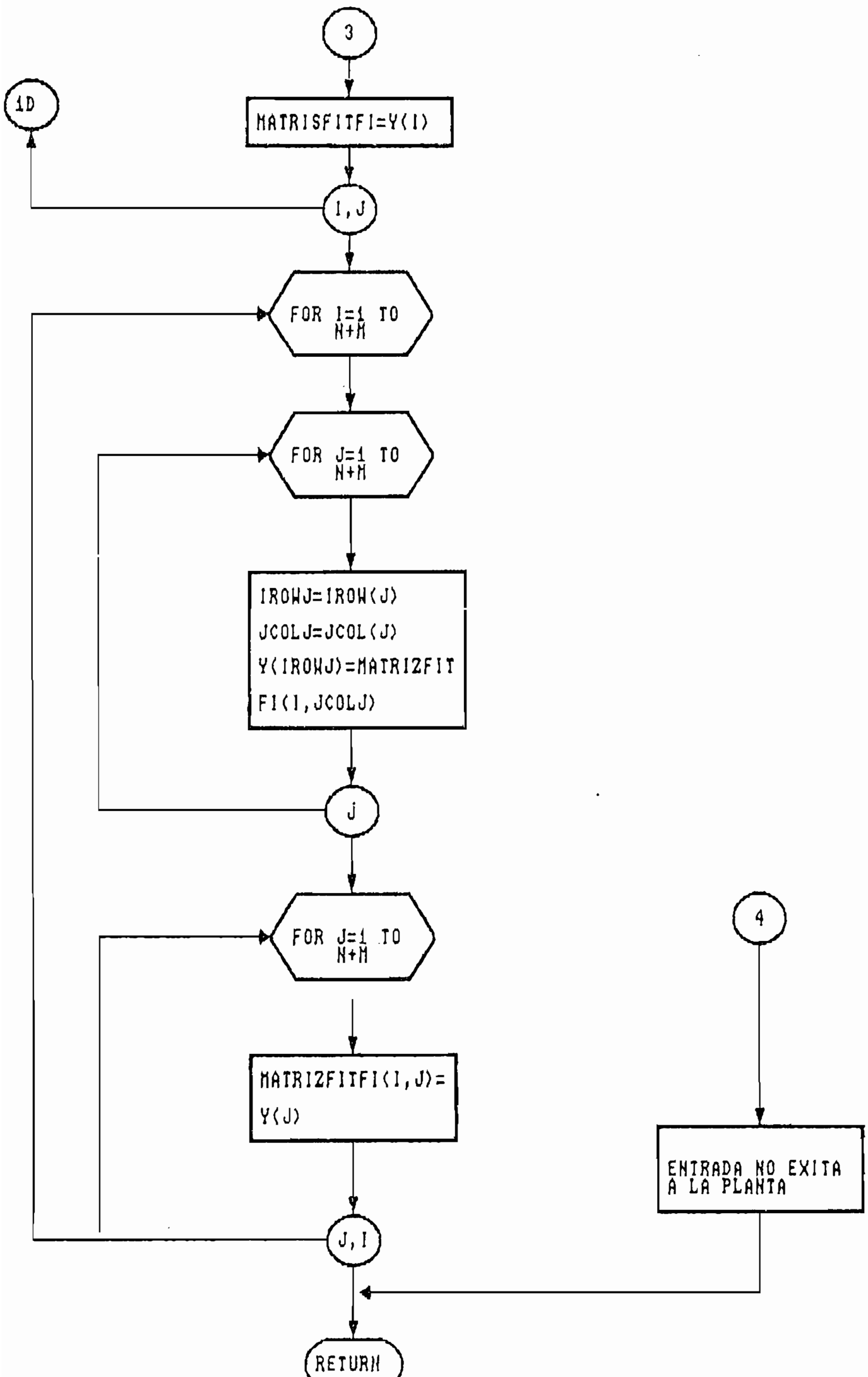

4
ENLACE CON SUBRUTINAS DE SALIDA CALCULDA Y DE ARCHIVAR RESULTADOS
PRESENTACION DE RESULTADOS EN PANTALLA
IMPRIMIR S/N
NO
SI
FORMAR LAS MATRICES (A, B, C) E IMPRIMIR
REGRESAR A PROGRAMA DEL MENU PRINCIPAL
MENSAJE DE ERROR DEL ENLACE

SUBRUTINA CALCULO DE LA MATRIZ INVERSA

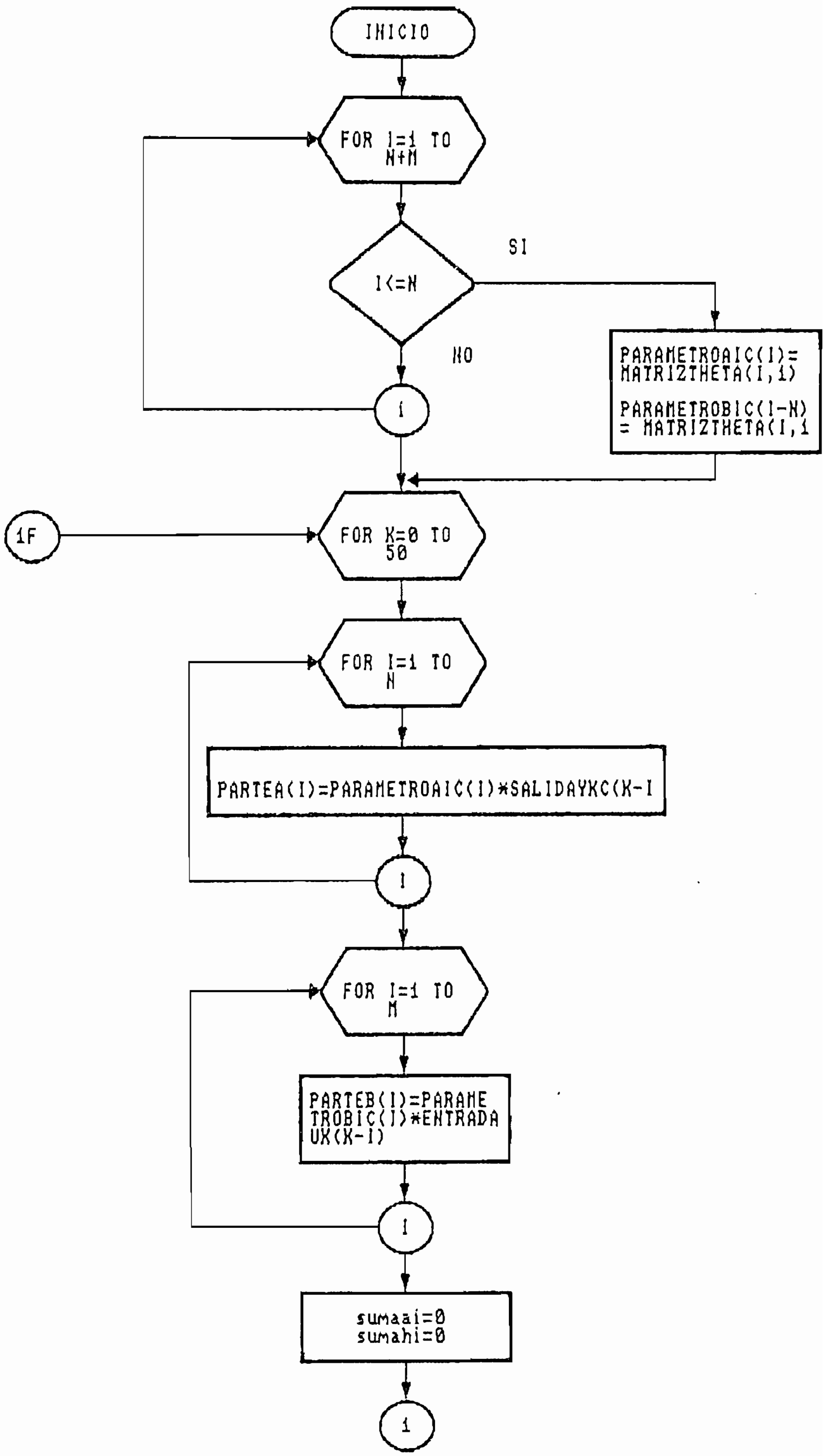


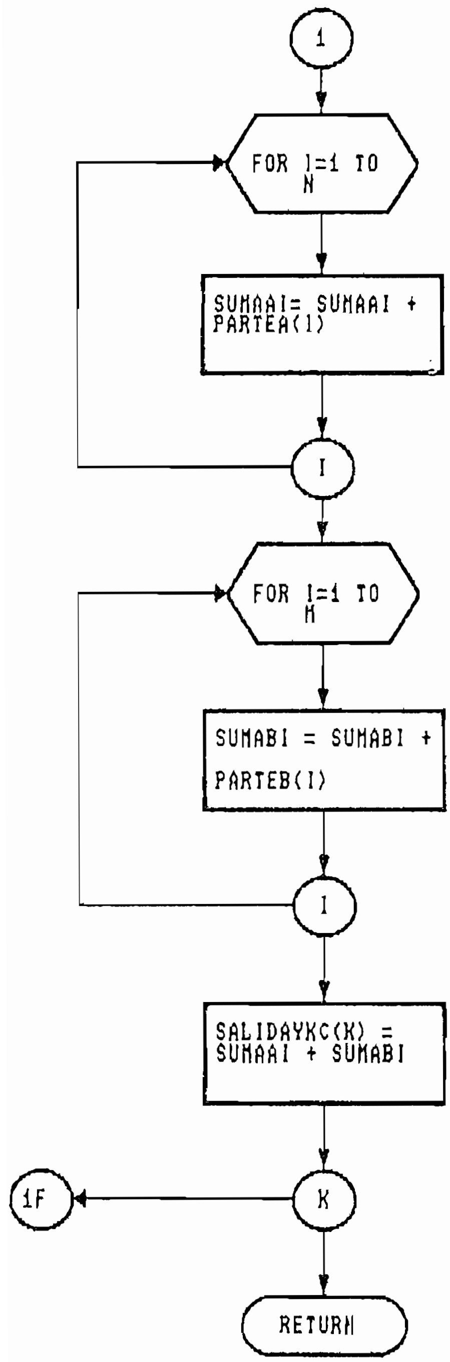

1
I=IROW
J=JCOL
SI
NO
A
I,J
1A
FITFI<=PI
VOTE
SI
PIVOTE=MATRIZFITFI(I,J)
IROW(K)=I,JCOL(K)=J
1B
J,I
PIVOTE>=EP
SILON
NO
IROWK=IROW(K),JCOLK=JCOL(K)
DETERMINANTE=DETERMINANTE * PIVOTE
4
DETER-
MINANTE<
E-6
SI
NO
MATRIZFITFI(IROW,
JCOLK)= 1/PIVOTE
1C
FOR I=1 TO
N+M
AIJCK=MATRIZFITFI(I,JCOLK
2


2
I=IROWK
SI
NO
MATRIZFITFI(I,
JCOLK)=-AIJCK/
PIVOTE
FOR J=1 TO
MAX
J<>JCOLK
SI
NO
MATRIZFITFI(I,J)=
MATRIZFITFI(I,J)-
AIJCK*MATRIZFITFI
(IROWK,J)
J
1C
I,K
FOR J=1 TO
N+M
FOR I=1 TO
N+M
IROWI=IROW(I) JCOLI=JCOL(I)
Y(JCOLI)=MATRIZFITFI(IROW,J)
I
FOR I=1 TO
N+M
1D
1D
3


3
1D
MATRISFITFI=Y(I)
I,J
FOR I=1 TO N+M
FOR J=1 TO N+M
IROWJ=IROW(J)
JCOLJ=JCOL(J)
Y(IROWJ)=MATRIZFIT
FI(I,JCOLJ)
J
FOR J=1 TO N+M
4
MATRIZFITFI(I,J)=
Y(J)
ENTRADA NO EXITA
A LA PLANTA
J,I
RETURN

SUBRUTINA GENERACION SALIDA CALCULADA




1
FOR I=1 TO N
SUMAAI= SUMAAI + PARTEA(I)
I
FOR I=1 TO M
SUMABI = SUMABI + PARTEB(I)
I
SALIDAYKC(K) = SUMAAI + SUMABI
1F
K
RETURN

