

01170

1ej. 1



UNIVERSIDAD NACIONAL
AUTÓNOMA

DIVISION DE ESTUDIOS DE POSGRADO
FACULTAD DE INGENIERIA

PAQUETE DE PROGRAMAS PARA COMPUTADORA
PARA IDENTIFICACION DE SISTEMAS DINAMICOS

Tesis que presenta:

MIGUEL GILBERTO GUEVARA RODRIGUEZ

para obtener el grado de Maestro en Ingeniería. (CONTROL)

Créditos asignados a la Tesis 11 (ONCE)

JURADO

- PRESIDENTE: DR. ROMEO ORTEGA MARTINEZ
- VOCAL: DR. GUSTAVO MEDRANO CERDA
- SECRETARIO: DR. STANISLAW RACZYNSKI
- SUPLENTE: DR. GUILLERMO REBOLLEDO C.
- SUPLENTE: DR. LUIS A. BUZO DE LA PENA

[Handwritten signatures and initials over the jury list]

TESIS CON
EVALUACION



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

TESIS CON FALLA DE ORIGEN

INDICE

Prefacio	1
Introducción	2
1 Teoría sobre Mínimos Cuadrados.	3
1.1 Principios Básicos.	4
1.2 Mínimos Cuadrados Recursivos.	8
1.3 Modelo Paramétrico por medio de Mínimos Cuadrados.	11
1.4 Propiedades Estadísticas del Estimador de Mínimos Cuadrados.	16
1.5 Mínimos Cuadrados Generalizados	18
1.6 Algoritmo de Mínimos Cuadrados Generalizados.	21
1.7 Mínimos Cuadrados Generalizados Recursivos.	24
1.8 Aplicación de la Técnica de Mínimos Cuadrados a Sistemas no Lineales.	26
1.9 Estimación en Lazo Cerrado.	32
2 Descripción de la Estructura del Programa.	45
2.1 Programa Principal.	45
2.2 Simulación.	47
2.2.1 Generación de Datos para el Sistema.	48
2.2.2 Generación de una Secuencia Pseudoaleatoria.	51

2.2.3 Generación de Ruido.	53
2.3 Arreglo de Datos.	56
2.3.1 Método Paramétrico.	56
2.3.2 Generación de Residuos.	58
2.3.3 Arreglo de Residuos.	59
2.4 Análisis de Datos.	60
2.4.1 Correlación de Archivos.	60
2.4.2 Media de Archivos.	62
2.4.3 Graficación.	63
3 Principales Algoritmos Utilizados en la Elaboración del Programa.	64
3.1 Resolución del Problema de Mínimos Cuadrados Mediante el Algoritmo de Householder.	64
3.2 Generación de una Señal Pseudoaleatoria.	72
3.3 Generador de una Secuencia Aleatoria con Distribución Uniforme.	74
3.4 Simulación de un Sistema Lineal Discreto.	75
3.5 Generación de una Serie de Tiempo con Distribución Normal.	76
3.6 Ejemplos Prácticos	78
Apéndice 1 Programa para Computadora	97
Bibliografía	123

PREFACIO

Una gran variedad de técnicas han sido desarrolladas para identificar sistemas. Estas técnicas se basan en las teorías de optimización y estimación. El propósito de este trabajo es el de hacer un paquete de identificación centrado en el método de mínimos cuadrados como solución al problema de identificación de sistemas. Para esto se tomó en cuenta que los mínimos cuadrados es un método clásico, utilizado en varios campos como: economía, medicina, astronomía etc., y el hecho de que otras técnicas de identificación como correlación cruzada, máxima verosimilitud, el filtro de Kalman y la variable instrumental, en algunos casos pueden ser relacionados con el algoritmo de mínimos cuadrados. Dicho paquete debe ser lo suficientemente flexible como para poder estimar un sistema por medio de varios métodos, simular un sistema lineal discreto con distintos tipos de ruido, y tener una estructura que permita introducir fácilmente otros algoritmos de identificación.

Este trabajo esta orientado a servir de base a los alumnos que lleven un curso de identificación de sistemas o a ingenieros que deseen aprender identificación de sistemas dinámicos. Como la identificación es la unión entre el modelo matemático y el modelo real, el aprendizaje mediante la práctica es esencial en esta materia y debido a esto el objetivo de este trabajo es el de ser un puente entre la teoría y la práctica en la identificación de sistemas dinámicos.

INTRODUCCION

El problema de la identificación de sistemas consiste en la determinación de un modelo matemático para un sistema determinado, mediante la observación de las entradas y salidas. Históricamente la identificación de sistemas ha sido motivada por la necesidad de diseñar mejores estrategias de control; como la necesidad de los ingenieros industriales de tener mejores conocimientos de su plantas, el estudio del comportamiento de vehículos aéreos y espaciales, la simulación de funciones biológicas como son el control de una mano o un brazo, la respuesta de la pupila del ojo. Sin embargo hay situaciones donde el principal interés es el de analizar las propiedades de un sistema; por ejemplo la necesidad de determinar experimentalmente parámetros como: el coeficiente de transferencia de calor, la tasa de cambio en una reacción química, etc.

Desde el punto de vista de la teoría de sistemas podemos determinar exactamente los parámetros desconocidos de un sistema si las mediciones de entrada-salida y el modelo son exactas. Sin embargo en la realidad los datos de entrada-salida están contaminados con ruido debido a las mediciones y también existen imprecisiones en la ecuación del modelo; entre las que podemos citar los disturbios aleatorios en el sistema en sí. Por lo que la determinación de los parámetros del sistema es esencialmente un problema de estimación estadística.

Esta tesis esta dividida en tres capítulos, en el primer capítulo se analiza el problema de los mínimos cuadrados, desde principios básicos hasta su estimación en lazo cerrado. En el segundo se analiza y describe la estructura del programa; en el tercero se describen los principales algoritmos utilizados en la elaboración del programa y unos ejemplos que muestran la potencialidad del paquete.

1 TEORIA SOBRE MINIMOS CUADRADOS

La teoría de mínimos cuadrados fue propuesta por Karl Gauss en su trabajo sobre la predicción de la órbita de los planetas. Esta teoría es muy popular debido a que es más fácil de comprender que otras y no requiere un conocimiento de estadística profundo, mas aún los mínimos cuadrados nos dan soluciones en casos donde otros métodos fallan. Estimaciones obtenidas mediante los mínimos cuadrados tienen propiedades estadísticas óptimas: son consistentes, insesgados y eficientes. También algunos algoritmos de estimación, que son usados en la identificación de sistemas, pueden ser interpretados desde el punto de vista de mínimos cuadrados.

A continuación se muestra un párrafo del trabajo sobre la predicción de la órbita de los planetas de Gauss (ver [14] y [39]) y posteriormente se resaltan las ideas relevantes de dicho párrafo. "Si las observaciones astronómicas y demás cantidades en las que se basa el cálculo de la órbita fueran totalmente correctas, los parámetros deducidos de tres o cuatro observaciones serían inmejorables (suponiendo que el movimiento se lleva a cabo de acuerdo con las leyes de Kepler) y si se hicieran otras observaciones los parámetros se confirmarían, y no se corregirían. Pero como todas nuestras mediciones y observaciones no son más que aproximaciones a la realidad, lo mismo es cierto para todos los cálculos obtenidos a partir de ellos, cuyo objetivo será el de aproximarse, de la mejor manera, a la verdad. Pero esto se debe de llevar a cabo mediante una combinación apropiada de un número mayor de observaciones que el número absolutamente necesario para determinar los parámetros desconocidos. Este problema solamente puede ser resuelto satisfactoriamente si se tiene un conocimiento aproximado de la órbita, el cual es corregido de tal forma que se satisfacen, de la manera más precisa, todas las observaciones". Del párrafo anterior los conceptos que debemos de tener en cuenta son:

1. Gauss habla del número de observaciones que son absolutamente necesarias para determinar las cantidades desconocidas.
2. Gauss hace notar la necesidad de tener un número mayor de observaciones que dicho mínimo debido a que existen errores en la medición y en las observaciones.
3. Gauss dice que las ecuaciones del movimiento deben ser descripciones exactas de dicho movimiento, de donde deducimos que él previó el problema de la dinámica y su importancia.
4. Gauss requiere que se tenga un conocimiento a priori de la órbita, esto implica el uso de un procedimiento de linealización.

5. Gauss menciona que los estimados de los parámetros deben satisfacer las observaciones en la manera más precisa que sea posible. De donde se puede deducir el concepto de residuos (la diferencia entre los valores observados y los predichos de los estimados) que deben ser lo menor que sea posible.
6. Gauss habla de la imprecisión de las observaciones e indica que dichos errores no pueden ser conocidos, por lo que sienta las bases para consideraciones probabilísticas.
7. Por último Gauss habla de "combinaciones apropiadas" de las observaciones que darán los estimados más precisos. Esto está relacionado a la definición de la estructura de un procedimiento de estimación (filtrado lineal o no lineal).

1.1 PRINCIPIOS BASICOS

La técnica de mínimos cuadrados es un procedimiento matemático mediante el cual podemos ajustar datos experimentales a un esquema dado. Por ejemplo, supongamos que existe una variable relacionada linealmente a un conjunto de n variables $x=(x_1, \dots, x_n)$, esto es:

$$y = x_1 \theta_1 + x_2 \theta_2 + \dots + x_n \theta_n \tag{1.1}$$

Donde los θ_i son parámetros desconocidos y deseamos estimar sus valores por medio de la observación de la entrada y la salida en tiempos diferentes.

Supongamos que una secuencia de m observaciones en X y Y han sido hechas en los tiempos t_1, t_2, \dots, t_m y llamaremos a los datos medidos como $y(i)$ y $x(i), \dots, x(i)$ para $i= 1, 2, \dots, m$;

Ahora agrupemos todas las observaciones de la siguiente manera:

$$y(i) = \theta_1 x_1(i) + \theta_2 x_2(i) + \dots + \theta_n x_n(i) \quad i=1,2,\dots,m \quad (1.2)$$

El arreglo anterior puede ser puesto en forma matricial de la siguiente manera:

$$Y = X \theta \quad (1.3)$$

donde

$$Y = \begin{bmatrix} y(1) \\ \vdots \\ y(m) \end{bmatrix} \quad X = \begin{bmatrix} x_1(1) & \dots & x_n(1) \\ \vdots & & \vdots \\ x_1(m) & \dots & x_n(m) \end{bmatrix} \quad \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$$

En donde para poder estimar los n parámetros del vector θ es necesario que m sea mayor que n . Si $n=m$ entonces podemos resolver θ en forma única por medio de la ecuación (1.3); obteniendo $\hat{\theta} = X^{-1} Y$, donde $\hat{\theta}$ es un estimador de θ . Sin embargo si $m > n$ en general no es posible determinar un estimador $\hat{\theta}$ que satisfaga todas las m ecuaciones debido a que los datos pueden estar contaminados de ruido, error en el modelo, etc. No obstante podemos determinar $\hat{\theta}$ en base al criterio del mínimo error cuadrático. Definamos un vector error de la siguiente manera:

$$e = (e(1), e(2), \dots, e(m))' \\ e = Y - X \hat{\theta} \quad (1.4)$$

Ahora escogeremos θ de tal forma que J se minimiza, en donde:

$$J = e' e$$

$$J = (Y - X\theta)'(Y - X\theta)$$

$$J = Y'Y - \theta'X'Y - Y'X\theta + \theta'X'X\theta \quad (1.5)$$

Diferenciando J con respecto a θ e igualando el resultado a cero obtenemos:

$$\frac{\partial J}{\partial \theta} = -2X'Y + 2X'X\theta = 0$$

de donde

$$X'X\theta = X'Y$$

$$\theta = (X'X)^{-1} X'Y \quad (1.6)$$

A este resultado se le denomina estimador de mínimos cuadrados de θ ; y a la ecuación (1.6) se le llama normal; a "e" se le conoce como residuo en estadística.

Para la obtención del resultado anterior se consideró el mismo peso para todos los $e(i)$, en otras palabras que todas las mediciones tuvieron la misma precisión. Sin embargo esta formulación se puede generalizar para permitir que cada término de error tenga un peso diferente. Si W es la matriz de peso deseada; entonces el criterio de error queda:

$$J = e' W e$$

$$= (Y - X\theta)' W (Y - X\theta) \quad (1.7)$$

En donde W es una matriz simétrica positiva definida. La minimización de Jw con respecto a θ nos proporciona el estimador pesado de mínimos cuadrados.

$$\theta = (X'WX)^{-1} X'WY$$

(1.8)

1.2 MINIMOS CUADRADOS RECURSIVOS

En esta sección deduciremos un algoritmo recursivo para la solución de mínimos cuadrados. La necesidad de contar con una solución recursiva se tiene cuando los datos experimentales se proporcionan continuamente y deseamos ajustar nuestros parámetros haciendo uso de esta última información.

Con una fórmula recursiva el estimado puede ser actualizado por intervalos de tiempo sin tener la necesidad de realizar la solución matricial de la ecuación (1.6). Ahora bien supongamos que tenemos m conjuntos de datos de tal forma que:

$$Y = X \theta$$

$$\theta[m] = (X_m' X_m)^{-1} X_m' Y_m \quad (1.9)$$

y una nueva ecuación es introducida de tal suerte que:

$$y(m+1) = \theta_1 x_1(m+1) + \dots + \theta_n x_n(m+1) \quad (1.10)$$

Definiendo

$$x'(m+1) = (x_1(m+1), x_2(m+1), \dots, x_n(m+1)) \quad (1.11)$$

Entonces podemos expresar

$$y(m+1) = x'(m+1) \theta \quad (1.12)$$

y el sistema de $(m+1)$ ecuaciones como:

$$Y[m+1] = X[m+1]\theta \quad (1.13)$$

en donde

$$Y[m+1] = \begin{bmatrix} y(1) \\ \vdots \\ y(m) \\ y(m+1) \end{bmatrix} = \begin{bmatrix} x(1) \\ \vdots \\ x(m) \\ x(m+1) \end{bmatrix} \theta$$

$$X[m+1] = \begin{bmatrix} x(1) & \dots & x(1) \\ 1 & & r_1 \\ \vdots & & \vdots \\ x(m) & \dots & x(m) \\ 1 & & r_m \\ x(m+1) & \dots & x(m+1) \\ 1 & & r_{m+1} \end{bmatrix} = \begin{bmatrix} X[m] \\ \dots \\ X'(m+1) \end{bmatrix}$$

El nuevo estimador es:

$$\theta(m+1) = (X'(m+1) X[m+1])^{-1} X'(m+1) Y[m+1] \quad (1.14)$$

Del resultado anterior pudieramos pensar que para obtener el estimado $\theta(m+1)$ debemos de invertir una matriz de $n \times n$. La pregunta que surge es de si podemos calcular $\theta(m+1)$ a partir de la actualización de $\theta(m)$ sin efectuar la inversion de la matriz. La respuesta es que si y los resultados se muestran a continuación:

$$\theta(m+1) = \theta(m) + A_1(m+1)P(m)x(m+1) * (y(m+1) - x'(m+1)\theta(m)) \quad (1.15)$$

donde

--1

$$P(m) = \{x(m') \times m\}$$

$$P(m+1) = P(m) - A1(m+1)P(m) \times (m+1) \times x'(m+1)P(m)$$

$$A1(m+1) = 1 / \{1 + x'(m+1)P(m) \times (m+1)\}$$

La ecuación (1.15) intuitivamente se puede interpretar de la siguiente manera: El estimado $\theta(m+1)$ es formado mediante la suma de un factor de corrección al estimado anterior, dicho factor es proporcional a $y(m+1) - x'(m+1)\theta(m)$; hay que hacer notar que el término $x'(m+1)\theta(m)$ sería el valor de $y(m+1)$ si el modelo fuera perfecto y no hubiera disturbios. Por lo que el factor de corrección es proporcional a la diferencia entre el valor medido $y(m+1)$ y el valor predicho de $y(m+1)$.

El lector interesado en la demostración puede consultar la ref(19), pp

20-26.

1.3 MODELO PARAMETRICO POR MEDIO DE MINIMOS CUADRADOS

Los modelos paramétricos de sistemas usualmente son de gran utilidad debido a que la teoría moderna de control y diseño de sistemas requieren una descripción en variables de estado de la dinámica del sistema. Aunque casi todos los sistemas son continuos, es más práctico y conveniente el aproximar estos sistemas por medio de modelos discretos; ya que las señales discretas pueden ser procesadas por medio de una computadora digital. Otra razón por la que se prefieren sistemas discretos es la de que las ecuaciones en diferencias son de naturaleza algebraica y por lo tanto son más fáciles de manipular e identificar que las ecuaciones diferenciales. Por ejemplo, si un sistema está descrito por una ecuación diferencial de orden n con los coeficientes constantes, entonces:

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_0 u(k) + \dots + b_n u(k-n) + u(k) \quad (1.16)$$

Si definimos $q^{-1} y(k) = y(k-1)$, entonces la ecuación anterior se puede escribir

$$A(q^{-1})y(k) = B(q^{-1})u(k) \quad (1.17)$$

en donde

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_n q^{-n}$$

La ecuación (1.16) la podemos escribir

$$y(k) = -\sum_{i=1}^n a_i y(k-i) + \sum_{i=1}^n b_i u(k-i) + e(k) \quad i=1..n \quad (1.18)$$

Si definimos el vector $x'(k) = [-y(k-1) \dots -y(k-n), u(k) \dots u(k-n)]$ de orden $2n+1$, y el vector $\theta = [a_1 \dots a_n, b_1 \dots b_n]'$; entonces podemos reescribir

$$y(k) = x'(k)\theta + e(k) \quad (1.19)$$

Debido a que tenemos una cadena de datos $\{y(k), u(k)\}$, para $k=1 \dots (N+n)$, podemos obtener un sistema de N ecuaciones ($N > 2n$) como

$$Y = X\theta + e \quad (1.20)$$

en donde

$$Y = [y(n+1) \dots y(n+N)]$$

$$e = [e(n+1) \dots e(n+N)]$$

$$X = \begin{bmatrix} x(n+1) \\ x(n+2) \\ \vdots \\ x(n+N) \end{bmatrix} = \begin{bmatrix} -y(n) \dots -y(1), u(n+1) \dots u(n) \\ -y(n+1) \dots -y(2), u(n+2) \dots u(n+1) \\ \vdots \\ -y(n+N-1) \dots -y(N), u(n+N) \dots u(n) \end{bmatrix}$$

Al sistema de ecuaciones (1.20) le podemos aplicar el algoritmo de mínimos cuadrados descrito anteriormente. Dependiendo de las características de "e", el estimador de mínimos cuadrados tendrá distintas propiedades. A manera de ilustración veremos que la ecuación 1.20 puede representar a diferentes estructuras Ej #1 Consideremos el

siguiente modelo con ruido

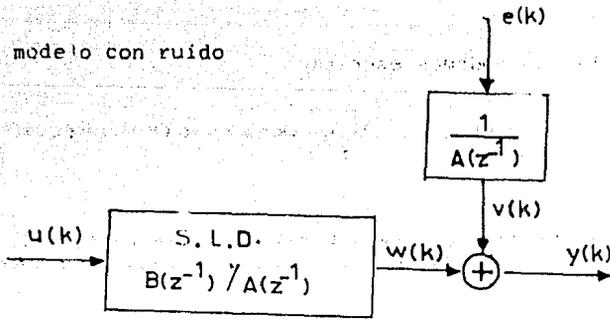


fig 1.1

En donde $v(k)$ es un proceso aleatorio estacionario con media cero

El sistema discreto está descrito por la ecuación en diferencias

$$A(q^{-1})w(k) = B(q^{-1})u(k) \quad (1.21)$$

$$y(k) = w(k) + v(k) \quad (1.22)$$

$$A(q^{-1})y(k) = B(q^{-1})u(k) + A(q^{-1})v(k) \quad (1.23)$$

definiendo

$$e(k) = A(q^{-1})v(k) \quad (1.24)$$

obtenemos la forma familiar

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k) \quad (1.25)$$

En donde $e(k)$ es en general un proceso aleatorio correlacionado.

Ej #2 si tenemos la siguiente estructura:

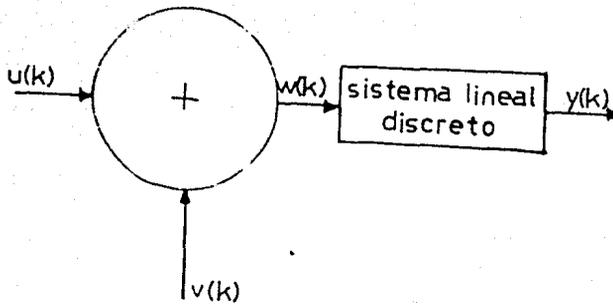


fig 1.2

$$A(q^{-1})y(k) = B(q^{-1})w(k) \quad (1.26)$$

$$w(k) = u(k) + v(k) \quad (1.27)$$

$$A(q^{-1})y(k) = B(q^{-1})u(k) + B(q^{-1})v(k) \quad (1.28)$$

Definiendo $e(k) = B(q^{-1})v(k)$ obtenemos:

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k) \quad (1.29)$$

que es igual a la ecuación (1.25)

Ej #3 si tenemos un sistema como el siguiente:

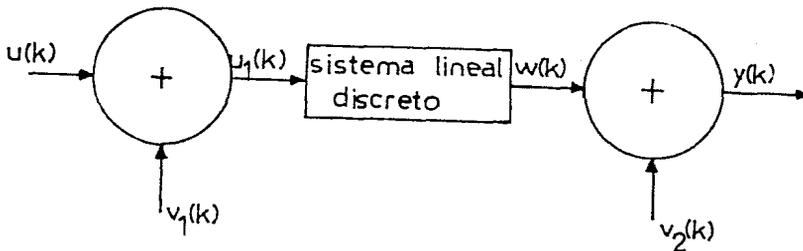


fig 1.3

$$A(q^{-1})w_1(k) = B(q^{-1})u_1(k) \quad (1.30)$$

$$w_2(k) + v_2(k) = \gamma(k) \quad (1.31)$$

$$u_1(k) + v_1(k) = u(k) \quad (1.32)$$

$$A(q^{-1})y(k) - A(q^{-1})v_2(k) = B(q^{-1})u(k) + B(q^{-1})v_1(k) \quad (1.33)$$

definiendo

$$e(k) = B(q^{-1})v_1(k) + A(q^{-1})v_2(k) \quad (1.34)$$

obtenemos

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k) \quad (1.35)$$

Como vemos los tres casos se pueden reducir a un sistema con una estructura similar.



1.4 PROPIEDADES ESTADÍSTICAS DEL ESTIMADOR DE MÍNIMOS CUADRADOS

Si los procesos a estimar por medio de mínimos cuadrados tienen las siguientes propiedades:

$$E\{y(m)\} = X'(m)\theta \quad (1.36)$$

$$E\{(Y - X\theta)(Y - X\theta)'\} = V^{-2}I \quad (1.37)$$

El estimado de mínimos cuadrados $\hat{\theta}$ tiene las siguientes propiedades (donde θ es un estimador que resulta de la solución de la ecuación normal).

1.- $\hat{\theta}$ es una función lineal de los datos (1.38)

2.- $E\{\hat{\theta}\} = \theta$ (estimador insesgado) (1.39)

3.- $E\{(\hat{\theta} - \theta)(\hat{\theta} - \theta)'\} = (X'X)^{-1} V^{-2}$ (1.40)

4.- $E\{(\hat{\theta} - \theta)(\theta - \theta)'\} = E\{(\hat{\theta} - \theta)(\theta - \theta)'\}$ (1.41)

En donde θ es cualquier otro estimador insesgado y lineal.

Un estimado $\hat{\theta}$ es llamado consistente si $\hat{\theta}(m)$ tiende a θ con probabilidad 1 conforme m tiende a infinito. Supongamos que un sistema está descrito por $Y = X\theta + e$; unas condiciones dadas en (3) para que el estimador sea consistente son:

- $e(k)$ es una secuencia de variables aleatorias independientes con media cero y momentos acotados de cuarto orden.

& El lector interesado en las demostraciones puede consultar la ref(16), pag 26-28

- Todas las raíces de $z^n + a_1 z^{n-1} + \dots + a_n$ estén dentro del círculo unitario.
- La secuencia de entrada $u(k)$ es independiente de $e(k)$.
- La secuencia de entrada es una señal persistente de orden n .

Hay que hacer notar que la segunda condición implica que el sistema a identificar es estable. Ljung en (24) dio unas condiciones más débiles y generales para la consistencia, las cuales son:

- $E\{e(k)/e(k-1), \dots, e(0)\} = 0$.
- $E\{|e(k)|^2 / e(k-1), \dots, e(0)\} < C$.
- La entrada $u(k)$ es independiente de $e(k-1), e(k-2), \dots, e(0)$.
- $\lim_{m \rightarrow \infty} \sup (1/m) \sum_{k=1}^m (|y(k)|^2 + |u(k)|^2) < \infty$ c.p.l.
- $X'X$ tiende a una matriz positiva definida c.p.l.

1.5 MINIMOS CUADRADOS GENERALIZADOS

Para poder explicar el concepto de mínimos cuadrados generalizados tenemos que poner el estimador de mínimos cuadrados en la siguiente forma (21):

$$\hat{\theta}(m) = \theta + (1/m) \sum_{k=1}^m x(k)x'(k) \quad -1 \quad 1/m \sum_{k=1}^m x(k)e(k) \quad (1.42)$$

Sabemos que una propiedad deseada del estimador de mínimos cuadrados es de que $\hat{\theta}(m)$ converga a θ cuando m tiende a infinito. De la ecuación (1.42) vemos que esto es cierto si las series $e(k)$ y $x(k)$ no están correlacionadas. Esto es cierto en los siguientes casos:

- $e(k)$ es una secuencia de variables aleatorias independientes con media cero (ruido blanco). Entonces $e(k)$ no depende de lo que ha pasado hasta $k-1$ y por lo tanto $E\{e(k)x(k)} = 0$.
- La secuencia de entrada $u(k)$ es independiente de la secuencia de ruido $e(k)$ y que la matriz X esté compuesta por puras entradas, en este caso $E\{e(t)x(t)} = 0$.

Si para un proceso determinado no se cumple que $e(k)$ y $x(k)$ no estén correlacionados, entonces los estimados tendrán un sesgo; para solucionar este problema existen varios métodos, uno de ellos es el método de los mínimos cuadrados generalizados cuyo objetivo es el de convertir los residuos correlacionados en residuos blancos; esto lo hace mediante el filtrado de los datos de entrada y de salida. En esta sección consideraremos que $e(k)$ tiene un espectro de potencia racional de tal forma que satisface el siguiente modelo autorregresivo:

$$e(k) + \sum_{i=1}^p c_i e(k-i) = \epsilon e(k) \quad (1.43)$$

en donde los c_i son coeficientes constantes y p es el orden del modelo. En general c_i y p son parámetros desconocidos a priori. No obstante, buenos resultados pueden obtenerse si suponemos p como 2 o 3 y luego optimizamos c_i . Reescribiendo la ecuación (1.43)

$$C(q^{-1})e(k) = ee(k) \quad (1.44)$$

donde

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_p q^{-p} \quad (1.45)$$

Esto implica que $e(k)$ es un ruido blanco filtrado con función de transferencia

$$e(z)/ee(z) = 1/C(z^{-1}) \quad (1.46)$$

Combinando la ecuación (1.24) con la ecuación (1.44) se obtiene:

$$v(z)/ee(z) = C(z^{-1})/A(z^{-1}) \quad (1.47)$$

Relacionando la ecuación (1.25) con la (1.44) se tiene:

$$C(q^{-1})A(q^{-1})v(k) = C(q^{-1})B(q^{-1})u(k) + ee(k) \quad (1.48)$$

Este modelo tiene la característica de tener ruido blanco $ee(k)$; y estimadores consistentes de los parámetros a_i, b_i, c_i se pueden obtener si se minimiza la siguiente función de error:

$$J = \sum_{k=1}^N ee(k)^2$$

$$- \sum_{k=0}^{\infty} \lambda^k \left[\sum_{i=0}^k a_i y(k-i) - \sum_{j=0}^k b_j u(k-j) \right] \quad (1.99)$$

En el contexto de identificación de sistemas a $e(k)$ se le denomina ecuación generalizada del error y al método por medio del cual estimamos los parámetros a_i, b_i, c_i se le denomina procedimiento de mínimos cuadrados generalizados. Al filtro $(1/z^{-1})$ se le denomina filtro blanqueador.

1.6 ALGORITMO DE MINIMOS CUADRADOS GENERALIZADOS

En esta sección trataremos la estimación de los parámetros del sistema por medio del algoritmo de mínimos cuadrados generalizados. Debido a que la ecuación (1.48) es no lineal en los polinomios $A(q^{-1})$, $B(q^{-1})$ y $C(q^{-1})$; los parámetros no se pueden estimar mediante procedimientos lineales por esos utilizaremos un algoritmo numérico. Los mínimos cuadrados generalizados son un método en el cual el criterio J es minimizado con respecto a a_i, b_i y luego con respecto a c_i ; este procedimiento es iterado como se describe a continuación:

Paso 1 Hacer $C(q^{-1}) = 1$, por lo que

$$\begin{aligned}
 J &= \sum_{k=1}^m [A(q^{-1})y(k) - B(q^{-1})u(k)]^2 \\
 &= (Y - XB)'(Y - XB)
 \end{aligned}
 \tag{1.50}$$

Si minimizamos con respecto a θ obtenemos:

$$\theta = (X'X)^{-1} X'Y
 \tag{1.51}$$

Paso 2 Con los estimadores de $A(q^{-1})$ y $B(q^{-1})$ definimos el residuo $r(k)$ como:

$$r(k) = A(q^{-1})y(k) - B(q^{-1})u(k)
 \tag{1.52}$$

en donde A y B son los estimados de A y B respectivamente. Ahora hacemos la prueba de autocorrelación de los residuos y si los residuos resultan ser blancos detenemos el algoritmo y habremos obtenido unos estimadores eficientes, sino vamos al Paso 3.

Paso 3 Hacemos

$$J = \sum_{k=1}^M [C(q^{-1})e(k)]^2 \quad (1.53)$$

$$= (re - HC)'(re - HC)$$

donde

$$C = [c_1 \ c_2 \ \dots \ c_p]$$

$$re = [re(p+1) \ \dots \ re(p+N)]'$$

$$H = \begin{bmatrix} -re(p) & \dots & -re(1) \\ \vdots & & \vdots \\ -re(p+N) & \dots & -re(N) \end{bmatrix}$$

El estimador $\hat{\theta}$ que minimiza J_2 esta dado por los mínimos cuadrados normales

$$\hat{\theta} = (H'H)^{-1} H're \quad (1.54)$$

Paso 4 Definimos

$$\hat{u}(k) = \hat{\theta}(q^{-1})u(k) \quad (1.55)$$

$$\hat{y}(k) = \hat{\theta}(q^{-1})y(k) \quad (1.56)$$

y

$$J_3 = \sum_{k=1}^n [A(q^{-1})y(k) - B(q^{-1})u(k)]^2$$

$$= (Y - X\theta)'(Y - X\theta)$$

(1.57)

En donde \underline{Y} y \underline{X} resultán de la substitución de $y(k)$ y $u(k)$ en las matrices Y y X . Ahora obtenemos el siguiente estimado

$$\hat{\theta} = (\underline{X}'\underline{X})^{-1} \underline{X}'\underline{Y}$$

(1.58)

y regresamos al Paso 2

1.7 MÍNIMOS CUADRADOS GENERALIZADOS RECURSIVOS

Es posible deducir una versión de mínimos cuadrados generalizados recursivos. Este algoritmo de estimación consiste en 2 rutinas recursivas de mínimos cuadrados, una para ajustar $\theta(N)$ y otra para $\epsilon(N)$ a medida que N se incrementa. Debido al hecho que $\theta(N)$ y $\epsilon(N)$ son vectores que varían con el tiempo; las señales filtradas $\underline{y}(k)$, $\underline{u}(k)$ y el residuo $e(k)$ son señales generadas por sistemas cuyos parámetros varían en el tiempo; es por esto que se le da mayor peso a las señales más recientes. Los resultados se muestran a continuación:

$$\theta(N+1) = \theta(N) + G(N+1)P(N)X(N+1)[Y(N+1) - X'(N+1)\theta(N)] \quad (1.59)$$

$$\epsilon(N+1) = \epsilon(N) + B(N+1)L(N)S(N+1)[e(N+1) - S'(N+1)\epsilon(N)] \quad (1.60)$$

donde

$$\underline{y}(N+1) = y(N+1) + \sum_{i=1}^p \epsilon(N)y(N+1-i)$$

$$\underline{u}(N+1) = u(N+1) + \sum_{i=1}^p \epsilon(N)u(N+1-i)$$

$$\underline{x}(N+1) = [-y(N), \dots, -y(N+1-n), u(N+1), \dots, u(N+1-n)]$$

$$P(N+1) = 1/hh[P(N) - G(N+1)P(N)X(N+1)X'(N+1)P(N)]$$

para $0 < hh < 1$

$$S(N+1) = 1/[1 + S'(N+1)P(N)\underline{x}(N+1)]$$

$$e(N+1) = y(N+1) + \sum_{i=1}^N A_i(N+1)y(N+1-i) - \sum_{i=1}^N K_i(N+1)u(N+1-i)$$

$$S(N+1) = [1 - e(N), -e(N-1), \dots, -e(N-p+1)]'$$

$$L(N+1) = 1/hh[L(N) - B(N+1)L(N)S(N+1)S'(N+1)L(N)]$$

$$B(N+1) = 1/[1 + S'(N+1)L(N)S(N+1)] \quad (1.61)$$

La función error correspondiente es

$$J = \sum_{i=1}^N h_i^2 \frac{(N-i)^2}{e(i)} \quad (1.62)$$

De la ecuación anterior vemos que mientras mas recientes sean los datos, mas se toman en cuenta.

Para mayores detalles ver ref(19), pag 2057-2062

1.8 APLICACION DE LA TECNICA DE MINIMOS CUADRADOS A SISTEMAS NO LINEALES

La técnica de mínimos cuadrados se puede utilizar en sistemas no lineales, siempre y cuando la relación con respecto a los parámetros sea lineal; por ejemplo: si tuviéramos un sistema no lineal con parámetros lineales, cuyo diagrama de bloques se muestra a continuación:

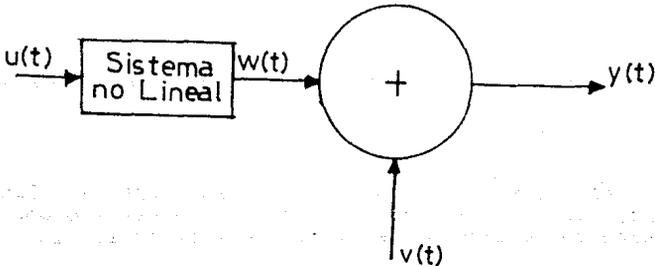


fig 1.4

y conociéramos exactamente la estructura del sistema, entonces podríamos aplicar las técnicas de identificación vistas anteriormente. Ej si:

$$w(k) = \theta_1 w^2(k-1) + \theta_2 u(k-1) + \theta_3 u(k-1)w(k) \quad (1.53)$$

como (En donde los parámetros θ_i son lineales)

$$y(k) = w(k) + v(k) \quad (1.54)$$

y si definimos

$$\theta = \{\theta_1, \theta_2, \theta_3\}' \quad (1.65)$$

$$x(k) = \{y(k-1), u(k-1), u(k-1)y(k)\}' \quad (1.66)$$

podemos escribir

$$y(k) = x'(k)\theta + e(k) \quad (1.67)$$

en donde $e(k)$ es la ecuación de error

$$e(k) = \theta_1 [v(k-1) - 2y(k-1)v(k-1)] + \theta_3 u(k-1)v(k) + v(k) \quad (1.68)$$

como la ecuación (1.67) tiene una estructura idéntica a las vistas con anterioridad; los parámetros del sistema, descrito por la ecuación (1.63), pueden ser estimados mediante el criterio de los mínimos cuadrados.

En el siguiente ejemplo veremos una configuración no lineal denominada configuración de Hammerstein; cuyo diagrama de bloques aparece a continuación:

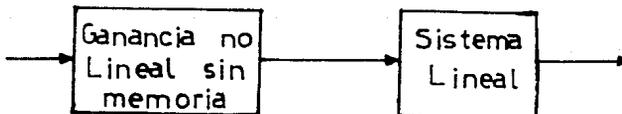


fig 1.5

La ganancia no lineal se puede aproximar por medio de un polinomio de potencias que tenga un orden p.

$$x(k) = u(k) + Z^2 u(k) + \dots + Z^m u(k) \quad (1.69)$$

en donde los Z^i $i=1 \dots m$ se seleccionan de tal forma que nos den una buena aproximación de un sistema con ganancia no lineal y sin memoria. El subsistema lineal tiene una ecuación en diferencias de orden n.

$$A(q^{-1})y(k) = B(q^{-1})x(k) \quad (1.70)$$

donde

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_n q^{-n}$$

en donde se supone que el sistema lineal es estable y que el ruido aditivo de salida $v(k)$ es una variable aleatoria con media cero. El problema de identificación se traduce en estimar los parámetros a_i, b_i y Z^i tomando en cuenta que n y p son conocidos.

Combinando las ecuaciones (1.69) y (1.70) obtenemos la siguiente ecuación del sistema

$$A(q^{-1})y(k) = B(q^{-1})\{u(k) + \sum_i Z^i u(k)\} + e(k) \quad (1.71)$$

donde

$$e(k) = A(q^{-1})v(k)$$

En donde vemos que los Z_i multiplican a las b_i de tal forma que obtenemos un producto cruzado. Esto significa que tenemos que resolver una identificación con parámetros no lineales. Sin embargo, si se toman los productos cruzados como parámetros nuevos; la identificación se vuelve lineal en los parámetros. A continuación se presenta la solución mediante mínimos cuadrados generalizados.

Primeramente se definen los polinomios $S_i(q^{-1})$

$$\begin{aligned} S_i(q^{-1}) &= Z_i B_i(q^{-1}) \\ &= s_{i0} + s_{i1} q^{-1} + \dots + s_{in} q^{-n} \end{aligned} \quad (1.72)$$

donde

$$s_{ij} = Z_{ij} b_{ij} \quad i = 1, \dots, m \quad j = 1, \dots, n \quad (1.73)$$

tenemos

$$A(q^{-1})y(k) = B(q^{-1})u(k) + \sum_i S_i(q^{-1})u_i(k) + e(k) \quad (1.74)$$

en donde $e(k)$ satisface un modelo autorregresivo

$$C(q^{-1})e(k) = ee(k) \quad (1.75)$$

siendo $ee(k)$ una variable aleatoria independiente con media cero, y

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_p q^{-p}$$

El algoritmo de mínimos cuadrados generalizados se puede aplicar para estimar los parámetros a_i, b_i y S_i y los $b_i \cdot Z_i$ pueden ser separados de los S_i en una etapa posterior. Definamos un conjunto de señales filtradas

$$\underline{u}(k) = C(q^{-1}) u(k) \quad (1.76)$$

$$\underline{y}(k) = C(q^{-1}) y(k) \quad (1.77)$$

La ecuación (1.74) se convierte en

$$A(q^{-1}) \underline{y}(k) = B(q^{-1}) \underline{u}(k) + \sum_i S_i(q^{-1}) \underline{u}(k) + e(k) \quad (1.78)$$

Esta ecuación y la ecuación 4 son la que se aplican en el algoritmo de mínimos cuadrados generalizados, como se muestra a continuación:

Paso 1

Hagamos $c_i = 0$ y obtenemos un estimado por mínimos cuadrados de los parámetros $(\hat{a}_i, \hat{b}_i, \hat{S}_i)$, usando los datos sin filtrar $y(k)$ y $u(k)$.

Paso 2

Generar los residuos $e(k)$ por medio de la ecuación (1.74) y si los residuos son blancos pasamos al paso 5.

Paso 3

Obtener una estimación por medio de mínimos cuadrados de c_i .

Paso 4

Generar las señales filtradas $y^{(i)}$ y $y^{(k)}$ $i=1, \dots, n$
por medio de las ecuaciones (1.74) y (1.77) e irse al paso 2

Paso 5

Estimar los Z_i por medio de la ecuación (1.73) obteniendo:

$$Z_i = \begin{pmatrix} 2 & -1 \\ 1 & j i \end{pmatrix} K \xi \quad i=2, \dots, n$$

1.9 ESTIMACION EN LAZO CERRADO

1.9.1 INTRODUCCION

Como ya sabemos el objetivo de un experimento de identificación es el de determinar la dinámica de un proceso determinado. Es muy importante estudiar el fenómeno de malla cerrada ya que la mayoría de los procesos forman parte de una configuración de control y la entrada al proceso se determina, parcialmente, de la realimentación de ciertas señales. Mas aún, en algunos casos la seguridad o motivos de producción no permiten que los reguladores sean quitados durante el experimento de identificación y en otros casos como economía o sistemas biológicos la realimentación es un efecto que esta dentro del mismo sistema.

1.9.2 PRINCIPIO BASICOS

Se podría decir que el resultado de un experimento de identificación depende de ciertos elementos como lo son:

- a) El sistema
- b) La estructura del modelo o parametrización
- c) El método de identificación utilizado
- d) Las condiciones del experimento

a) El sistema

Para esto consideraremos sistemas lineales, multivariables, discretos y estocástico; que designaremos con la letra Ψ y que tienen la siguiente forma general:

$$y(t) = G_{\Psi}(q^{-1})u(t) + H_{\Psi}q^{-1}e(t) \quad (1.81)$$

La salida, $y(t)$, es un vector de dimensión n_y y la entrada, $u(t)$, tiene dimensión n_u . Las variables, $e(t)$, son secuencias independientes y aleatorias con media cero y covarianza $E[e(t)e'(t)] = \Lambda$. Se supone que $G_{\Psi}(z)$ y $H_{\Psi}(z)$ son matrices que tienen dimensiones adecuadas.

b) El modelo

Para determinar el modelo del sistema las funciones $G(z)$ y $H(z)$ tienen que ser parametrizadas de una manera conveniente por medio del vector paramétrico θ . Un modelo que corresponde a un cierto valor de θ lo designaremos $\Delta(\theta)$ y estará dado por:

$$y(t) = G_{\theta}(q-1)u(t) + H_{\theta}(q-1)\epsilon(t) \quad (1.82)$$

donde $[\epsilon(t)]$ es una secuencia de vectores independientes y aleatorios, con media cero y covarianza Λ . Cuando θ es variada en una región de posibles valores, la ecuación (1.82) representa una familia de modelos que denotaremos por Δ . A ésta familia algunas veces la llamaremos "estructura del modelo". De secciones anteriores sabemos que el problema de identificación consiste en el de determinar θ de tal forma que $\Delta(\theta)$ describa, en cierto sentido, al sistema Ψ , dado por la ecuación (1.81).

La parametrización de $G_{\theta}(q-1)$ y $H_{\theta}(q-1)$ puede hacerse mediante una ecuación vectorial en diferencias:

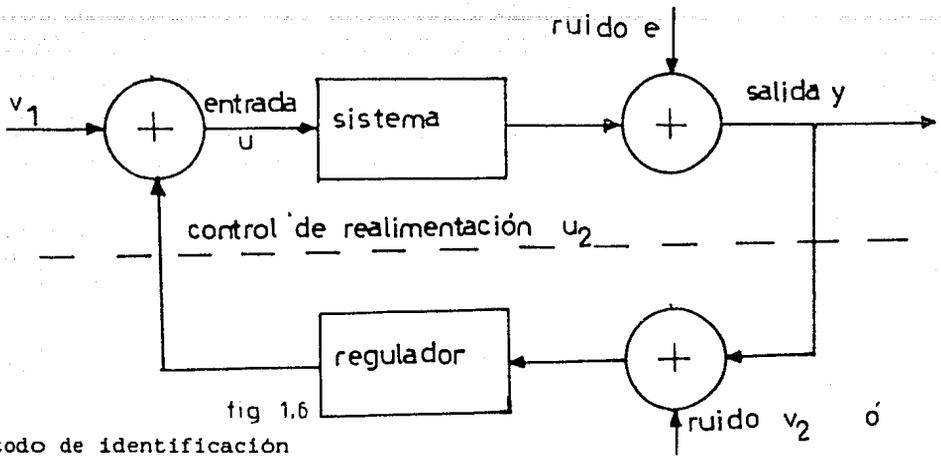
$$A_{\theta}(q-1)y(t) = B_{\theta}(q-1)u(t) + C_{\theta}(q-1)\epsilon(t) \quad (1.83)$$

o también puede representarse mediante la representación en variables de estado como se muestra a continuación:

$$\begin{aligned} x(t+1) &= A_{\theta}x(t) + B_{\theta}u(t) + K_{\theta}\epsilon(t) \\ y(t) &= C_{\theta}x(t) + D_{\theta}u(t) + \epsilon(t) \end{aligned} \quad (1.84)$$

c) Condiciones del experimento

La entrada $u(t)$ a el sistema dado por (1.81), puede ser escogida de distintas maneras. Puede, como en los experimentos de malla abierta, ser determinada libremente por el diseñador del experimento. O puede ser determinada parcialmente de la salida de un regulador, etc. La manera como la entrada es determinada la designaremos como "condiciones del experimento" denotandola por σ . La configuración que analizaremos es como la que se muestra a continuación:



d) Método de identificación

El procedimiento mediante el cual θ es determinado es llamado método de identificación, ζ . Para sistemas que operan en malla cerrada, es posible aplicar los métodos de identificación. Una manera es la de tratar a los datos de entrada y salida como si hubiesen sido obtenidos de un experimento de malla abierta. A éste procedimiento lo llamaremos identificación directa, ζ_1 .

Si el regulador es lineal, libre de ruido e invariante con el tiempo; entonces un método indirecto de estimación de los parámetros puede ser aplicado. El sistema de malla cerrada puede ser considerado como un todo y sus parámetros determinados mediante algún método. A ésta aproximación la llamaremos identificación indirecta, ζ_2 .

Más aún, en algunos casos puede ser provechoso el considerar la entrada y la salida conjuntamente; como si fueran la salida de un sistema manejado por puro ruido. A éste tipo de identificación lo designaremos como identificación conjunta entrada-salida, ζ_3 .

Diferentes métodos pueden ser empleados con este tipo de aproximaciones, como: análisis de correlación y de espectro, métodos paramétricos de identificación, etc. En este capítulo consideraremos el método de identificación del error de predicción. Para este método el vector de parámetros θ en el modelo es escogido de manera que la función:

$$g(Q_N(\Psi, \Delta(e)))$$

(1.85)

de la matriz

$$Q_N(\Psi, \Delta(\theta)) = 1/N \sum [y(t) - \hat{y}(t/t-1; \Delta(\theta))]^2$$

es minimizado. Aquí $\hat{y}(t/t-1; \Delta(\theta))$ denota la predicción de mínimos cuadrados de $y(t)$ basada en los datos hasta el tiempo $t-1$. Comúnmente la función g es la traza o el determinante de la matriz Q_N .

Si designamos al vector paramétrico que minimiza (1.85) en el tiempo N , como: $\theta(N; \Psi, \Delta, \sigma)$. En Ljung [24] se demuestra que bajo condiciones muy débiles:

$$\theta(N; \Psi, \Delta, \sigma) \rightarrow \theta_1(\Psi, \Delta, \sigma) \text{ con probabilidad 1 conforme } N \rightarrow \infty$$

donde $D(\Psi, \Delta, \sigma)$ es el conjunto de θ , que contiene un modelo-predicor que se comporta asintoticamente como un predicor verdadero en el sentido que:

$$D_1(\Psi, \Delta, \sigma) = \{ \theta / 1/N \sum E [\hat{y}(t/t-1; \Psi) - \hat{y}(t/t-1; \Delta(\theta))]^2 = 0 \}$$

Es claro que el estimador deseado esta dado por el conjunto:

$$D_T(\Psi, \Delta) = \{ \theta / G_\theta(z) = G_\Psi(z) \text{ y } H_\theta(z) = H_\Psi(z) \} \text{ para casi todas las } z.$$

Este conjunto está integrado por parámetros que nos dan modelos, Δ , que tienen la misma función de transferencia y las mismas características de ruido que el sistema Ψ .

1.9.3 ALGUNOS PROBLEMAS DE IDENTIFICACION

A continuación mostraremos algunos problemas interesantes en la identificación de sistemas:

ejemplo a.1. Consideremos un sistema como el que se muestra en la siguiente figura:

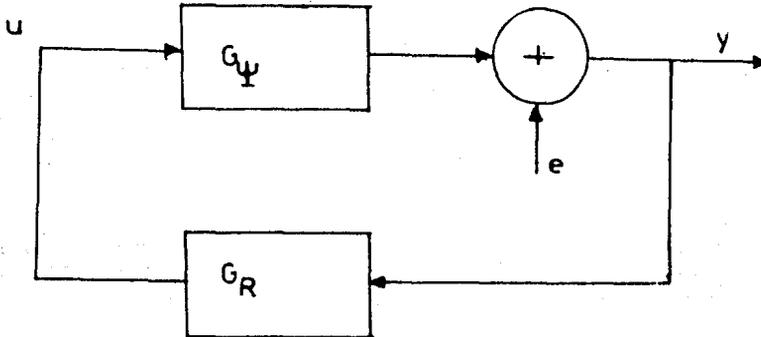


fig 1.7

Consideremos que la identificación se realiza mediante métodos que no implican una estructura causal (como los métodos basados en correlación o análisis espectral). Entonces la identificación de G_Ψ de las mediciones "y" y "u" nos darían, $G_\Psi = 1/G_R$, es decir la función inversa de la transferencia de la realimentación [1]. Si usamos otros métodos ej la identificación del error de predicción y si suponemos un modelo causal y paramétrico; entonces, en algunos casos, podremos lograr una identificación correcta.

ejemplo a.2. Consideremos el sistema:

$$y(t) + ay(t-1) = bu(t-1) + e(t) \quad (1.86)$$

con un regulador proporcional

$$u(t) = gy(t)$$

Si identificamos los parámetros a, b por medio de mínimos cuadrados, entonces todas las estimaciones de los parámetros

$$z = a + \gamma y(t)$$

$$k = b + \gamma$$

con γ arbitrario, nos darían el mismo valor del criterio de identificación. Para $\gamma \neq 0$ obtenemos una descripción errónea del proceso en malla abierta. Pero si conociéramos el parámetro "a" de antemano, obtendríamos una buena descripción del proceso en malla abierta. En estos ejemplos vemos la importancia de escoger un método apropiado para la identificación de sistemas (ejemplo 7.1) y que aún si escogemos una estructura adecuada podemos no obtener una buena descripción del proceso en malla abierta (ejemplo 7.2).

En cuanto a la determinación del orden del sistema, la pregunta que nos podría surgir es la de si podemos determinar el orden del sistema después de haber hecho el experimento. Esto lo trataremos de ilustrar mediante un ejemplo.

ejemplo a.3. Consideremos el sistema

$$A(q-1)y(t) = q-kB(q-1)u(t) + C(q-1)e(t) \quad (1.87)$$

con realimentación

$$F(q-1)u(t) = G(q-1)y(t) \quad (1.88)$$

y el sistema:

$$[A(q-1) + L(q-1)G(q-1)]y(t) = [q-kB(q-1) + L(q-1)F(q-1)]u(t) + C(q-1)e(t) \quad (1.89)$$

con realimentación (1.88) y con un polinomio arbitrario $L(q-1)$. Bajo estas condiciones el sistema (1.89) tendrá la misma relación entrada salida en malla cerrada que el sistema (1.87). Por lo que bajo estas circunstancias el sistema (1.87) no podrá ser distinguido del sistema (1.89). Esto significa que si tratamos de obtener el orden del sistema de los datos obtenidos en nuestro experimento, no podemos asegurar que el modelo obtenido tiene la misma función de

transferencia, en malla abierta, que el sistema a identificar.