SUBRUTINA PARA EL ALMACENAMIENTO DE DATOS

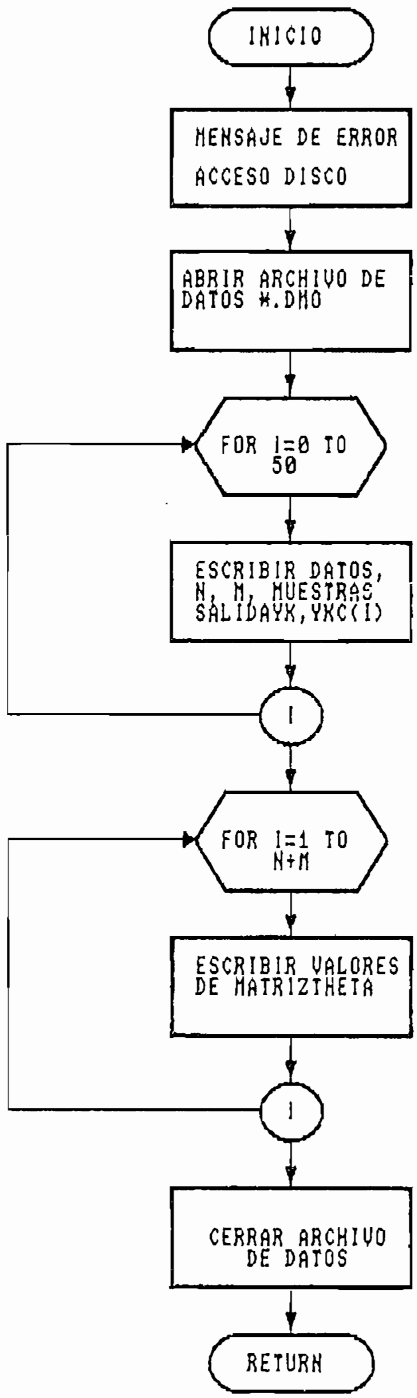
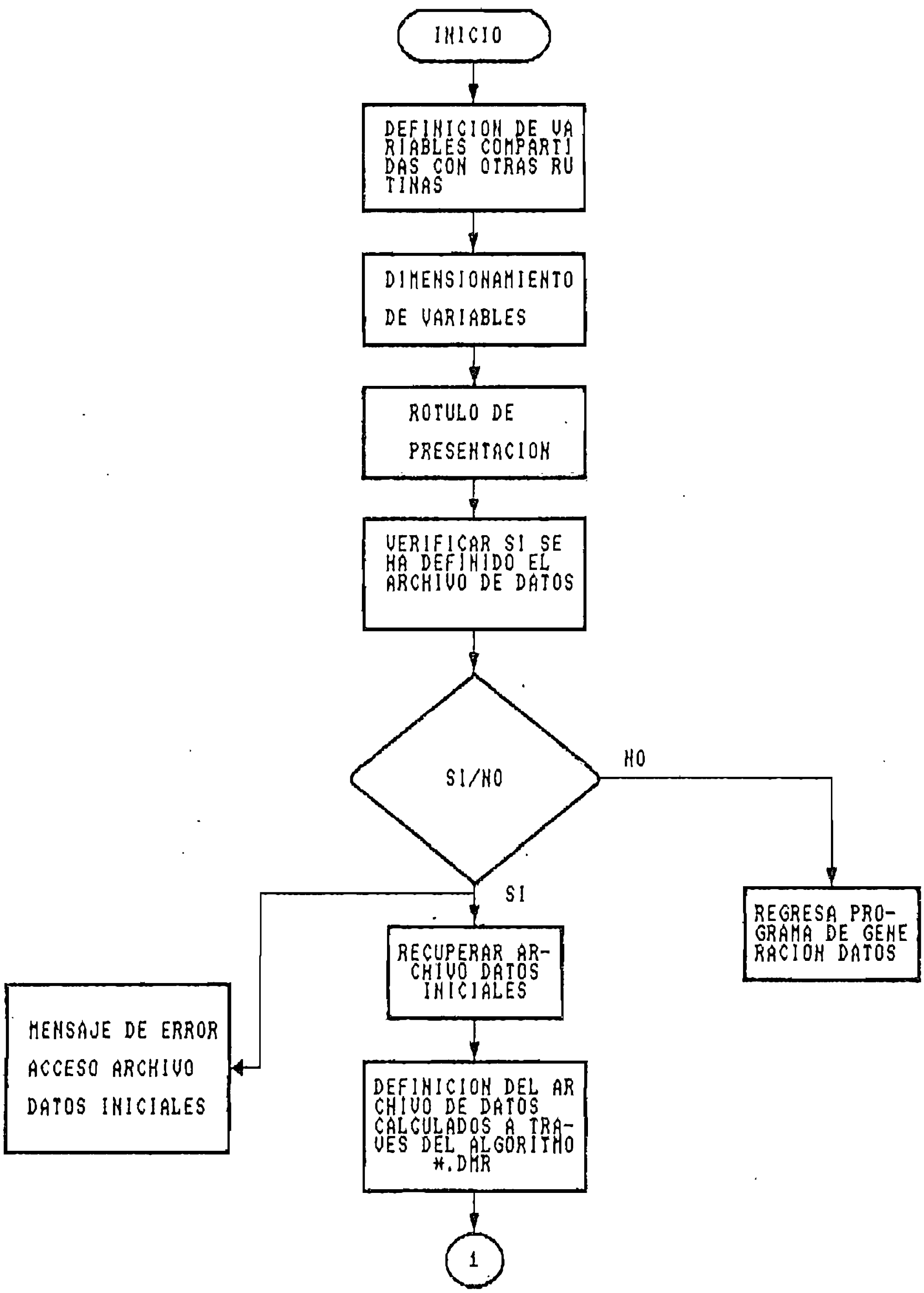
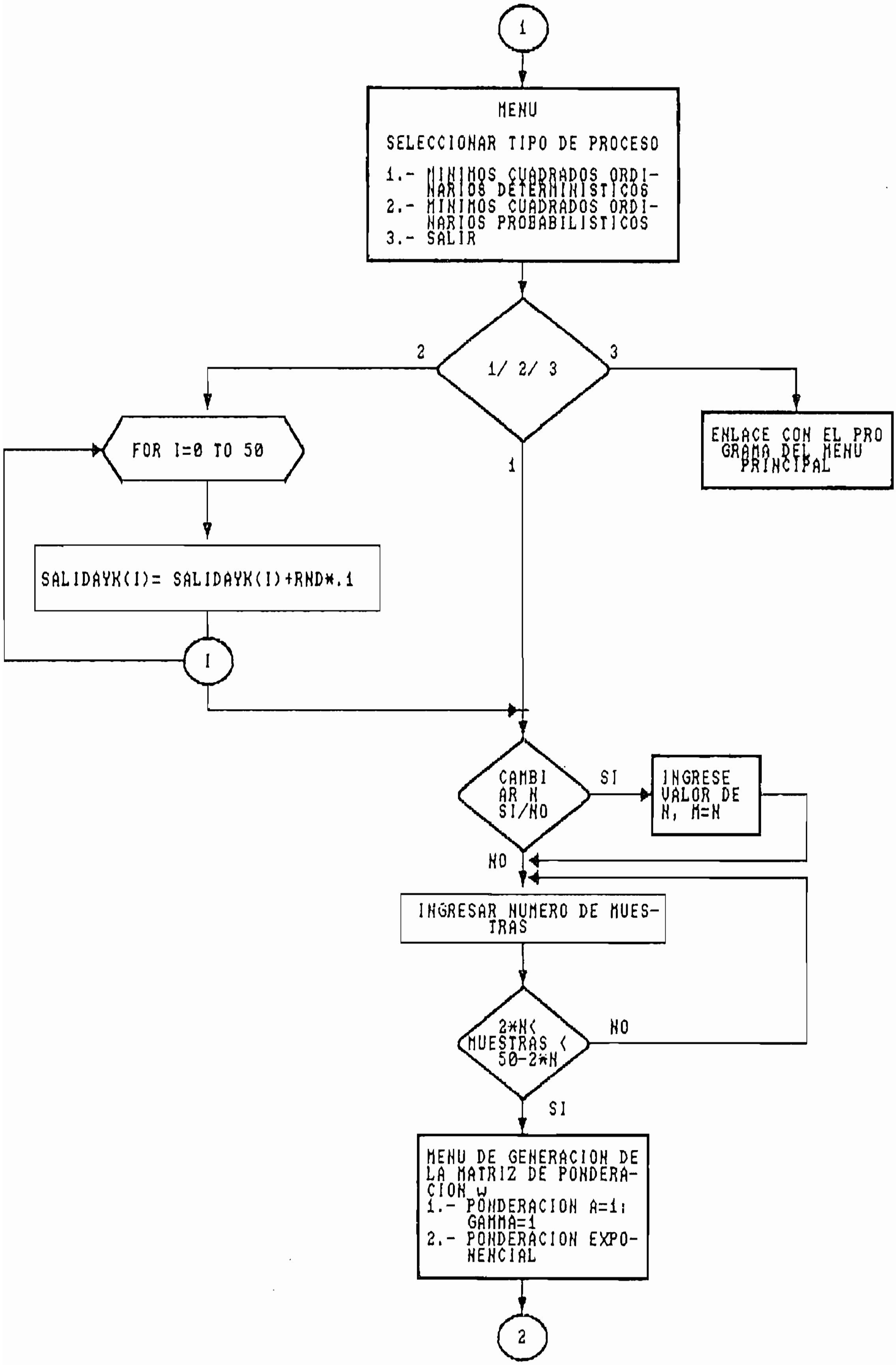
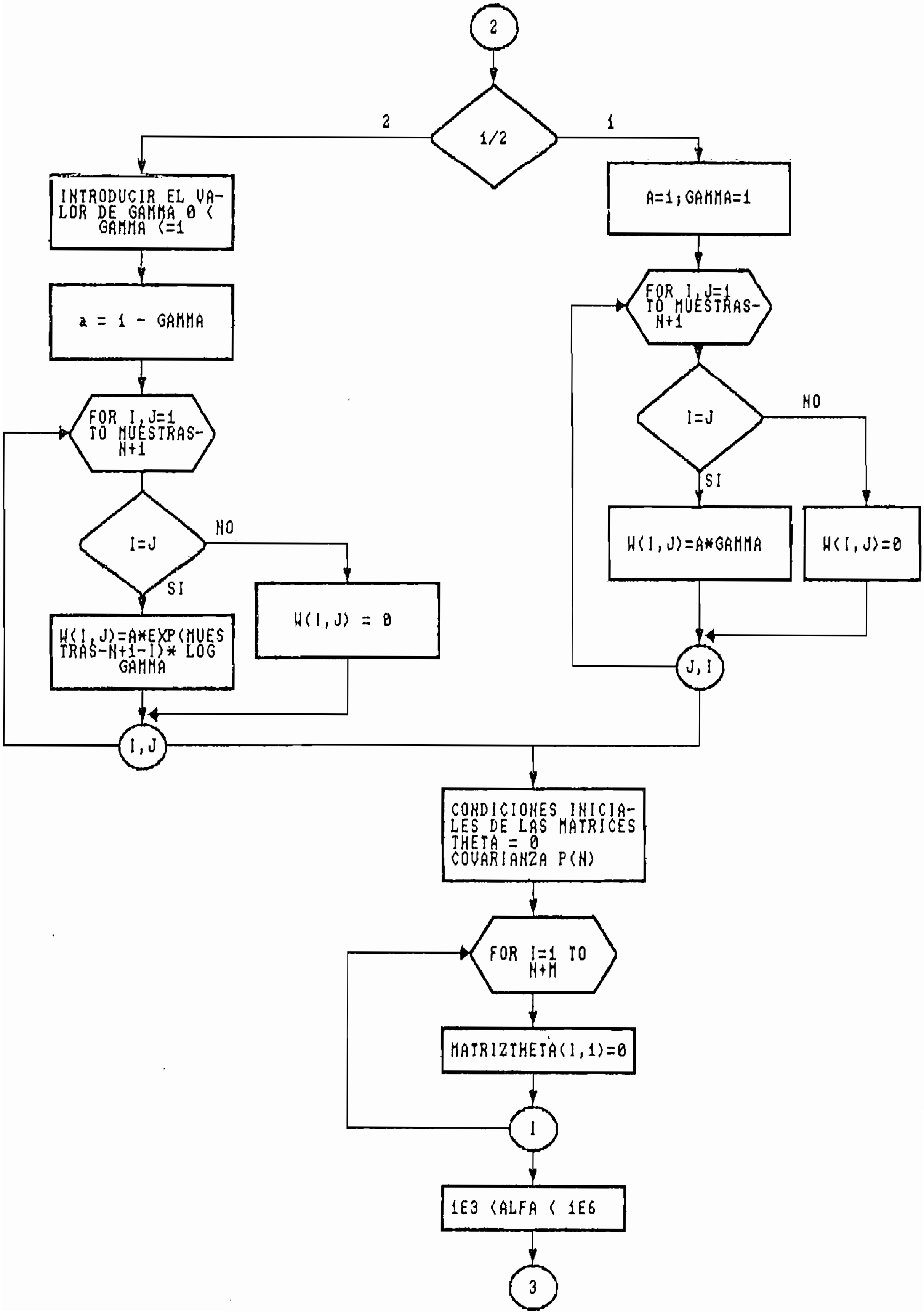
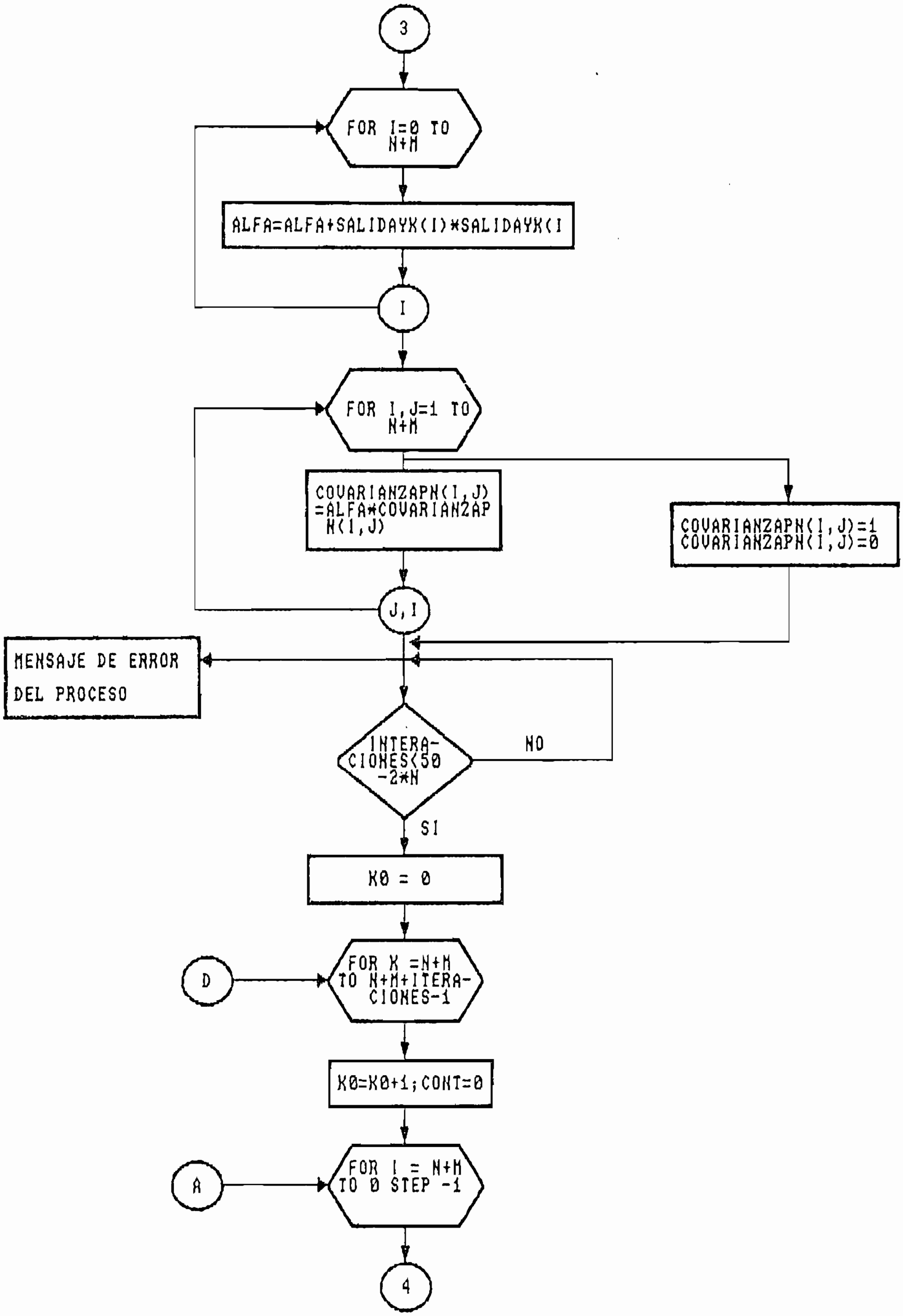


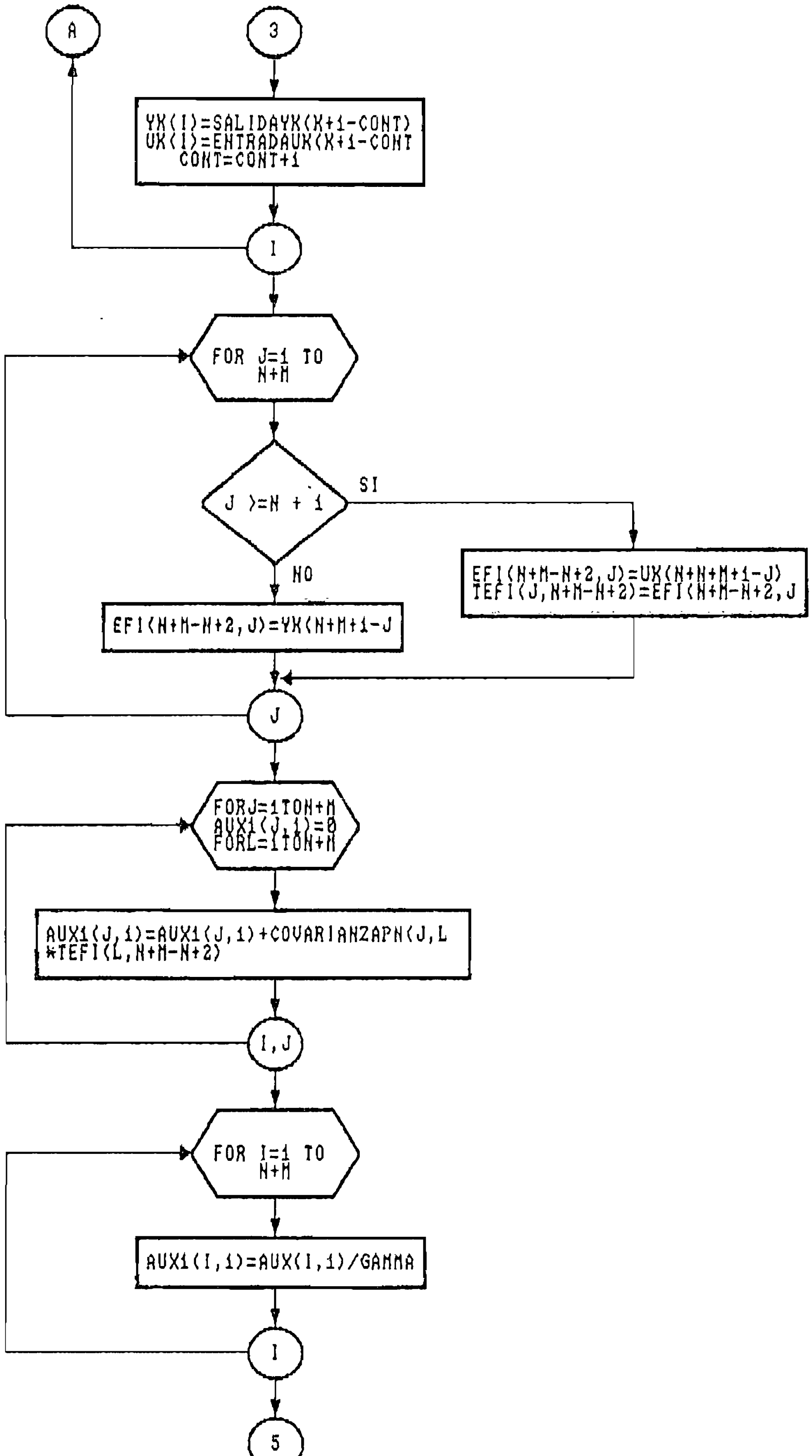
DIAGRAMA DE FLUJO PROGRAMA DE MINIMOS CUADRADOS RECURSIVOS

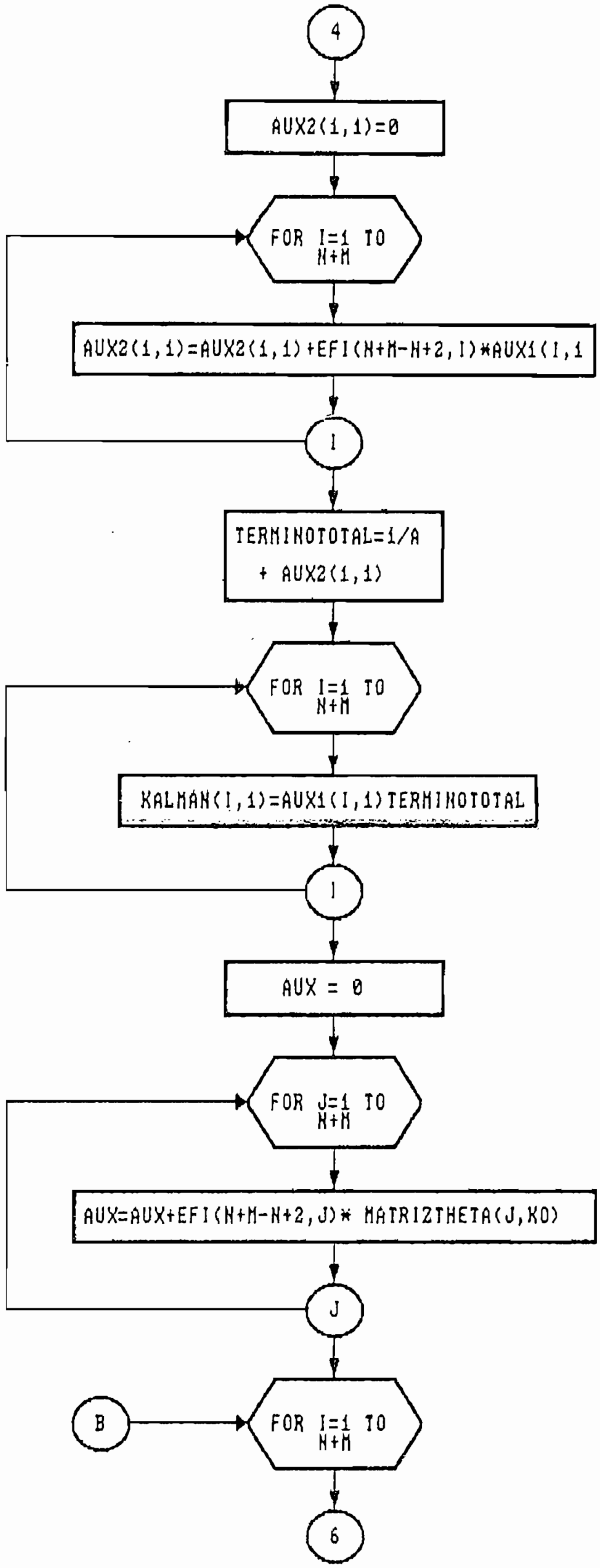


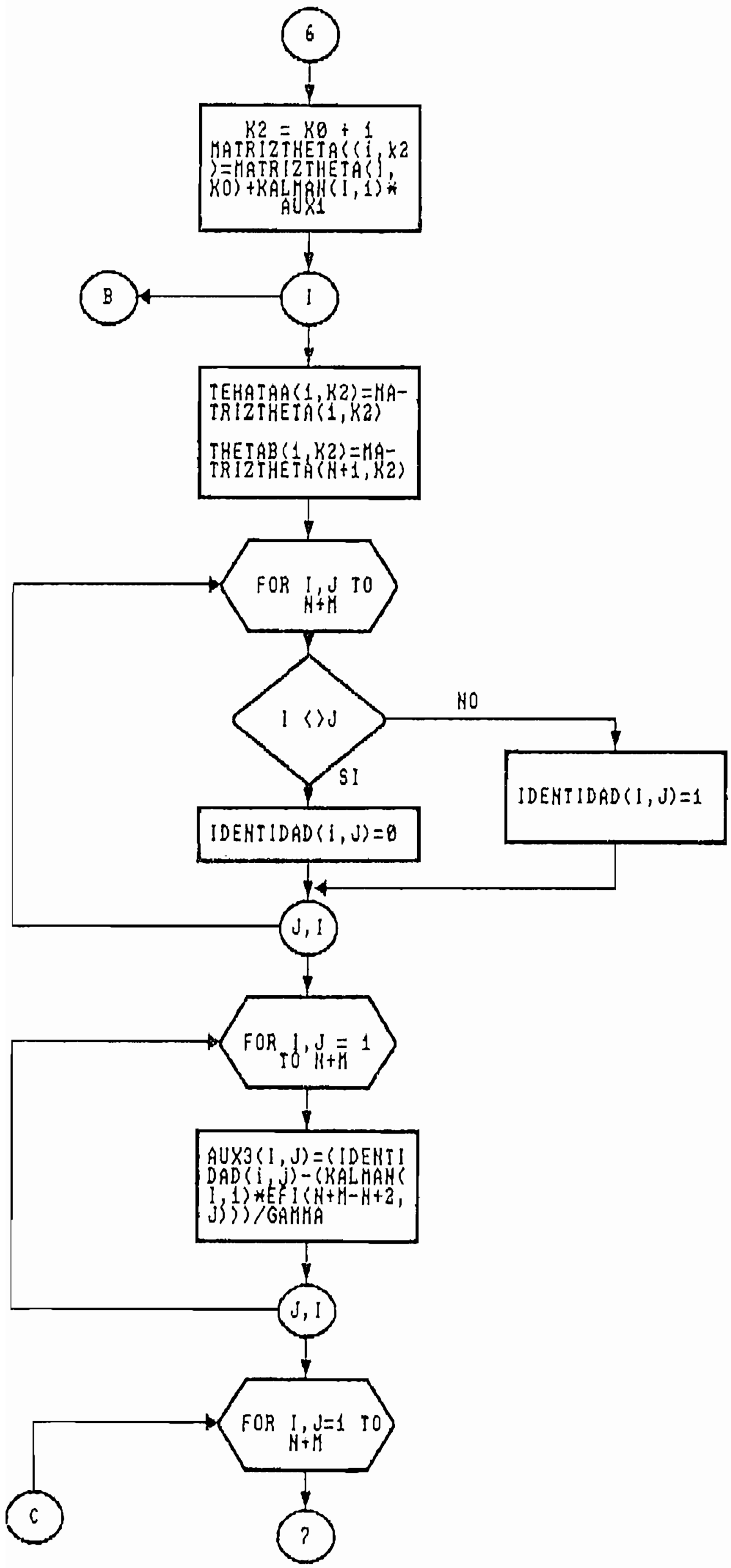

1
MENU
SELECCIONAR TIPO DE PROCESO
1.- MINIMOS CUADRADOS ORDINARIOS DETERMINISTICOS
2.- MINIMOS CUADRADOS ORDINARIOS PROBABILISTICOS
3.- SALIR
1/ 2/ 3
2
3
1
ENLACE CON EL PROGRAMA DEL MENU PRINCIPAL
FOR I=0 TO 50
SALIDAYK(I)= SALIDAYK(I)+RND*.1
I
CAMBIAR N SI/NO
SI
NO
INGRESE VALOR DE N, M=N
INGRESAR NUMERO DE MUESTRAS
2*N< MUESTRAS < 50-2*N
NO
SI
MENU DE GENERACION DE LA MATRIZ DE PONDERACION w
1.- PONDERACION A=1; GAMMA=1
2.- PONDERACION EXPONENCIAL
2