1.9.4 DESARROLLO

Otro concepto importante es el de identificabilidad. Los parámetros verdaderos de un sistema θ se dice que son identificables, si la secuencia de estimados $\hat{\theta}(N)$ convergen a θ en un sentido estocástico. A continuación daremos una serie de definiciones que tienen por objeto que entendamos el concepto de identificabilidad.

Definición 1. El sistema Ψ se dice que es un "Sistema Identificable bajo Δ, ζ , y \square ", $SI(\Delta, \zeta, \square)$ si $\theta(N; \Psi, \Delta, \zeta, \square) \rightarrow D_T(\Psi, \Delta)$ con probabilidad uno conforme $N \rightarrow \infty$

Definición 2. El sistema Ψ se dice que es un "Sistema Fuertemente Identificable bajo ζ, \square ", $SFI(\zeta, \square)$ si es $SI(\Delta, \zeta, \square)$ para todo Δ donde $D_T(\Psi, \Delta)$ es un conjunto no vacío.

Definición 3. El sistema Ψ se dice que es "Identificable Parametricamente bajo Δ, ζ , y \square ", $IP(\Delta, \zeta, \square)$, si es $SI(\Delta, \zeta, \square)$ y $D_T(\Psi, \Delta)$ tiene solamente un elemento.

Estas definiciones las ilustraremos mediante los siguientes ejemplos:

ejemplo 7.4

Consideremos el sistema Ψ

$$y(t) = bu(t-1) + e(t) \quad (1.90)$$

Donde las condiciones del experimento, \square_1 , son las que se muestran en (1.91)

$$u(t) = gy(t) \quad (1.91)$$

Si primeramente suponemos una estructura del sistema Δ_1

$$y(t) = bu(t-1) + e(t) \quad (1.92)$$

y el parametro b es estimado mediante mínimos cuadrados $\zeta_{m.c.}$

Facilmente podemos ver que \hat{b} converge a b . Por lo que el sistema Ψ es $SI(\Delta_1, \zeta_{m.c.}, \square_1)$ y también $PI(\Delta_1, \zeta_{m.c.}, \square_1)$.

Ahora supongamos que tenemos una estructura del sistema Δ_2

$$y(t) + ay(t-1) = bu(t-1) + \varepsilon(t). \quad (1.93)$$

y la realimentación descrita por (1.91) la sustituimos en (1.93), obtenemos:

$$y(t) + (a-gb)y(t-1) = \varepsilon(t) \quad (1.94)$$

de donde todos a y b tales que $a - gb = -gb$; nos darán una descripción correcta del sistema de malla cerrada con la realimentación (1.91). El conjunto $D_1(\Psi, \Delta_2, \zeta_{m.c.}, \square_1)$ consiste en en todas las a y b que satisfacen la ecuación anterior. Sin embargo si sustituimos $a=1$ y $b = b + 1/g$ en (1.93) no obtenemos una función de transferencia correcta del sistema Ψ . Por lo que el sistema no es ni $SI(\Delta_2, \zeta_{m.c.}, \square_1)$, ni $IP(\Delta_2, \zeta_{m.c.}, \square_1)$. En particular Ψ no es $SFI(\zeta_{m.c.}, \square_1)$ para la configuración experimental propuesta. A manera de ejercicio el lector puede verificar que un sistema es $PI(\Delta, \zeta, \square)$ si es $SI(\Delta, \zeta, \square)$ y $D_1(\Psi, \Delta)$ tiene solamente un elemento; también puede verificar que si un sistema es $SFI(\zeta, \square)$ entonces también sera $SI(\Delta, \zeta, \square)$.

A continuación analizaremos la influencia del método de identificación en la identificabilidad y en la precisión. Como mencionamos anteriormente existen 3 procedimientos diferentes para hacer la identificación en malla cerrada; la identificación directa, ζ_1 , identificación indirecta, ζ_2 y la identificación conjunta entrada-salida, ζ_3 . Con cada uno de estos procedimientos diferentes métodos de identificación y estimación pueden ser usados. Debe hacerse notar que mientras ζ_1 y ζ_3 no imponen ninguna restricción a las condiciones del experimento, la identificación indirecta ζ_2 requiere que conozcamos la estructura del regulador y que no tenga ruido ($v_n = 0$). Con respecto a la identificación conjunta entrada-salida, ζ_3 , debemos hacer notar que aunque este método en principio es aplicable a configuraciones arbitrarias de realimentación, requiere que la realimentación pueda ser modelada mediante modelos paramétricos y por lo tanto este método no es recomendable cuando tenemos una realimentación no lineal. En seguida mostraremos algunas conclusiones sobre identificabilidad y precisión

de sistemas debido a el método de identificación dadas en [17] :

1. Si en ζ_1 y ζ_2 , usamos el método del error de predicción entonces obtendremos modelos idénticos si: hay un retraso en el regulador o en el sistema y el ruido en el regulador es independiente del ruido en el sistema.
2. Si ζ_1 es usado con el método del error de predicción y ζ_2 con cualquier otro método, entonces obtendremos las mismas propiedades de identificabilidad.
3. La precisión obtenida en ζ_1 y ζ_2 si usamos el método del error de predicción es la misma.
4. La precisión obtenida con el método de identificación indirecta no tiene ninguna ventaja a la obtenida por el método de identificación directa.

La influencia de la estructura del modelo en la precisión es de que el modelo debe de tener el menor número de parámetros como sea posible, para obtener mejor precisión [7].

A continuación analizaremos la influencia de las condiciones del experimento, \square , tanto en la identificabilidad como en la precisión de los parámetros del sistema. En cierto sentido \square , tiene una influencia mayor en los resultados que la que tienen Δ y ζ ; ya que si \square no nos proporciona identificabilidad o una precisión adecuada entonces tendremos que repetir todo el experimento. Por otra parte si Δ y ζ no fueron escogidos de manera adecuada, es posible probar otra Δ y ζ con los mismos datos. La influencia de las condiciones del experimento sobre la identificabilidad la ilustraremos mediante 2 teoremas.

teorema 1. Consideremos es sistema Ψ , (7.1) con la siguiente señal de entrada

$$u(t) = \sum_{i=1}^r F_i(q^{-1})y(t) + \sum_{i=1}^r K_i(q^{-1})v(t) \quad (1.95)$$

donde

$$v(t) = \delta + R(q^{-1})e(t)$$

$v(t)$ y $e(s)$ son procesos independientes para toda t y s ; R es un filtro causal y δ es una señal persistentemente excitada (ver [2] para obtener mayor información sobre señales persistentemente

excitadas). La ecuación (1.95) nos da a entender que hay "r" cambios de filtros durante el experimento de identificación de tal forma que cada uno es usado una parte del tiempo total. También supondremos que hay un retraso ya sea en el sistema o en el regulador ($G(0)F_1(0) = 0$) y que el método de identificación utilizado es el del error de predicción.

Un sistema Y es SFI(ζ, \square) si y solo si :

$$\text{rango} \begin{bmatrix} K_1(z) \dots K_r(z) & F_1(z) \dots F_r(z) \\ 0 \dots \dots \dots 0 & I \dots \dots I \end{bmatrix} \quad (1.96)$$

es $n_y + n_u$

La dimensión de K_i es n_u/n_y , de F_i n_u/n_y , de 0 n_u/n_y y de I n_y/n_y ; i de tal manera que la dimensión de la matriz (1.96) es $(n_u + n_y n_u) / (n_y + n_y)$. La prueba de este teorema está dada en [37].

Una condición necesaria para que (1.96) se cumpla es la de que $r \geq (n_u + n_y) / (n_u + n_y)$. Ahora discutamos algunos casos particulares; si $n_u = n_y$ y $K(z)$ es no singular, entonces (1.96) se satisface para cualquier r (incluyendo $r=1$) sin importar los filtros de realimentación F_i . Si $n_u = 0$ entonces (1.96) solamente se puede satisfacer si:

$$r \geq 1 + n_u/n_y \quad (1.97)$$

Esto significa que si alternamos en el experimento varios reguladores, podemos obtener SFI aún si $n_u = 0$. Si $n_u = n_y$, entonces es suficiente usar 2 reguladores. Estos se pueden escoger de tal forma que $\det(F(z) - F(z)) \neq 0$. A manera de conclusión podemos decir que cuando (1.96) se satisface entonces no existen dificultades adicionales para encontrar una estructura adecuada del modelo debido a la realimentación. En otras palabras, toda la identificación puede ser efectuada como si los datos fueran obtenidos durante un experimento de malla abierta, de hecho el caso de malla abierta está incluido en el teorema ($F_i = 0$).

Cuando (1.96) no se satisface, entonces el sistema no es SFI. Sin embargo puede ser SI para ciertas estructuras de modelos. Un caso importante en el cual SFI es imposible es cuando tenemos un regulador que es invariante con el tiempo, lineal, sin ruido y $n_y = 0$; ej

$r=1$, $n_v=0$. A continuación investigaremos el caso que además de cumplir las condiciones anteriores $n_u=n_y=1$ y Δ está descrito por:

$$\begin{aligned}
 y(t) + a_1 y(t-1) + \dots + a_{n_a} y(t-n_a) &= \\
 &= b_1 u(t-k-1) + \dots + b_{n_b} u(t-k-n_b) \\
 &\quad + e(t) + c_1 e(t-1) + \dots + c_{n_c} e(t-n_c) \quad (1.98) \\
 k &\geq 0, a_{n_a} \neq 0, b_1 \neq 0, b_{n_b} \neq 0, c_{n_c} \neq 0;
 \end{aligned}$$

y el regulador es

$$\begin{aligned}
 u(t) = -f_1 u(t-1) - \dots - f_{n_f} u(t-n_f) \\
 \quad + g_0 y(t) + \dots + g_{n_g} y(t-n_g) \quad (1.99) \\
 f_{n_f} \neq 0, g_{n_g} \neq 0.
 \end{aligned}$$

Si definimos los polinomios $A(z)$, $B(z)$, $C(z)$, $F(z)$ y $G(z)$ como

$$A(z) = 1 + a_1 z + \dots + a_{n_a} z^{n_a}$$

$C(z)$ y $F(z)$ analogamente

$$B(z) = b_1 z + \dots + b_{n_b} z^{n_b}$$

$$G(z) = g_0 + g_1 z + \dots + g_{n_g} z^{n_g}$$

Si suponemos que $F(z)$ y $G(z)$ son relativamente primos y que los polinomios $A(z)$, $B(z)$ y $C(z)$ no tienen ningún valor en común; $C(z)$ y $A(z)F(z) - z^k B(z)G(z)$ tienen exactamente n_p factores comunes. Entonces tendremos el siguiente resultado:

teorema 7.2 Consideremos el sistema descrito por (1.98) con realimentación dada por (1.99); y la clase de modelos Δ también descritos por (1.98). entonces Ψ es $SI(\Delta, \zeta, \rho)$ si y solo si:

$$\max(n_f - n_b, n_g + k - n_a) - n_p \geq 0. \quad (1.100)$$

la prueba se demuestra en [35]

En seguida veremos la influencia de las condiciones del experimento en la precisión. Frecuentemente se dice que si la identificación de sistemas que operan en malla cerrada es teóricamente posible, los estimados obtenidos en la práctica son muy pobres debido al efecto del término de realimentación en la señal de entrada. Normalmente la razón de la realimentación es la de disminuir la variación de la salida; por lo que es natural que los estimados obtenidos tengan menos precisión que los estimados obtenidos en malla abierta. Por esta razón los experimentos en malla cerrada pudieran ser considerados como inferiores a los de malla abierta; sin embargo esta comparación es injusta ya que una variable que determina la identificación de un experimento es la varianza de la salida, ya que muchas veces esta variación debe ser mantenida por debajo de un cierto nivel, debido a razones de producción. Por lo que diferentes condiciones experimentales deben de ser comparadas para la misma varianza en la salida.

Bajo estas suposiciones las preguntas a contestar son :

1. Si los experimentos en malla abierta son mejores que los experimentos en malla cerrada cuando la varianza en la salida es la misma.
2. Si hay alguna diferencia sistemática entre varias condiciones experimentales que nos proporcionan identificabilidad.

En [34] y [36] estos problemas son estudiados mediante simulaciones y análisis de sistemas con ordenes bajos. Las respuestas son las siguientes

1. Experimentos en malla abierta no necesariamente son mejores a los de malla cerrada si la varianza en la salida es la misma.
2. No hay una diferencia sistemática entre las condiciones de los experimentos que nos dan identificabilidad.

- 1.- Simulación de un sistema lineal discreto.
- 2.- Arreglo de los datos del sistema y ejecución del algoritmo de mínimos cuadrados.
- 3.- Análisis de datos.

La subrutina de simulación tiene como finalidad la de generar datos de entrada y de salida. Para esto tiene un algoritmo que genera ruido blanco y filtrado; otro que genera una secuencia pseudoaleatoria; y uno más que genere un sistema lineal discreto con o sin retraso. A su vez el subprograma de arreglo de los datos tiene como finalidad el obtener estimados mediante el criterio de mínimos cuadrados; ej forma las matrices "X" y "H" de tal forma que podamos obtener los estimados $\hat{\theta}$ y \hat{C} respectivamente. La subrutina análisis de datos tiene la función de correlacionar archivos, la de obtener la media de un archivo, la de graficar parte de un archivo. Dichas funciones se pueden emplear para ver si los residuos están correlacionados.



3.2 SIMULACION

El algoritmo de simulación tiene las siguientes opciones

- 1.- Generación de ruido con distintas distribuciones.
- 2.- Generación de una secuencia pseudoaleatoria.
- 3.- Generación de datos para el sistema
- 4.- Generación de residuos para el sistema.

Un diagrama esquemático lo podemos ver en la siguiente figura;

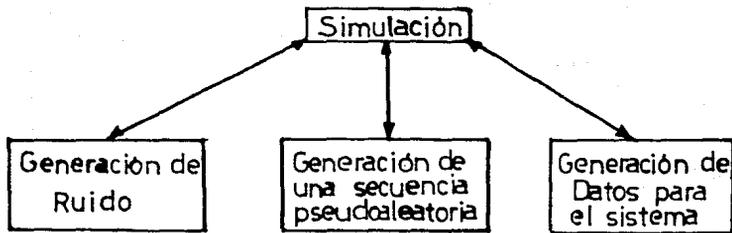


fig 2.2

2.2.1 GENERACION DE DATOS PARA EL SISTEMA

Un diagrama de la subrutina de generación de datos para el sistema, se muestra a continuación:

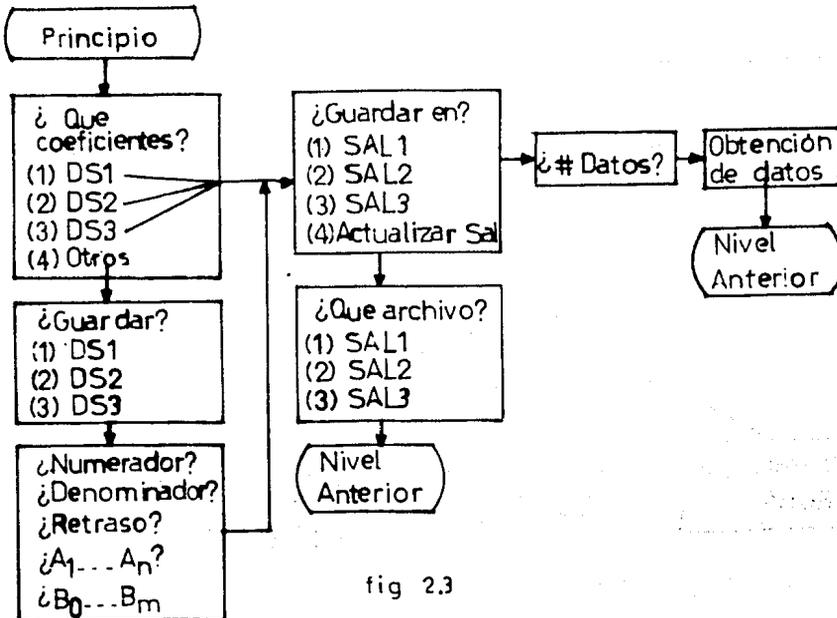


fig 2.3

Al acceder la subrutina Generación de datos para el sistema el programa nos pregunta:

Quisieres emplear los coeficientes del sistema almacenados en el archivo

- 1.- DS1
- 2.- DS2
- 3.- DS3
- 4.- Prefieres emplear otros coeficientes

Esto se debe basicamente a que esta subrutina simula un sistema discreto como se indica a continuaci3n:

$$y(k) = -\sum_{i=1}^n a_i y(k-i) + \sum_{i=0}^m b_i u(k-i-r) + e(k) \quad K= 0,1,2,\dots$$

(2.1)

donde r es el retraso. Si simulamos un sistema y luego queremos saber los coeficientes a_i y b_i con los que simulamos dicho sistema; lo 3nico que tenemos que hacer es mandar imprimir el contenido del archivo DS1. En dicho archivo existe la siguiente informaci3n: orden del numerador, orden del denominador, retraso, los coeficientes a_1,\dots,a_n y b_0,\dots,b_m . Tambi3n es 3til esta estructuraci3n en el caso de que necesitemos mas muestras de las simuladas anteriormente. Ej si en una simulaci3n generamos 100 datos y ahora necesitamos 150 datos, lo 3nico que tendr3amos que hacer es seleccionar la opci3n 1 y cambiar el n3mero de datos deseados.

Si seleccionamos la opci3n de emplear otros coeficientes lo que se nos preguntaría ser3a:

Escribe el n3mero del archivo en el cual quieres guardar tus datos

- (1) Archivo DS1
- (2) Archivo DS2
- (3) Archivo DS3

y despu3s

Escribe el orden del denominador....

Escribe el orden del numerador....

Escribe cuanto vale el retraso....

Escribe el valor de A(1).....A(4)

Escribe el valor de B(1).....B(4)

posteriormente aparece en la pantalla

Escribe el número del archivo en el cual
quieres guardar tus datos

- (1) Archivo SAL1
- (2) Archivo SAL2
- (3) Archivo SAL3
- (4) Quieres actualizar el archivo SAL

En donde la función de actualizar el archivo SAL se describe a
continuación

Cual de los siguientes archivos lo quieres
guardar en el archivo SAL

- (1) SAL 1
- (2) SAL 2
- (3) SAL 3

La razón de hacer estos dos pasos de almacenamiento de datos para el sistema es el de que en algunos algoritmos de identificación el archivo de salida tiene que ser filtrado y consecuentemente modificado. Si en este algoritmo quisiéramos después de pasar por un filtro los datos de salida, acceder los datos originales, lo único que tendríamos que hacer es invocar a cualquiera de los archivos SAL1, SAL2, SAL3.

2.2.2 GENERACION DE UNA SECUENCIA PSEUDOALEATORIA

Un diagrama de esta subrutina se muestra a continuación:

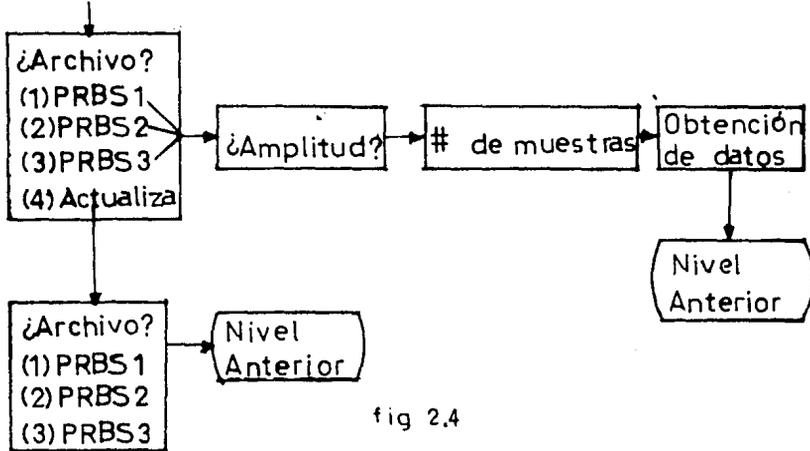


fig 2.4

Al invocar la subrutina de generación de una secuencia pseudoaleatoria, el programa nos pregunta:

En que archivo quieres guardar tus datos

- (1) PRBS1
- (2) PRBS2
- (3) PRBS3
- (4) Actualizar el archivo ENT

Si escogemos la opción (4) tendremos

Cual de los siguientes archivos lo quieres guardar en el archivo ENT

- (1) PRBS1

(2) PRBS2
(3) PRBS2

En caso de escoger cualquier otra opción nos aparecería en la pantalla

Escribe la amplitud de la secuencia pseudoaleatoria....

Cuántas muestras....

2.2.3 GENERACION DE RUIDO

Un diagrama esquemático del procedimiento sería el siguiente:

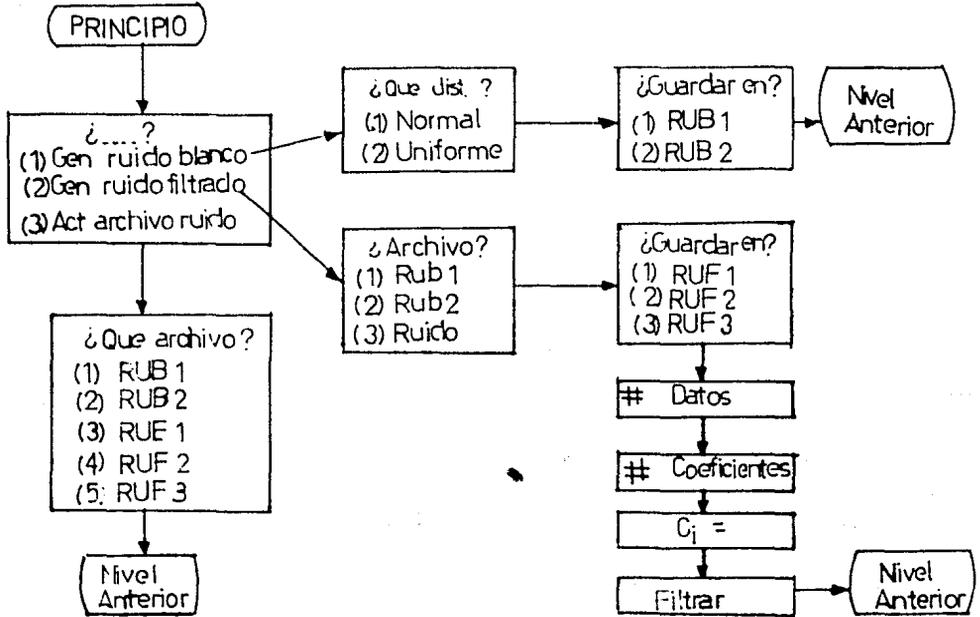


fig 2.5

Debido a la necesidad de generar ruido blanco y de color con distintas distribuciones; la subrutina ruido esta organizada de la siguiente manera:

En la tabla se describen los pasos:

- (1) Generación de ruido blanco
- (2) Generación de ruido filtrado
- (3) Actualiza el archivo de

Si se selecciona la opción 3 lo que se despliega en la pantalla es lo siguiente:

Escribe el número del archivo que
quieres poner en el archivo ruido

- (1) RUF1
- (2) RUF2
- (3) RUF1
- (4) RUF2
- (5) RUF3

En caso de escoger la opción 1 obtendríamos:

Escribe el número de la opción que quieras

- (1) Distribución uniforme
- (2) Distribución normal

Escribe en cual de los siguientes archivos
quieres guardar tus datos:

- (1) RUF1
- (2) RUF2

Escribe semilla y número de resultados

Si se hubiese seleccionado la opción #2 entonces lo que veríamos sería:

Escribe cual de los siguientes archivos quieres filtrar

- (1) RUF1
- (2) RUF2
- (3) RUF10

En cual de los siguientes archivos quieres guardar el ruido filtrado

- (1) RUF1
- (2) RUF2
- (3) RUF3

Dame el número de datos que quieras generar:

Escribe el número de coeficientes del filtro

El valor de $c_1 \dots c_p$ es:

2.3 ARREGLO DE DATOS

La subrutina de arreglo de los datos del sistema esta estructurada como lo indica la siguiente figura:

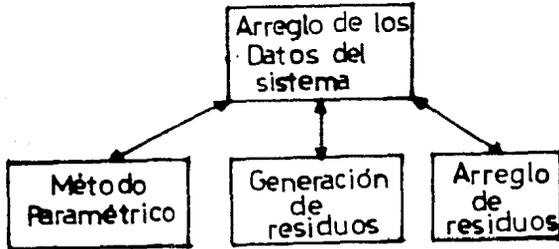


fig 2.6

2.3.1 METODO PARAMETRICO

El arreglo para el método paramétrico se hace de la siguiente forma; dado que sabemos que el sistema simulado tiene la siguiente estructura

$$y(k) = \sum_{i=1}^n a_i y(k-i) + \sum_{i=0}^m b_i u(k-i-r) + e(k) \quad (2.2)$$

Con estos datos formamos el sistema:

$$Y = XB + e \quad (2.3)$$

en donde

$$X = \left[\begin{array}{cccc|cccc} -y(m+r-1) & \dots & -y(m+r-r) & & u(m) & \dots & u(0) & \\ \vdots & & \vdots & & \vdots & & \vdots & \\ \vdots & & \vdots & & \vdots & & \vdots & \\ -y(m+r+p-2) & \dots & -y(m+r-r+p-1) & & u(m+p-1) & \dots & u(p-1) & \end{array} \right] \quad p$$

si $n \leq m+r$ (2.4)

$$Y = [y(m+r) \dots y(m+r+p-1)] \quad (2.5)$$

$$e = [e(m+r) \dots e(m+r+p-1)] \quad (2.6)$$

Donde p representa el número de renglones

Si $m+r > n$ tenemos las siguientes formulas

$$X = \left[\begin{array}{cccc|cccc} -y(n-1) & \dots & y(0) & & u(n-r) & \dots & u(r-(m+r)) & \\ \vdots & & \vdots & & \vdots & & \vdots & \\ \vdots & & \vdots & & \vdots & & \vdots & \\ -y(n+p-2) & \dots & -y(p-1) & & u(n+p-(r+1)) & \dots & u(p+n-(m+r+1)) & \end{array} \right] \quad (2.7)$$

$$Y = [Y(n) \dots Y(n+p-1)] \quad (2.8)$$

$$e = [e(n) \dots e(n+p-1)] \quad (2.9)$$

Comparando las estructuras anteriores vemos que si $n = m+r$ las matrices (2.4), (2.5) y (2.6) son iguales a (2.7), (2.8) y (2.9) respectivamente.