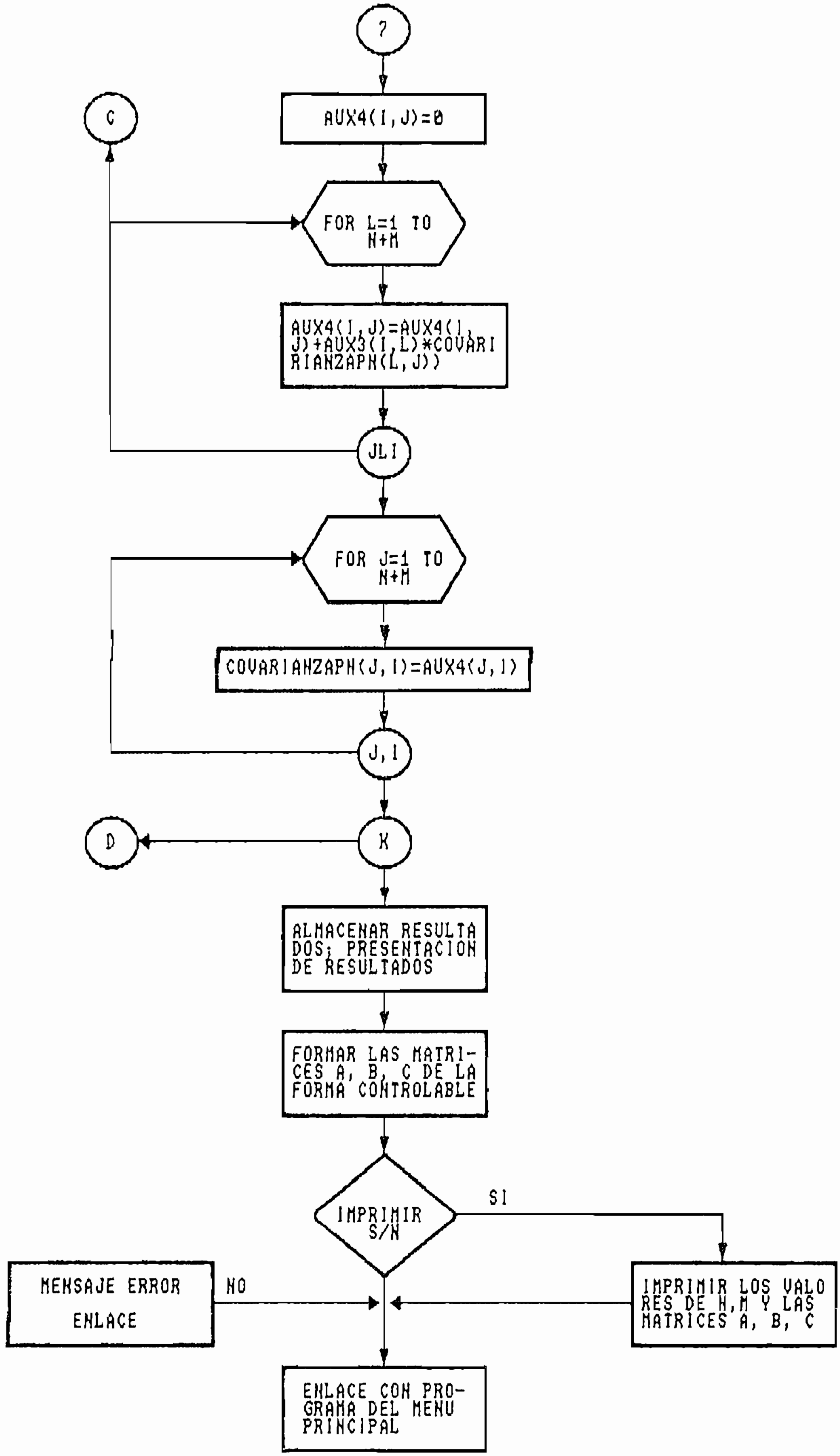

2
1/2
2
1
INTRODUCIR EL VALOR DE GAMMA 0 < GAMMA <=1
a = 1 - GAMMA
FOR I,J=1 TO MUESTRAS-N+1
I=J
NO
SI
W(I,J)=A*EXP(MUESTRAS-N+1-I)* LOG GAMMA
W(I,J) = 0
I,J
A=1;GAMMA=1
FOR I,J=1 TO MUESTRAS-N+1
I=J
NO
SI
W(I,J)=A*GAMMA
W(I,J)=0
J,I
CONDICIONES INICIALES DE LAS MATRICES THETA = 0 COVARIANZA P(N)
FOR I=1 TO N+M
MATRIZTHETA(I,1)=0
I
1E3 <ALFA < 1E6
3


3
FOR I=0 TO N+M
ALFA=ALFA+SALIDAYK(I)*SALIDAYK(I
I
FOR I,J=1 TO N+M
COVARIANZAPN(I,J) =ALFA*COVARIANZAPN(I,J)
COVARIANZAPN(I,J)=1
COVARIANZAPN(I,J)=0
J,I
MENSAJE DE ERROR DEL PROCESO
INTERA-CIONES<50 -2*N
NO
SI
K0 = 0
D
FOR K =N+M TO N+M+ITERA-CIONES-1
K0=K0+1;CONT=0
A
FOR I = N+M TO 0 STEP -1
4


A
3
YK(I)=SALIDAYK(K+1-CONT)
UK(I)=ENTRADAUK(K+1-CONT
CONT=CONT+1
I
FOR J=1 TO N+M
J >=N + 1
SI
NO
EFI(N+M-N+2,J)=YK(N+M+1-J
EFI(N+M-N+2,J)=UK(N+N+M+1-J)
TEFI(J,N+M-N+2)=EFI(N+M-N+2,J
J
FORJ=1TON+M
AUX1(J,1)=0
FORL=1TON+M
AUX1(J,1)=AUX1(J,1)+COVARIANZAPN(J,L
*TEFI(L,N+M-N+2)
I,J
FOR I=1 TO N+M
AUX1(I,1)=AUX(I,1)/GAMMA
I
5


4
AUX2(1,1)=0
FOR I=1 TO N+M
AUX2(1,1)=AUX2(1,1)+EFI(N+M-N+2,I)*AUX1(I,1
I
TERMINOTOTAL=1/A + AUX2(1,1)
FOR I=1 TO N+M
KALMAN(I,1)=AUX1(I,1)TERMINOTOTAL
I
AUX = 0
FOR J=1 TO N+M
AUX=AUX+EFI(N+M-N+2,J)* MATRIZTHETA(J,KO)
J
B
FOR I=1 TO N+M
6


6
K2 = X0 + 1
MATRIZTHETA((1,K2
)=MATRIZTHETA(1,
K0)+KALMAN(I,1)*
AUX1
B
I
TEHATAA(1,K2)=MA-
TRIZTHETA(1,K2)
THETAB(1,K2)=MA-
TRIZTHETA(N+1,K2)
FOR I,J TO
N+M
I <>J
NO
SI
IDENTIDAD(I,J)=1
IDENTIDAD(I,J)=0
J,I
FOR I,J = 1
TO N+M
AUX3(I,J)=(IDENTI
DAD(I,J)-(KALMAN(
I,1)*EFI(N+M-H+2,
J)))/GAMMA
J,I
FOR I,J=1 TO
N+M
C
7


7
AUX4(I,J)=0
C
FOR L=1 TO N+M
AUX4(I,J)=AUX4(I,J)+AUX3(I,L)*COVARIANZAPN(L,J))
JLI
FOR J=1 TO N+M
COVARIANZAPN(J,I)=AUX4(J,I)
J,I
D
K
ALMACENAR RESULTADOS; PRESENTACION DE RESULTADOS
FORMAR LAS MATRICES A, B, C DE LA FORMA CONTROLABLE
IMPRIMIR S/N
SI
NO
MENSAJE ERROR ENLACE
IMPRIMIR LOS VALORES DE N,M Y LAS MATRICES A, B, C
ENLACE CON PROGRAMA DEL MENU PRINCIPAL

SUBRUTINA PARA EL ALMACENAMIENTO DE DATOS

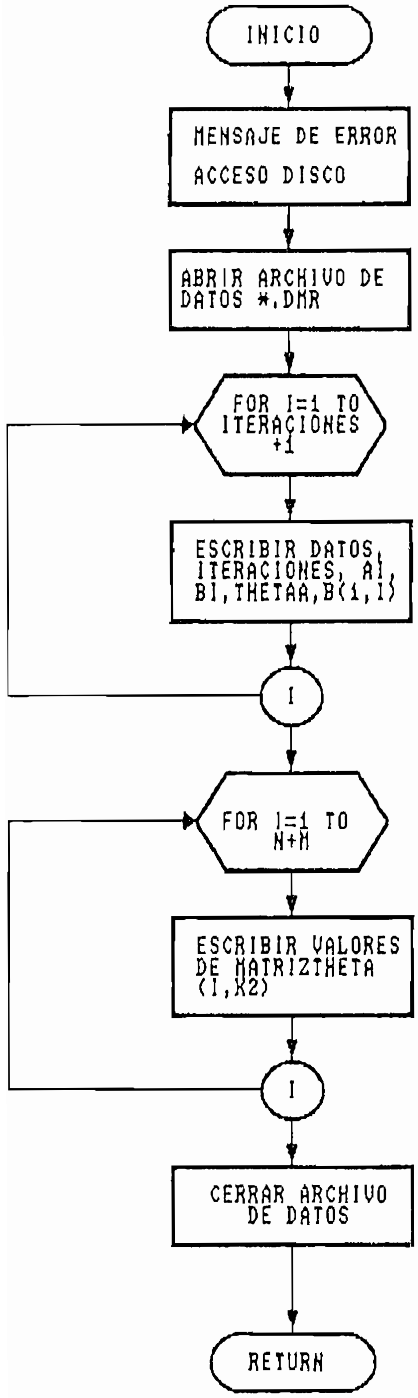
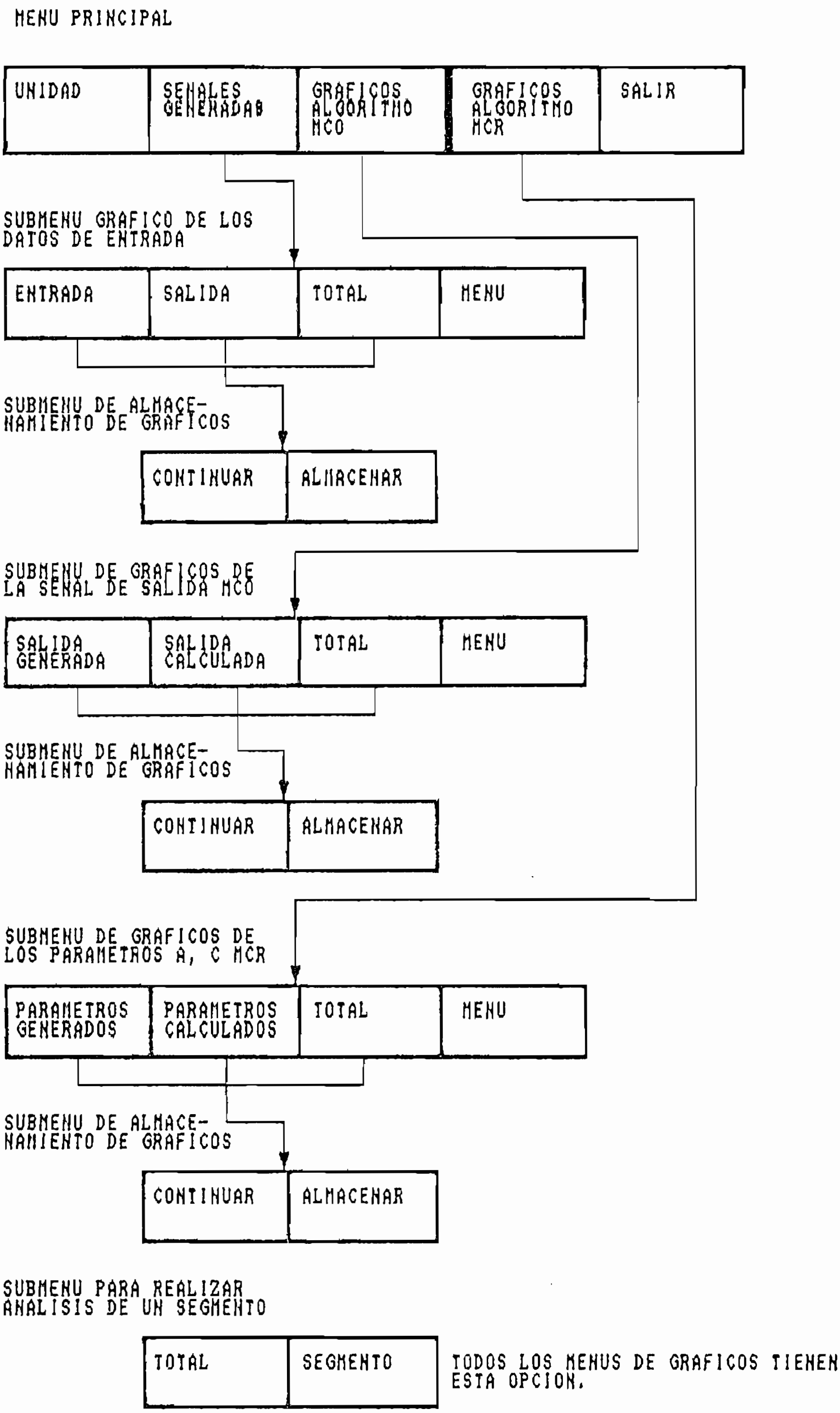


DIAGRAMA DE FLUJO PROGRAMA DE GRAFICOS



LISTADOS DE LOS PROGRAMAS

3.-

PROGRAMAS EN QBASIC 4.5 (MICROSOFT)

```
'*********************************************************
'
'          PROGRAMA MAESTRO DE CONTROL DE RUTINAS
'
'                JOSE RAUL MEJIA RECALDE
'
'ARCHIVO:IPVE.BAS
'
'*********************************************************
COMMON unidad$, nombre$, nombres$, N, m, datos
DIM entradauk(-5 TO 50), salidayk(-5 TO 50), parametroai(4),
parametrobi(4)
DIM  ruido(-5  TO  50),   matriza(5,  5),   matrizb(5,  5),
matrizc(5, 5)
datos = 50
CLS
menu:
DO
     CLS : COLOR 3, 5, 0
     FOR  i = 1 TO 25: LOCATE  i, 1: PRINT STRING$(79, " "):
NEXT i
     COLOR 15, 5: LOCATE 2, 60: PRINT "Fecha: "; DATE$
     LOCATE 3, 60: PRINT "Hora:  "; TIME$
     LOCATE 3, 1: PRINT "FACULTAD DE INGENIERIA ELECTRICA"
     LOCATE  4,  1: PRINT  "IDENTIFICACION DE  PARAMETROS EN
VARIABLES DE ESTADO"
     COLOR 3, 5: LOCATE 5, 1: PRINT STRING$(79, 205)
     COLOR  15, 5:  LOCATE  6, 25:  PRINT  "JOSE RAUL  MEJIA
RECALDE"
     LOCATE 8, 25: PRINT "MENU PRINCIPAL ":
     LOCATE 10, 20: COLOR 15, 5: PRINT  "1.-"; : COLOR 3, 5:
PRINT " DETERMINAR UNIDAD DE DATOS"
     LOCATE 12, 20: COLOR 15, 5: PRINT "2.- "; : COLOR 3, 5:
PRINT "GENERACION DE LOS DATOS MUESTREADOS"
     LOCATE 14, 20: COLOR  15, 5: PRINT "3.-"; : COLOR 3, 5:
PRINT " MINIMOS CUADRADOS ORDINARIOS"
     LOCATE 16, 20: COLOR 15, 5: PRINT "4.-";  : COLOR 3, 5:
PRINT " MINIMOS CUADRADOS RECURSIVOS"
     LOCATE  18, 20: COLOR 15, 5: PRINT "5.-"; : COLOR 3, 5:
PRINT " GRAFICAR RESULTADOS "
     LOCATE 20, 20: COLOR 15, 5:  PRINT "6.-"; : COLOR 3, 5:
PRINT " SALIR AL SISTEMA OPERATIVO"
     LOCATE 22, 30: PRINT "Seleccione una de las  opciones";
: COLOR 15, 5: PRINT " (1 a 6)..."; : COLOR 7, 5
seleccionar: LET  tecla$ = INKEY$: IF tecla$  = "" THEN GOTO
seleccionar
     SELECT CASE tecla$
     CASE "1"
          CLS : COLOR 3, 5, 0
          FOR  i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, "
"): NEXT i
```

```
          COLOR 15, 5: LOCATE 3, 12: PRINT ; "IDENTIFICACION
DE PARAMETROS EN VARIABLES DE ESTADO"
          LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
          LOCATE 7, 24: PRINT "UNIDAD DEL DISCO DE DATOS"
          LOCATE 14, 20: PRINT  "INGRESE LA UNIDAD DEL DISCO
DE DATOS..(A,B,C)"
unidad:
          LOCATE 17,  35: INPUT "UNIDAD =  ", unidad$: COLOR
7, 5
          IF  unidad$   =  ""  THEN  GOTO  unidad  ELSE  IF
INSTR("ABCabc", unidad$) <> 0 THEN GOTO nombre ELSE BEEP
          LOCATE 17, 43: PRINT STRING$(30, " "): GOTO unidad
nombre:
          nombre$ = "": N = 0: m = 0:  muestras = 0: gamma =
0
          GOTO menu
     CASE "2"
          CLS : COLOR 3, 5, 0
          FOR i = 1  TO 25: LOCATE i, 1: PRINT STRING$(79, "
"): NEXT i
          LOCATE 3, 12:  COLOR 15, 5:  PRINT "IDENTIFICACION
DE PARAMETROS EN VARIABLES DE ESTADO"
          LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
          LOCATE 7, 28: PRINT "GENERACION DE DATOS"
          IF (unidad$ <> "") THEN GOTO continuar
          LOCATE  20, 18: PRINT "No se  conoce la unidad del
disco de datos";
          LOCATE  23,  18: COLOR  15,  0:  PRINT "***  Pulse
alguna tecla para continuar ***": BEEP
pulsar1:
          LET tecla$ = INKEY$
          IF tecla$ = "" THEN GOTO pulsar1 ELSE GOTO menu
continuar:
          LOCATE  12,  14:  PRINT  "INGRESE  EL  NOMBRE  DEL
ARCHIVO (máximo 8 caracteres)"
repetir:
          LOCATE 14, 32: INPUT "NOMBRE = ", nombre$
          IF (LEN(nombre$) < 9)  AND (LEN(nombre$) > 0) THEN
GOTO verfdatos
          LOCATE  14,  43:  PRINT  STRING$(30,  "  "):  GOTO
repetir
verfdatos:      ON ERROR GOTO erroropen
          OPEN unidad$ + ":" + nombre$ + ".din" FOR INPUT AS
#1:
          CLOSE #1
          LOCATE 16, 20:  PRINT "El archivo  de datos:    ";
unidad$; ":"; nombre$; ".DIN ya existe"
          LOCATE 18, 20: COLOR 15, 5: PRINT " 1.-";   : COLOR
3, 5: PRINT " Recuperar los datos desde el archivo"
          LOCATE 20, 20: COLOR 15, 5: PRINT " 2.-";   : COLOR
3, 5: PRINT " Reemplazar los datos"
```

```
          LOCATE  22,  25:  PRINT  "Seleccione  una  de  las
opciones"; : COLOR 15, 5: PRINT " (1 a 2..)"
unodos:          tecla$ =  INKEY$: IF tecla$ = "" THEN  GOTO
unodos
          IF tecla$ = "1" THEN GOTO recuperar
          IF tecla$ = "" THEN GOTO unodos
          IF  tecla$ =  "2" THEN  GOTO generacion  ELSE GOTO
unodos
recuperar:
          PRINT : PRINT : CLOSE
          ON ERROR GOTO error1
          LOCATE  23, 25: COLOR  15, 5:  PRINT "***  Se está
recuperando los datos ***"
          OPEN unidad$ + ":" + nombre$ + ".din" FOR INPUT AS
#1
               DO UNTIL EOF(1)
                    INPUT #1, datos, N, m
                    FOR i = 0 TO 50
                         INPUT #1, entradauk(i), salidayk(i)
                    NEXT i
                    FOR i = 1 TO N
                         INPUT #1, parametroai(i)
                    NEXT i
                    PRINT : PRINT
                    FOR i = 1 TO m
                         INPUT #1, parametrobi(i)
                    NEXT i
                    INPUT #1, escalon, varianza, ruido
                    CLS
               LOOP
          CLOSE #1
ON N GOSUB primer, segundo, tercer, cuarto
primer:
matriza(1, 1) = parametroai(1)
matrizb(1, 1) = 1
matrizc(1, 1) = parametrobi(1)
GOTO seguir
segundo:
matriza(1, 1) = 0: matriza(1, 2) = 1
FOR j = 1 TO N
     matriza(2, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 1
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
tercer:
matriza(1, 1)  = 0: matriza(1,  3) = 0:  matriza(2, 1)  = 0:
matriza(2, 2) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1
FOR j = 1 TO N
```

```
     matriza(3, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 1
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
cuarto:
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(1, 4) = 0
matriza(2, 1) = 0: matriza(2, 2) = 0: matriza(2, 4) = 0
matriza(3, 1) = 0: matriza(3, 2) = 0: matriza(3, 3) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1: matriza(3, 4) = 1
FOR j = 1 TO N
     matriza(4, j) = parametroai(j)
NEXT j
matrizb(1, 1)  = 0: matrizb(2,  1) = 0:  matrizb(3, 1)  = 0:
matrizb(4, 1) = 1
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
seguir:
          CLS : COLOR 3, 5, 0
          FOR i =  1 TO 25: LOCATE i, 1: PRINT STRING$(79, "
"): NEXT i
          LOCATE 2,  12: COLOR 15,  5: PRINT "IDENTIFICACION
DE PARAMETROS EN VARIABLES DE ESTADO"
          LOCATE 4, 25: PRINT "JOSE RAUL MEJIA RECALDE"
          LOCATE 5, 28: PRINT "DATOS RECUPERADOS"
          LOCATE 6, 12: PRINT  "Orden del modelo =    ";  N:
LOCATE 6, 40: PRINT "Valor de m   ="; m
          LOCATE  7, 12: PRINT  "Escalón            =    ";
escalon: LOCATE 7, 40: PRINT "Varianza     ="; varianza
          LOCATE 8, 40: PRINT "Ruido        ="; ruido
          COLOR 3, 5: LOCATE 9, 10: PRINT " Matriz  A"
          LOCATE 10, 10: PRINT " *********": COLOR 15, 5
          fila = 0
          FOR i = 1 TO N
               col = 0
               FOR j = 1 TO N
                    col = col + 8
                    LOCATE 11 + fila,  10 + col: PRINT USING
"+#.###"; matriza(i, j)
                    IF j = N THEN fila = fila + 1
               NEXT j
          NEXT i
          COLOR 3, 5: LOCATE 16, 10: PRINT "Matriz B"
          LOCATE 17, 10: PRINT "********": COLOR 15, 5
          fila = 0
          FOR i = 1 TO N
               LOCATE 18 + fila,  18: PRINT USING  "+#.###";
matrizb(i, 1)
```

```
                fila = fila + 1
          NEXT i
          COLOR 3, 5: LOCATE 16, 27: PRINT "Matriz C"
          LOCATE 17, 27: PRINT "********": COLOR 15, 5
          col = 0
          FOR j = 1 TO m
               col = col + 8
               LOCATE 18, 26  + col:  PRINT USING  "+#.###";
matrizc(1, j)
          NEXT j
          FOR i = 22 TO 25: LOCATE i, 1: PRINT STRING$(79, "
"); : NEXT i
          LOCATE  22,  18: COLOR  15,  5:  PRINT "***  Pulse
alguna tecla para continuar ***"
          DO: LOOP WHILE INKEY$ = ""
          LOCATE 24,  24: PRINT "*** Espere  por favor ***":
FOR i = 1 TO 1000: NEXT i: GOTO menu
generacion:
          ON ERROR GOTO error1
          CHAIN "GENDATOS"
     CASE "3"
          CLS : COLOR 3, 5, 0
          FOR  i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, "
"): NEXT i
          LOCATE 3,  12: COLOR 15, 5:  PRINT "IDENTIFICACION
DE PARAMETROS EN VARIABLES DE ESTADO"
          LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
          LOCATE   7,  18:   PRINT  "ALGORITMO   DE  MINIMOS
CUADRADOS ORDINARIOS"
          IF (unidad$ <> "") AND  (nombre$ <> "") THEN  GOTO
seguir3
          LOCATE 20, 18: PRINT  "No se conoce la unidad  del
disco de datos": BEEP
          LOCATE  23,  18: COLOR  15,  5:  PRINT "***  Pulse
alguna tecla para continuar ***"
pulsar2:
          LET tecla$ = INKEY$
          IF tecla$ = "" THEN GOTO pulsar2 ELSE GOTO menu
seguir3:
          FOR i = 10 TO 25: LOCATE i, 1: PRINT STRING$(78, "
"); : NEXT i
          LOCATE 23,  18: COLOR  15, 5:  PRINT "***  Se está
enlazando al programa de MCO ***"
          ON ERROR GOTO error1
          CHAIN "MICUAOR"
     CASE "4"
          CLS : COLOR 3, 5, 0
          FOR i = 1 TO 25: LOCATE  i, 1: PRINT STRING$(79, "
"): NEXT i
          LOCATE 3, 12:  COLOR 15, 5: PRINT  "IDENTIFICACION
DE PARAMETROS EN VARIABLES DE ESTADO"
          LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
```

```
            LOCATE  7,  18:  PRINT  "ALGORITMO  DE  MINIMOS
CUADRADOS RECURSIVOS"
            IF (unidad$ <> "") AND  (nombre$ <> "") THEN  GOTO
seguir4
            LOCATE 20, 18: PRINT  "No se conoce la unidad  del
disco de datos ": BEEP
            LOCATE 23, 18: PRINT  "*** Pulse alguna tecla para
continuar ***"
pulsar3:
            LET tecla$ = INKEY$
            IF tecla$ = "" THEN GOTO pulsar3 ELSE GOTO menu
seguir4:
            FOR i = 10 TO 25: LOCATE i, 1: PRINT STRING$(78, "
"); : NEXT i
            LOCATE 23,  18: COLOR  15, 5:  PRINT "***  Se está
enlazando al programa de MCR ***"
            ON ERROR GOTO error1
            CHAIN "MICUARE"
            CASE "5"
            ON ERROR GOTO error1
            SHELL "GRAFICO.BAT"
            GOTO menu
error1:
            FOR i = 20 TO 25:
                 LOCATE i, 1: PRINT STRING$(79, " "); :
            NEXT i:
            BEEP:  COLOR  15,  5:  LOCATE 20,  10:  PRINT  "Ha
ocurrido algun problema durante el enlace con las rutinas "
            LOCATE 22,  13: PRINT  "Verifique que el  disco de
programas sea el correcto"
            LOCATE    23,    15:     PRINT   "Reinicialice    el
sistema..Presione cualquier tecla"
            LOCATE 24, 24: PRINT "Codigo de error = "; ERR;
pulsar7:          IF INKEY$ = "" THEN GOTO pulsar7 ELSE RESUME
NEXT
erroropen:
            FOR i = 20 TO 25
                 LOCATE i, 1: PRINT STRING$(79, " ");
            NEXT i
            BEEP: LOCATE 20, 15
            PRINT  "El archivo  de datos a  utilizar es:     ";
unidad$; ":"; nombre$; ".DIN"
            FOR i = 1 TO 5: NEXT i
            RESUME generacion
       CASE "6"
            EXIT DO
       CASE ELSE
            BEEP
       END SELECT
LOOP
END
```

```
'*************************************************************
'
'                 RUTINA DE GENERACION DE DATOS
'
'                   JOSE RAUL MEJIA RECALDE
'
'ARCHIVO:GENDATOS.BAS
'
'*************************************************************
COMMON unidad$, nombre$, nombres$, n, m, datos
DIM entradauk(-5 TO 50), salidayk(-5 TO 50), ruido(-5 TO 50), parametroai(4), parametrobi(4)
DIM partea(4), parteb(4)
DIM matriza(5, 5), matrizb(5, 5), matrizc(5, 5)
CLS : COLOR 3, 5, 0
FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE 3, 12: COLOR 15, 5: PRINT "IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 23: PRINT "RUTINA DE GENERACION DE DATOS"
IF (unidad$ <> "") AND (nombre$ <> "") THEN GOTO generardatos ELSE BEEP
LOCATE 20, 17: PRINT "*** No se ha definido el archivo de datos ***"
LOCATE 22, 20: PRINT "*** Espere un momento por favor ***"
FOR i = 1 TO 2000: NEXT i: GOTO regresarmenu
generardatos:
LOCATE 10, 9: PRINT " Se va a generar las señales de entrada y salida para un modelo  "
LOCATE 11, 20: PRINT " Representado en Variables de Estado"
LOCATE 13, 22: PRINT "de la forma canónica controlable"
LOCATE 15, 24: COLOR 3, 5: PRINT " X(k+1) = A.X(k) + B.U(k)"
LOCATE 17, 24: PRINT " Y(k+1) = C.X(k) "
LOCATE 23, 20: PRINT "*** Pulse  alguna tecla para continuar ***"
DO: LOOP WHILE INKEY$ = ""
'FOR i = 1 TO 4000: NEXT i
CLS : COLOR 3, 5, 0
FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE 3, 12: COLOR 15, 5: PRINT "IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 23: PRINT "RUTINA DE GENERACION DE DATOS"
LOCATE 12, 18: PRINT " Introducir el  orden del modelo,"; : COLOR 3, 5: PRINT " (0<n<=4)"
repetirn:
FOR i = 13 TO 20: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i: BEEP
COLOR 15, 5: LOCATE 14, 10: INPUT " El valor de n = "; n
IF n <= 0 THEN PRINT : PRINT TAB(15); "No se puede generar un modelo, intente de nuevo": FOR i = 1 TO 1000: NEXT i: GOTO repetirn: BEEP
```

```
IF  n > 4 THEN PRINT :  PRINT TAB(15); "Orden del modelo muy
alto, intente  de nuevo": FOR  i = 1  TO 1000: NEXT  i: GOTO
repetirn:
BEEP
LOCATE 16, 24: PRINT "Introducir el valor de m";  : COLOR 3,
5: PRINT " (0<m<=n)": COLOR 15, 5
repetirm:
FOR i =  18 TO 25: LOCATE  i, 1: PRINT  STRING$(79, " ");  :
NEXT i: BEEP
LOCATE 18, 10: INPUT "El valor de m = "; m
IF m <=  0 THEN PRINT : PRINT TAB(15);  "No se puede generar
un  modelo, intente de  nuevo": FOR i  = 1 TO  1000: NEXT i:
GOTO repetirm: BEEP
IF  m  > n  THEN  PRINT  :  PRINT  TAB(15); "Valor  alto  de
m..Intente  de  nuevo": FOR  i  = 1  TO 1000:  NEXT  i: GOTO
repetirm: BEEP
cambiar:
CLS : COLOR 3, 5, 0
FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE  3,  12:  COLOR  15,  5:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 23: PRINT "RUTINA DE GENERACION DE DATOS"
LOCATE 9,  5: PRINT  "Ingresar la  matriz A  (forma canónica
controlable)"; : COLOR 3, 5: PRINT " -4=< an,j <=4 "
COLOR 15, 5
fuerarango1:
FOR i =  10 TO 24:  LOCATE i, 1:  PRINT STRING$(79, "  "); :
NEXT i: BEEP
ON n GOSUB primer, segundo, tercer, cuarto
primer:
FOR i = 1 TO n
     LOCATE 10  + i, 5:  PRINT "a (1,";  i; ") =  "; : INPUT
parametroai(i)
     IF parametroai(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo,  valor  bajo": FOR  j  =  1  TO  1000: NEXT  j:  GOTO
fuerarango1
     IF parametroai(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo,  valor  alto": FOR  j  =  1  TO 1000:  NEXT  j:  GOTO
fuerarango1
NEXT i
matriza(1, 1) = parametroai(1)
matrizb(1, 1) = 1
LOCATE 16,  5: PRINT "Ingresar  la matriz C  (forma canónica
controlable)"; : COLOR 3, 5: PRINT " (-4=<c1,j<=4)"
COLOR 15, 5
fuerarango21:
FOR i  = 17 TO  24: LOCATE i,  1: PRINT STRING$(79, "  "); :
NEXT i: BEEP
FOR i = 1 TO m
     LOCATE 17  + i, 5:  PRINT "c  (1,"; i; ")  ="; :  INPUT
parametrobi(i)
```

```
     IF parametrobi(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo, valor bajo": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango21
     IF parametrobi(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo, valor alto": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango21
NEXT i
matrizc(1, 1) = parametrobi(1)
GOTO seguir
segundo:
FOR i = 1 TO n
     LOCATE  10 + i,  5: PRINT "a  (2,"; i; ")  = "; : INPUT
parametroai(i)
     IF parametroai(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo, valor bajo": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango1
     IF parametroai(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo, valor alto": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango1
NEXT i
matriza(1, 1) = 0: matriza(1, 2) = 1
FOR j = 1 TO n
     matriza(2, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 1
LOCATE 16,  5: PRINT "Ingresar  la matriz C  (forma canónica
controlable)"; : COLOR 3, 5: PRINT " -4=< c1,j <=4 "
COLOR 15, 5
fuerarango22:
FOR i =  17 TO 24:  LOCATE i, 1:  PRINT STRING$(79, " ");  :
NEXT i: BEEP
FOR i = 1 TO m
     LOCATE 17  + i, 5:  PRINT "c  (1,"; i; ")  ="; :  INPUT
parametrobi(i)
     IF parametrobi(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo, valor bajo": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango22
     IF parametrobi(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo, valor alto": FOR j = 1 TO 1000: NEXT j: GOTO
fuerarango22
NEXT i
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
tercer:
matriza(1, 1)  = 0: matriza(1,  3) =  0: matriza(2, 1)  = 0:
matriza(2, 2) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1
FOR i = 1 TO n
     LOCATE  10 + i,  5: PRINT "a  (3,"; i; ")  = "; : INPUT
parametroai(i)
```

```
     IF parametroai(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo,  valor  bajo":  FOR  j  = 1  TO  1000:  NEXT  j: GOTO