2.3.2 GENERACION DE RESIDUOS

El subprograma de residuos lo que hace es:

$$re(k) = A(q^{-1}) - B(q^{-1})u(k-r) \quad (2.10)$$

en donde

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}$$

$$B(q^{-1}) = b_0 + \dots + b_m q^{-m}$$

$r = \text{retraso}$

2.3.3 ARREGLO DE RESIDUOS

Para el arreglo de residuos se utiliza la siguiente estructura

$$re = H\theta + f \quad (2.11)$$

en donde

$$H = \begin{bmatrix} -re(p-1) & \dots & -re(0) \\ \vdots & & \vdots \\ -re(p+q-2) & \dots & -re(q-1) \end{bmatrix} \quad Q \quad (2.12)$$

$$re = [re(p) \dots re(p+q-1)] \quad (2.13)$$

$$C = \begin{bmatrix} c_0 & c_1 & \dots & c_p \end{bmatrix} \quad (2.14)$$

Y el estimado $\hat{\theta}$ se obtiene mediante la aplicación del criterio de mínimos cuadrados a la ecuación (2.11).

2.4 ANALISIS DE DATOS

La subrutina de arreglo de los datos del sistema tiene la siguiente estructura:

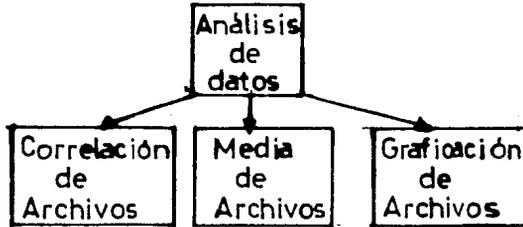


fig 2.7

2.4.1 CORRELACION DE ARCHIVOS

Un diagrama esquemático se muestra a continuación

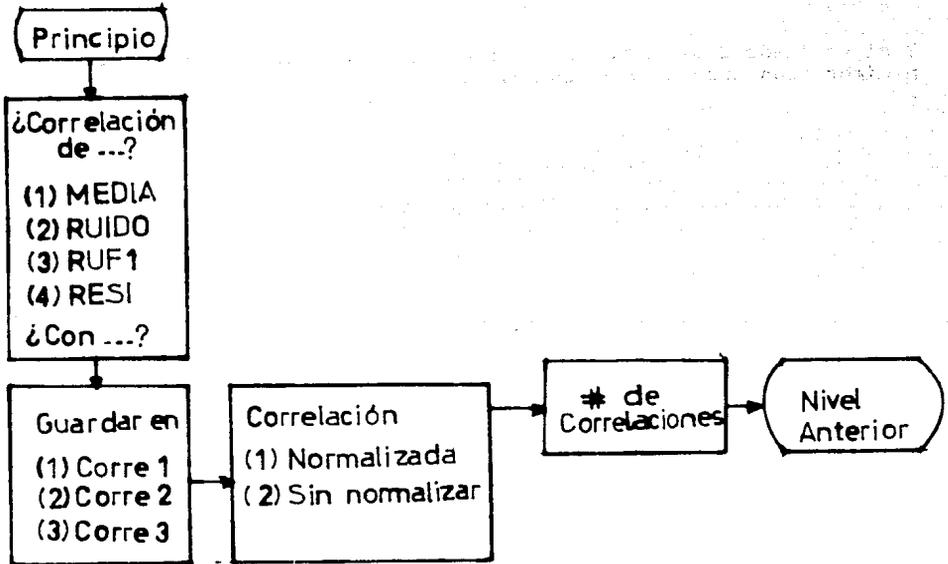


fig 2.8

Al llamar a la subrutina correlación lo que aparece en la pantalla es lo siguiente:

- (1) Media
- (2) Ruido
- (3) kuf1
- (4) Resi

Quieres hacer la correlación del archivo #

Con el archivo #

posteriormente aparece:

Escribe el número del archivo en el cual quieres guardar los datos correlacionados.

- (1) Correlación1
- (2) Correlación2
- (3) Correlación3

.....

y luego

Escribe el #1 si quieres correlación normalizada

Escribe el #2 si no la quieres normalizada

.....

Escribe el número de correlaciones que quieras

.....

3.4.2 MEDIA DE ARCHIVOS

Este subprograma lo podemos ver graficamente como sigue:

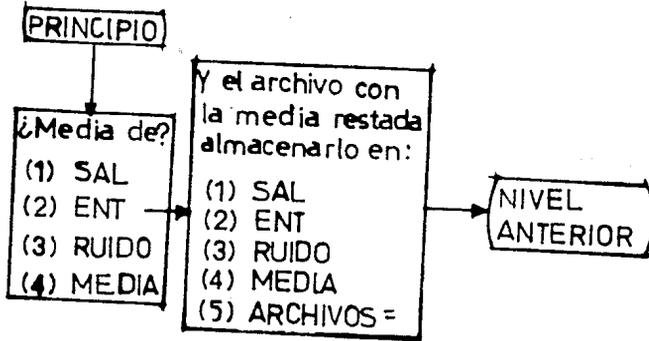


fig 2.9

Al principio se nos pregunta los siguiente:

Quieres obtener la media del archivo

- (1) Sal
- (2) Ent
- (3) Ruido
- (4) Media
-

y el archivo con la media restada almacenarlo en:

- (1) Sal
- (2) Ent
- (3) Ruido
- (4) Media
- (5) Quieres dejar los archivos iguales
-

2.4.3. GRAFICACION

Un diagrama esquemático de esta subrutina es el siguiente:

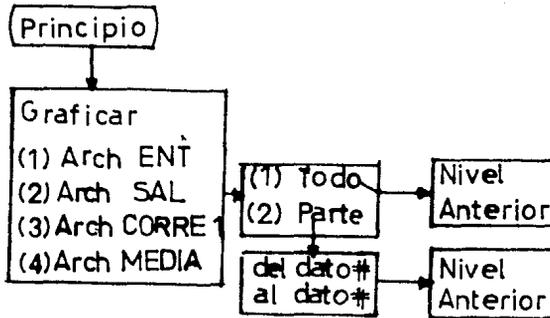


fig 2.10

Al entrar en esta subrutina el programa nos pregunta:

Cual de los siguientes archivos quieres graficar

- (1) Archivo Ent
- (2) Archivo Sal
- (3) Archivo Correl
- (4) Archivo Media

y luego

Escribe el número de la opción que desees

- (1) Graficar todo el archivo
- (2) Graficar parte del archivo

si se selecciona la opción #2 aparece lo siguiente:

Quieres graficar del dato #...

Al dato #....

3 PRINCIPALES ALGORITMOS UTILIZADOS EN LA ELABORACION DEL PROGRAMA

Basicamente son 2 los algoritmos mas importantes en el paquete, uno es la solución del problema de minimos cuadrados mediante el algoritmo de Householder; su importancia radica en que es numericamente estable. Y el otro algoritmo genera una serie de tiempo con distribución normal y su fuerza reside en que la mayoría de los ruidos reales pueden ser modelados mediante una distribución normal.

3.1 RESOLUCION DEL PROBLEMA DE MINIMOS CUADRADOS MEDIANTE EL ALGORITMO DE HOUSEHOLDER

De secciones anteriores sabemos que $(X'X)\theta = X'Y$, si denotamos a $X'X$ como Z y a $X'Y$ como L obtenemos:

$$Z\theta = L \quad (3.1)$$

Si este sistema, su lado izquierdo es multiplicado por unas matrices apropiadas obtenemos:

$$Z_n \theta = L_n \quad (3.2)$$

en donde $Z_j, L_j = P_j(Z[j-1], L[j-1])$; es decir que el sistema $Z_j\theta = L_j$ se puede obtener a partir del sistema $Z[j-1]\theta = L[j-1]$. Y la sensibilidad de los cambios en θ debidos a los arreglos Z_j, L_j para $j = 1, \dots, n$ de los estados intermedios esta dado por:

$$\text{cond}(Z_j) = \text{lub}(Z_j) \text{lub}(Z_j)^{-1} \quad (3.3)$$

en donde

$$\text{lub}(Z) = \max_{B \neq 0} \sqrt{B'Z'ZB / (B'B)}$$

Para las matrices Z_j donde $\text{cond}(Z_j) \gg \text{cond}(Z_0)$, la secuencia de cálculos no es numéricamente estable; ya que el error de redondeo ej tiene una influencia mas fuerte en el resultado final que el error inicial. Por esta razón es muy importante escoger P de tal forma que los números $\text{cond}(Z_j)$ no crezcan.

Ahora bien si la matriz P es unitaria ($P'P = I$) entonces:

$$\begin{aligned} \text{lub}(Z) &= \text{lub}(P'PZ) \\ &= \text{lub}(P')\text{lub}(PZ) = \text{lub}(PZ) \\ \text{lub}(PZ) &= \text{lub}(P)\text{lub}(Z) = \text{lub}(Z) \end{aligned}$$

por lo que

$$\text{lub}(PZ) = \text{lub}(Z) \quad (3.4)$$

y similarmente

$$\text{lub}(ZP) = \text{lub}(Z) \quad (3.5)$$

De lo anterior podemos concluir que $\text{cond}(Z) = \text{lub}(Z)\text{lub}(Z^{-1}) = \text{cond}(PZ)$, para una matriz unitaria P. Es decir que al multiplicar una matriz cualquiera por una matriz unitaria los números de condición no aumentan. Mas aún las transformaciones unitarias dejan la norma de un vector x invariante $\|Px\| = \sqrt{x'P'Px} = \sqrt{x'x} = \|x\|$ y como el criterio de mínimos cuadrados trata de minimizar $\|Y-X\theta\|^2 = (Y-X\theta)'(Y-X\theta)$; si P es unitaria y es obtenida del producto de n matrices unitarias ($P'P = P_1'P_1 \dots P_n'P_n \dots P = I$) entonces: $\|Y-X\theta\| = \|P(Y-X\theta)\| = \|Y[n]-X[n]\theta\|$ y además los números de condición asociados con el sistema $Z[j]\theta = L[j]$ no cambian.

Más aún las matrices $P[i,j]$ se pueden escoger de tal manera que las matrices $X[i,j]$ sean más sencillas. Como fue sugerido por Householder esto se puede hacer de la siguiente manera:

Se escoge la matriz P de la siguiente manera:

$$P = I - 2WW' \quad \text{donde } W'W = 1, \quad W \in \mathbb{R}^n \quad (3.6)$$

esto trae como consecuencia que la matriz P tenga las siguientes propiedades:

$$\begin{aligned} P' &= I - (2WW')' \\ &= I - 2WW' \\ &= P \end{aligned} \quad (3.7)$$

$$\begin{aligned} P'P &= PP \\ &= (I - 2WW')(I - 2WW') \\ &= I - 2WW' - 2WW' + 4WW'WW' \\ &= I \end{aligned} \quad (3.8)$$

Si queremos transformar un vector x (puede ser la primera columna de X), de tal manera que $x[*] = Px$ en donde $x[*] = ke_2$, donde $e_2 = (1, 0, 0, \dots)$. como $x[*]'x[*] = x'x$, esto implica que $|k| = \|x\| = \sqrt{x'x}$ y como $x'e_1$ debe ser real ya que $x'x[*] = x'Px = (x'Px)'$ se desprende que:

$$k = \pm e_1 \quad \text{ta donde} \quad t_1 = \sqrt{x'x} \quad (3.9)$$

De la restricción

$$\begin{aligned} Px &= x - 2(W'x)W \\ &= k \cdot e_1 \end{aligned}$$

y de la condición

$$W'W = 1$$

se deduce que

$$W = (x - k e) / \|x - k e\| \quad (3.10)$$

$$\text{si } x_1 = e^{(i\lambda a)} \quad |x_1|$$

$$\|x - k e\| = \|x + ta e^{(i\lambda a)}\|$$

$$= \sqrt{|x_1 + ta e^{(i\lambda a)}|^2 + |x_2|^2 + \dots + |x_n|^2}$$

$$= \sqrt{(|x_1| + ta)^2 + |x_2|^2 + \dots + |x_n|^2} \quad (3.11)$$

escogiendo $k = -ta e^{(i\lambda a)}$ resulta:

$$\begin{aligned} |x_1 - k| &= |x_1 + ta e^{(i\lambda a)}|^2 \\ &= |ta + |x_1||^2 \\ &= ta^2 + 2ta|x_1| + |x_1|^2 \end{aligned} \quad (3.12)$$

de donde

$$\|x - k e\|_1^2 = 2ta + 2ta\|x\|_1 \quad (3.13)$$

$$2WW' = (x - k e)(x - k e)' / \|x - k e\|_1^2 \quad (3.14)$$

La matriz $P = I - 2WW'$ se puede escribir en la siguiente forma:

$$P = I - QM'M' \quad (3.15)$$

con

$$ta = \sqrt{\frac{n}{\sum_{i=1}^n |x_i|^2}}, \quad x_i = e^{(i\lambda a)} |x_i|$$

$$k = -ta e^{(i\lambda a)}$$

$$M = x - k e = \begin{bmatrix} e^{(i\lambda a)} \\ (|x| + ta) \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$Q = (ta(ta + |x|))^{-1} \quad (3.16)$$

$$\begin{aligned}
 X[j-1] &= \begin{bmatrix} + & \dots & + \\ \vdots & & \vdots \\ + & \dots & + \\ \times [j-1] & \dots & \times [j-1] \\ \vdots & & \vdots \\ 0 & & \vdots \\ \vdots & & \vdots \\ \times [j-1] & \dots & \times [j-1] \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \end{bmatrix} \\
 &= \begin{bmatrix} D & B \\ 0 & X[j-1] \end{bmatrix} \quad (3.19)
 \end{aligned}$$

entonces determinamos la matriz unitaria $P[j]$ de orden $(n-j+1) \times (n-j+1)$ de acuerdo a (3.16)

$$P[j] \begin{bmatrix} \times [j-1] \\ j j \\ \vdots \\ \vdots \\ \times [j-1] \\ n j \end{bmatrix} = k \begin{bmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix} \quad (3.20)$$

La matriz unitaria deseada la construimos a partir de $P[j]$ como sigue

$$P_j = \begin{bmatrix} I[j-1] & 0 \\ 0 & P[j] \end{bmatrix} \begin{matrix} j-1 \\ n-j-1 \end{matrix} \quad (3.21)$$

Siguiendo este procedimiento una matriz como la indicada en la ecuación (3.26) se obtiene despues de n pasos. Ahora bien como $m \geq n$; podemos definir $h = Y[n]$ donde:

$$H = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \quad \begin{matrix} H_1 \in R^{n \times n} \\ H_2 \in R^{(m-n) \times n} \end{matrix} \quad (3.22)$$

de donde

$$Y[n] - X[n]\theta = \begin{bmatrix} H_1 - R\theta \\ H_2 \end{bmatrix} \quad \begin{matrix} n \\ m-n \end{matrix} \quad (3.23)$$

y como queremos minimizar $\|Y - X\theta\| = \|Y[n] - X[n]\theta\|$,
 escogemos θ de tal forma que

$$H_1 = R\theta \quad (3.24)$$

Si las columnas de X son linealmente independientes entonces el sistema descrito por la ecuación (3.23) puede ser resuelto en forma única para θ y esta θ es el único punto mínimo para el problema de los mínimos cuadrados; y el residuo de $\|Y - X\theta\|$ es $\|h_2\|$. Si tenemos h_1 y R , los elementos de θ se pueden obtener mediante la siguiente ecuación recursiva:

$$\theta_i = h_i - \sum_{k=i+1}^n r_{ik} \theta_k / r_{ii} \quad (3.25)$$

para $i=n, n-1, \dots, 1$.

3.3 GENERACION DE UNA SEÑAL PSEUDOALEATORIA

Como señal de entrada al simulador del sistema lineal discreto se escogió una secuencia binaria pseudoaleatoria. Dicha secuencia tiene las siguientes características:

- Solamente puede tomar 2 valores $\pm A$, y los cambios de un valor a otro solamente pueden ocurrir a intervalos constantes múltiplos de un periodo determinado t ($t = t, 2t, \dots$):
- La señal es determinística y periódica con periodo determinado Nt .
- La función de autocorrelación se muestra a continuación

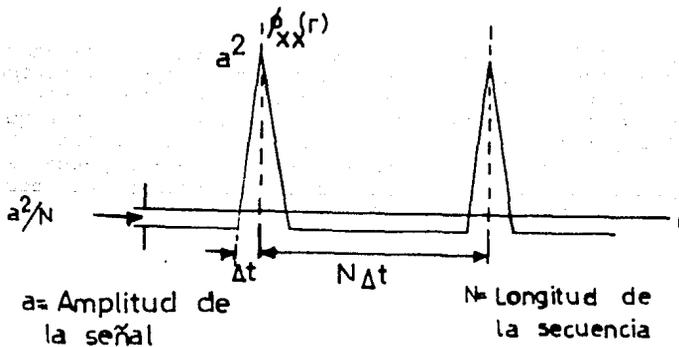


fig 3.1

De donde vemos que su función de autocorrelación se aproxima muy bien a la función autocorrelación del ruido blanco (Levin demostró que una secuencia de ruido blanco minimiza el determinante de la matriz de covarianza).

Una secuencia como la descrita anteriormente se puede generar mediante registros de corrimiento como se ilustra en la siguiente figura:

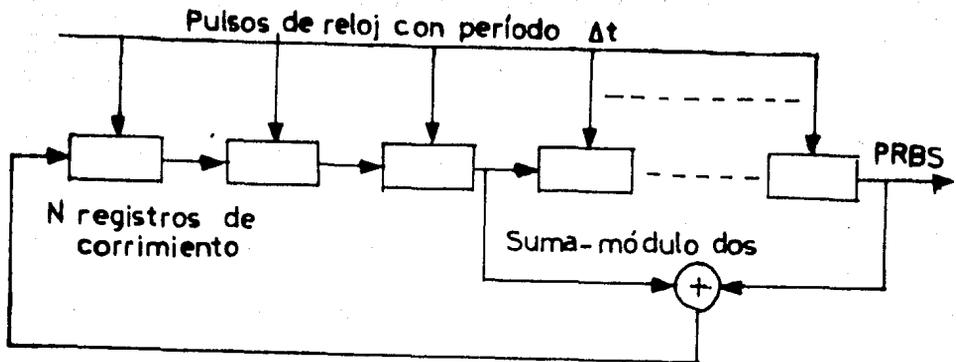


fig 3.2

La salida del Or exclusivo representa los dos estados de la secuencia pseudoaleatoria. Además es necesario escoger el periodo de la secuencia pseudoaleatoria $T(T=N\Delta t)$ lo suficientemente grande, de tal forma que $R_{uy}(w)$ es aproximadamente cero para valores de w que no son múltiplos del periodo, es decir que la entrada y la salida no estén correlacionadas. Computacionalmente las secuencias pseudoaleatorias tienen la ventaja de que podemos hacer coincidir el periodo de muestreo, k , de los datos entrada-salida con Δt .

3.3 GENERADOR DE UNA SECUENCIA ALEATORIA CON DISTRIBUCION UNIFORME

Para esto se utilizó un método congruencial lineal. La estructura de un generador de este tipo se muestra a continuación.

$$X_{n+1} = (aX_n + C) \text{ mod } m \quad (3.26)$$

en donde

X_0 es el valor inicial $X_0 \geq 0$
 a el múltiplo $a \geq 0$
 c el incremento $c \geq 0$
 m el módulo $m \geq X_0, m \geq a, m \geq c.$

Una secuencia generada mediante la ecuación (3.26) tiene la propiedad que se repite con período T; el período máximo es m. Para la elección de los coeficientes X_0, a, c, m se escogió una secuencia que pasara la prueba de análisis espectral, debido a que no solo nos proporciona buenos generadores de números aleatorios, sino que todas las secuencias que se consideran como malas o no aleatorias, no pasan esta prueba &. Los coeficientes que se escogieron fueron:

$a = 3141592621$

$m = 10^{10}$

$c = 1$

X_0 dato de entrada llamado semilla

Para generar esta secuencia se tuvo que generar la función módulo y hacer uso de variables de doble precisión.

& ver (21) pag 82.

3.4 SIMULACION DE UN SISTEMA LINEAL DISCRETO

Esta subrutina simula un sistema lineal discreto de una entrada-una salida, como el que se describe a continuación.

$$y(k) + \dots + a_n y(k-n) = b_0 u(k-r) + \dots + b_m u(k-m-r) + e(k) \quad (3.27)$$

en donde los datos que le proporcionamos son: $n, m, r, a_1, \dots, a_n, b_0, \dots, b_m$. Y $e(k)$ es el ruido que se le suma.

Para la generación de N datos se simulan $(N+127)$ datos con condiciones iniciales nulas y posteriormente se eliminan los 127 datos iniciales, para minimizar los efectos transitorios. Se escogió el número 127 debido a que es el período de la secuencia pseudoaleatoria.

3.5 GENERACION DE UNA SERIE DE TIEMPO CON DISTRIBUCION NORMAL

Quizas la distribución continua mas importante sea la distribución normal con media cero y desviación estandar igual a uno.

$$F(x) = 1 / \sqrt{2\pi} \int_{-\infty}^x e^{-t^2/2} dt \quad (3.28)$$

Para esto utilizaremos el método polar; el cual calcula dos variables aleatorias normales e independientes, dadas dos distribuciones U_1, U_2 uniformemente distribuidas

P1. Genere dos variables aleatorias U_1, U_2 , uniformemente distribuidas entre cero y uno.

Haga $V_1 = 2U_1 - 1$ y $V_2 = 2U_2 - 1$ (Ahora están uniformemente distribuidas entre -1 y 1).

1 1 2 2

P2. Haga $S = V_1^2 + V_2^2$

1 2

P3. Si $S > 1$, entonces regresamos al paso 1.

P4. $X_1 = V_1 \sqrt{(-2 \ln S / S)}$; $X_2 = V_2 \sqrt{(-2 \ln S / S)}$

1 1 2 2

Que son las variables aleatorias deseadas.

Prueba Si $S < 1$ en el paso 3, entonces el punto (V_1, V_2) (en coordenadas cartesianas) es un punto aleatorio uniformemente distribuido dentro del círculo unitario. Transformando a coordenadas polares $V_1 = R \cos(L)$, $V_2 = R \sin(L)$. Donde $S = R^2$. Definiendo las siguientes variables en coordenadas polares, $X_1 = \sqrt{-2 \ln S} \cos(L)$, $X_2 = \sqrt{-2 \ln S} \sin(L)$, $X_1 = R' \cos L'$ y $X_2 = R' \sin L'$ encontramos que $L' = L$ y $R' = \sqrt{-2 \ln S}$.

Es claro que R' y L' son independientes dentro del círculo unitario. L está uniformemente distribuido entre 0 y 2π ; y la probabilidad de que $R' = r$ es la probabilidad $-2 \ln S = r^2$. Por ejemplo la probabilidad de que $S = e^{-(r^2/2)}$ es igual a $1 - e^{-(r^2/2)}$ ya que $S = R^2$ está uniformemente distribuida entre cero y uno. La probabilidad de que R' esté entre r y $r+dr$ es por lo tanto la derivada de $1 - e^{-(r^2/2)}$ y que es igual a $[re^{-(r^2/2)}]dr$. Análogamente la probabilidad de que L' esté entre L y $L+dL$ es $(1/2\pi)dL$. Por lo que la probabilidad de que $X_1 = x_1$ y que $X_2 = x_2$ es:

$$\int_{L=0}^{2\pi} \int_{R=0}^1 \frac{1}{2\pi} e^{-(r^2/2)} r dr dL \quad (3.29)$$

$\{(R,L)/r \cos(L) = x_1, r \sin(L) = x_2\}$

$$= \frac{1}{2\pi} \int_{Y=0}^1 \int_{X=0}^1 e^{-(X^2+Y^2)/2} dx dy \quad (3.30)$$

$$= \left[\frac{1}{2\pi} \int_{-\infty}^{x_1} e^{-x^2/2} dx \right] \left[\frac{1}{2\pi} \int_{-\infty}^{x_2} e^{-y^2/2} dy \right] \quad (3.31)$$

Esto nos muestra que X_1 y X_2 son dos variables aleatorias independientes y normalmente distribuidas. &

¡ El lector interesado en otros métodos de generación de una serie de tiempo con distribución Normal; en la generación de otras distribuciones; puede consultar la referencia [2].

3.6 EJEMPLOS PRACTICOS

1.- Si un proceso esta descrito por el siguiente modelo:

$$y(k-1) - 0.5y(k) = u(k-1) + ee(k) + 0.8 ee(k-1)$$

(3.32)

donde $ee(k)$ es una secuencia de variables aleatorias independientes uniformemente distribuidas $(-0.5, 0.5)$. Si la amplitud de la secuencia PRBS = 0.7, identificar el sistema por medio de mínimos cuadrados y mínimos cuadrados generalizados y comparar resultados.