fuerarango1
     IF parametroai(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo, valor  alto":  FOR  j  = 1  TO  1000:  NEXT  j:  GOTO
fuerarango1
NEXT i
FOR j = 1 TO n
     matriza(3, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 1
LOCATE  16, 5: PRINT  "Ingresar la matriz  C (forma canónica
controlable)"; : COLOR 3, 5: PRINT " (-4=<ci,j<=4)"
COLOR 15, 5
fuerarango23:
FOR i  = 17 TO  24: LOCATE i, 1:  PRINT STRING$(79, "  "); :
NEXT i: BEEP
FOR i = 1 TO m
     LOCATE  17 +  i, 5: PRINT  "c (1,";  i; ")  ="; : INPUT
parametrobi(i)
     IF parametrobi(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo,  valor bajo":  FOR  j  =  1 TO  1000:  NEXT  j:  GOTO
fuerarango23
     IF parametrobi(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo,  valor  alto": FOR  j  =  1  TO  1000: NEXT  j:  GOTO
fuerarango23
NEXT i
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
cuarto:
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(1, 4) = 0
matriza(2, 1) = 0: matriza(2, 2) = 0: matriza(2, 4) = 0
matriza(3, 1) = 0: matriza(3, 2) = 0: matriza(3, 3) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1: matriza(3, 4) = 1
FOR i = 1 TO n
     LOCATE  10 + i,  5: PRINT "a  (4,"; i; ")  = "; : INPUT
parametroai(i)
     IF parametroai(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo, valor  bajo":  FOR  j  =  1 TO  1000:  NEXT  j:  GOTO
fuerarango1
     IF parametroai(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo,  valor alto":  FOR  j  =  1 TO  1000:  NEXT  j:  GOTO
fuerarango1
NEXT i
FOR j = 1 TO n
     matriza(4, j) = parametroai(j)
NEXT j
matrizb(1,  1) = 0:  matrizb(2, 1) =  0: matrizb(3, 1)  = 0:
matrizb(4, 1) = 1
```

```
LOCATE 16, 5:  PRINT "Ingresar la  matriz C (forma  canónica
controlable)"; : COLOR 3, 5: PRINT " (-4=<oi,j<=4)"
COLOR 15, 5
fuerarango24:
FOR  i = 17 TO  24: LOCATE i,  1: PRINT STRING$(79,  " "); :
NEXT i: BEEP
FOR i = 1 TO m
     LOCATE 17  + i,  5: PRINT  "c (1,"; i;  ") =";  : INPUT
parametrobi(i)
     IF parametrobi(i) < -4  THEN PRINT TAB(20); "Intente de
nuevo,  valor bajo":  FOR  j  =  1 TO  1000:  NEXT  j:  GOTO
fuerarango24
     IF parametrobi(i)  > 4 THEN PRINT  TAB(20); "Intente de
nuevo,  valor  alto": FOR  j  =  1  TO  1000: NEXT  j:  GOTO
fuerarango24
NEXT i
FOR j = 1 TO m
     matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir
seguir:
CLS : COLOR 3, 5, 0
FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE  3,  12:  COLOR  15,  5:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 23: PRINT "RUTINA DE GENERACION DE DATOS"
COLOR 3, 5: LOCATE 9, 10: PRINT " Matriz  A"
LOCATE 10, 10: PRINT " *********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
     col = 0
       FOR j = 1 TO n
          col = col + 8
          LOCATE 11 + fila, 10 + col: PRINT  USING "+#.###";
matriza(i, j)
          IF j = n THEN fila = fila + 1
     NEXT j
NEXT i
COLOR 3, 5: LOCATE 16, 10: PRINT "Matriz B"
LOCATE 17, 10: PRINT "********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
     LOCATE 18 + fila,  18: PRINT USING "+#.###"; matrizb(i,
1)
     fila = fila + 1
NEXT i
COLOR 3, 5: LOCATE 16, 27: PRINT "Matriz C"
LOCATE 17, 27: PRINT "********": COLOR 15, 5
col = 0
FOR j = 1 TO m
     col = col + 8
```

```
     LOCATE 18,  26 + col: PRINT  USING "+#.###"; matrizc(1,
j)
NEXT j
COLOR 7, 5:  LOCATE 24, 15:  PRINT "Desea cambiar  los datos
introducidos"; : COLOR 15, 5: PRINT " (S/N)? ";
sino: tecla$ = INKEY$:
     IF tecla$ = "s" OR tecla$ = "S" THEN GOTO cambiar
     IF tecla$ = "" THEN GOTO sino
     IF  tecla$ =  "n" OR tecla$  = "N"  THEN GOTO continuar
ELSE GOTO sino
continuar:
CLS : COLOR 3, 5, 0
FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
DO
COLOR 15, 5:  LOCATE 4, 11:  PRINT "  Menu de generación  de
señales de entrada": COLOR 7, 5
PRINT
COLOR  15, 5: PRINT  TAB(20); " 1.-  "; : COLOR  3, 5: PRINT
"Entrada Escalón"
COLOR 15,  5: PRINT TAB(20); "  2.-"; : COLOR 3,  5: PRINT "
Ruido blanco"
COLOR 15,  5: PRINT TAB(20); "  3.-"; : COLOR 3,  5: PRINT "
Ruido"
PRINT
COLOR 15,  5: PRINT TAB(20); "  4.-"; : COLOR 3,  5: PRINT "
Salir"
PRINT
PRINT : PRINT TAB(15); " Seleccione una de las  opciones"; :
COLOR 15, 5: PRINT " (1 a 4):"
ch$ = INPUT$(1)
     SELECT CASE ch$
          CASE "1"
               CLS : FOR  i = 1  TO 25: LOCATE  i, 1:  PRINT
STRING$(79, " "): NEXT i
               COLOR 15, 5: LOCATE 3, 15: INPUT " Ingrese la
amplitud del escalón: "; escalon
               LOCATE   22,  11:   PRINT  "***   Espere  por
favor...Se está generando la señal de entrada ***": FOR  i =
1 TO 2000: NEXT i
               FOR i = 0 TO 50
                    entradauk(i) = escalon
               NEXT i
               GOSUB generarsalida
               GOSUB archivodatos
               CLS :
          EXIT DO
          CASE "2"
                CLS : FOR  i = 1 TO  25: LOCATE i, 1:  PRINT
STRING$(79, " "): NEXT i
               LOCATE 5, 20: COLOR 15,  5: PRINT "Introducir
el valor de la varianza (V > 0)"
```

```
               LOCATE 7,  22: INPUT "El valor  de varianza =
"; varianza
               IF varianza = 0 THEN
               LOCATE 20, 20: PRINT "No existe ruido"
               ELSEIF varianza < 0 THEN
                    LOCATE 20,  20: PRINT " La  varianza del
ruido es > 0 "
               ELSE
               LOCATE   22,   11:   PRINT  "***   Espere  por
favor...se está generando la señal de entrada ***"
                    D1 = SQR(varianza)
                    FOR i = 0 TO 50
                         Z = 0
                         FOR j = 1 TO 12
                         RANDOMIZE TIMER
                         Z = Z + RND: NEXT j
                         ruido(i) = (Z - 6) * D1 / 2
                    NEXT i
                    FOR i = 0 TO 50
                         entradauk(i) = ruido(i)
                    NEXT i
               END IF
               GOSUB generarsalida
               GOSUB archivodatos
               CLS :
          EXIT DO
          CASE "3"
                CLS : FOR i  = 1 TO  25: LOCATE i, 1:  PRINT
STRING$(79, " "): NEXT i
               LOCATE 3, 15: COLOR 15, 5: INPUT " Ingrese la
amplitud del ruido "; ruido
               LOCATE     22,    11:    PRINT    "***    Espere
porfavor...se está generando la señal de entrada ***"
               FOR i = 0 TO 50
                    entradauk(i) = ruido * RND
               NEXT i
               GOSUB generarsalida
               GOSUB archivodatos
               CLS :
          EXIT DO
          CASE "4"
               GOTO regresarmenu
          CASE ELSE
               BEEP
     END SELECT
LOOP
ON n GOSUB uno, dos, tres, cuatro
uno:
matriza(1,  1)  =  parametroai(1):  matrizb(1,  1)  =  1:
matrizc(1, 1) = parametrobi(1)
GOTO seguir1
dos:
```

```
matriza(1, 1) = 0: matriza(1, 2) = 1
FOR j = 1 TO n
      matriza(2, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 1
FOR j = 1 TO m
      matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir1
tres:
matriza(1, 1) =  0: matriza(1, 3)  = 0: matriza(2,  1) =  0:
matriza(2, 2) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1
FOR j = 1 TO n
      matriza(3, j) = parametroai(j)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 1
FOR j = 1 TO m
      matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir1
cuatro:
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(1, 4) = 0
matriza(2, 1) = 0: matriza(2, 2) = 0: matriza(2, 4) = 0
matriza(3, 1) = 0: matriza(3, 2) = 0: matriza(3, 3) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1: matriza(3, 4) = 1
FOR j = 1 TO n
      matriza(4, j) = parametroai(j)
NEXT j
matrizb(1, 1) =  0: matrizb(2,  1) = 0:  matrizb(3, 1) =  0:
matrizb(4, 1) = 1
FOR j = 1 TO m
      matrizc(1, j) = parametrobi(j)
NEXT j
GOTO seguir1
seguir1:
CLS : FOR i =  1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "):
NEXT i
COLOR  15,  5:  LOCATE  2,  12:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 4, 30: PRINT "Archivo de  datos es:  "; unidad$ + ":"
+ nombre$ + ".DIN"; CHR$(13)
COLOR 3, 5: LOCATE 5, 17: PRINT "****  D a t o s   I n i c i
a l e s ***": COLOR 15, 5
LOCATE 6, 10: PRINT  "Orden del modelo =    "; n:  LOCATE 6,
40: PRINT "Valor de m          =  "; m
LOCATE 7, 10: PRINT "Escalón             =   "; escalon: LOCATE
7, 40: PRINT "Varianza          =  "; varianza
LOCATE 8, 40: PRINT "Ruido               =  "; ruido
COLOR 3, 5: LOCATE 9, 10: PRINT " Matriz  A"
LOCATE 10, 10: PRINT " *********": COLOR 15, 5
fila = 0
```

```
FOR i = 1 TO n
      col = 0
        FOR j = 1 TO n
            col = col + 8
            LOCATE 11  + fila, 10 + col: PRINT USING "+#.###";
matriza(i, j)
            IF j = n THEN fila = fila + 1
      NEXT j
NEXT i
COLOR 3, 5: LOCATE 16, 10: PRINT "Matriz B"
LOCATE 17, 10: PRINT "********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
      LOCATE 18 + fila,  18: PRINT USING "+#.###"; matrizb(i,
1)
      fila = fila + 1
NEXT i
COLOR 3, 5: LOCATE 16, 27: PRINT "Matriz C"
LOCATE 17, 27: PRINT "********": COLOR 15, 5
col = 0
FOR j = 1 TO m
      col = col + 8
      LOCATE 18,  26 + col: PRINT  USING "+#.###"; matrizc(1,
j)
NEXT j
LOCATE 23, 20: PRINT "*** Pulse alguna tecla  para continuar
***"
DO: LOOP WHILE INKEY$ = ""
LOCATE 24, 25: PRINT "Desea  un reporte impreso..."; : PRINT
"(S/N)"; : BEEP
sino1: tecla$ = INKEY$:
        IF tecla$ = "s" OR tecla$ = "S" THEN GOTO imprimir
        IF tecla$ = " " THEN GOTO sino1
        IF  tecla$  =  "n"  OR  tecla$   =  "N"  THEN   GOTO
regresarmenu ELSE GOTO sino1
imprimir:
ON ERROR GOTO errorimp:
LPRINT , TAB(13); "E S C U E L A   P O L I T E C N I C A   N
A C I O N A L"; CHR$(13)
LPRINT , TAB(9); "F A C U  L T A D   D E   I N G E N I E R I
A   E L E C T R I C A"; CHR$(13)
LPRINT , TAB(30); "TESIS DE GRADO"; CHR$(13)
LPRINT , TAB(15); "IDENTIFICACION DE PARAMETROS EN VARIABLES
DE ESTADO"; CHR$(13)
LPRINT , TAB(26); "JOSE RAUL MEJIA RECALDE"; CHR$(13)
LPRINT , TAB(23); "RUTINA DE GENERACION DE DATOS"; CHR$(13)
LPRINT ,  TAB(22); "Archivo de  datos :  "; unidad$ +  ":" +
nombre$ + ".DIN"; CHR$(13)
LPRINT , TAB(17); "**** D a t o s   i n i c i a l e s ***";
CHR$(13)
LPRINT , TAB(24); "Orden del modelo       = "; n
LPRINT , TAB(24); "Valor de m             = "; m
```

```
LPRINT , TAB(24); "Magnitud Escalón        = "; escalon
LPRINT , TAB(24); "Varianza (Ruido Blanco)= "; varianza
LPRINT , TAB(24); "Magnitud Ruido               =  "; ruido;
CHR$(13)
LPRINT , TAB(10); "Matriz  A"
LPRINT , TAB(10); "*********"
FOR i = 1 TO n
     IF n = 1 THEN
          LPRINT , TAB(18); USING "+#.###"; matriza(i, 1)
     ELSEIF n = 2 THEN
          LPRINT , TAB(18); USING "+#.###"; matriza(i, 1);
TAB(30); matriza(i, 2)
     ELSEIF n = 3 THEN
          LPRINT , TAB(18); USING "+#.###"; matriza(i, 1);
TAB(30); matriza(i, 2); TAB(42); matriza(i, 3)
     ELSEIF n = 4 THEN
          LPRINT , TAB(18); USING "+#.###"; matriza(i, 1);
TAB(30); matriza(i, 2); TAB(42); matriza(i, 3); TAB(54);
matriza(i, 4)
     END IF
NEXT i
LPRINT , TAB(10); "Matriz B"
LPRINT , TAB(10); "********"
FOR i = 1 TO n
     LPRINT , TAB(18); USING "+#.###"; matrizb(i, 1)
NEXT i
LPRINT , TAB(10); "Matriz C"
LPRINT , TAB(10); "********"
IF m = 1 THEN
     LPRINT , TAB(18); USING "+#.###"; matrizc(1, 1)
ELSEIF m = 2 THEN
     LPRINT , TAB(18); USING "+#.###"; matrizc(1, 1);
TAB(30); matrizc(1, 2)
ELSEIF m = 3 THEN
     LPRINT , TAB(18); USING "+#.###"; matrizc(1, 1);
TAB(30); matrizc(1, 2); TAB(42); matrizc(1, 3)
ELSEIF m = 4 THEN
     LPRINT , TAB(18); USING "+#.###"; matrizc(1, 1);
TAB(30); matrizc(1, 2); TAB(42); matrizc(1, 3); TAB(54);
matrizc(1, 4)
END IF
regresarmenu:
ON ERROR GOTO errorenlace
CLS : LOCATE 24, 20: COLOR 15, 5: PRINT "*** Regresar al
menú principal ***"
CHAIN "ipve"
generarsalida:
FOR i = 1 TO 25
     LOCATE i, 1: PRINT STRING$(79, " ");
NEXT i
LOCATE 16, 24: PRINT "La señal de salida tiene la forma:"
LOCATE 18, 28: PRINT "Y(k+1) = C.X(k)"
```

```
LOCATE 20, 20: PRINT "C = [a1......an b1.......bm]"
LOCATE 22, 20: PRINT "X(k) = [y(k-1)......y(k-n) u(k-1)......u(k-m)]"
LOCATE 24, 10: PRINT "*** Espere porfavor...se está generando la señal de salida ***": COLOR 7, 5
FOR k = 0 TO 50
     FOR i = 1 TO n
          partea(i) = parametroai(i) * salidayk(k - i)
     NEXT i
     FOR i = 1 TO m
          parteb(i) = parametrobi(i) * entradauk(k - i)
     NEXT i
     sumaai = 0
     sumabi = 0
     FOR i = 1 TO n
          sumaai = sumaai + partea(i)
     NEXT i
     FOR i = 1 TO m
          sumabi = sumabi + parteb(i)
     NEXT i
     salidayk(k) = sumaai + sumabi
NEXT k
RETURN
archivodatos:
CLS : COLOR 3, 5, 0: FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE 24, 21: COLOR 15, 5: PRINT "*** Se está almacenando los datos ***": COLOR 7, 5
ON ERROR GOTO errordisco: CLOSE
OPEN unidad$ + ":" + nombre$ + ".din" FOR OUTPUT AS #1
     WRITE #1, datos, n, m
     FOR i = 0 TO 50
          WRITE #1, entradauk(i), salidayk(i)
     NEXT i
     FOR i = 1 TO n
          WRITE #1, parametroai(i)
     NEXT i
     FOR i = 1 TO m
          WRITE #1, parametrobi(i)
     NEXT i
     WRITE #1, escalon, varianza, ruido
CLOSE #1
PRINT : PRINT :
RETURN
errordisco:
FOR i = 20 TO 25: LOCATE i, 1: PRINT STRING$(79, " "); :
NEXT i
BEEP: LOCATE 22, 17: COLOR 15, 5: PRINT "Existe algun error durante el acceso al disco"
PRINT TAB(21); "*** Pulse alguna tecla para continuar ***"
LOCATE 24, 28: PRINT "Código de error = "; ERR;
```

```
pulsar3: IF  INKEY$  =  "" THEN  GOTO  pulsar3  ELSE  RESUME
archivodatos
errorimp:
FOR i = 20 TO 25
     LOCATE i, 1: PRINT STRING$(79, " ");
NEXT i
DEF SEG = &H40
statusport& = PEEK(9) * 256 + PEEK(8) + 1
IF INP(statusport&) <> 233 THEN
     LOCATE 23, 27: PRINT "La impresora no esta lista"
     LOCATE 25, 21: COLOR 15,  5: PRINT "Desea continuar con
la impresión...S/N";
pulsar5:  LET tecla$  = INKEY$:  IF tecla$  = " "  THEN GOTO
pulsar5
     IF tecla$ = "s" OR tecla$ = "S" THEN GOTO contimp
     IF tecla$ = " " THEN GOTO pulsar5
     IF tecla$ = "n" OR tecla$ = "N" THEN  GOTO regresarmenu
ELSE RESUME errorimp
contimp:
     FOR i = 24 TO 25
          LOCATE i, 1: PRINT STRING$(79, " ");
     NEXT i
     LOCATE  25,  10:  PRINT  "Aliste  la  impresora...pulse
cualquier tecla para continuar";
     DO: LOOP WHILE INKEY$ = " "
     DO: LOOP UNTIL INP(statusport&) = 223
END IF
DEF SEG
RESUME regresarmenu:
errorenlace:
FOR  i = 20 TO  25: LOCATE i,  1: PRINT STRING$(79,  " "); :
NEXT i
BEEP:  LOCATE  22, 15:  COLOR  15,  5: PRINT  "Existe  error
durante el enlace con el programa"
PRINT TAB(20); "Principal... presiona cualquier tecla"
LOCATE 24, 28: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar4: IF INKEY$ = "" THEN GOTO pulsar4 ELSE RESUME NEXT
CLS :  LOCATE 23,  24:  COLOR 15,  5: PRINT  "***Espere  por
favor***"
CHAIN "IPVE"
```

```
'***********************************************************
'
' RUTINA DE IDENTIFICACION DE PARAMETROS POR EL ALGORITMO DE
'
'                MINIMOS CUADRADOS ORDINARIOS
'
'                   JOSE RAUL MEJIA RECALDE
'ARCHIVO:MICUAOR.BAS
'
'***********************************************************
COMMON unidad$, nombre$, nombres$, n, m, muestras, datos
DIM entradauk(-5 TO 50), salidayk(-5 TO 50), ruido(-5 TO 50)
DIM parametroai(4), parametrobi(4), partea(4), parteb(4)
DIM parametroaic(4), parametrobic(4), salidaykc(-5 TO 50)
CLS : COLOR 3, 5, 0
LOCATE 3, 12: COLOR 15, 5: PRINT "IDENTIFICACION DE
PARAMETROS EN VARIABLES DE ESTADO"
PRINT : PRINT TAB(25); "JOSE RAUL MEJIA RECALDE": COLOR 7, 5
PRINT : COLOR 15, 5: PRINT TAB(18); "ALGORITMO DE MINIMOS
CUADRADOS ORDINARIOS"
IF (unidad$ <> "") AND (nombre$ <> "") THEN GOTO
recuperardatos ELSE BEEP
LOCATE 20, 17: PRINT "*** No se ha definido el archivo de
datos ***"
LOCATE 22, 20: PRINT "*** Espere un momento por favor ***"
FOR i = 1 TO 2000: NEXT i: GOTO regresarmenu
recuperardatos:
ON ERROR GOTO recdatos
COLOR 15, 5:
LOCATE 10, 17: PRINT "El archivo de datos es          : ";
unidad$ + ":" + nombre$ + ".DIN"
LOCATE 20, 14: PRINT "*** Se está recuperando el archivo de
datos ***":
OPEN unidad$ + ":" + nombre$ + ".din" FOR INPUT AS #1
INPUT #1, datos, n, m
DO UNTIL EOF(1)
     FOR i = 0 TO 50: INPUT #1, entradauk(i), salidayk(i)
          entradauk(i) = entradauk(i)
          salidayk(i) = salidayk(i)
     NEXT i
     FOR i = 1 TO n: INPUT #1, parametroai(i)
          parametroai(i) = parametroai(i)
     NEXT i
     FOR i = 1 TO m: INPUT #1, parametrobi(i)
          parametrobi(i) = parametrobi(i)
     NEXT i
INPUT #1, escalon, varianza, ruido
LOOP: CLOSE #1
nombres$ = nombre$
FOR i  = 12 TO  25: LOCATE i, 1:  PRINT STRING$(79, "  "); :
NEXT i
```

```
LOCATE 12, 17: PRINT "El archivo de salida se llama : ";
unidad$ + ":" + nombres$ + ".DMO"
LOCATE 14, 17: PRINT "Desea cambiar el nombre del archivo de
salida..(S/N)?": BEEP
sn1: LET tecla$ = INKEY$: IF tecla$ = "" THEN GOTO sn1