Los datos de entrada para $k=127 \dots 226$ se muestran en la figura (3,3) y los datos de salida en la figura (3,4)

a) Con Mínimos cuadrados normales (M.C.G la iteración)

$$a = -0.33097$$

1

$$b = 0.9763083$$

0

La correlación de residuos se indica en la figura(3,5) y de ahí vemos que los residuos estan correlacionados.

b) Con M.C.G. 2a iteración

$$a = -0.48455$$

1

$$b = 1.02189$$

0

La correlación de residuos se indica en la figura(3,6) y de ahí vemos que los residuos estan correlacionados.

c) Con M.C.G. 3a iteración

ESTA TESIS HA DEBE
SER DEBIDA

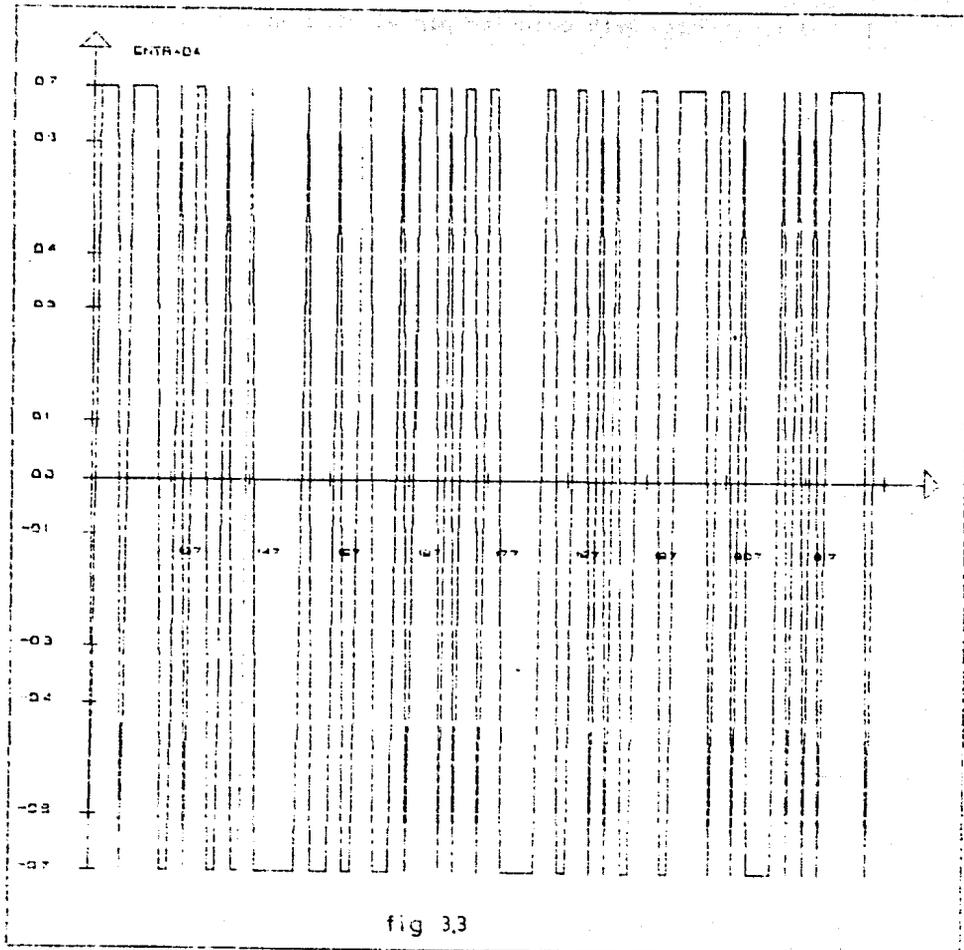


fig 3.3

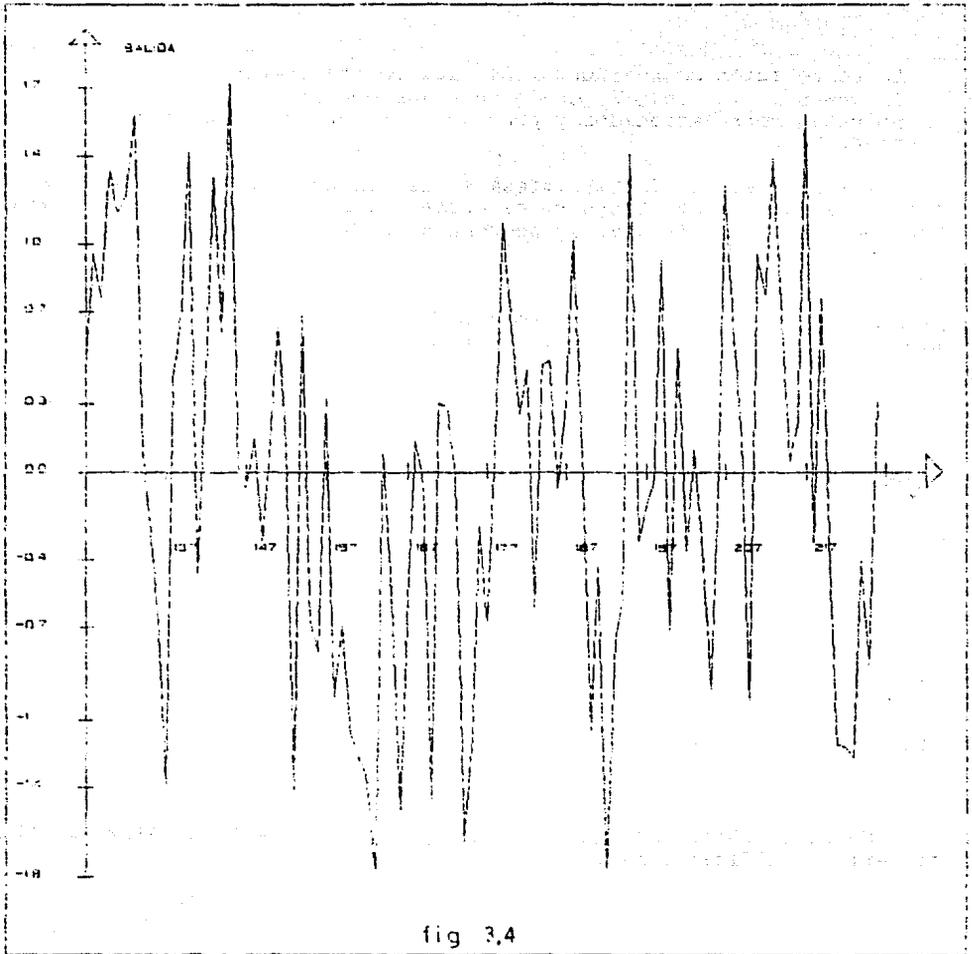


fig 3,4

a = -0.4937
1

b = 1.02064
0

La correlación de residuos para esta cuarta iteración la vemos en la fig(3.7), de donde vemos que los residuos no están correlacionados y por eso aquí detenemos nuestro algoritmo.

2.- Para el siguiente sistema lineal discreto graficar el error debido a las inexactitudes en el orden. Con ruido no correlacionado, 500 muestras y amplitud de la secuencia PRBS=5.

$$y(k)+2.25y(k-1)+2.75y(k-2)+2.09375y(k-3)+0.96875y(k-4)+0.234375y(k-5)=$$
$$u(k)-2.25u(k-1)+2.75u(k-2)-2.09375u(k-3)+0.96875u(k-4)-.234375u(k-5)$$

(3.33)

Resultados

Orden (n=a)	h
1	1.55963 E(4)
2	4.24922 E(3)
3	6.31418 E(2)
4	7.86514 E(1)
5	4.15369 E(1)
6	4.11673 E(1)
7	4.09775 E(1)

De donde observamos que para n mayor o igual a 5 el valor de h2 no varía significativamente.

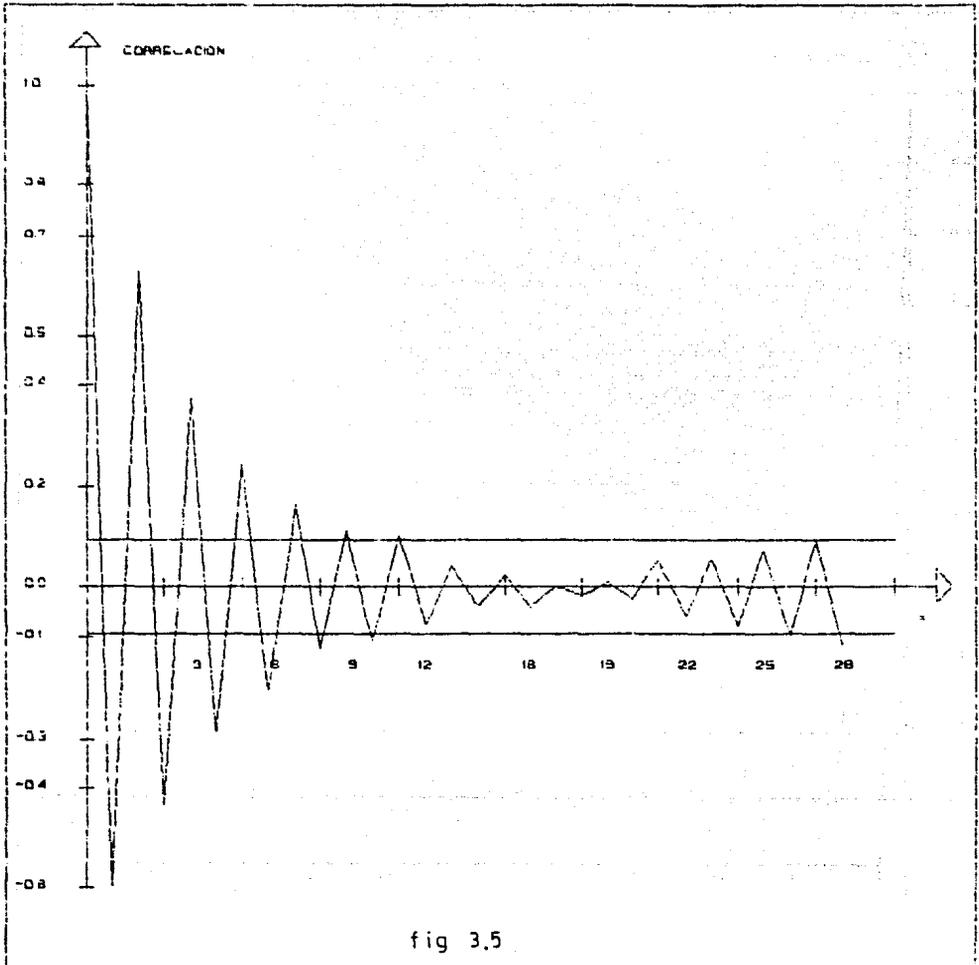
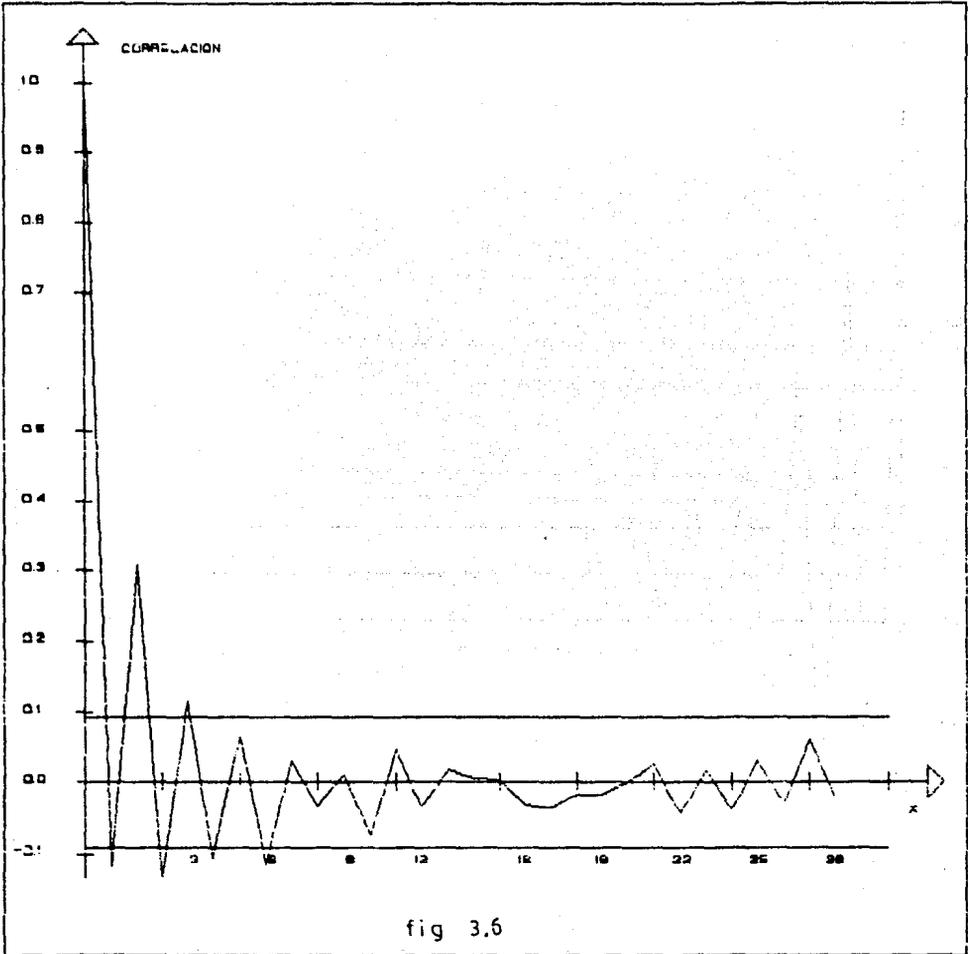
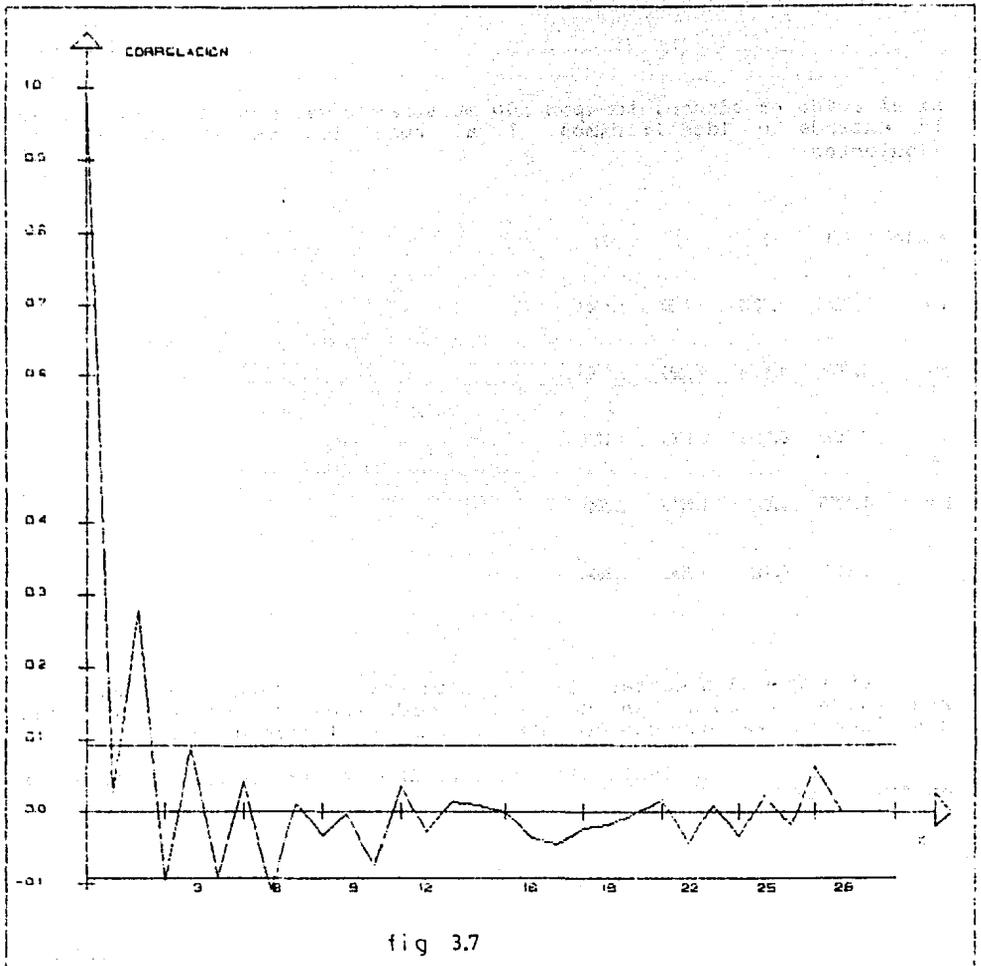


fig 3.5





3.- Para el siguiente sistema lineal discreto:

$$y(k)+1.69y(k-2)+0.938y(k-3)+0.1715= u(k)+0.3u(k-1)+ee(k)$$

(3.34)

si el ruido es blanco, tomamos 500 muestras y variamos la amplitud de la entrada e identificamos el sistema; los resultados son los siguientes:

AMPLITUD=	0.1	1	10	30
a = 1	1.7258	1.7239	1.7055	1.6904
a = 2	0.9846	0.9804	0.9589	0.9385
a = 3	0.1839	0.1836	0.1765	0.1716
b = 0	1.1259	1.0128	1.0013	1.0004
b = 1	0.4427	0.3445	0.3165	0.3008

Vemos que al aumentar la amplitud de la secuencia PRES los resultados se aproximan más a los verdaderos, algunos coeficientes convergen más rápidamente que otros, como es el caso de b₀.

4.- Para el siguiente sistema descrito por la siguiente ecuación en diferencias:

$$y(k)+0.8y(k-1)+0.07y(k-2)= u(k)+0.8u(k-1)+0.15u(k-2)+ee(k)$$

(3.35)

Si la amplitud de la secuencia pseudoaleatoria es de 3 y el ruido es blanco, variar el número de muestras para ver su influencia en el valor de los coeficientes.

MUESTRAS=	50	100	300	500	700
a = 1	0.6425	0.8053	0.8304	0.8453	0.8207
a = 2	0.17791	0.0227	0.1020	0.1074	0.0923
b = 0	0.9924	0.9829	1.0002	1.0039	1.0017
b = 1	0.8498	0.8140	0.8381	0.8467	0.8225
b = 2	0.2545	0.0920	0.18080	0.18720	0.1728

Vemos que el número de muestras debe ser al menos de 300 para obtener resultados confiables.

5. A continuación tendremos un ejemplo aeroespacial (ver [40]) en donde un control típico que tiene tres componentes. Un diagrama de dicho sistema se muestra a continuación:

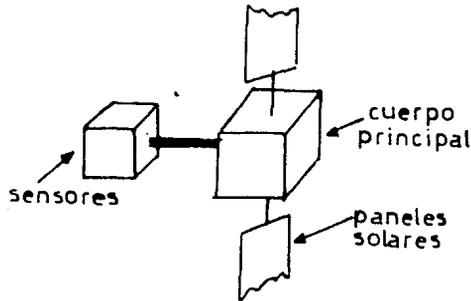


fig 3.8

La primer componente consta de aquellas partes cuyas ecuaciones son conocidas completamente. La segunda consta de partes con estructura conocida pero parámetros desconocidos, como la frecuencia de resonancia. La tercera parte del modelo no tiene dinámica conocida en estructura ni en complejidad. La formulación de éste modelo se puede resumir si asumimos que todo el modelo está descrito por:

$$G(s) = G_m(s, \theta) (1 + l(s)) \quad (3.36)$$

donde θ es un vector de parámetros desconocidos y $l(s)$ es desconocido, pero está acotado

$$|l(j\omega)| \leq |l_0(j\omega)| \quad (3.37)$$

En éste ejemplo tratamos de identificar

$$G_m(s, \theta) = \frac{C(s + w/2\Gamma)}{s^2 [s^2 + (2\Gamma w)s + w^2]} \quad (3.38)$$

si discretizamos por medio de un retén de orden cero y con una frecuencia de muestreo $T_m = 2\pi/10$ segundos y $C = 0.072$, $w = 2.0$, $\Gamma = 0.02$, entonces:

$$G_m(q^{-1}) =$$

$$10^{-3} (0.481 + 5.98q^{-1} + 3.137q^{-2} + 1.923q^{-3} + 4.593q^{-4} + 0.3202q^{-5})$$

$$1 - 3.148q^{-1} + 5.057q^{-2} - 6.259q^{-3} + 4.783q^{-4} - 2.475q^{-5} + 0.7428q^{-6}$$

(3.39)

Si a éste sistema lo excitamos con distintas entradas, almacenamos los datos de entrada y salida y luego identificamos el sistema (sin ruido), suponiendo que conocemos el orden del sistema a identificar, los resultados obtenidos (con 900 muestras) se dan a continuación:

1. Si la entrada es una señal senoidal con $T_s = 10 T_m$

A(1) = -3.1454	B(0) = 0.00481
A(2) = 5.3693	B(1) = 0.05985
A(3) = -6.3001	B(2) = 0.03161
A(4) = 4.8362	B(3) = 0.02057
A(5) = -2.5198	B(4) = 0.04709
A(6) = 0.7607	B(5) = 0.00329

2. Si la entrada es una señal cuadrada con $T_c = 12 T_m$

A(1) = -3.1589	B(0) = 0.00481
A(2) = 5.3868	B(1) = 0.05979
A(3) = -6.2970	B(2) = 0.03069
A(4) = 4.7835	B(3) = 0.01864
A(5) = -2.4752	B(4) = 0.04597
A(6) = 0.7441	B(5) = 0.00321

3. Si la entrada es una señal pseudoaleatoria de longitud 127.

A(1) = -3.1481	B(0) = 0.00481
A(2) = 5.3574	B(1) = 0.05984
A(3) = -6.2596	B(2) = 0.03136
A(4) = 4.7835	B(3) = 0.01922
A(5) = -2.4752	B(4) = 0.04593
A(6) = 0.7428	B(5) = 0.00320

De donde vemos que si conocemos exactamente el orden del sistema podemos identificar el sistema bastante bien.

Los parámetros de orden reducido obtenidos si la entrada al sistema es una señal senoidal ($T_s = 10T_m$) son los siguientes:

1. Orden 2

A(1)= -1.9806	B(0)= 0.04678
A(2)= 1.6473	B(1)= 0.03858
	B(2)= 0.02851

2. Orden 3

A(1)= -2.3204	B(0)= 0.03867
A(2)= 1.6473	B(1)= 0.02812
A(3)= -0.3265	B(2)= 0.04193
	B(3)= -0.04224

3. Orden 4

A(1)= -2.6022	B(0)= 0.00479
A(2)= 3.1556	B(1)= 0.06244
A(3)= -2.5035	B(2)= 0.06172
A(4)= 0.9509	B(3)= 0.00414
	B(4)= -0.00003

Y si la entrada es una señal PRBS los parámetros obtenidos son los siguientes:

1. Orden 2

A(1)= -1.8194	B(0)= -0.00088
A(2)= 0.8198	B(1)= 0.05537
	B(2)= 0.09697

2. Orden 3

A(1)= -2.1657	B(0)= -0.00608
A(2)= 1.5850	B(1)= 0.04974
A(3)= -0.4191	B(2)= 0.07285
	B(3)= -0.00785

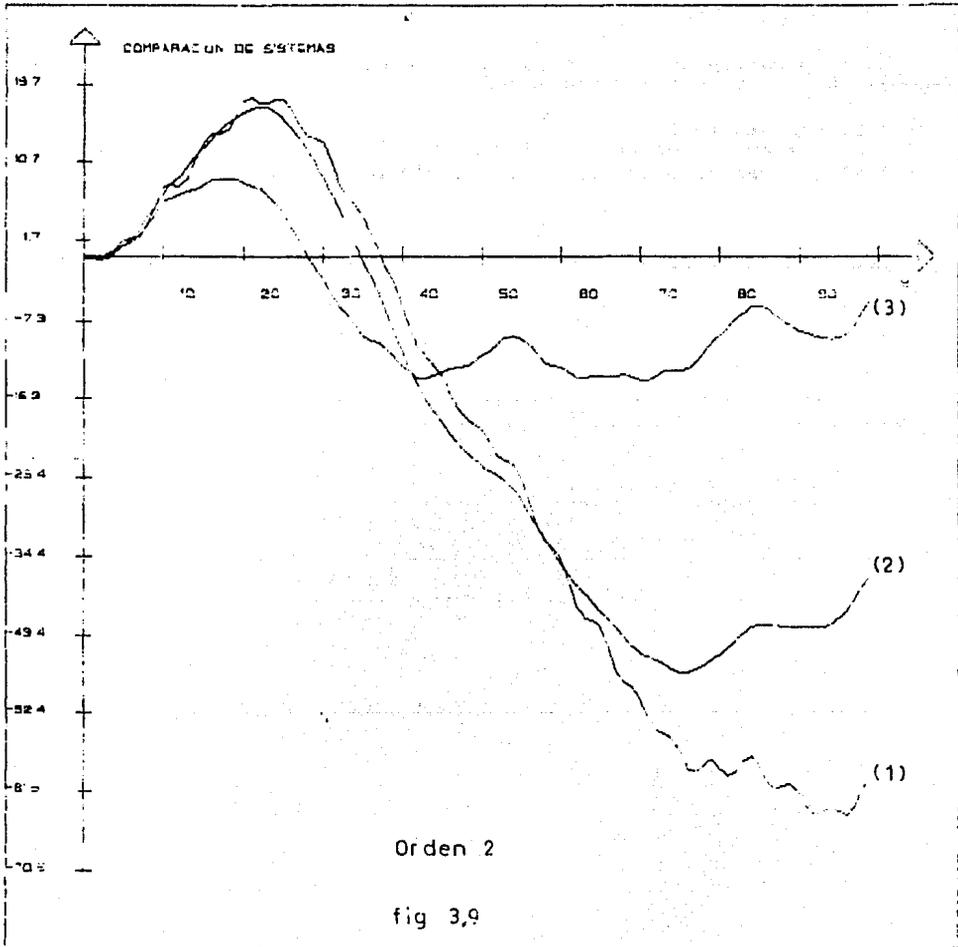
3. Orden 4

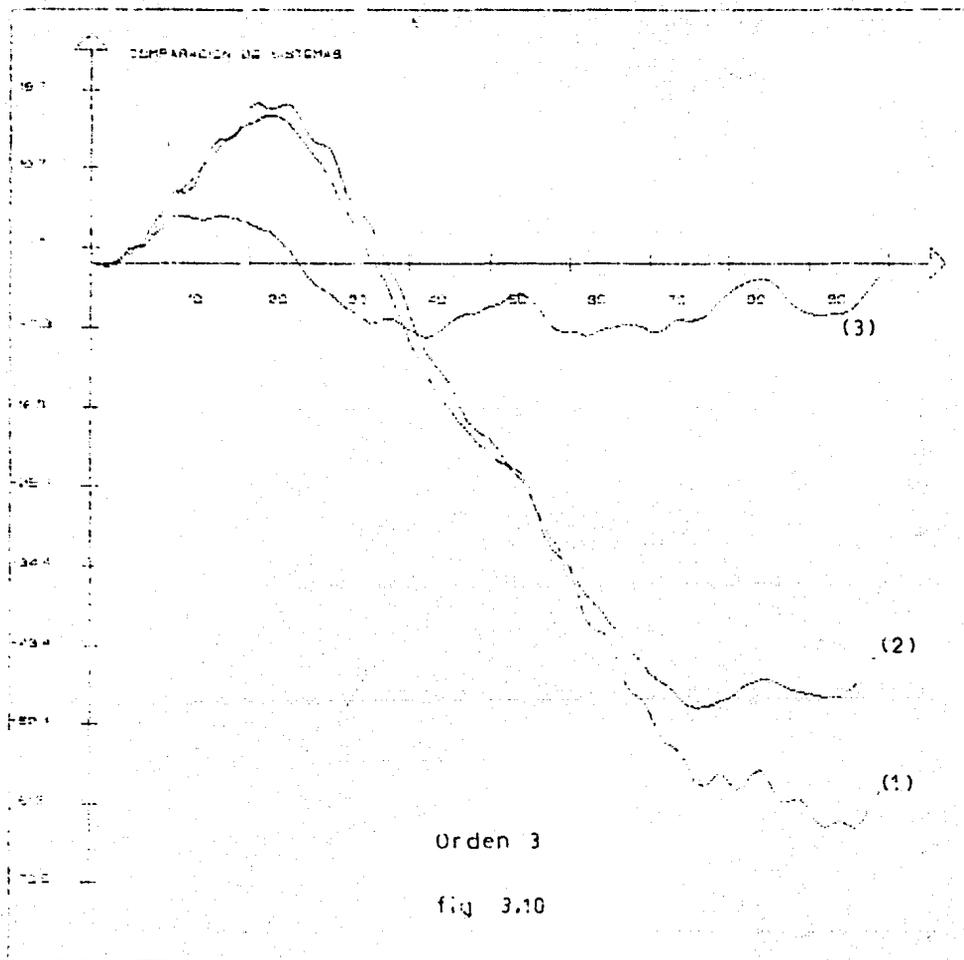
A(1)=	-2.6008	B(0)=	0.00481
A(2)=	3.1526	B(1)=	0.06247
A(3)=	-2.5019	B(2)=	0.06179
A(4)=	0.9507	B(3)=	0.00423
		B(4)=	-0.00002