     IF tecla$ = "s" OR tecla$ = "S" THEN GOTO nuevoarchivo
     IF tecla$ = "n" OR tecla$ = "N" THEN GOTO igualarchivo
ELSE GOTO sn1
nuevoarchivo:
LOCATE 16, 17: PRINT "Nuevo nombre del archivo (máx. 8
caracteres)"
repetir1:
LOCATE 18, 32: INPUT "Nombre = ", nombres$:
IF (LEN(nombres$) < 9) AND (LEN(nombres$) > 0) THEN GOTO
verfdatos
LOCATE 18, 43: PRINT STRING$(30, " "): BEEP: GOTO repetir1
verfdatos:
ON ERROR GOTO recdatos
LOCATE 20, 17: PRINT "El archivo de salida se llama:"; uni$
+ ":" + nombres$ + ".DMO"
igualarchivo:
LOCATE 24, 22: PRINT "*** Espere un momento por favor ***"
FOR i = 1 TO 2500: NEXT i
CLS : COLOR 3, 5, 0: COLOR 15, 5
LOCATE 3, 12: PRINT "IDENTIFICACION DE PARAMETROS EN
VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 18: PRINT "ALGORITMO DE MINIMOS CUADRADOS
ORDINARIOS"
LOCATE 9, 20: PRINT "Seleccionar tipo de proceso"
LOCATE 11, 20: PRINT "1.-"; : COLOR 3, 5: PRINT " Mínimos
Cuadrados Ordinarios Determinístico"
LOCATE 12, 20: COLOR 15, 5: PRINT "2.-"; : COLOR 3, 5: PRINT
" Mínimos Cuadrados Ordinarios Probabilísticos"
LOCATE 14, 20: COLOR 15, 5: PRINT "3.-"; : COLOR 3, 5: PRINT
" Salir"
PRINT : PRINT TAB(30); "Seleccione una de las opciones"; :
COLOR 15, 5: PRINT " (1 a 3)...": COLOR 7, 5
seleccionar: LET tecla$ = INKEY$: IF tecla$ = "" THEN GOTO
seleccionar
     IF tecla$ = "1" THEN GOTO deterministicos
     IF tecla$ = " " THEN GOTO seleccionar
     IF tecla$ = "2" THEN GOTO estocasticos
     IF tecla$ = "3" THEN GOTO regresarmenu ELSE GOTO
seleccionar
estocasticos:
FOR i = 0 TO 50
     RANDOMIZE TIMER
     salidayk(i) = salidayk(i) + RND * .1
NEXT i
deterministicos:
```

```
LOCATE 18, 25: COLOR  15, 5: PRINT "El orden  del modelo es=
"; n
LOCATE 19, 25: PRINT "Desea cambiar el orden (S/N)?": BEEP
sn: LET tecla$ = INKEY$: IF tecla$ = "" THEN GOTO sn
     IF tecla$ = "s" OR tecla$ = "S" THEN GOTO nuevomodelo
     IF tecla$  = "n" OR tecla$ =  "N" THEN GOTO igualmodelo
ELSE GOTO sn
nuevomodelo:
LOCATE 20, 25: INPUT "Ingrese el valor de n="; n
LET m = n
igualmodelo:
LOCATE 21, 2:  PRINT "El número  de muestras debe  ser mayor
que el doble del orden del modelo"
LOCATE 22,  15: PRINT  "Y  menor que  un valor  tomado  como
50-2*n"
repetirmuestras:
LOCATE  23,  20: INPUT  "Ingrese  el  número de  muestras=";
muestras: BEEP
IF muestras >= 50 - 2 * n THEN GOTO repetirmuestras
IF muestras < 2 * n THEN GOTO repetirmuestras
DIM  matrizyn(1  TO muestras  -  n +  1,  1),  matrizfi(1 TO
muestras - n + 1, 1  TO n + m), matrizfit(1  TO n + m, 1  TO
muestras - n + 1)
DIM matrizfitfi(1 TO n + m, 1 TO n + m), matrizfity(1 TO n +
m, 1), matriztheta(1 TO n + m, 1)
ON ERROR GOTO errorproceso
CLS  : LOCATE  23,  18: COLOR  15,  5: PRINT  "**** Se  está
ejecutando el Algoritmo ****": COLOR 7, 5
TINICIO = TIMER
FOR i = 1 TO muestras - n + 1
     matrizyn(i, 1) = salidayk(n - 1 + i)
NEXT i
FOR i = 1 TO muestras - n + 1
     FOR j = 1 TO n + m
          IF j >=  n + 1 THEN matrizfi(i, j) = entradauk(2 *
n - j + i - 1): GOTO continuar
          matrizfi(i, j) = salidayk(n - j + i - 1)
continuar:
NEXT j, i
FOR i = 1 TO n + m
     FOR j = 1 TO muestras - n + 1
     matrizfit(i, j) = matrizfi(j, i)
NEXT j, i
FOR i = 1 TO n + m
     FOR j = 1 TO n + m
     matrizfitfi(i, j) = 0
          FOR k = 1 TO muestras - n + 1
               matrizfitfi(i,  j)  =  matrizfitfi(i,  j)  +
matrizfit(i, k) * matrizfi(k, j)
          NEXT k
     NEXT j
NEXT i
```

```
GOSUB matrizinversa
FOR i = 1 TO n + m
     matrizfity(i, 1) = 0
     FOR k = 1 TO muestras - n + 1
          matrizfity(i, 1)  = matrizfit(i, k)  * matrizyn(k,
1) + matrizfity(i, 1)
      NEXT k
NEXT i
FOR i = 1 TO n + m
     matriztheta(i, 1) = 0
     FOR k = 1 TO n + m
          matriztheta(i,   1)   =   matrizfitfi(i,   k)   *
matrizfity(k, 1) + matriztheta(i, 1)
     NEXT k
NEXT i
GOSUB salidacalculada
GOSUB archivoresultados
GOSUB pantalla
regresarmenu:
ON ERROR GOTO errorenlace
CLS  : LOCATE 24,  20: COLOR 15,  5: PRINT  "*** Regresar al
menú principal ***": COLOR 7, 5
CHAIN "ipve"
archivoresultados:
ON ERROR GOTO errordisco
LOCATE 23,  18: COLOR 15, 5: PRINT  "*** Se está almacenando
los resultados ***": COLOR 7, 5
CLOSE
OPEN unidad$ + ":" + nombres$ + ".dmo" FOR OUTPUT AS #2
     WRITE #2, datos, n, m, muestras
     FOR i = 0 TO 50
          WRITE #2, salidayk(i), salidaykc(i)
     NEXT i
     FOR i = 1 TO n + m
          WRITE #2, matriztheta(i, 1)
     NEXT i
CLOSE #2
CLOSE : TFINAL = TIMER
RETURN
salidacalculada:
FOR i = 1 TO n + m
     IF i <= n THEN parametroaic(i) = matriztheta(i, 1) ELSE
parametrobic(i - n) = matriztheta(i, 1)
NEXT i
FOR k = 0 TO 50
     FOR i = 1 TO n
          partea(i) = parametroaic(i) * salidaykc(k - i)
     NEXT i
     FOR i = 1 TO m
          parteb(i) = parametrobic(i) * entradauk(k - i)
     NEXT i
     sumaai = 0
```

```
      sumabi = 0
      FOR i = 1 TO n
           sumaai = sumaai + partea(i)
      NEXT i
      FOR i = 1 TO m
           sumabi = sumabi + parteb(i)
      NEXT i
      salidaykc(k) = sumaai + sumabi
NEXT k
RETURN
matrizinversa:
valorepsilon:
epsilon = .0001
max = n + m
pivote = 0
DIM irow(8), jcol(8), jord(8), y(8)
repetirvalor:
IF n + m >= 50 THEN GOTO repetirvalor
determinante = 1
FOR k = 1 TO n + m
      km1 = k - 1
      pivote = 0
      FOR i = 1 TO n + m
           FOR j = 1 TO n + m
                IF k = 1 THEN GOTO valorabsoluto
                FOR iscan = 1 TO km1
                     FOR jscan = 1 TO km1
                          IF i = irow(iscan) GOTO continuar1
                          IF j = jcol(jscan) GOTO continuar1
                     NEXT jscan
                NEXT iscan
valorabsoluto:
                IF  (ABS(matrizfitfi(i,  j)) <=  ABS(pivote))
THEN GOTO continuar1
                pivote = matrizfitfi(i, j)
                irow(k) = i
                jcol(k) = j
continuar1:
           NEXT j
      NEXT i
      IF ABS(pivote) >= epsilon GOTO actualizar
      GOTO fin
actualizar:
      irowk = irow(k)
      jcolk = jcol(k)
      determinante = determinante * pivote
      IF determinante < .0001 THEN GOTO fin
      FOR j = 1 TO max
           matrizfitfi(irowk,  j) =  matrizfitfi(irowk, j)  /
pivote
      NEXT j
      matrizfitfi(irowk, jcolk) = 1 / pivote
```

```
     FOR i = 1 TO n + m
          aijck = matrizfitfi(i, jcolk)
               IF i = irowk GOTO fininv
          matrizfitfi(i, jcolk) = -aijck / pivote
          FOR j = 1 TO max
               IF  j  <>  jcolk  THEN  matrizfitfi(i,  j)  =
matrizfitfi(i, j) - aijck * matrizfitfi(irowk, j)
          NEXT j
fininv:
     NEXT i
NEXT k
FOR j = 1 TO n + m
     FOR i = 1 TO n + m
          irowi = irow(i)
          jcoli = jcol(i)
          y(jcoli) = matrizfitfi(irowi, j)
     NEXT i
     FOR i = 1 TO n + m
          matrizfitfi(i, j) = y(i)
     NEXT i
NEXT j
FOR i = 1 TO n + m
     FOR j = 1 TO n + m
          irowj = irow(j)
          jcolj = jcol(j)
          y(irowj) = matrizfitfi(i, jcolj)
     NEXT j
     FOR j = 1 TO n + m
          matrizfitfi(i, j) = y(j)
     NEXT j
NEXT i
GOTO regresar
fin:
PRINT  : COLOR 15, 5: PRINT TAB(12); "La señal de entrada no
exita a la dinámica de la planta":  FOR i = 1 TO 5000:  NEXT
i: COLOR 7, 5:
GOTO regresarmenu
regresar:
RETURN
pantalla:
ON n GOSUB primer, segundo, tercer, cuarto
primer:
matriza(1, 1) = matriztheta(1, 1)
matrizb(1, 1) = 1
matrizc(1, 1) = matriztheta(2, 1)
GOTO seguir
segundo:
matriza(1, 1) = 0: matriza(1, 2) = 1
FOR j = 1 TO n
     matriza(2, j) = matriztheta(j, 1)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 1
```

```
FOR j = n + 1 TO n + m
     matrizc(1, j - n) = matriztheta(j, 1)
NEXT j
GOTO seguir
tercer:
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(2, 1) = 0:
matriza(2, 2) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1
FOR j = 1 TO n
     matriza(3, j) = matriztheta(j, 1)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 1
FOR j = n + 1 TO n + m
     matrizc(1, j - n) = matriztheta(j, 1)
NEXT j
GOTO seguir
cuarto:
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(1, 4) = 0
matriza(2, 1) = 0: matriza(2, 2) = 0: matriza(2, 4) = 0
matriza(3, 1) = 0: matriza(3, 2) = 0: matriza(3, 3) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1: matriza(3, 4) = 1
FOR j = 1 TO n
     matriza(4, j) = matriztheta(j, 1)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 0:
matrizb(4, 1) = 1
FOR j = n + 1 TO n + m
     matrizc(1, j - n) = matriztheta(j, 1)
NEXT j
GOTO seguir
seguir:
CLS : COLOR 3, 5, 0: FOR i = 1 TO 25: LOCATE i, 1: PRINT
STRING$(79, " "): NEXT i
COLOR 15, 5: LOCATE 2, 12: PRINT "IDENTIFICACION DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 4, 23: PRINT "Archivo de datos es: "; unidad$ + ":"
+ nombres$ + ".DMO"; CHR$(13)
COLOR 3, 5: LOCATE 6, 21: PRINT "**** Parámetros Calculados
***": COLOR 15, 5
LOCATE 7, 10: PRINT "Orden del modelo =    "; n: LOCATE 7,
40: PRINT "Valor de m          = "; m
LOCATE 8, 40: PRINT "Número de muestras = "; muestras
COLOR 3, 5: LOCATE 9, 10: PRINT " Matriz A"
LOCATE 10, 10: PRINT " *********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
     col = 0
        FOR j = 1 TO n
           col = col + 11
           LOCATE 11 + fila, 10 + col: PRINT USING
"+#.######"; matriza(i, j)
           IF j = n THEN fila = fila + 1
```

```
    NEXT j
NEXT i
COLOR 3, 5: LOCATE 16, 10: PRINT "Matriz B"
LOCATE 17, 10: PRINT "********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
    LOCATE 18 + fila, 18: PRINT USING "+#.######";
matrizb(i, 1)
    fila = fila + 1
NEXT i
COLOR 3, 5: LOCATE 16, 27: PRINT "Matriz C"
LOCATE 17, 27: PRINT "********": COLOR 15, 5
col = 0
FOR j = 1 TO m
    col = col + 11
    LOCATE 18, 25 + col: PRINT USING "+#.######";
matrizc(1, j)
NEXT j
TPROC = TFINAL - TINICIO: minutos = FIX(TPROC / 60):
segundos = FIX(TPROC - minutos * 60)
LOCATE 23, 17: PRINT ; TAB(17); "** Tiempo de procesamiento
: "; minutos; " min. "; segundos; " seg. **"
LOCATE 24, 24: PRINT "Desea un reporte impreso..."; : PRINT
"(S/N)"; : BEEP
sino1: tecla$ = INKEY$:
      IF tecla$ = "s" OR tecla$ = "S" THEN GOTO imprimir
      IF tecla$ = "" THEN GOTO sino1
      IF tecla$ = "n" OR tecla$ = "N" THEN GOTO
regresarmenu ELSE GOTO sino1
imprimir:
ON ERROR GOTO errorimp:
LPRINT , TAB(13); "E S C U E L A  P O L I T E C N I C A  N
A C I O N A L"; CHR$(13)
LPRINT , TAB(9); "F A C U  L T A D  D E  I N G E N I E R I
A  E L E C T R I C A"; CHR$(13)
LPRINT , TAB(30); "TESIS DE GRADO"; CHR$(13)
LPRINT , TAB(15); "IDENTIFICACION DE PARAMETROS EN VARIABLES
DE ESTADO"; CHR$(13)
LPRINT , TAB(26); "JOSE RAUL MEJIA RECALDE"; CHR$(13)
LPRINT , TAB(17); "ALGORITMO DE MINIMOS CUADRADOS
ORDINARIOS"; CHR$(13)
LPRINT , TAB(22); "Archivo de datos : "; unidad$ + ":" +
nombres$ + ".DMO"; CHR$(13)
LPRINT , TAB(21); "**** Parámetros Calculados ***"; CHR$(13)
LPRINT , TAB(24); "Orden del modelo   = "; n
LPRINT , TAB(24); "Valor de m         = "; m
LPRINT , TAB(24); "Número de muestras = "; muestras;
CHR$(13)
LPRINT , TAB(10); "Matriz  A"
LPRINT , TAB(10); "*********"
FOR i = 1 TO n
    IF n = 1 THEN
```

```
        LPRINT , TAB(18); USING "+#.######"; matriza(i, 1)
    ELSEIF n = 2 THEN
        LPRINT  , TAB(18);  USING "+#.######";  matriza(i,
1); TAB(30); matriza(i, 2)
    ELSEIF n = 3 THEN
        LPRINT  , TAB(18);  USING "+#.######";  matriza(i,
1); TAB(30); matriza(i, 2); TAB(42); matriza(i, 3)
    ELSEIF n = 4 THEN
        LPRINT  , TAB(18);  USING "+#.######";  matriza(i,
1); TAB(30); matriza(i, 2); TAB(42); matriza(i, 3); TAB(54);
matriza(i, 4)
    END IF
NEXT i
LPRINT , TAB(10); "Matriz B"
LPRINT , TAB(10); "********"
FOR i = 1 TO n
    LPRINT , TAB(18); USING "+#.######"; matrizb(i, 1)
NEXT i
LPRINT , TAB(10); "Matriz C"
LPRINT , TAB(10); "********"
IF m = 1 THEN
    LPRINT , TAB(18); USING "+#.######"; matrizc(1, 1)
ELSEIF m = 2 THEN
    LPRINT ,  TAB(18);  USING "+#.######";  matrizc(1,  1);
TAB(30); matrizc(1, 2)
ELSEIF m = 3 THEN
    LPRINT ,  TAB(18);  USING "+#.######";  matrizc(1,  1);
TAB(30); matrizc(1, 2); TAB(42); matrizc(1, 3)
ELSEIF m = 4 THEN
    LPRINT ,  TAB(18);  USING "+#.######";  matrizc(1,  1);
TAB(30);  matrizc(1,  2); TAB(42);  matrizc(1,  3); TAB(54);
matrizc(1, 4)
END IF
TPROC  =  TFINAL  -  TINICIO:  minutos =  FIX(TPROC  /  60):
segundos = FIX(TPROC - minutos * 60)
LPRINT : LPRINT ; TAB(17); "** Tiempo de procesamiento  : ";
minutos; " min. "; segundos; " seg. **"
LOCATE  24,  20:  PRINT  ;  "***  Pulse  alguna  tecla  para
continuar ***": COLOR 7, 5
DO: LOOP WHILE INKEY$ = ""
GOTO regresarmenu:
errordisco:
FOR i =  20 TO 25:  LOCATE i, 1:  PRINT STRING$(79, " ");  :
NEXT i
BEEP: LOCATE 22, 16: COLOR 15, 5: PRINT "Existe  algun error
durante el acceso al disco"
PRINT TAB(20); ".Pulse alguna tecla para continuar"
LOCATE 24, 24: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar3:  IF INKEY$  =  ""  THEN  GOTO pulsar3  ELSE  RESUME
archivoresultados
errorenlace:
```

```
FOR  i = 20  TO 25: LOCATE  i, 1: PRINT STRING$(79,  " "); :
NEXT i
BEEP: LOCATE  20, 10: COLOR 15, 5:  PRINT "Ha ocurrido algun
problema durante el enlace con el programa"
PRINT TAB(20); "Principal... presione cualquier tecla"
LOCATE 24, 24: PRINT "Código de error = "; ERR;
pulsar4: IF INKEY$ = "" THEN GOTO pulsar4 ELSE RESUME NEXT
CLS  : LOCATE 23, 24: PRINT  "***Espere por favor***": COLOR
7, 5
CHAIN "IPVE"
recdatos:
FOR  i = 20  TO 25: LOCATE  i, 1: PRINT  STRING$(79, " "); :
NEXT i
BEEP: LOCATE 20, 16: COLOR 15,  5: PRINT "Existe error en la
recuperación de los datos"
PRINT TAB(20); "Presione alguna tecla para continuar"
LOCATE 24, 24: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar5:  IF  INKEY$ =  ""  THEN  GOTO  pulsar5 ELSE  RESUME
recuperardatos
errorproceso:
FOR i =  20 TO 25: LOCATE  i, 1: PRINT  STRING$(79, " ");  :
NEXT i
BEEP:  LOCATE  20,  16: COLOR  15,  5:  PRINT  "Existe error
durante el procesamiento";
PRINT  TAB(11);  "Revise las  datos...presione  alguna tecla
para continuar";
LOCATE 24, 24: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar6: IF  INKEY$  =  "" THEN  GOTO  pulsar6  ELSE  RESUME
recuperardatos
errorimp:
FOR i = 20 TO 25
     LOCATE i, 1: PRINT STRING$(79, " ");
NEXT i
DEF SEG = &H40
statusport& = PEEK(9) * 256 + PEEK(8) + 1
IF INP(statusport&) <> 233 THEN
     LOCATE 22, 23: PRINT " La impresora no esta lista"
     LOCATE  25,   21:   PRINT  "Desea   continuar  con   la
impresión..."; : COLOR 15, 5: PRINT "s/n"; : COLOR 7, 5
pulsar2: LET  tecla$ =  INKEY$: IF  tecla$ = "  " THEN  GOTO
pulsar2
        IF tecla$ = "s" OR tecla$ = "S" THEN GOTO contimp
        IF tecla$ = "" THEN GOTO pulsar2
        IF  tecla$   =  "n"  OR  tecla$  =  "N"   THEN  GOTO
regresarmenu ELSE RESUME errorimp
contimp:
     FOR i = 24 TO 25
          LOCATE i, 1: PRINT STRING$(79, " ");
     NEXT i
     LOCATE  25,  10:  PRINT  "Aliste  la  impresora...pulse
alguna tecla para continuar";
     DO: LOOP WHILE INKEY$ = ""
```

```
    DO: LOOP UNTIL INP(statusport&) = 223
END IF
DEF SEG
RESUME NEXT
```

```
'**************************************************************
'
' RUTINA DE IDENTIFICACION DE PARAMETROS POR EL ALGORITMO DE
'
'                MINIMOS CUADRADOS RECURSIVOS
'
'                  JOSE RAUL MEJIA RECALDE
'ARCHIVO:MICUARE.BAS
'
'**************************************************************
COMMON unidad$, nombre$, nombres$, n, m, datos
DIM entradauk(-5 TO 50), salidayk(-5 TO 50), parametroai(4),
parametrobi(4)
DIM partea(4), parteb(4)
DIM matriza(5, 5), matrizb(5, 5), matrizc(5, 5)
DIM efi(10, 8), tefi(8, 10), covarianzapn(8, 8)
DIM   matriztheta(8,   50),   kalman(8,  50),   w(50,   50),
identidad(8, 8)
CLS : COLOR  3, 5, 0:  FOR i = 1  TO 25: LOCATE i,  1: PRINT
STRING$(78, " "): NEXT i
COLOR  15,  5:  LOCATE   3,  12:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE   7,  18:  PRINT   "ALGORITMO  DE  MINIMOS  CUADRADOS
RECURSIVOS"
IF  (unidad$   <>   "")  AND   (nombre$  <>  "")   THEN  GOTO
recuperardatos ELSE BEEP
LOCATE 20, 18: PRINT  "*** No se  ha definido el archivo  de
datos ***"
LOCATE 22, 24: PRINT "*** Espere por favor ***"
FOR i = 1 TO 2000: NEXT i: GOTO regresarmenu
recuperardatos:
ON ERROR GOTO recdatos
LOCATE 11, 17: PRINT  "El archivo de datos  es          :  ";
unidad$ + ":" + nombre$ + ".DIN"
LOCATE 24, 14: PRINT "*** Se está recuperando el  archivo de
datos ***"
OPEN unidad$ + ":" + nombre$ + ".din" FOR INPUT AS #1
INPUT #1, datos, n, m
DO UNTIL EOF(1)
     FOR i = 0 TO 50: INPUT #1, entradauk(i), salidayk(i)
       entradauk(i) = entradauk(i)
       salidayk(i) = salidayk(i)
     NEXT i
     FOR i = 1 TO n: INPUT #1, parametroai(i)
       parametroai(i) = parametroai(i)
     NEXT i
     FOR i = 1 TO m: INPUT #1, parametrobi(i)
       parametrobi(i) = parametrobi(i)
     NEXT i
INPUT #1, escalon, varianza, ruido
LOOP: CLOSE #1
```

```
nombres$ = nombre$
FOR i =  12 TO 25:  LOCATE i, 1:  PRINT STRING$(79, "  "); :
NEXT i
LOCATE 12, 17:  PRINT "El archivo  de salida  se llama :  ";
unidad$ + ":" + nombres$ + ".DMR"
LOCATE 14, 17: PRINT "Desea cambiar el nombre del archivo de
salida...(S/N)?": BEEP
sn1: LET tecla$ = INKEY$: IF tecla$ = "" THEN GOTO sn1
     IF tecla$ = "s" OR tecla$ = "S" THEN GOTO nuevoarchivo
     IF tecla$ = "n" OR tecla$ = "N" THEN  GOTO igualarchivo
ELSE GOTO sn1
nuevoarchivo:
LOCATE  16,  11: PRINT  "Nuevo  nombre del  archivo  (máx. 8
caracteres)"
repetir1:
LOCATE 18, 32: INPUT "Nombre = ", nombres$:
IF (LEN(nombres$)  < 9) AND  (LEN(nombres$) >  0) THEN  GOTO
verfdatos
LOCATE 18, 43: PRINT STRING$(30, " "): BEEP: GOTO repetir1
verfdatos:
ON ERROR GOTO recdatos
LOCATE 20, 17: PRINT "El archivo de salida se  llama:"; uni$
+ ":" + nombres$ + ".DMR"
igualarchivo:
LOCATE 24, 22: PRINT "*** Espere un momento por favor ***"
FOR i = 1 TO 2500: NEXT i
CLS :  COLOR 3, 5, 0:  FOR i = 1  TO 25: LOCATE i,  1: PRINT
STRING$(79, " "): NEXT i
COLOR  15,  5:  LOCATE   3,  12:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT ; "JOSE RAUL MEJIA RECALDE"; CHR$(13)
LOCATE  7,  19:  PRINT  ; "ALGORITMO  DE  MINIMOS  CUADRADOS
RECURSIVOS"
LOCATE 9, 20: PRINT "Seleccionar tipo de proceso"
LOCATE 11, 20: COLOR 15, 5: PRINT "1.-"; : COLOR 3, 5: PRINT
" Mínimos Cuadrados Recursivos Determinísticos"
LOCATE 12, 20: COLOR 15, 5: PRINT "2.-"; : COLOR 3, 5: PRINT
" Mínimos Cuadrados Recursivos Probabilísticos"
LOCATE 14, 20: COLOR 15, 5: PRINT "3.-"; : COLOR 3, 5: PRINT
" Salir"
LOCATE 16,  30: PRINT "Seleccionar una de las  opciones"; :
COLOR 15, 5: PRINT " (1 a 3)...": BEEP
seleccionar: LET  tecla$ = INKEY$: IF tecla$  = "" THEN GOTO
seleccionar:
IF tecla$ = "1" THEN GOTO deterministicos
     IF tecla$ = "" THEN GOTO seleccionar
     IF tecla$ = "2" THEN GOTO estocasticos
     IF  tecla$  =  "3"  THEN GOTO  regresarmenu  ELSE  GOTO
seleccionar
estocasticos:
FOR i = 0 TO 50
     RANDOMIZE TIMER
```

```
      saliday k(i) = saliday k(i) + RND * .1
NEXT i
deterministicos:
LOCATE 18, 25: PRINT "El orden del modelo es"; n
LOCATE 19, 20: PRINT "Desea cambiar el orden (S/N)?": BEEP
sn: LET tecla$ = INKEY$: IF tecla$ = "" THEN GOTO sn
      IF tecla$ = "s" OR tecla$ = "S" THEN GOTO nuevomodelo
      IF tecla$  = "n" OR tecla$ =  "N" THEN GOTO igualmodelo
ELSE GOTO sn
nuevomodelo:
LOCATE 20, 25: INPUT "Ingrese el valor de n="; n
m = n
igualmodelo:
LOCATE 20, 2:  PRINT "El número  de muestras debe  ser mayor
que el doble del orden del modelo"
LOCATE 21,  15: PRINT  "Y  menor que  un valor  tomado  como
50-2*n"
repetirmuestras:
LOCATE  22,  20:  INPUT  "Ingrese  número  de  muestras  ";
muestras: BEEP
IF muestras >= 50 - 2 * n THEN GOTO repetirmuestras
IF muestras < 2 * n THEN GOTO repetirmuestras
CLS : COLOR 3,  5, 0: FOR i  = 1 TO  25: LOCATE i, 1:  PRINT
STRING$(79, " "): NEXT i
LOCATE 3, 20: COLOR 15, 5: PRINT "Escoger algoritmo"
LOCATE 5, 25: COLOR 15, 5:  PRINT "1.-"; : COLOR 3, 5: PRINT
" Ponderación a=1: gamma = 1"
LOCATE 6, 25: COLOR 15, 5: PRINT "2.-"; : COLOR  3, 5: PRINT
" Ponderación Exponencial"
LOCATE 8,  30:  PRINT "Seleccione  una de  las opciones";  :
COLOR 15, 5: PRINT " (1 o 2).."
escoger: tecla$ = INKEY$: IF tecla$ = "" THEN GOTO escoger:
IF tecla$ = "1" THEN GOTO valores
IF tecla$ = "" THEN GOTO escoger
IF tecla$ = "2" THEN GOTO valoragamma ELSE GOTO escoger
valoragamma:
LOCATE   10,  20:   PRINT   "La   función   de   Ponderación
es:A*Gamma^(N-K)"
LOCATE 12, 32: PRINT " A = 1 - gamma"
repetirgamma:
LOCATE 14, 5:  PRINT : INPUT  "Introducir el valor  de gamma
(0<gamma<1) "; gamma: BEEP
IF gamma >= 1 THEN GOTO repetirgamma
IF gamma <= 0 THEN GOTO repetirgamma
LOCATE  24, 20:  PRINT  "*** Se está  generando la matriz de
ponderación W ***": FOR i = 1 TO 2000: NEXT i
a = 1 - gamma
FOR i = 1 TO muestras - n + 1
      FOR j = 1 TO muestras - n + 1
         IF i = j THEN w(i, j) = a * EXP(muestras - n + 1 - i)
* LOG(gamma) ELSE w(i, j) = 0
      NEXT j
```

```
NEXT i
GOTO algoritmo
valores:
LOCATE 10, 20: PRINT "La función de Ponderación es:A*Gamma^(N-K)"
LOCATE 25, 15: PRINT "*** Se está generando la matriz de ponderación W ***"
FOR i = 1 TO 2000: NEXT i
a = 1: gamma = 1
FOR i = 1 TO muestras - n + 1
     FOR j = 1 TO muestras - n + 1
       IF i = j THEN w(i, j) = a * gamma ELSE w(i, j) = 0
     NEXT j
NEXT i
algoritmo:
CLS : COLOR 3, 5, 0: FOR i = 1 TO 25: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
COLOR 15, 5: LOCATE 3, 12: PRINT "IDENTIFICACION DE PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 5, 25: PRINT "JOSE RAUL MEJIA RECALDE"
LOCATE 7, 18: PRINT "ALGORITMO DE MINIMOS CUADRADOS RECURSIVOS"
LOCATE 9, 16: PRINT "Condición inicial de Theta(N)= 0"
FOR i = 1 TO n + m
     matriztheta(i, 1) = 0
NEXT i
fila = 0
FOR i = 1 TO n + m
     LOCATE 11 + fila, 24: PRINT "Matriz Theta0("; i; ", 1)="; matriztheta(i, 1)
     fila = fila + 1
NEXT i
LOCATE 20, 16: PRINT "Condición inicial de P(N)"
LOCATE 21, 15: PRINT "Introducir el valor de Alfa"; : COLOR 3, 5: PRINT " (1.000 <= alfa < 1´000.000)"
repetiralfa: BEEP
COLOR 15, 5: LOCATE 23, 20: INPUT "El valor de alfa = "; alfa
IF alfa >= 1000000 THEN GOTO repetiralfa
IF alfa < 1000 THEN GOTO repetiralfa
inicialalfa = alfa
LOCATE 24, 10: PRINT "*** Se está generando la condición inicial para P(N) ***"
FOR i = 1 TO 3000: NEXT i
FOR i = 0 TO n + m
     alfa = alfa + salidayk(i) * salidayk(i)
NEXT i
alfa = alfa / (n + m + 1)
FOR i = 1 TO n + m
     FOR j = 1 TO n + m
       IF i = j THEN covarianzapn(i, j) = 1: GOTO continuar
       covarianzapn(i, j) = 0
```

```
continuar:       covarianzapn(i, j) = alfa * covarianzapn(i,
j)
     NEXT j
NEXT i
ON ERROR GOTO errorproceso
CLS : COLOR 3, 5, 0:
COLOR 15, 5: LOCATE 3, 5: PRINT "El número de iteraciones es
menor que 50-2*n"
cambiar:
FOR i = 5 TO 10: LOCATE i, 1: PRINT STRING$(79, " "): NEXT i
LOCATE 5, 5: INPUT "Introducir  el número de Iteraciones= ";
iteraciones: BEEP
     IF iteraciones <  1 OR iteraciones  > 50 -  2 * n  THEN
PRINT TAB(15);  "Cambie  el  número  de  Iteraciones":  GOTO
cambiar
CLS : COLOR  3, 5, 0:  FOR i = 1  TO 25: LOCATE i,  1: PRINT
STRING$(79, " "): NEXT i
LOCATE 23, 18: COLOR 15, 5: PRINT "*** Se está ejecutando el
Algoritmo de MCR ***": COLOR 7, 5
TINICIO = TIMER
DIM aux1(n + m, 1), aux2(1, 1)
DIM aux3(n + m, n + m), aux4(n + m, n + m)
DIM thetaa(1, iteraciones + 1), thetab(1, iteraciones + 1)
k0 = 0
FOR k = n + m TO n + m + iteraciones - 1
     k0 = k0 + 1
     cont = 0
     FOR i = n + m + 1 TO 0 STEP -1
       yk(i) = salidayk(k + 1 - cont)
       uk(i) = entradauk(k + 1 - cont)
       cont = cont + 1
     NEXT i
     FOR j = 1 TO n + m
       IF j >= n + 1 THEN efi(n + m - n + 2, j) = uk(n + n +
m + 1 - j): GOTO seguir
            efi(n + m - n + 2, j) = yk(n + m + 1 - j)
seguir:          tefi(j, n + m - n +  2) = efi(n + m - n + 2,
j)
     NEXT j
     FOR j = 1 TO n + m
       aux1(j, 1) = 0
       FOR l = 1 TO n + m
            aux1(j, l)  = aux1(j, 1) +  covarianzapn(j, l) *
tefi(l, n + m - n + 2)
       NEXT l
     NEXT j
     FOR i = 1 TO n + m
        aux1(i, 1) = aux1(i, 1) / gamma
     NEXT i
     aux2(1, 1) = 0
     FOR i = 1 TO n + m
```

```
        aux2(1, 1) =  aux2(1, 1) + efi(n  + m -  n + 2, i)  *
aux1(i, 1)
      NEXT i
         terminototal = 1 / a + aux2(1, 1)
      FOR i = 1 TO n + m
        kalman(i, 1) = aux1(i, 1) / terminototal
      NEXT i
      aux = 0
      FOR j = 1 TO n + m
        aux =  aux + efi(n +  m - n + 2,  j) * matriztheta(j,
k0)
      NEXT j
      aux1 = salidayk(k + 1) - aux
      FOR i = 1 TO n + m
        k2 = k0 + 1
        matriztheta(i, k2) =  matriztheta(i, k0) +  kalman(i,
1) * aux1
      NEXT i
      FOR i = 1 TO n + m
        FOR j = 1 TO n + m
              IF  i  <>  j  THEN  identidad(i,  j)  =  0  ELSE
identidad(i, j) = 1
        NEXT j
      NEXT i
      FOR i = 1 TO n + m
        FOR j = 1 TO n + m
              aux3(i, j) = (identidad(i,  j) - (kalman(i, 1) *
efi(n + m - n + 2, j))) / gamma
        NEXT j
      NEXT i
      FOR i = 1 TO n + m
        FOR j = 1 TO n + m
              aux4(i, j) = 0
              FOR l = 1 TO n + m
                aux4(i,  j)  =  aux4(i,  j)  +  aux3(i,  l)  *
covarianzapn(l, j)
              NEXT l
        NEXT j
      NEXT i
      FOR j = 1 TO n + m
        FOR i = 1 TO n + m
              covarianzapn(j, i) = aux4(j, i)
        NEXT i
      NEXT j
NEXT k
GOSUB archivoresultados
GOSUB pantalla
regresarmenu:
ON ERROR GOTO errorenlace
CLS  : COLOR 15,  5: LOCATE 24,  20: PRINT "***  Regresar al
menú principal ***"
CHAIN "ipve"
```

```
archivoresultados:
DIM ai(iteraciones + 1), bi(iteraciones + 1)
FOR i = 1 TO iteraciones + 1
      ai(i) = parametroai(1)
      bi(i) = parametrobi(1)
NEXT i
ON ERROR GOTO errordisco
LOCATE 23, 18: COLOR 15, 5: PRINT "  *** Se está almacenando
los resultados ***  ": COLOR 7, 5
CLOSE
OPEN unidad$ + ":" + nombres$ + ".dmr" FOR OUTPUT AS #2
      WRITE #2, iteraciones
      FOR i = 1 TO iteraciones + 1
        WRITE #2, ai(i), bi(i), thetaa(1, i), thetab(1, i)
      NEXT i
      FOR i = 1 TO n + m
        WRITE #2, matriztheta(i, k2)
      NEXT i
      WRITE #2, muestras, alfa, gamma
CLOSE #2
CLOSE : TFINAL = TIMER
RETURN
pantalla:
ON n GOSUB primer, segundo, tercer, cuarto
primer:
matriza(1, 1) = matriztheta(1, k2)
matrizb(1, 1) = 1:
matrizc(1, 1) = matriztheta(2, k2)
GOTO rseguir
segundo:
matriza(1, 1) = 0: matriza(1, 2) = 1
FOR j = 1 TO n
      matriza(2, j) = matriztheta(j, k2)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 1
FOR j = n + 1 TO n + m
      matrizc(1, j - n) = matriztheta(j, k2)
NEXT j
GOTO rseguir
tercer:
matriza(1, 1) =  0: matriza(1,  3) = 0:  matriza(2, 1) =  0:
matriza(2, 2) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1
FOR j = 1 TO n
      matriza(3, j) = matriztheta(j, k2)
NEXT j
matrizb(1, 1) = 0: matrizb(2, 1) = 0: matrizb(3, 1) = 1
FOR j = n + 1 TO n + m
      matrizc(1, j - n) = matriztheta(j, k2)
NEXT j
GOTO rseguir
cuarto:
```

```
matriza(1, 1) = 0: matriza(1, 3) = 0: matriza(1, 4) = 0
matriza(2, 1) = 0: matriza(2, 2) = 0: matriza(2, 4) = 0
matriza(3, 1) = 0: matriza(3, 2) = 0: matriza(3, 3) = 0
matriza(1, 2) = 1: matriza(2, 3) = 1: matriza(3, 4) = 1
FOR j = 1 TO n
     matriza(4, j) = matriztheta(j, k2)
NEXT j
matrizb(1, 1) =  0: matrizb(2,  1) = 0:  matrizb(3, 1) =  0:
matrizb(4, 1) = 1
FOR j = n + 1 TO n + m
     matrizc(1, j - n) = matriztheta(j, k2)
NEXT j
GOTO rseguir
rseguir:
CLS :  COLOR 3, 5, 0:  FOR i = 1  TO 25: LOCATE i,  1: PRINT
STRING$(79, " "): NEXT i
COLOR  15,  5:  LOCATE  2,  12:  PRINT  "IDENTIFICACION  DE
PARAMETROS EN VARIABLES DE ESTADO"
LOCATE 4, 23: PRINT "Archivo de datos es:  ";  unidad$ + ":"
+ nombres$ + ".DMR"
COLOR 3, 5: LOCATE 5, 21: PRINT  "**** Parámetros Calculados
***": COLOR 15, 5
LOCATE 6,  10: PRINT "Orden del modelo     = "; n: LOCATE 6,
40: PRINT "Valor de m          = "; m
LOCATE 7, 10: PRINT  "Número de iteraciones= "; iteraciones:
LOCATE 7, 40: PRINT "Valor de Gamma     = "; gamma
LOCATE 8, 40:  PRINT "Valor de alfa =";  inicialalfa: LOCATE
9, 40: PRINT "Número de muestras= "; muestras
COLOR 3, 5: LOCATE 9, 10: PRINT " Matriz  A"
LOCATE 10, 10: PRINT " *********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
     col = 0
       FOR j = 1 TO n
       col = col + 11
       LOCATE  11 + fila, 10 + col: PRINT USING "+#.######";
matriza(i, j)
       IF j = n THEN fila = fila + 1
     NEXT j
NEXT i
COLOR 3, 5: LOCATE 16, 10: PRINT "Matriz B"
LOCATE 17, 10: PRINT "********": COLOR 15, 5
fila = 0
FOR i = 1 TO n
     LOCATE   18  +  fila,   18:  PRINT  USING  "+#.######";
matrizb(i, 1)
     fila = fila + 1
NEXT i
COLOR 3, 5: LOCATE 16, 27: PRINT "Matriz C"
LOCATE 17, 27: PRINT "********": COLOR 15, 5
col = 0
FOR j = 1 TO m
```

```
     col = col + 11
     LOCATE 18,  26 +  col: PRINT USING  "+#.######";
matrizc(1, j)
NEXT j
TPROC  = TFINAL  - TINICIO: minutos  = FIX(TPROC  /  60):
segundos = FIX(TPROC - minutos * 60)
LOCATE 23, 17: PRINT "** Tiempo de procesamiento : ";
minutos; " min. "; segundos; " seg. **"
LOCATE 24, 24: PRINT "Desea un reporte impreso..."; : PRINT
"(S/N)"; : BEEP
sinol: tecla$ = INKEY$:
       IF tecla$ = "s" OR tecla$ = "S" THEN GOTO imprimir
       IF tecla$ = "" THEN GOTO sinol
       IF  tecla$  =  "n"  OR  tecla$  =  "N"  THEN  GOTO
regresarmenu ELSE GOTO sinol
imprimir:
ON ERROR GOTO errorimp:
LPRINT , TAB(13); "E S C U E L A  P O L I T E C N I C A  N
A C I O N A L"; CHR$(13)
LPRINT , TAB(9); "F A C U  L T A D  D E  I N G E N I E R I
A  E L E C T R I C A"; CHR$(13)
LPRINT , TAB(30); "TESIS DE GRADO"; CHR$(13)
LPRINT , TAB(15); "IDENTIFICACION DE PARAMETROS EN VARIABLES
DE ESTADO"; CHR$(13)
LPRINT , TAB(26); "JOSE RAUL MEJIA RECALDE"; CHR$(13)
LPRINT  ,  TAB(17);  "ALGORITMO  DE  MINIMOS  CUADRADOS
RECURSIVOS"; CHR$(13)
LPRINT  , TAB(22); "Archivo  de datos :  "; unidad$ +  ":" +
nombres$ + ".DMR"; CHR$(13)
LPRINT , TAB(21); "**** Parámetros Calculados ***"; CHR$(13)
LPRINT , TAB(24); "Orden del modelo        = "; n
LPRINT , TAB(24); "Valor de m              = "; m
LPRINT , TAB(24); "Número de muestras      = "; muestras
LPRINT , TAB(24); "Número de iteraciones = "; iteraciones
LPRINT , TAB(24); "Valor de Gamma          = "; gamma
LPRINT , TAB(24); "Valor  de alfa             = "; inicialalfa;
CHR$(13)
LPRINT , TAB(10); "Matriz  A"
LPRINT , TAB(10); "*********"
FOR i = 1 TO n
     IF n = 1 THEN
       LPRINT , TAB(18); USING "+#.######"; matriza(i, 1)
     ELSEIF n = 2 THEN
       LPRINT ,  TAB(18); USING "+#.######";  matriza(i, 1);
TAB(30); matriza(i, 2)
     ELSEIF n = 3 THEN
       LPRINT ,  TAB(18); USING "+#.######";  matriza(i, 1);
TAB(30); matriza(i, 2); TAB(42); matriza(i, 3)
     ELSEIF n = 4 THEN
       LPRINT ,  TAB(18); USING "+#.######";  matriza(i, 1);
TAB(30); matriza(i,  2);  TAB(42); matriza(i,  3);  TAB(54);
matriza(i, 4)
```

```
    END IF
NEXT i
LPRINT , TAB(10); "Matriz B"
LPRINT , TAB(10); "********"
FOR i = 1 TO n
    LPRINT , TAB(18); USING "+#.######"; matrizb(i, 1)
NEXT i
LPRINT , TAB(10); "Matriz C"
LPRINT , TAB(10); "********"
IF m = 1 THEN
    LPRINT , TAB(18); USING "+#.######"; matrizc(1, 1)
ELSEIF m = 2 THEN
    LPRINT , TAB(18); USING "+#.######"; matrizc(1, 1);
TAB(30); matrizc(1, 2)
ELSEIF m = 3 THEN
    LPRINT , TAB(18); USING "+#.######"; matrizc(1, 1);
TAB(30); matrizc(1, 2); TAB(42); matrizc(1, 3)
ELSEIF m = 4 THEN
    LPRINT , TAB(18); USING "+#.######"; matrizc(1, 1);
TAB(30); matrizc(1, 2); TAB(42); matrizc(1, 3); TAB(54);
matrizc(1, 4)
END IF
TPROC = TFINAL - TINICIO: minutos = FIX(TPROC / 60):
segundos = FIX(TPROC - minutos * 60)
LPRINT : LPRINT TAB(17); "** Tiempo de procesamiento : ";
minutos; " min. "; segundos; " seg. **"
LOCATE 24, 20: PRINT "*** Pulse alguna tecla para continuar
***": COLOR 7, 5
DO: LOOP WHILE INKEY$ = ""
GOTO regresarmenu:
errordisco:
FOR i = 20 TO 25: LOCATE i, 1: PRINT STRING$(79, " "); :
NEXT i
BEEP: LOCATE 22, 16: COLOR 15, 5: PRINT "Existe algun error
durante el acceso al disco"
PRINT TAB(25); ".Pulse alguna tecla para continuar"
LOCATE 24, 24: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar3: IF INKEY$ = "" THEN GOTO pulsar3 ELSE RESUME
archivoresultados
errorenlace:
FOR i = 20 TO 25: LOCATE i, 1: PRINT STRING$(79, " "); :
NEXT i
BEEP: LOCATE 22, 16: COLOR 15, 5: PRINT "Existe error
durante el enlace con el programa"
PRINT TAB(20); "Principal... presione cualquier tecla"
LOCATE 24, 24: PRINT "Código de error = "; ERR;
pulsar4: IF INKEY$ = "" THEN GOTO pulsar4 ELSE RESUME NEXT
CLS : LOCATE 23, 24: PRINT "***Espere por favor***"
CHAIN "IPVE"
recdatos:
FOR i = 20 TO 25: LOCATE i, 1: PRINT STRING$(79, " "); :
NEXT i
```

```
BEEP:  LOCATE 20, 16: COLOR 15, 5: PRINT "Existe error en la
recuperación de los datos"
PRINT TAB(20); "Presione alguna tecla para continuar"
LOCATE 24, 24: PRINT "Código de error = "; ERR; : COLOR 7, 5
pulsar5:  IF  INKEY$  = ""  THEN  GOTO  pulsar5  ELSE RESUME
recuperardatos
errorproceso:
FOR i =  20 TO 25: LOCATE  i, 1: PRINT  STRING$(79, " ");  :
NEXT i
BEEP: LOCATE  20,  16:  COLOR 15,  5:  PRINT  "Existe  error
durante el procesamiento";
PRINT TAB(15);  "Revise  las datos...presione  alguna  tecla
para continuar";
LOCATE 24, 24: PRINT "Código de error = "; ERR;
pulsar6:  IF  INKEY$ =  ""  THEN  GOTO pulsar6  ELSE  RESUME
recuperardatos
errorimp:
FOR i = 20 TO 25
     LOCATE i, 1: PRINT STRING$(79, " ");
NEXT i
DEF SEG = &H40
statusport& = PEEK(9) * 256 + PEEK(8) + 1
IF INP(statusport&) <> 233 THEN
     LOCATE 24, 23: PRINT " La impresora no esta lista"
     LOCATE 25, 21: COLOR 15, 5:  PRINT "Desea continuar con
la impresión...s/n"; : COLOR 7, 5
pulsar2: tecla$ = INKEY$
       IF tecla$ = "s" OR tecla$ = "S" THEN GOTO contimp
       IF tecla$ = "" THEN GOTO pulsar2
       IF   tecla$  =  "n"   OR  tecla$  =   "N"  THEN  GOTO
regresarmenu ELSE RESUME errorimp
contimp:
END IF
RESUME NEXT
```