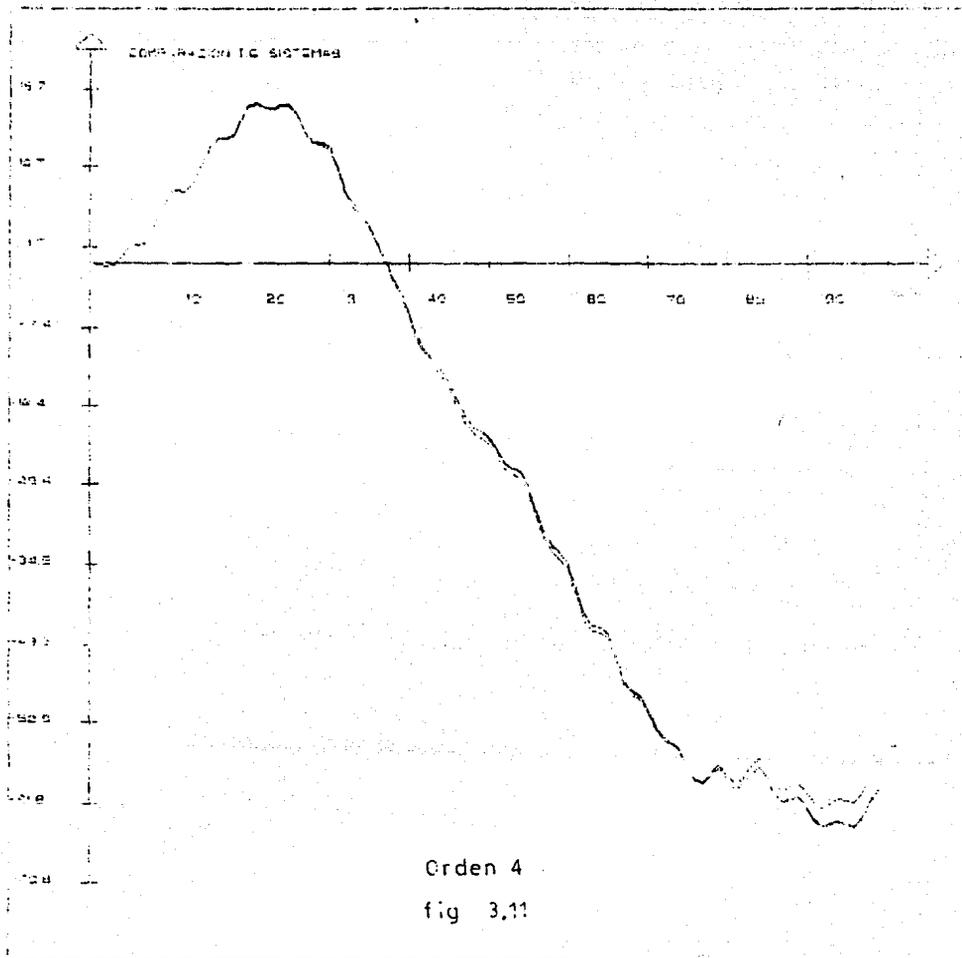
La comparación de éstos parámetros (cuando la entrada es una señal pseudoaleatoria) la hacemos graficando:

- (1) El sistema real.
- (2) El sistema de orden reducido cuando la entrada es senoidal.
- (3) El sistema de orden reducido cuando la entrada es PRBS.

Los resultados para órdenes 2,3 y 4 se muestran a continuación:







De las gráficas anteriores vemos que si el orden es 2 o 3 los coeficientes obtenidos cuando la entrada es senoidal son mejores a los obtenidos cuando la entrada es una señal PRBS, pero si el orden es 4 los coeficientes obtenidos mediante una señal PRBS son mejores.

Las gráficas (3.12) y (3.13) nos muestran como los sistemas de orden reducido se van pareciendo al sistema original. La gráfica (3.a.d) es obtenida de parámetros deducidos cuando el sistema tenía una onda senoidal y la (3.13) cuando la identificación fué de un sistema que tenía una entrada pseudoaleatoria.

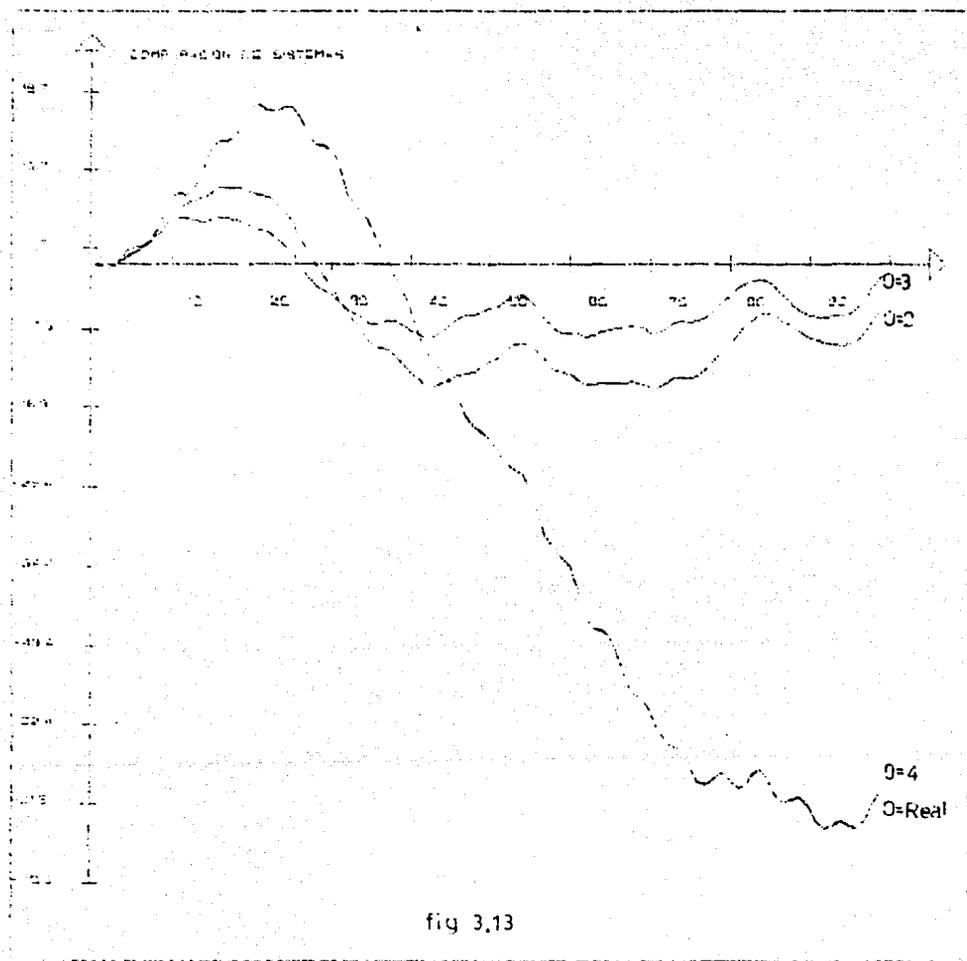


fig 3,13

APENDICE I

PROGRAMA PARA COMPUTADORA

+ Este programa fue hecho en una máquina
Vax en lenguaje Pascal versión V3.0-2

45:195:195-PL-SL-

```

00001 0 0 (INHERIT ('SYS:PAGRA:PAGIN.ERF.PEN'))
00002 0 0 PROGRAM PRUEBA (INPUT,OUTPUT,SAL,ENT,ISE,ENTI,DS1,DS2,DS3,RUIDO,
00003 0 0 FRBS1,PRBS ,PPBS1,RUB1,RUB2,RUF1,RUF2,RUF3,REST,
00004 0 0 CORRE1,CORRE2,CORRE3,SAL1,SAL2,SAL3,MEDIA,A,B,C);
00005 0 0 CONST RE=1100; CO=30;
00006 C 0 0 'RE' ES EL NUMERO DE REGIONES Y CO EL DE COLUMNAS EN UN
00007 C 0 0 ARREGLO BIDIMENSIONAL;
00008 0 0 TYPE CC= FILE OF REAL;
00009 0 0 VAR SI:BOOLEAN;
00010 0 0 SAL,ENT,ISE,ENTI,DS1,DS2,DS3,RUIDO,
00011 0 0 FRBS1,PRBS,CORRE3,RUB1,RUB2,RUF1,RUF2,RUF3,REST,
00012 0 0 CORRE1,CORRE2,CORRE3,SAL1,SAL2,SAL3,
00013 0 0 MEDIA,A,B,CC;
00014 0 0 I:INTEGER;
00015 0 0
00016 1 0 PROCEDURE COPIA_3 (VAR BR,R:CC);
00017 C 1 0 (ESTA SUBROUTINA COPIA LOS DATOS DE UN ARCHIVO 'X' EN UN ARCHIVO 'Y')
00018 1 1 BEGIN
00019 1 1 REWRITE(B); RESL(R);
00020 1 1 WHILE NOT EOF(R) DO
00021 1 2 BEGIN
00022 1 2 R:=BR;
00023 1 2 PUT(L); GET(R);
00024 1 1 END;
00025 1 1 CLOSE(R);
00026 0 0 END;
00027 0 0
00028 1 0 PROCEDURE PAGINA;
00029 C 1 0 (SUBROUTINA PARA CAMBIAR DE PAGINA EN LA PANTALLA)
00030 1 1 BEGIN
00031 1 1 PAGE(OUTPUT);
00032 0 0 END;
00033 0 0
00034 1 0 PROCEDURE SIMULACION;
00035 1 0 VAR SI:BOOLEAN;
00036 1 0 I:INTEGER;
00037 1 0
00038 2 0 PROCEDURE SIMULA;
00039 2 0 VAR I:INTEGER;
00040 2 0
00041 3 0 PROCEDURE DATOS SISTEMA;
00042 3 0 VAR I:INTEGER;
00043 3 0
00044 4 0 PROCEDURE ACTUALIZA DATOS SISTEMA (VAR A:CC);
00045 C 4 0 (SUBROUTINA QUE GUARDA EN UN ARCHIVO LOS DATOS DEL SISTEMA QUE
00046 C 4 0 RESEAMOS SIMULARE)
00047 4 0 VAR P,M,N,I: INTEGER;
00048 4 1 BEGIN
00049 4 1 REWRITE (A);
00050 4 1 PAGINA;
00051 4 1 WRITELN ('ESCRIBE EL ORDEN DEL DENOMINADOR');
00052 4 1 READLN;
00053 4 1 A:=N;PUT(A);
00054 4 1 WRITELN ('ESCRIBE EL ORDEN DEL NUMERADOR');
00055 4 1 READLN;

```

PRUEBA
01

Source Listing

3-Oct-1985 14:51:31 VAX/VMS: V3.0-2 178
3-Oct-1985 12:42:59 DMS:1PRJ1985.01711PRUEBA. AS:3 17

LINK-TO-PL-5L-

```
00050 4 1 A:=R; PUT(A);
00051 4 1 WRITELN (' ESCRIBE CUANTO VALE EL RETRASO');
00052 4 1 R:=R(R);
00053 4 1 A:=R; PUT(A);
00054 4 1 FOR I1:=1 TO M DO
00055 4 2 BEGIN
00056 4 2 WRITELN (' ESCRIBE EL VALOR DE A(' ,I1:2,') ');
00057 4 2 READ(A);
00058 4 2 PUT(A);
00059 4 1 END;
00060 4 1 FOR I1:=0 TO M DO
00061 4 2 BEGIN
00062 4 2 WRITELN (' ESCRIBE EL VALOR DE R(' ,I1:2,') ');
00063 4 2 READ(A);
00064 4 2 PUT(A);
00065 4 1 END;
00066 4 1 CLOSE(A);
00067 4 1 COPIA_3_ (A,R);
00068 3 0 END;
00069 3 0
00070 3 1 BEGIN
00071 3 1 PAGINA;
00072 3 1 WRITELN (' ESCRIBE EL NUMERO DEL ARCHIVO EN EL CUAL');
00073 3 1 WRITELN (' QUIERES GUARDAR TUS DATOS');
00074 3 1 WRITELN; WRITELN;
00075 3 1 WRITELN (' (1) ARCHIVO DS1');
00076 3 1 WRITELN (' (2) ARCHIVO DS2');
00077 3 1 WRITELN (' (3) ARCHIVO DS3');
00078 3 1 READ (I);
00079 3 2 CASE I OF
00080 3 2 1:ACTUALIZA DATOS SISTEMA(DS1);
00081 3 2 2: ACTUALIZA DATOS SISTEMA(DS2);
00082 3 2 3: ACTUALIZA DATOS SISTEMA(DS3);
00083 3 1 END;
00084 2 0 END;
00085 2 0
00086 3 0 PROCEDURE ACTUALIZA ARCHIVO SALIDA;
00087 3 0 VAR I: INTEGER;
00088 3 1 BEGIN
00089 3 1 PAGINA;
00090 3 1 WRITELN (' CUAL DE LOS SIGUIENTES ARCHIVOS LO QUIERES');
00091 3 1 WRITELN (' GUARDAR EN EL ARCHIVO SAL');
00092 3 1 WRITELN; WRITELN;
00093 3 1 WRITELN ('(1) SAL1');
00094 3 1 WRITELN ('(2) SAL2');
00095 3 1 WRITELN ('(3) SAL3');
00096 3 1 WRITELN; WRITELN;
00097 3 1 WRITELN ('OPCION #...');
00098 3 1 READ (I);
00099 3 1 IF I=OK (1-3) THEN I:=4;
00100 3 2 CASE I OF
00101 3 2 1: COPIA_1 (SAL1,SAL);
00102 3 2 2: COPIA_2 (SAL2,SAL);
00103 3 2 3: COPIA_3 (SAL3,SAL);
00104 3 2 4: WRITELN ('NO SE VALE');
```

LINE-IDC-PL-SL-

```

00111 3 1 END;
00112 3 0 END;
00113 3 0
00114 3 0 PROCEDURE SIMULACION SISTEMA DISCRETO; VAR SAL:CC;
00115 C 3 0 S:SERVICIO QUE SIMULA UN SISTEMA LINEAL DISCRETO CON RUIDO;
00116 3 0 TYPE M:=ARRAY[0..RE] OF REAL;
00117 3 0 VAR Y,U,A,C:M;
00118 3 0 K,DATOS,N,M,N1,J1:INTEGER;
00119 3 0 S1,S2:REAL;
00120 3 1 BEGIN
00121 3 1 RESET(R);
00122 3 1 M:=ROUND(R*);
00123 3 1 GET(R);
00124 3 1 M:=ROUND(R*);
00125 3 1 GET(R);
00126 3 1 K:=ROUND(R*);
00127 3 1 FOR J1:=1 TO N DO
00128 3 2 BEGIN
00129 3 2 GET(R);
00130 3 2 A[J1]:=R*);
00131 3 1 END;
00132 3 1 FOR I1:=0 TO n-1 DO
00133 3 2 BEGIN
00134 3 2 GET(R);
00135 3 2 U[I1]:=R*);
00136 3 1 END;
00137 3 1 WRITELN('ESCRIBIR EL NUMERO DE DATOS QUE QUIERAS ');
00138 3 1 READ(DATOS);
00139 3 1 DATOS:=DATOS+127;
00140 3 1 FWRITE(SAL); RESET(RUIDO); RESET(ENT);
00141 3 1 FOR J1:=1 TO n DO
00142 3 2 BEGIN
00143 3 2 S[J1]:=0;PUT(SAL);
00144 3 1 END;
00145 3 1 FOR J1:=1 TO n DO
00146 3 2 BEGIN
00147 3 2 Y[J1]:=0;
00148 3 1 END;
00149 3 1 UC[J1]:=0;
00150 3 1 FOR J1:=1 TO DATOS DO
00151 3 2 BEGIN
00152 3 2 I1:=0;
00153 3 2 S2:=ENT*ACC;
00154 3 2 UC[J1]:=ENT;
00155 3 2 FOR I1:=1 TO N DO
00156 3 3 BEGIN
00157 3 3 S1:=S1+UC[I1]*I1+ENT*I1+I1;
00158 3 3 EN:=I1+I1:=I1-I1;
00159 3 2 END;
00160 3 2 F: I1:=1 TO n DO
00161 3 3 BEGIN
00162 3 3 S2:=S2+C*(M-I1)+I1*(M-I1+I1);
00163 3 3 UC(M-I1+I1):=UC(M-I1);
00164 3 2 END;
00165 3 2 S1:=S1;S2:=PUT(R*);

```

*****SOURCE*****

```

00166      2      SAL2= 1.1;
00167      2      PUT(SAL);
00168      3      GET=KODUD;
00169      3      END;
00170      3      FOR I=1 TO SAL2;
00171      3      +ESQ(I);
00172      3      FOR II=1 TO 127 DO
00173      3      +GET=9;
00174      3      +WRITE(9);
00175      3      WHILE NOT GET=9 DO
00176      3      +END;
00177      3      SAL2= I;
00178      3      PUT(SAL);GET=9;
00179      3      END;
00180      3      CLOSE(SAL);
00181      2      END;
00182      2      BEGIN
00183      2      + PAGINA;
00184      2      + WRITELN ( QUIERES EMPLEAR LOS COEFICIENTES DEL SISTEMA );
00185      2      + WRITELN ( ALMACENAR EN EL ARCHIVO? );
00186      2      + WRITELN;WRITELN;
00187      2      + WRITELN ( 1)   DS1 );
00188      2      + WRITELN ( 2)   DS2 );
00189      2      + WRITELN ( 3)   DS3 );
00190      2      + WRITELN ( 4)   PREFERIR EMPLEAR OTROS COEFICIENTES );
00191      2      + WRITELN ( 5)   QUIERES ACTUALIZAR EL ARCHIVO SAL );
00192      2      + WRITELN;WRITELN;
00193      2      + WRITELN ( SELECCIONA EL NUMERO DE LA OPCION QUE QUIERAS... );
00194      2      + READ(I);
00195      2      + CASE I OF
00196      2      + 1: COPIA 1 (DS1);
00197      2      + 2: COPIA 2 (DS2);
00198      2      + 3: COPIA 3 (DS3);
00199      2      + 4: DATOS SISTEMA;
00200      2      + 5: ACTUALIZA ARCHIVO SALIDA;
00201      2      + END;
00202      2      + IF I=5 THEN
00203      2      + BEGIN
00204      2      + WRITELN ( EN QUE ARCHIVO QUIERES GUARDAR TUS DATOS? );
00205      2      + WRITELN; WRITELN;
00206      2      + WRITELN ( (1) SAL1 );
00207      2      + WRITELN ( (2) SAL2 );
00208      2      + WRITELN ( (3) SAL3 );
00209      2      + WRITELN; WRITELN;
00210      2      + WRITELN ( OPCION ... );
00211      2      + READ (I);
00212      2      + IF (I=1 OR I=3) THEN I:=4;
00213      2      + CASE I OF
00214      2      + 1: SIMULACION SISTEMA DISCRETO (SAL1);
00215      2      + 2: SIMULACION SISTEMA DISCRETO (SAL2);
00216      2      + 3: SIMULACION SISTEMA DISCRETO (SAL3);
00217      2      + 4: WRITELN ( NO SE VALE );
00218      2      + END;
00219      2      + END;
00220      1      END;

```

-LINF:06-PL-5L-

```

00221 1 0
00222 2 0 PROCEDURE PSEUDOALEATORIO;
00223 C 2 0 ( SUBROUTINA QUE PONE EN EL ARCHIVO 'ENT' UNA SECUENCIA
00224 C 2 0 PSEUDOALEATORIA DE AMPLITUD VARIABLE);
00225 2 0 VAR I: INTEGER;
00226 2 0
00227 5 0 PROCEDURE ACTUALIZA_ARCHIVO_ENTRADA;
00228 3 0 VAR I: INTEGER;
00229 2 1 BEGIN
00230 3 1 PAGINA;
00231 3 1 WRITELN (' CUAL DE LOS SIGUIENTES ARCHIVOS LO QUIERES?');
00232 3 1 WRITELN (' GUARDAR EN EL ARCHIVO ENT');
00233 3 1 WRITELN; WRITELN;
00234 3 1 WRITELN ('(1) PRBS1');
00235 3 1 WRITELN ('(2) PRBS2');
00236 3 1 WRITELN ('(3) PRBS3');
00237 3 1 WRITELN; WRITELN;
00238 3 1 WRITELN (' Opcion 0...');
00239 3 1 READ (I);
00240 3 1 IF (I<1) OR (I>3) THEN I:=4;
00241 3 2 CASE I OF
00242 3 2 1: COPIA (' PRBS1,ENT);
00243 3 2 2: COPIA (' PRBS2,ENT);
00244 3 2 3: COPIA (' PRBS3,ENT);
00245 3 2 4: WRITELN (' NO SE VALE');
00246 3 1 END;
00247 2 0 END;
00248 2 0
00249 3 0 PROCEDURE GENEPA_SECUENCIA(VAR PRBS:CC);
00250 C 3 0 (AQUI SE GENEPA LA SECUENCIA PSEUDOALEATORIA)
00251 3 0 VAR REG:ARRAY[0..700] OF INTEGER;
00252 3 0 I,J,L:INTEGER;
00253 3 0 G: REAL;
00254 3 1 BEGIN
00255 3 1 REWRITE(PRBS);
00256 3 1 WRITELN (' ESCRIBE LA AMPLITUD DE LA SECUENCIA PSEUDOALEATORIA');
00257 3 1 READ(G);
00258 3 1 WRITELN (' CUANTAS MUESTRAS?');
00259 3 1 READ(L);
00260 3 1 FOR I:=1 TO 3 DO REG[I]:=0;
00261 3 1 FOR I:=4 TO 7 DO REG[I]:=1;
00262 3 1 FOR I:=1 TO L DO
00263 3 2 BEGIN
00264 3 2 REG[0]:=(REG[7]+REG[4]) MOD 2;
00265 3 2 FOR J:=7 DOWNTO 1 DO REG[J]:=REG[J-1]+J;
00266 3 2 PRBS:=CHR(REG[0]+1)*G;
00267 3 2 PUT(PRBS);
00268 3 1 END;
00269 3 1 CLOSE(PRBS);
00270 2 0 END;
00271 2 1 BEGIN
00272 2 1 PAGINA;
00273 2 1 WRITELN (' EN QUE ARCHIVO QUIERES GUARDAR TUS DATOS?');
00274 2 1 WRITELN; WRITELN;
00275 2 1 WRITELN ('(1) PRBS1');

```


PRUEBA
01

Source Listing

3-uct-1985 14:51:31 04: Pascal 03.0-2 1299
3-uct-1985 13:42:59 03:1 (PRO)1955.03: 10000000.F45:3 (1)

-LINE-LOC-PL-SL-

```
00331 4 0
00332 5 0 PROCEDURE RAIDOM(VAR X,Y: DOUBLE);
00333 C 5 0 (SUBROUTINA QUE GENERA NUMEROS ALEATORIOS UNIFORMEMENTE
00334 C 5 0 DISTRIBUIDOS ENTRE CERO Y UNO)
00335 5 0 VAR A,C,M: DOUBLE;
00336 5 0
00337 6 0 PROCEDURE MODULO(VAR D:DOUBLE);
00338 E 6 0 (SUBROUTINA QUE GENERA LA FUNCION MODULO)
00339 6 0 VAR I,COM: INTEGER;
00340 6 0 DO: REAL;
00341 6 0
00342 7 0 PROCEDURE RECORRE(VAR B:DOUBLE; VAR COM:INTEGER);
00343 7 0 VAR A: DOUBLE;
00344 7 1 BEGIN
00345 7 1 A:=10; COM:= 0;
00346 7 1 WHILE B>A DO
00347 7 2 BEGIN
00348 7 2 COM:= COM+1;
00349 7 2 B:= B/10;
00350 7 1 END;
00351 6 0 END;
00352 6 1 BEGIN
00353 6 1 RECORRE(D.COM);
00354 6 1 I:=4;
00355 6 1 WHILE COM I DO
00356 6 2 BEGIN
00357 6 2 D:= DA(10*A*I);
00358 6 2 DO:= TRUNC(D);
00359 6 2 D:= D-DO;
00360 6 2 COM:= COM-I;
00361 6 1 END;
00362 6 1 I:=1;
00363 6 1 WHILE COM 0 DO
00364 6 2 BEGIN
00365 6 2 D:= DA(10*A*I);
00366 6 2 DO:= TRUNC(D);
00367 6 2 D:= D-DO;
00368 6 2 COM:= COM-1;
00369 6 1 END;
00370 5 0 END;
00371 5 1 BEGIN
00372 5 1 A:= 10000; A:= AAAA30; A:=A+ 141592621;
00373 5 1 C:=1; M:= 10000; M:= MAHA100;
00374 5 1 Y:=(AAX+C)/M;
00375 5 1 MODULO(Y);
00376 5 1 A:= YAM;
00377 4 0 END;
00378 4 1 BEGIN
00379 4 1 WRITELN('ESCRIBE SEMILLA Y NUMERO DE RESULTADOS ');
00380 4 1 READ(S);READ(N);
00381 4 1 REWRITE(ERUIBO);
00382 4 1 IF A=2 THEN MU:=0 ELSE
00383 4 2 BEGIN
00384 4 2 WRITELN('QUE MEDIA ?');
00385 4 2 READ(MU);
```