LISTADOS DE LOS PROGRAMAS DE ARCHIVO DE LOTES

ARCHIVO GRAFICO.BAT

```
rem
echo off
cls
echo .                              Tesis de grado
echo .
echo  .       "Identificación de  Parámetros en  Variables de
Estado"
echo .
echo .
goto lazo2
:lazo1
echo .                      Este no es el disco correcto
:lazo2
echo .
echo .                     Inserte el Disco #2  (GRAFICOS)
echo .
echo .                  Presione cualquier tecla para continuar
echo .
echo .
echo .
echo .
pause
if not exist 123.EXE goto lazo1
if not exist grafico.wk1 goto lazo1
if not exist key-fake.com goto lazo1
if not exist grafico.bat goto lazo1

echo off
cls
:again
be menu.dat
be ask "" rgcm DEF=r bright yellow
if errorlevel 4 goto MONO
if errorlevel 3 goto COLOR
if errorlevel 2 goto GRAFICO
if errorlevel 1 goto REGRESAR

:MONO
echo off
cls
copy 123mono.set 123.set
goto again

:COLOR
echo off
```

```
cls
copy 123color.set 123.set
goto again

:GRAFICO
key-fake "/FRGRAFICO" 13 @30
123
cls
goto again


:REGRESAR
cls
goto seguir

:seguir
cls
echo .                          Tesis de grado
echo .
Echo  .      "Identificación de  Parámetros en  Variables de
Estado"
echo .
Echo .
Echo .
Echo .
Goto lazo3
:lazo4
echo .                     Este no es el disco correcto
:lazo3
Fecho .
Echo .                   Inserte el disco #1 (PROGRAMAS)
echo .
Echo .              Presione cualquier tecla para continuar
echo .
Echo .
Echo .
Echo .
Pause
if not exist ipve.exe goto lazo4
if not exist gendatos.exe goto lazo4
if not exist micuaor.exe goto lazo4
if not exist micuare.exe goto lazo4
if not exist grafico.bat goto lazo4
if not exist brun45.exe goto lazo4
exit
```

MENU.DAT

```
cls
be sa bright yellow on black
```

C

```
window 0,0,24,79 bright cyan on magenta explode
window 4,11,20,68 bright cyan on magenta explode shadow
rowcol 6,33 "TESIS DE GRADO" bright yellow
rowcol 8,25 "IDENTIFICACION DE PARAMETROS" bright white
rowcol 9,27 "EN VARIABLES DE ESTADO" bright white
rowcol 11,21 "Seleccionar tipo de monitor" bright white
rowcol 12,25 "M - MONITOR MONOCROMATICO"
rowcol 13,25 "C - MONITOR COLOR"
rowcol 15,21 "Seleccionar tipo de proceso" bright white
rowcol 16,25 "G - GRAFICAR"
rowcol 17,25 "R - REGRESAR AL MENU PRINCIPAL"
rowcol 12,25 "M" bright yellow
rowcol 13,25 "C" bright yellow
rowcol 16,25 "G" bright yellow
rowcol 17,25 "R" bright yellow
rowcol  19,25 "SELECCIONE  UNA DE  LAS OPCIONES  ..." bright
yellow
```

ID.BAT

```
echo off
cls
key-fake
be id.dat
ipve
cls
```

ID.DAT

```
cls
be sa bright yellow on black
window 2,2,23,78 bright cyan on magenta explode
window 8,10,19,69 bright cyan on magenta explode

rowcol 4,26 "ESCUELA POLITECNICA NACIONAL" bright white
rowcol 6,24 "FACULTAD DE INGENIERIA ELECTRICA" bright white
rowcol 10,32"TESIS DE GRADO" bright white
rowcol 12,24 "IDENTIFICACION DE PARAMETROS EN" bright white
rowcol 13,30 "VARIABLES DE ESTADO" bright white
rowcol  15,12 "Autor:      JOSE  RAUL MEJIA  RECALDE" bright
white
rowcol 17,12 "Director: ING. MARCO BARRAGAN" bright white
delay 2
window 18,48,23,79 black on white explode shadow
rowcol 20,51 "Pulse alguna tecla para" black on white
rowcol 21,58 "continuar" black on white
be ask ""
```

LISTADO DE LAS MACROS DE GRAFICOS EN LOTUS

MACRO INICIAL DE ARRANQUE
```
{HOME}/WGRM
{GOTO}I21~{WINDOWSOFF}{ONERROR \ERRORES,L38}
/FDB:~/GRGTXOFGLQ
TFIDENTIFICACION DE PARAMETROS~
TXMUESTRAS y/o ITERACIONES~TYAMPLITUD~
TSGRAFICOS~LANADA~LBNADA~QQ
{MENUBRANCH \MENU}
```

TESIS DE GRADO

"IDENTIFICACION DE PARAMETROS POR VARIABLES DE ESTADO"

MACROS DE GRAFIZACION DE RESULTADOS

AUTOR : JOSE RAUL MEJIA RECALDE

DIRECTOR : Ing. MARCO BARRAGAN

RUTINA DE INTERCEPCION DE ERRORES
```
{RESTART}{BEEP}
{GOTO}K38~ERROR :~
{GOTO}K39~PRESIONE CUALQUIER TECLA PARA CONTINUAR~
{GOTO}I21~{GET I50}/REK38..L39~
{WINDOWSOFF}{BRANCH \A}
```

MENU PRINCIPAL (MENU)
```
UNIDAD
Unidad del disco de datos
/FD{?}~
{BRANCH I12}
```

SUBMENU DE GRAFICOS DE LOS DATOS DE ENTRADA (MENU1E}
```
ENTRADA
Gráfico de la señal de entrada generada
{MENUCALL \MENUSEG}
```

```
/GRXABQX{SALTO4}A{SALTO1}
OTS{ESC}SEÑAL DE ENTRADA~LA{ESC}
ENTRADA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2E}


SUBMENU DE ALMACENAMIENTO DE GRAFICOS (MENU2E)
CONTINUAR
Continuar el programa
{MENUBRANCH \MENU1E}


SUBMENU DE GRAFICOS DE LOS DATOS DE  LA SEÑAL DE SALIDA PARA
MCO (MENU1O}
SALIDA GENERADA
Gráfico de la señal de salida generada
{MENUCALL \MENUSEG}
/GRXABQX{SALTO4}A{SALTO1}
OTS{ESC}SALIDA GENERADA~LA{ESC}
SALIDA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2O}


SUBMENU DE ALMACENAMIENTO DE GRAFICOS (MENU2O)
CONTINUAR
Continuar el programa
{MENUBRANCH \MENU1O}


SUBMENU  DE GRAFICOS  DE LOS  PARAMETROS A(n,1),  C(1,1) MCR
(MENU1S)
PARAMETROS GENERADOS
Gráficos de los parámetros a(n,1), c(1,1) generados
{MENUCALL \MENUSEG}
/GRXABQX{SALTO5}A{SALTO1}B{SALTO2}
OTS{ESC}PARAMETROS GENERADOS~
LA{ESC}PARAMETRO a(n,1)~LB{ESC}PARAMETRO c(1,1)~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2S}


SUBMENU DE ALMACENAMIENTO DE GRAFICOS (MENU2S)
CONTINUAR
Continuar con el programa
{MENUBRANCH \MENU1S}
```

SUBMENU PARA REALIZAR EL ANALISIS DE UN SEGMENTO (MENUSEG)

```
TOTAL
Gráfico completo
{GOTO}AA3~'2~
{GOTO}AA5~@STRING($A$1+1,0)~
/CAA3~AB3..AH3~
/CAA5~AB5..AH5~
{RETURN}


SEÑALES GENERADAS
Gráficos de las señales de entrada y/o salida generadas
{HOME}
/RE.{END}{DOWN}{RIGHT}{RIGHT}
{RIGHT}{RIGHT}{RIGHT}{RIGHT}~
/FIN*.DIN~{?}~/CA1~AA1~
{GOTO}C2~
{GOTO}D2~/DFD2~~~~
/DF.{LEFT}{LEFT}{END}
{DOWN}{RIGHT}{RIGHT}~~~~
{GOTO}D2~
/GRXABCDEFQQ
{GOTO}I21~
{MENUBRANCH \MENU1E}


SALIDA
Gráfico de la señal de salida generada
{MENUCALL \MENUSEG}
/GRXABQX{SALTO4}A{SALTO2}
OTS{ESC}SEÑAL DE SALIDA~
LA{ESC}SALIDA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2E}


ALMACENAR
Almacenar el gráfico
/GS{?}~RQ{ESC}{ESC}
{MENUBRANCH \MENU1E}


SALIDA CALCULADA
Gráfico de la señal de salida calculada
{MENUCALL \MENUSEG}
/GRXABQX{SALTO4}A{SALTO2}
OTS{ESC}SALIDA CALCULADA~
LA{ESC}SALIDA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
```

```
{MENUBRANCH \MENU2O}


ALMACENAR
Almacenar el gráfico
/GS{?}~RQ{ESC}{ESC}
{MENUBRANCH \MENU1O}


PARAMETROS CALCULADOS
Gráficos de los parámetros a(n,1), c(1,1) calculados
{MENUCALL \MENUSEG}
/GRXABQX{SALTO5}A{SALTO3}
B{SALTO4}OTS{ESC}
PARAMETROS CALCULADOS~LA
{ESC}PARAMETRO a(n,1)~LB{ESC}PARAMETRO c(1,1)~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2S}


ALMACENAR
Almacenar el gráfico
/GS{?}~RQ{ESC}{ESC}
{MENUBRANCH \MENU1S}


SEGMENTO
Gráfico de un segmento
{GETNUMBER "LIMITE INFERIOR = ",AB1}
{IF AB1<1#OR#AB1>=AA1}{BEEP 1}{BRANCH R90}
{GETNUMBER "LIMITE SUPERIOR = ",AC1}
{IF AC1<=AB1#OR#AC1>AA1}{BEEP 1}{BRANCH R92}
{GOTO}AA3~@STRING($AB$1+1,0)~
/C~AB3..AH3~
{GOTO}AA5~@STRING($AC$1+1,0)~


GRAFICOS ALGORITMO MCO
Gráficos de la señal de salida generada y/o calculada
{HOME}
/RE.{END}{DOWN}{RIGHT}{RIGHT}
{RIGHT}{RIGHT}{RIGHT}{RIGHT}~
/FIN*.DMO~{?}~/CA1~AA1~
{GOTO}C2~
{GOTO}D2~/DFD2~~~~
/DF.{LEFT}{LEFT}{END}
{DOWN}{RIGHT}{RIGHT}~~~~
{GOTO}D2~
/GRXABCDEFQQ
{GOTO}I21~
{MENUBRANCH \MENU1O}
```

```
TOTAL
Gráficos de las señales de entrada y salida generadas
{MENUCALL \MENUSEG}
/GRXABQX{SALTO4}A{SALTO1}
B{SALTO2}
OTS{ESC}ENTRADA Y SALIDA~
LA{ESC}ENTRADA~LB{ESC}SALIDA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2E}

TOTAL
Gráficos de las señales de salida generada y calculada
{MENUCALL \MENUSEG}
/GRXABQX{SALTO4}A{SALTO1}
B{SALTO2}
OTS{ESC}SEÑALES DE SALIDA~
LA{ESC}GENERADA~LB{ESC}CALCULADA~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2O}

TOTAL
Gráficos de los parámetros a(n,1), c(1,1) generados y
calculados
{MENUCALL \MENUSEG}
/GRXABQX{SALTO5}A{SALTO1}B{SALTO2}
C{SALTO3}D{SALTO4}
OTS{ESC}TOTAL PARAMETROS~
LA{ESC}PARAMETROS a(n,1)~LB{ESC}PARAMETROS c(1,1)~QVQ
{GOTO}I21~{WINDOWSON}
{WINDOWSOFF}
{MENUBRANCH \MENU2S}

GRAFICOS ALGORITMO MCR
Gráficos de los parámetros Theta
{HOME}
/RE.{END}{DOWN}{RIGHT}{RIGHT}
{RIGHT}{RIGHT}{RIGHT}{RIGHT}~
/FIN*.DMR~{?}~/CA1~AA1~
{GOTO}D2~
{GOTO}E2~/DFE2~~~~
/DF.{LEFT}{LEFT}{END}
{DOWN}{RIGHT}{RIGHT}~~~~
{GOTO}E2~
/GRXABCDEFQQ
{GOTO}I21~
```

```
{MENUBRANCH \MENU1S}


MENU
Regresar al menu anterior
{MENUBRANCH \MENU}


MENU
Regresar al menu anterior
{MENUBRANCH \MENU}


SALIR
Volver al programa principal
{HOME}
{GOTO}I21~/QY~




MENU
Regresar al menu anterior
{MENUBRANCH \MENU}
```

BIBLIOGRAFIA

[1] GENE F. FRANKLIN-J. DAVID POWELL, Digital Control of Dynamic Systems, Addison-Wesley Publishing Company, 1980.

[2] M.H. DAVIS-R.B. VINTER, Stochastic Modelling and Control, Chapman and Hall, 1985.

[3] B.P. LATHI, An Introduction to Random Signals and Comunication Theory, International Text Book Company, 1968.

[4] OGATA KATSUHITO, Ingeniería de Control Moderno, Mac-Graw-Hill, 1974.

[5] KUO BENJAMIN, Digital Control Systems, Halt, Rinehart and Winston, 1980.

[6] D.W. CLARKE, Generalized-Least-Squares Estimation of the Parameters of a Dynamic Model, New College, Oxford.

[7] MEYER PAUL L., Probabilidad y Aplicaciones Estadísticas, Fondo Educativo Interamericano S.A., 1973.