LINE-DE PL-SL-

```

00386 4 1 END;
00387 4 1 IF(A<=1/A*(A-2)) THEN Writeln('NO SE VALE A');
00388 4 1 ELSE
00389 4 2 BEGIN
00390 4 3 I:= 0;
00391 4 2 WHILE I<N DO
00392 4 3 BEGIN
00393 4 3 FOR J:=1 TO A DO
00394 4 4 BEGIN
00395 4 4 RANDOM (S,Y); R:=Y;
00396 4 4 IF J=1 THEN X:=R;
00397 4 4 IF A=1 THEN R:=R-0.5*NU;
00398 4 4 I:=I+1
00399 4 3 END;
00400 4 3 IF A<= THEN
00401 4 4 BEGIN
00402 4 4 Z:=SORT(-2*LN(X))ASIN(2*PI*A);
00403 4 4 W:=SORT(-2*LN(X))ASIN(2*PI*A);
00404 4 4 R:=Z;
00405 4 3 END;
00406 4 3 FOR J:=1 TO A DO
00407 4 4 BEGIN
00408 4 4 RRUIDO:= SNG(R);
00409 4 4 IF (IGN)OR(J=1) THEN PUT(RRUIDO);
00410 4 4 IF --=2 THEN R:=W
00411 4 3 END;
00412 4 2 END;
00413 4 1 END;
00414 4 1 CLOSE(RRUIDO);
00415 3 0 END;
00416 3 1 BEGIN
00417 3 1 PAGINA;
00418 3 1 Writeln ('(1) DISTRIBUCION UNIFORME');
00419 3 1 Writeln ('(2) DISTRIBUCION NORMAL');
00420 3 1 Writeln (' ESCRIBE EL NUMERO DE LA OPCION QUE QUIERAS');
00421 3 1 READ(J);
00422 3 1 Writeln (' ESCRIBE EN CUAL DE LOS SIGUIENTES ARCHIVOS');
00423 3 1 Writeln (' QUIERES GUARDAR TUS DATOS');
00424 3 1 Writeln (' WRITELN;
00425 3 1 Writeln ('(1) RUB1');
00426 3 1 Writeln ('(2) RUB2');
00427 3 1 READ (I);
00428 3 1 IF (I<1) OR (I>2) THEN I:=3;
00429 3 2 CASE I OF
00430 3 2 1: ALEATORIO (RUB1,J);
00431 3 2 2: ALEATORIO (RUB2,J);
00432 3 2 3: Writeln ('NO SE VALE');
00433 3 1 END;
00434 2 0 END;
00435 2 0
00436 3 0 PROCEDURE RUIDO_FILTRADO;
00437 C 3 0 (SUBROUTINA QUE GENERA RUIDO DE COLOR A PARTIR DEL RUIDO BLANCO
00438 C 3 0 ALMACENADO EN LOS ARCHIVOS RUB1 O RUB2.)
00439 3 0 VAR I,J: INTEGER;
00440 3 0

```

PROBLEMA
01

Source Listing

7-Oct-1985 14:51:31

04 - Pascal - 4.0 -

9-Oct-1985 14:43:59

001116K071955.21111TROJER 433 (1)

-LINE-770-FL-56-

```
00440 4 0 PROCEDURE F DE FILTRADO_1 VAR B8:CC;
00441 4 0 VAR DATO,K,P: INTEGER;
00441 4 0 SI: REAL;
00444 4 0 C,E: ARRAY [0..3] OF REAL;
00445 4 1 BEGIN
00446 4 1 KESULT:=0;WRITE(B8);
00447 4 1 PAGINA;
00448 4 1 WRITELN ( ' dame el numero de DATOS QUE QUIERAS GUARDAR :');
00449 4 1 READ (DAT 3);
00450 4 1 WRITELN ( ' ESCRIBE EL NUMERO DE COEFICIENTES DEL FILTRO :');
00451 4 1 READ (P);
00452 4 1 FOR K:=1 TO P DO
00453 4 2 BEGIN
00454 4 2 WRITELN ( ' EL VALOR DE C[K],K:3, ES: ');
00455 4 2 READ (C[K]);
00456 4 1 END;
00457 4 1 FOR K:=1 TO P DO
00458 4 1 E[K]:=0;
00459 4 1 FOR K:=1 TO DATOS DO
00460 4 2 BEGIN
00461 4 2 SI:=0;
00462 4 2 FOR I:=1 TO P DO
00463 4 3 BEGIN
00464 4 3 SI:= 1+EXP(-I)*AECP-I+1;
00465 4 3 E[(I+1)]:=ECP-I;
00466 4 2 END;
00467 4 2 E[I]:=SI*B;
00468 4 2 B:=E[I];
00469 4 2 PUT(KB);GET(B);
00470 4 1 END;
00471 4 1 CLOSE (B);
00472 3 0 END;
00473 3 0
00474 3 1 BEGIN
00475 3 1 PAGINA;
00476 3 1 WRITELN ( ' ESCRIBE CUAL DE LOS SIGUIENTES ARCHIVOS QUIERES FILTRAR :');
00477 3 1 WRITELN ( '(1) RUB1');
00478 3 1 WRITELN ( '(2) RUB2');
00479 3 1 WRITELN ( '(3) RUIDO');
00480 3 1 READ (I);
00481 3 1 IF (I=1) OR (I=2) THEN I:=4;
00482 3 2 CASE I OF
00483 3 2 1: COPIA_1 (RUB1,B);
00484 3 2 2: COPIA_1 (RUB2,B);
00485 3 2 3: COPIA_1 (RUIDO,B);
00486 3 2 4: WRITELN ('NO SE VALE');
00487 3 1 END;
00488 3 1 WRITELN:WRITELN;
00489 3 1 PAGINA;
00490 3 1 WRITELN ( ' EN CUAL DE LOS SIGUIENTES ARCHIVOS QUIERES :');
00491 3 1 WRITELN ( ' GUARDAR EL RUIDO FILTRADO:');
00492 3 1 WRITELN:WRITELN;
00493 3 1 WRITELN ( '(1) RUB1');
00494 3 1 WRITELN ( '(2) RUB2');
00495 3 1 WRITELN ( '(3) RUIDO');
```


-LINE- I/O-PL-SL-

```

00551 1 0 T:=E RDA= A*RAY (I..RE..I, 0) OF DOUBLE;
00552 1 0 UDA= A RAY (I..RE) OF DOUBLE;
00553 1 0 VAR SI:POO DAN;
00554 1 0 II:INT GER;
00555 1 0
00556 2 0 PFCENUSE TE ANGULARIZACI,M: INTEGER;VAR A:RDA; VAR I..X:UDA;
00557 C 2 0 SUBROUTINA DE TRIANGULACI M LA PARTE SUPERIOR DE UNA MATRIZ
00558 C 2 0 'A', EMPLEANDO EL METODO DE HOUSEHOLDER. PARA EMPLEAR ESTA
00559 C 2 0 SUBROUTINA EL NUMERO DE REFLEXIONES DEBE SER MAYOR O IGUAL QUE
00560 C 2 0 EL NUMERO DE COLUMNAS. ESTA SUBROUTINA TAMBIEN OBTIENE LOS
00561 C 2 0 COEFICIENTES DEL VECTOR 'Y', DE TAL FORMA QUE LOS RESIDUOS
00562 C 2 0 'R=Y-AX' SEAN MINIMOS SGUIENDO EL CRITERIO DEL ERROR
00563 C 2 0 CUADRATICO;
00564 2 0 VAR SIGMA,I..S,SUM,SUMI,BETA,RES:DOUBLE;
00565 2 0 J,I,K:INTEGER;
00566 2 0 D:UDA;
00567 2 1 BEGIN
00568 2 1 FOR J:=1 TO M DO
00569 2 2 BEGIN
00570 2 2 SIGMA:=0;
00571 2 2 FOR I:=1 TO M DO
00572 2 2 SIGMA:=SIGMA+SQRT(A(I,J));
00573 2 2 IF SIGMA=0 THEN WRITELN ('MATRIZ SINGULAR');
00574 C 2 2 (SI SIGMA ES IGUAL A CERO ESTO SIGNIFICA QUE NO TODAS LAS
00575 C 2 2 COLUMNAS DE MI ARREGLO BIDIMENSIONAL SON LINEALMENTE
00576 C 2 2 INDEPENDIENTES);
00577 2 2 IF A(J, I) THEN D(J):=SORT(SIGMA) ELSE
00578 2 2 D(J):=-RT(SIGMA);
00579 2 2 S:=D(J);
00580 2 2 BETA:=-(S*A(I,J)-SIGMA);
00581 2 2 A(I,J):=A(I,J)+S;
00582 2 2 FOR K:=J+1 TO N DO
00583 2 3 BEGIN
00584 2 3 SUM:=0;
00585 2 3 FOR I:=J TO M DO
00586 2 3 SUM:=SUM+A(I,J)*A(I,K);
00587 2 3 SUM:=BETA+SUM;
00588 2 3 FOR I:=J TO M DO
00589 2 3 A(I,K):=A(I,K)+A(I,J)*SUM;
00590 2 2 END;
00591 2 2 SUMI:=0;
00592 2 2 FOR I:=1 TO M DO
00593 2 2 SUMI:=SUMI+A(I,J)*A(I,I);
00594 2 2 SUMI:=BETA+SUMI;
00595 2 2 FOR I:=J TO M DO
00596 2 2 Y(I):=Y(I)+A(I,J)*SUMI;
00597 2 1 END;
00598 2 1 FOR I:=1 TO M DO
00599 2 1 A(I,I):=D(I);
00600 2 1 FOR I:=N DOWN TO 1 DO
00601 2 2 BEGIN
00602 2 2 YI:=0;
00603 2 2 FOR J:=I+1 TO M DO
00604 2 2 YI:=YI+A(I,J)*Y(J);
00605 2 2 X(I):=(Y(I)-YI)/A(I,I);

```

FILE:PROE.PL:01

```

00606      2 1  FMD;
00607      2 1  RES:= 0;
00608      2 1  FOR I:=N+1 TO N DO
00609      2 1  RES:= RES+(I*IA);
00610      2 1  WRITELN ( CL VALOR DEL RESULTADO ES: ',RES)
00611      1 0 END;
00612      1 0
00613      2 0 PROCEDIMIENTO A: ELEGIR;
00614      C 2 0 SUBROUTINA QUE PONE LAS ENTRADAS Y LAS SALIDAS EN UNA FORMA
00615      C 2 0 ORDENADA DE TAL FORMA QUE PODAMOS OBTENER UNOS ESTIMADOS
00616      L 2 0 DE LOS PARAMETROS DEL SISTEMA EN ESTUDIO)
00617      2 0     VAR N,M,K,M,J,I: INTEGER;
00618      2 0     W1: DOUBLE;
00619      2 0     A: BD;
00620      2 0     I,J: IMA;
00621      2 1 BEGIN
00622      2 1     A:=A;
00623      2 1     WRITELN ( DAME EL VALOR DEL DENOMINADOR);
00624      2 1     READLN;
00625      2 1     WRITELN ( DAME EL VALOR DEL NUMERADOR);
00626      2 1     READLN;
00627      2 1     WRITELN ( DAME EL VALOR DEL RETRAZO);
00628      2 1     READLN;
00629      2 1     WRITELN ( ESCRIBIR EL NUMERO DE REPLICACIONES);
00630      2 1     READLN;
00631      2 1     RESET(SAL);
00632      2 1     FOR J:=1 TO M*N DO
00633      2 1     GET(SAL);
00634      2 1     FOR I:=N DOWNTO 1 DO
00635      2 2     BEGIN
00636      2 2         AC1(J):=DLE(SAL);
00637      2 2         GET(SAL);
00638      2 1     END;
00639      2 1     FOR I:=0 TO N DO
00640      2 2     BEGIN
00641      2 2         AC1(J):=DLE(SAL);
00642      2 2         GET(SAL);
00643      2 2         AC1(J):=DLE(SAL);
00644      2 2         FOR J:=2 TO N DO
00645      2 3             AC1(J):=AC1-I,J-1);
00646      2 1     END;
00647      2 1     (N1):=DLE(SAL);
00648      2 1     RESET(ENT);
00649      2 1     FOR J:=1 TO M*N DO
00650      2 1     GET(ENT);
00651      2 1     FOR I:=N DOWNTO N+1 DO
00652      2 2     BEGIN
00653      2 2         AC1(J):=DLE(ENT);
00654      2 2         GET(ENT);
00655      2 1     END;
00656      2 1     FOR I:=2 TO N DO
00657      2 2     BEGIN
00658      2 2         AC1,N+1:=DLE(ENT);
00659      2 2         GET(ENT);
00660      2 2         FOR J:=N+2 TO M*N+1 DO

```

LINE# COL# LINE#

```

00661 2 1  R(I,J) = I*(J+1);
00667 1 1  END;
00668 2 1  TR(ANGULA) = AN*(M+1,N),A,(1);
00669 1 1  WRITELN;
00670 2 1  FOR I:=1 TO N DO
00671 1 1  WRITELN (A(I,1), ' = ', C(I));
00672 1 1  FOR I:=1 TO M DO
00673 2 1  WRITELN (A(I,1), I, ' = ', C(I)*N+1);
00674 1 1  -LWRITE(0);
00675 1 1  USE(0) = 0;
00676 1 1  USE(0) = 0;
00677 1 1  USE(0) = 0;
00678 1 1  USE(0) = 0;
00679 2 1  FOR I:=1 TO N DO
00680 1 2  BEGIN
00681 2 2  W(I) = 0;
00682 1 2  DSA(I) = 0; PUT(DSA);
00683 1 1  END;
00684 1 1  LLOSE(DSA);
00685 1 0  END;
00686 1 0
00687 1 0 PROCEDURE RESIDUOS;
00688 C 1 0 (SUBROUTINA DE OBTENER LOS RESIDUOS "res = Y - X" DE TAL
00689 C 2 0 MANNER DE QUE PODAMOS HACER PRUEBAS ESTADISTICAS N
00690 C 2 0 "SE" AK FILTROS CON DIFER RESIDUOS)
00691 2 0 TYPE: REAL, REAL, REAL OF REAL;
00692 1 0 VAR: Y,U,A,C,M,B;
00693 2 0 RESIDUOS(I,N,M,K,I1,I2) INTEGER;
00694 1 0 SLS REAL;
00695 2 0
00696 1 1 BEGIN
00697 1 1 RESET(DSA);
00698 1 1 W := FOUND(DSA);
00699 1 1 W := 0;
00700 1 1 W := 0;
00701 1 1 W := 0;
00702 1 1 W := 0;
00703 1 1 W := 0;
00704 1 1 W := 0;
00705 1 1 W := 0;
00706 1 1 W := 0;
00707 1 1 W := 0;
00708 1 1 W := 0;
00709 1 1 W := 0;
00710 1 1 W := 0;
00711 1 1 W := 0;
00712 1 1 W := 0;
00713 1 1 W := 0;
00714 1 1 W := 0;
00715 2 2 BEGIN

```


-LINE-IDC-PL SL-

```

00771 3 1 FOR I:=1 TO P DO
00772 3 1   A(I):=0;
00773 3 1   RESET(R); REWRITE(AA);
00774 3 1   WHILE NOT EOF (R) DO
00775 3 1     BEGIN
00776 3 1       S2:=R;
00777 3 2     FOR I:=1 TO P DO
00778 3 3       BEGIN
00779 3 3         M1:=MCP-I+1;
00780 3 3         S2:=S2+SMGL(M1) A MCP-I+1;
00781 3 1       MCP-I+1:=MCP-I;
00782 3 2     END;
00783 3 2     M1:=P;GET(R);
00784 3 2     AA:=S+P;PUT(AA);
00785 3 1   END;
00786 3 1   (LOSE AA);
00787 3 0 END;
00788 3 0
00789 2 1 BEGIN
00790 3 1   NI:=0;
00791 3 1   F:=NA;
00792 3 1   WRITELN (' ESCRIBE EL ORDEN DEL FILTRO');
00793 3 1   READ(P);
00794 3 1   IF NI=0 THEN
00795 3 2     BEGIN
00796 3 2       WRITELN (' ESCRIBE EL NUMERO DE RENGLONES');
00797 3 2       EN(M);
00798 3 1     END;
00799 3 1     FE E (RES);
00800 3 1     FOR J:=P DOWNTO 1 DO
00801 3 2       BEGIN
00802 3 2         AC(J):=DBLE(RES); GET(RES);
00803 3 1       END;
00804 3 1     (CONTINUACION SE FORMAN LAS MATICES 'A' Y 'E' PARA OBTENER
00805 3 2     LOS COEFICIENTES DEL FILTRO)
00806 3 1     FOR I:=2 TO NI DO
00807 3 2       BEGIN
00808 3 2         AC(I):=DBLE(RES);
00809 3 2         LE(I):=DBLE(RES); GET(RES);
00810 3 2         FOR J:=1 TO P DO
00811 3 2           AC(I,J):=AC(I,J-1);
00812 3 1         END;
00813 3 2         E(NI):=DBLE(RES);
00814 3 1     (LLAMA A LA SUBROUTINA QUE ESTIMA LOS COEFICIENTES DEL FILTRO
00815 3 1     MEDIANTE EL ALGORITMO DE MINIMOS CUADRADOS)
00816 3 1     TRIANGULAR(P,I,A,E,X);
00817 3 1     FOR I:=1 TO P DO
00818 3 1       WRITELN (' C(I, I):=' ,X(I));
00819 3 1     ACTUALIZA ENT;
00820 3 1     ACTUALIZA SAL;
00821 3 1     END;
00822 3 0
00823 3 1
00824 3 1   IF TRUE;
00825 3 1   WHILE SI 0

```

-LINE-IDC-PL-SL-

```

00826 1 2 BEGIN
00827 1 2 PAGINA;
00828 1 2 WRITELN ('(1) METODO PARAMETRICO');
00829 1 2 WRITELN ('(2) GENERACION DE RESIDUOS');
00830 1 2 WRITELN ('(3) ARREGLO DE RESIDUOS PARA OBTENER LOS ');
00831 1 2 WRITELN (' COEFICIENTES DEL FILTRO');
00832 1 2 WRITELN ('(4) METODO RECURSIVO PARA M.P. ');
00833 1 2 WRITELN ('(5) REGRESAR AL NIVEL 1');
00834 1 2 WRITELN;WRITELN;
00835 1 2 WRITELN ('ESCRIBE EL NUMERO DE LA OPCION QUE QUIERAS...');
00836 1 2 READ (I);
00837 1 2 IF (I<1) OR (I>5) THEN I:=6;
00838 1 3 CASE I OF
00839 1 3 1: ARREGLO;
00840 1 3 2: RESIDUOS;
00841 1 3 3: RESI ARREGLO;
00842 1 3 4: WRITELN ('METODO RECURSIVO PARA M.P. ');
00843 1 3 5: SI:=FALSE;
00844 1 3 6: WRITELN ('ESTA OPCION NO SE VALE');
00845 1 2 END;
00846 1 1 END;
00847 0 0 END;
00848 0 0
00849 1 0 PROCEDURE ANALISIS_DATOS;
00850 1 0 VAR I: INTEGER;
00851 1 0 BB: BOOLEAN;
00852 1 0
00853 2 0 PROCEDURE CORRELACIONES;
00854 C 2 0 (SUBROUTINA QUE HACE LA CORRELACION DE UNA ARCHIVO CON
00855 C 2 0 OTRO ARCHIVO)
00856 2 0 VAR KE,COM1,I,J:INTEGER;
00857 2 0 BB: BOOLEAN;
00858 2 0
00859 3 0 PROCEDURE CALCULO (KE:INTEGER; VAR A,B,CORRELACIONES:CC);
00860 3 0 VAR SUM:ARRAY[0..RE] OF REAL;
00861 3 0 I,J,K:INTEGER;
00862 3 0 G: REAL;
00863 3 1 BEGIN
00864 3 1 FOR I:=0 TO KE DO
00865 3 2 BEGIN
00866 3 2 RESET(A);RESET(B);
00867 3 2 FOR J:=1 TO I DO
00868 3 2 GET (A);
00869 3 2 SUMC[I]:=0; K:=0;
00870 3 2 WHILE (NOT EOF(A)) AND (K< COM1-KE) DO
00871 3 3 BEGIN
00872 3 3 SUMC[I]:= SUMC[I] +A^AB";
00873 3 3 GET(A); GET(B);
00874 3 3 K:= K+1;
00875 3 3 END;
00876 3 2 SUMC[I]:=SUMC[I]/(COM1-KE);
00877 3 1 END;
00878 3 1 FOR I:=0 TO KE DO
00879 3 2 BEGIN
00880 3 2 IF BB= TRUE THEN

```

-LINE-IDC-FL-SL-

```

00881 3 3 BEGIN
00882 3 3 IF I=0 THEN G:=SUME0;
00883 3 3 SUMC1:=SUMC1/G;
00884 J 2 END;
00885 3 2 WRITELN ('LA CORRELACION EN EL INSTANTE ',I:3,' ES:',SUMC1:10);
00886 3 1 END;
00887 3 1 REWRITE(CORRELACIONES);
00888 3 1 CORRELACIONES:= CON1-KE; PUT(CORRELACIONES);
00889 3 1 FOR I:=0 TO KE DO
00890 3 2 BEGIN
00891 3 2 CORRELACIONES:= SUMC1;
00892 3 2 PUT(CORRELACIONES);
00893 3 1 END;
00894 3 1 CLOSE(CORRELACIONES);
00895 2 0 END;
00896 2 0
00897 2 1 BEGIN
00898 2 1 PAGRAINICIA('BASICO', 'NADA', '2D');
00899 2 1 VACIA(1);
00900 2 1 TERMINA;
00901 2 1 WRITELN('(1) MEDIA ');
00902 2 1 WRITELN('(2) RUIDO ');
00903 2 1 WRITELN('(3) RUF1 ');
00904 2 1 WRITELN('(4) RESI ');
00905 2 1 WRITELN; WRITELN;
00906 2 1 WRITELN ('QUIERES HACER LA CORRELACION DEL ARCHIVO #');
00907 2 1 READ(I);
00908 2 1 WRITELN ('CON EL ARCHIVO #');
00909 2 1 READ(J);
00910 2 2 CASE I OF
00911 2 2 1: COPIA_3 (MEDIA,A);
00912 2 2 2: COPIA_3 (RUIDO,A);
00913 2 2 3: COPIA_3 (RUF1,A);
00914 2 2 4: COPIA_3 (RESI,A);
00915 2 1 END;
00916 2 2 CASE J OF
00917 2 2 1: COPIA_3 (MEDIA,B);
00918 2 2 2: COPIA_3 (RUIDO,B);
00919 2 2 3: COPIA_3 (RUF1,B);
00920 2 2 4: COPIA_3 (RESI,B);
00921 2 1 END;
00922 2 1 WRITELN ('ESCRIBE EL NUMERO DEL ARCHIVO EN EL CUAL QUIERES');
00923 2 1 WRITELN ('GUARDAR LOS DATOS CORRELACIONADOS ');
00924 2 1 WRITELN; WRITELN;
00925 2 1 WRITELN ('(1) CORRELACION1 ');
00926 2 1 WRITELN ('(2) CORRELACION2 ');
00927 2 1 WRITELN ('(3) CORRELACION3 ');
00928 2 1 WRITELN; WRITELN;
00929 2 1 WRITELN ('OPCION#.. ');
00930 2 1 READ (I);
00931 2 1 IF (I<1) OR (I>3) THEN I:=4;
00932 2 1 WRITELN ('ESCRIBE EL #1 SI QUIERES CORRELACION NORMALIZADA');
00933 2 1 WRITELN ('ESCRIBE EL #2 SI NO LA QUIERES NORMALIZADA');
00934 2 1 READ (J);
00935 2 1 IF J=1 THEN RR:=TKUE ELSE RR:=FALSE;

```

-LINE-LOC-PL-SL-

```

00936 2 1 WRITELN ('ESCRIBE EL NUMERO DE CORRELACIONES QUE QUIERAS');
00937 2 1 READ (KF);
00938 2 1 RESET(B); COM1:=1;
00939 2 1 WHILE NOT EOF (B) DO
00940 2 2 BEGIN
00941 2 2     COM1:= COM1+1;
00942 2 2     GET(B);
00943 2 1 END;
00944 2 2 CASE 1 OF
00945 2 2 1: CALCULO (KF,A,B,CORRE1);
00946 2 2 2: CALCULO (KF,A,B,CORRE2);
00947 2 2 3: CALCULO (KF,A,B,CORRE3);
00948 2 2 4: WRITELN ('NO SE VALE');
00949 2 1 END;
00950 1 0 END;
00951 1 0
00952 2 0 PROCEDURE MEDIAA;
00953 C 2 0 (SUBROUTINA QUE CALCULA LA MEDIA DE UN ARCHIVO, PUDIENDO RESTAR A
00954 C 2 0 ESE ARCHIVO LA MEDIA OBTENIDA)
00955 2 0 VAR I,J: INTEGER;
00956 2 0     G: REAL;
00957 2 0
00958 3 0 PROCEDURE REN(VAR A,C:CC);
00959 3 0 VAR COM,COM1: INTEGER;
00960 3 0 VAR D:CC;
00961 3 1 BEGIN
00962 3 1 I:=0;G:=0;
00963 3 1 RESET(A);
00964 3 1 WHILE NOT EOF (A) DO
00965 3 2 BEGIN
00966 3 2     G:=G+A;
00967 3 2     I:= I+1;
00968 3 2     GET(A);
00969 3 1 END;
00970 3 1 G:=G/I;
00971 3 1 WRITELN ('EL VALOR DE I:=',I);
00972 3 1 WRITELN ('LA MEDIA DEL ARCHIVO ES: ', G:10);
00973 3 1 REWRITE(B);
00974 3 1 COM1:=0;
00975 3 1 RESET(A);
00976 3 1 WHILE NOT EOF (A) DO
00977 3 2 BEGIN
00978 3 2     D:=A^5;
00979 3 2     GET(A); PUT(D);
00980 3 2     COM1:= COM1+1;
00981 3 1 END;
00982 3 1 WRITELN ('EL VALOR DE COM1 ES ',COM1);
00983 3 1 REWRITE(C); RESET(D);
00984 3 1 COM:=0;
00985 3 1 WHILE NOT EOF (D) DO
00986 3 2 BEGIN
00987 3 2     C^:=D;
00988 3 2     PUT(C); GET(D);
00989 3 2     COM:= COM+1;
00990 3 1 END;

```

-LINE-IDC-PL-SL-

```

00991 3 1 CLOSE(C);
00992 3 1 WRITELN ('EL VALOR DEL CONTADOR ES',CON);
00993 2 0 END;
00994 2 0
00995 2 1 BEGIN
00996 2 1 PAGRATINICA('BASICO', 'NADA', '20');
00997 2 1 VACIA(1);
00998 2 1 TERMINA;
00999 2 1 WRITELN (' QUIERES OBTENER LA MEDIA DEL ARCHIVO ');
01000 2 1 WRITELN; WRITELN;
01001 2 1 WRITELN ('(1) SAL ');
01002 2 1 WRITELN ('(2) ENT ');
01003 2 1 WRITELN ('(3) RUIDO ');
01004 2 1 WRITELN ('(4) MEDIA ');
01005 2 1 WRITELN; WRITELN;
01006 2 1 WRITELN ('OPCIONE....');
01007 2 1 READ(I);
01008 2 1 IF (I<1) OR (I>4) THEN I:=5;
01009 2 1 WRITELN; WRITELN;
01010 2 1 WRITELN ('Y EL ARCHIVO CON LA MEDIA RESTADA LA ');
01011 2 1 WRITELN ('ALMACENAR EN EL ARCHIVO');
01012 2 1 WRITELN; WRITELN;
01013 2 1 WRITELN ('(1) SAL ');
01014 2 1 WRITELN ('(2) ENT ');
01015 2 1 WRITELN ('(3) RUIDO ');
01016 2 1 WRITELN ('(4) MEDIA ');
01017 2 1 WRITELN ('(5) QUIERES DEJAR LOS ARCHIVOS IGUALES');
01018 2 1 WRITELN; WRITELN;
01019 2 1 WRITELN ('OPCIONE... ');
01020 2 1 READ (J);
01021 2 1 IF (J<1) OR (J>5) THEN J:=6;
01022 2 2 CASE I OF
01023 2 2 1: COPIA_3 (SAL,B);
01024 2 2 2: COPIA_3 (ENT,B);
01025 2 2 3: COPIA_3 (RUIDO,B);
01026 2 2 4: COPIA_3 (MEDIA,B);
01027 2 2 5: WRITELN ('NO SE VALE');
01028 2 1 END;
01029 2 2 CASE J OF
01030 2 2 1: REM(B,SAL);
01031 2 2 2: REM(B,ENT);
01032 2 2 3: REM(B,RUIDO);
01033 2 2 4: REM (B,MEDIA);
01034 2 2 5: REM (B,B);
01035 2 2 6: WRITELN ('NO SE VALE');
01036 2 1 END;
01037 1 0 END;
01038 1 0
01039 2 0 PROCEDURE GRAFICACION;
01040 C 2 0 (SUBROUTINA QUE GRAFICA UN ARCHIVO DETERMINADO)
01041 2 0 CONST F=0.01;E1=0.1;F2=10;
01042 2 0 VAR I,J,K,L,NE: INTEGER;
01043 2 0 X1,X11,Y1,VMAX,VMIN,G2,H,H1,C1,C2: REAL;
01044 2 0 EX,EY,FX,FIY,G1,G,X,Y: REAL;
01045 2 0

```

-LINE-IDC-PL-SL-

```

01046 C 2 0 (A CONTINUAL IOM SE PRESENTAN SUBROUTINAS QUE FORMAN PARTE
01047 C 2 0 DE LA SUBROUTINA DE GRAFICACION)
01048 3 0 PROCEDURE ELECHADOR(A,R: REAL);
01049 3 1 BEGIN
01050 3 1 ABS2NUEVE(A,B+2AEY);
01051 3 1 ABS2LINEA(A,B+2AEY);
01052 3 1 ABS2LINEA(A+2AEX,B);
01053 3 1 ABS2LINEA(A,B+2AEY);
01054 2 0 END;
01055 2 0
01056 3 0 PROCEDURE ELECHAVER(A,B: REAL);
01057 3 1 BEGIN
01058 3 1 ABS2NUEVE(A+2AEX,B);
01059 3 1 ABS2LINEA(A+2AEX,B);
01060 3 1 ABS2LINEA(A,B+2AEY);
01061 3 1 ABS2LINEA(A+2AEX,B);
01062 2 0 END;
01063 2 0
01064 3 0 PROCEDURE RAVER(A,B: REAL);
01065 3 1 BEGIN
01066 3 1 ABS2NUEVE(A,B+EY);
01067 3 1 ABS2LINEA(A,B+EY);
01068 2 0 END;
01069 2 0
01070 3 0 PROCEDURE RAHOR(A,B: REAL);
01071 3 1 BEGIN
01072 3 1 ABS2NUEVE(A+EX,B);
01073 3 1 ABS2LINEA(A+EX,B);
01074 2 0 END;
01075 2 0
01076 3 0 PROCEDURE PAVER(A,B,C:REAL);
01077 3 1 BEGIN
01078 3 1 ABS2NUEVE(A,B+ELY);
01079 3 1 NUMEROS(C,'14');
01080 2 0 END;
01081 2 0
01082 3 0 PROCEDURE NUMOR(A,B: REAL);
01083 3 1 BEGIN
01084 3 1 ABS2NUEVE(A+EX,B);
01085 3 1 NUMEROS(B,'F5.1');
01086 2 0 END;
01087 2 0
01088 3 0 PROCEDURE MARCO_CORRELACION;
01089 3 0 VAR N,YA: REAL;
01090 3 1 BEGIN
01091 3 1 TEXTO('CORRELACION');
01092 3 1 RESET(A); N:=A;
01093 3 1 YA:= 2A UMAX/(SORT(N));
01094 3 1 ABS2NUEVE(0,YA); ABS2LINEA(KF,YA);
01095 3 1 ABS2NUEVE(0,-YA); ABS2LINEA(KF,-YA);
01096 2 0 END;
01097 2 0
01098 2 1 BEGIN
01099 2 1 K:= 0;
01100 2 1 WRITELN('CUAL DE LOS SIGUIENTES ARCHIVOS QUIERES GRAFICAR');

```

-LINE-IDC-PL-SL-

```

01101 2 1 WRITELN; WRITELN;
01102 2 1 WRITELN ('(1) ARCHIVO ENT');
01103 2 1 WRITELN ('(2) ARCHIVO SAL');
01104 2 1 WRITELN ('(3) ARCHIVO CORREI');
01105 2 1 WRITELN ('(4) ARCHIVO MEDIA');
01106 2 1 READ(I);
01107 2 1 WRITELN (' ESCRIBE EL NUMERO DE LA OPCION QUE VESEES');
01108 2 1 WRITELN; WRITELN;
01109 2 1 WRITELN ('(1) GRAFICAR TODO EL ARCHIVO');
01110 2 1 WRITELN ('(2) GRAFICAR PARTE DEL ARCHIVO');
01111 2 1 WRITELN; WRITELN;
01112 2 1 WRITELN ('OPCION ....');
01113 2 1 READ(J);
01114 2 1 IF (I<1) OR (I>4) THEN I:=5;
01115 2 2 CASE II OF
01116 2 2 1: COPIA_3 (ENT,B);
01117 2 2 2: COPIA_3 (SAL,B);
01118 2 2 3: COPIA_3 (CORREI,B);
01119 2 2 4: COPIA_3 (MEDIA,B);
01120 2 2 5: WRITELN ('NO SE VALE');
01121 2 1 END;
01122 2 1 IF J=2 THEN
01123 2 2 BEGIN
01124 2 2 WRITELN ('QUIERES GRAFICAR DEL DATO #');
01125 2 2 READ(K);
01126 2 2 WRITELN ('AL DATO #');
01127 2 2 READ(L);
01128 2 2 RESET(R);
01129 2 2 I:=0;
01130 2 2 WHILE (NOT EOF(B)) AND (I<K) DO
01131 2 3 BEGIN
01132 2 3 GET(P);
01133 2 3 IF EOF(B) THEN
01134 2 3 WRITELN ('HUBO FIN DEL ARCHIVO ANTES DEL LIMITE INFERIOR');
01135 2 3 I:= I+1;
01136 2 2 END;
01137 2 2 REWRITE(A);
01138 2 2 WHILE (NOT EOF(B)) AND (I<L+1) DO
01139 2 3 BEGIN
01140 2 3 A:= B;
01141 2 3 GET(R); PUT(A);
01142 2 3 IF EOF(B) THEN
01143 2 3 WRITELN ('HUBO FIN DEL ARCHIVO ANTES DEL LIMITE SUPERIOR');
01144 2 3 I:= I+1;
01145 2 2 END;
01146 2 2 END
01147 2 1 ELSE
01148 2 1 IF J=1 THEN COPIA_3 (R,A)
01149 2 1 ELSE WRITELN('NO SE VALE');
01150 2 1 RESET(A);
01151 2 1 IF II=3 THEN GET(A);
01152 2 1 VMAX:= A;VMIN:= A;
01153 2 1 WHILE NOT EOF(A) DO
01154 2 2 BEGIN
01155 2 2 GET(A);

```

-LINE-IRC-PL-SL-

```

01156 2 1 IF A > VMAX THEN VMAX:= A;
01157 2 2 IF A < VMIN THEN VMIN:= A;
01158 2 1 END;
01159 2 1 IF VMIN > 0 THEN VMIN:=0;
01160 2 1 C1:= (VMAX-VMIN)/E2;
01161 2 1 RESET (A);
01162 2 1 KE:= 0;
01163 2 1 WHILE NOT EOF(A) DO
01164 2 2 BEGIN
01165 2 2 GET(A);
01166 2 2 KE:= KE+1;
01167 2 1 END;
01168 2 1 C2:= KE/E2;
01169 2 1 FX:= EAC2*E2;
01170 2 1 FY:= EAC1*E2;
01171 2 1 FIY:= FIAC1AF2;
01172 2 1 F1X:= FIAC2AF2;
01173 2 1 PAGERADIL:=' BASIC', 'MADA', '2D';
01174 2 1 ABRE (0);
01175 2 1 G2:= +0.1*E2AC1;
01176 2 1 H:= -0.1*E2AC2; H1:= 1.1*E2AC2;
01177 2 1 G:=VMIN-G2 ;G1:=VMAX+G2;
01178 2 1 VENTANA(H,G,H1,G1);
01179 2 1 ABS2MUEVE(H,G);
01180 2 1 ABS2LINEA(H1,G); ABS2LINEA(H1,G1);
01181 2 1 ABS2LINEA(H,G); ABS2LINEA(H,G);
01182 2 1 X:= 1.05*E2AC2; Y:=0;
01183 2 1 FLECHAHOR(X,Y);
01184 2 1 ABS2MUEVE(X,Y); ABS2LINEA(0,0);
01185 2 1 ABS2MUEVE(0,VMIN);
01186 2 1 ABS2LINEA(0,VMAX+0.5*G2);
01187 2 1 X:= 0; Y:= VMAX+0.5*G2;
01188 2 1 FLECHAVEC(X,Y);
01189 2 1 X:=0; Y:=0;
01190 2 1 ABS2MUEVE(X,Y);
01191 2 1 FOR I:=1 TO E2 DO
01192 2 2 BEGIN
01193 2 2 X:= X+ C2;
01194 2 2 Y1:= ROUND(X);
01195 2 2 Y11:= Y+ F1;
01196 2 2 FAVEC(Y1,Y11);
01197 2 2 IF I<E2 THEN MUEVE(X1,Y,Y11);
01198 2 1 END;
01199 2 1 X:=0; Y:=VMIN; Y1:=ROUND(10*Y);
01200 2 1 ABS2MUEVE(X,Y);
01201 2 1 FOR I:=0 TO E2 DO
01202 2 2 BEGIN
01203 2 2 PAHOR(X,Y1/10);
01204 2 2 MUEVE(X,Y1/10);
01205 2 2 Y:= Y+ C1;
01206 2 2 Y1:= ROUND(10*Y);
01207 2 1 END;
01208 2 1 ABS2MUEVE 1.05*E2AC2,-0.04*E2AC1);
01209 2 1 TEXTB('X');
01210 2 1 ABS2MUEVE(0.05*E2AC2,VMAX+0.4*G2);

```

-LINE-IDC-PL-SL-

```

01211 2 2 CASE I OF
01212 2 2 1: TEXTO('ENTRADA');
01213 2 2 2: TEXTO('SALIDA');
01214 2 2 3: MAKCO('OPRELACION');
01215 2 2 4: TEXTO('MEDIA');
01216 2 1 END;
01217 2 1 RESET(A);
01218 2 1 IF I1=3 THEN GET(A);
01219 2 1 ABS2MUEV(0,A");
01220 2 1 GET(A); I:=1;
01221 2 1 WHILE NOT EOF(A) DO
01222 2 2 BEGIN
01223 2 2 ABS2LINEA(I,A");
01224 2 2 GET(A);
01225 2 2 I:= I+1;
01226 2 1 END;
01227 2 1 CIERRA; TERMINA;
01228 1 0 END;
01229 1 0
01230 2 0 PROCEDURE OTRO_NIVEL;
01231 2 1 BEGIN
01232 2 1 PAGRAINICIA('BASICO','MADA','2D');
01233 2 1 VACIA(1);
01234 2 1 TERMINA;
01235 2 1 BB:= FALSE;
01236 1 0 END;
01237 1 0
01238 1 1 BEGIN
01239 1 1 BB:=TRUE;
01240 1 1 WHILE BF DO
01241 1 2 BEGIN
01242 1 2 WRITELN ('ESCRIBE EL NUMERO DE LA OPCION QUE DESEES');
01243 1 2 WRITELN; WRITELN;
01244 1 2 WRITELN (('1) CORRELACION DE ARCHIVOS ');
01245 1 2 WRITELN (('2) MEDIA DE ARCHIVOS');
01246 1 2 WRITELN (('3) GRAFICACION DE ARCHIVOS');
01247 1 2 WRITELN (('4) QUIERES REGRESAR AL NIVEL ANTERIOR');
01248 1 2 WRITELN; WRITELN;
01249 1 2 WRITELN ('OPCION#... ');
01250 1 2 READ (I);
01251 1 2 IF (I<1) OR (I>4) THEN I:=5;
01252 1 3 CASE I OF
01253 1 3 1: CORRELACIONES;
01254 1 3 2: MEDIA;
01255 1 3 3: GRAFICACION;
01256 1 3 4: OTRO_NIVEL;
01257 1 3 5: WRITELN ('NO SE VALE');
01258 1 2 END;
01259 1 1 END;
01260 0 0 END;
01261 0 0
01262 0 1 BEGIN
01263 0 1 S:=TRUE;
01264 0 1 WHILE S DO
01265 0 2 BEGIN

```

-LINE-IDC-PL-SL-

```

01266 0 2  WRITELN ('(1) SIMULACION DE UN SISTEMA LINEAL (DISCRETO)');
01267 0 2  WRITELN ('(2) ARREGLO DE LOS DATOS DEL SISTEMA Y');
01268 0 2  WRITELN (' EJECUCION DEL ALGORITMO DE MINIMOS CUADRADOS');
01269 0 2  WRITELN ('(3) ANALISIS DE DATOS');
01270 0 2  WRITELN ('(4) FIN DEL PROGRAMA');
01271 0 2  WRITELN;WRITELN;
01272 0 2  WRITELN (' ESCRIBE EL NUMERO DE LA OPCION QUE QUIERAS...');
01273 0 2  READ(I);
01274 0 2  IF (I<1) OR (I>4) THEN I:=5;
01275 0 3  CASE I OF
01276 0 3  1: SIMULACION;
01277 0 3  2: ESTIMACION DE COEFICIENTES;
01278 0 3  3: ANALISIS DATOS;
01279 0 3  4: S:= FALSE;
01280 0 3  5: WRITELN ('NO SE VALE ');
01281 0 2  END;
01282 0 1  END;
01283 0 0  END.

```

PRUEBA
01

Pascal Compilation Statistics

3-Oct-1985 14:51:31
3-Oct-1985 12:42:59

VAX Pascal V3.0-2
DL51:CPRO1955.21713PRUEBA4.PAS;3 (1)

Page

PSECT SUMMARY

Name	Bytes	Attributes
SCODE	31198 MOVEC,MOVRT, RD, EXE, SHR, LCL, REL, COM, PIC,ALIGN(2)	
LOCAL	500 MOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, COM, PIC,ALIGN(2)	

ENVIRONMENT STATISTICS

File	----- Symbols -----		
	Total	Loaded	Percent
DBAO:CPACKAGES.PACRA3PAGINTERF.PEN;27	648	31	5

COMMAND QUALIFIERS

PASCAL/MOOP/LIST PRUEBA4

/CHECK=(BOUNDS,MOCASE_SEL1,TURS,MOOVERFLOW,MOPIINTERK,MOSUBXANGE)
/DEBUG=(NOSYMBOLS,TRACEBACK)
/SHOW=(DICTIONARY,INCLUDE.MOTINLINE,HEADER,SOURCE,STATISTICS)
/MOOPTIMIZE
/MOENVIRONMENT
/LIST=DL51:CPRO1955.21713PRUEBA4.LIS;3
/OBJECT=DL51:CPRO1955.21713PRUEBA4.OBJ;2
/MOCKUS_REFERENCE /ERROR_LIMIT=30 /MOG_ELIDATING /MOMACHINE_CODE /MOOLD_VERSION /MOSTANDARD /WARNINGS

COMPILER INTERNAL TIMING

Phase	Faults	CPU Time	Elapsed Time
Initialization	148	00:00.4	00:00.9
Source Analysis	29.3	00:07.8	00:10.9
Source Listing	.1	00:02.0	00:03.4
Tree Construction	45.1	00:05.6	00:08.4
Flow Analysis	0	00:00.0	00:00.0
Value Propagation	0	00:00.0	00:00.0
Profit Analysis	0	00:00.0	00:00.0
Context Analysis	12391	00:30.6	00:41.3
Name Packing	543	00:03.0	00:03.7
Code Selection	45.3	00:07.8	00:10.2
Final	12.9	00:10.1	00:11.7
TOTAL	265.1	01:07.4	01:30.8

COMPILATION STATISTICS

CPU Time: 01:07.4 (1143 Lines/Minute)
Elapsed Time: 01:30.8
Page Faults: 26591
Compilation Complete

BIBLIOGRAPHIA

1. Akaike H. 'Some problems in the application of the cross-spectral method. In Spectral Analysis of Time Series. (B. Harris, Ed.) Wiley, New York (1976).
2. Astrom K.J. and Bohlin T. 'Numerical identification of linear dynamic systems from normal operating records'. Preprints IFAC Symposium on Theory of Self-Adaptive Control Systems, Teddington England, 1965.
3. Astrom K.J. and Soderstrom T. 'System Identification - a Survey', Automatica, Vol 7, pp 123-162, 1971.
4. Astrom K.J. and Kollstrom C. 'Application of system identification techniques to the determination of ship dynamics.' From 3rd IFAC Symposium on Identification and System Parameter Estimation, the Hague (North-Holland), 1973.
5. Astrom K.J. and Wittenmark B. 'Problems of identification and control.' Journal of Mathematical Analysis and Applications, vol 34, pp 90-113 1971.
6. Astrom K.J. 'Introduction to Stochastic Control Theory', Academic Press, London, 1970.
7. Box G.E.P. and Jenkins G.M. 'Time Series Analysis. Forecasting and Control'. Holden-Day, San Francisco (1976).
8. Clarke R.W. 'Generalized least squares estimation of parameter of a dynamic model.' 1st IFAC Symposium on Identification in Automatic Control Systems, Prague, 1967.
9. Chang F.H.L. and Luus R. 'A Noniterative Method for Identification Using Hammerstein Model'. IEE Transactions on Automatic Control, Vol AC-16, pp 454-468, Oct 1971.
10. Chen C.T. 'Introduction to Linear System Theory', Holt, Rinehart and Winston, New York, 1970.
11. Eykhoff P. System Identification, Wiley, London, 1974.
12. Eykhoff Pieter 'Trends and Progress in System Identification', Pergamon Press, New York, 1981.
13. Fredriksson B. 'A modified pre-filter for some recursive parameter estimation algorithms.' IEEE Transactions on Automatic Control, Vol AC-17, pp 91-946.
14. Gauss, K.G. 'Theory of Motion of the Heavenly Bodies'. New York, Dover, 1963.
15. Gevers M and Wertz V. 'A recursive least squares d-step-ahead predictor in lattice and ladder form.' IEEE Transactions on Automatic Control, vol 28, no 4, 1983.
16. Goodwin and Payne 'Dynamic System Identification', Academic Press, New York, 1970.
17. Gustavsson L., Chung L. and Soderstrom T. 'Identification of Processes in Close Loop- Identifiability and Accuracy Aspects'. Automatica Vol 13, pp 19-25, 1977.
18. Hjalton-Jones K. and Soderstrom T. 'Recursive Generalized Least Squares for On-line Identification of Process Parameters'. Proceedings IEE, vol 116, pp 2057-2062, 1969.

19. Hsia T.C. 'System Identification- Least Squares Methods', D.C Heath and Company, U.S.Am 1977.
20. Kailath T. 'Lectures on Linear Least-Squares Estimation,' Springer-Verlag, Wieh 1976.
21. Knuth D 'The Art of Computer Programming', Addison-Wesley, California, 1969.
22. Lee D.T. and Morf M. 'Recursive least squares ladder estimation algorithms.' IEEE Transactions on Acoustic, Speech and Signal Processing, vol 29, pp 627-641, 1981.
23. Ljung L. 'Consistency of the least squares identification method.' IEEE transactions on Automatic Control; 21-1976.
24. Ljung L. 'On consistency for prediction error identification methods, Report 7405, Dept of Automatic Control, Lund Institute of Technology, Sweden (March 1974).
25. Ljung L. 'Convergence analysis of parametric identification methods.' IEEE Transactions on Automatic Control, vol 27, pp 770-783, 1978.
26. Ljung L. and Söderström I. 'Theory and Practice of Recursive Identification.' MIT Press, 1983.
27. Mehra, R.K. 'Optimal Input Signals for Parameter Estimation in Dynamic Systems- Survey and New Results' IEEE Transactions on Automatic Control, Vol 17, No 1, pp 45-55 1973.
28. Mehra R.K. and Tyler J.S. 'Case studies in aircraft parameter identification.' Proc 3rd IFAC Symposium on Identification and System Parameter Estimation, the Hague (North-Holland) 1973.
29. Mendel, J.M. Discrete Techniques of Parameter Estimation, Marcel Dekker, New York, 1973.
30. Papulis A 'Probability Random Variables and Stochastic Process' Mc Graw-Hill, 1965.
31. Sage. A.P. and Melsa J.L., 'Estimation Theory with Application to Communications and Control.' Mc Graw-Hill, New York, 1971.
32. Satorius E.H and Pack J.D. 'Application of least squares lattice algorithms to adaptive equalization.' IEEE Transactions on Communications, vol 29, pp 136-142, 1981.
33. Söderström. I. 'Convergence Properties of the Generalized Least Squares Identification Method', Automatica, Vol 10, pp 617-626, 1974.
34. Söderström I., Ljung L. and Gustavsson I. 'On the accuracy of identification and the design of identification experiments'. Report 7428 Dept of Automatic Control, Lund Institute of Technology, Sweden (Dec. 1974).
35. Söderström I., Gustavsson I. and Ljung L. 'Identifiability conditions for linear systems operating in closed loop'. Int. J. Control 21, 243-255 (1975).
36. Söderström I., Gustavsson I. and Ljung L. 'On the accuracy problem in identification'. Proc 6th IFAC Congress, Boston, MA (1975).
37. Söderström I. Ljung L and Gustavsson I. 'Identifiability conditions for linear multivariable systems operating under feedback. IEEE Trans. Aut. Control, AC-21 (Dec 1976).

39. Soderstrom T., Ljung L and Gustavson I. 'A theoretical analysis of recursive identification methods.' Automatica, vol 14, pp 231-244, 1978.
40. Sorenson R.W. 'Least-squares estimation: from Gauss to Kalman'. IEEE spectrum: July 1970.
41. Stein G., Johnson K. 'Toward Development of a Practical Benchmark Example for Adaptive Control'. Control Systems Magazine, Dec. 1981).
42. Sternby J. 'On consistency for the method of least squares using martingales theory.' IEEE Transactions on Automatic Control, vol 21, pp 346-352.
43. Stoer J. and Burlish B. 'Introduction to Numerical Analysis' Springer-Verlag New York 1980.
44. Unbehauen H. and Gotsing B. 'Test for Determining Order in Parameter Estimation' Automatica, Vol 10, pp 233-244, May 1974.