

# SIMULADOR DE SISTEMAS DINÁMICOS CON PROCESADO DIGITAL DE SEÑALES<sup>1</sup>

M. BERTRÁN, J. M. JOVER

*Se describe un simulador de sistemas dinámicos basado en técnicas de procesado digital de señales. El simulador está orientado a diagramas de bloques y - junto con otros programas de diseño de los bloques permite abordar la simulación de sistemas dinámicos determinísticos con un enfoque distinto del que reduce e integra las ecuaciones de estado o del que se basa en el calculador -- analógico.*

*La problemática y uso de la metodología se discuten a través de dos ejemplos: uno de comunicaciones y otro de regulación. El estudio de las fuentes de error inherentes en el método y el de la forma de reducir su influencia, permite -- una primera comparación con los demás.*

## 1. INTRODUCCION

El diseño de un sistema complejo se facilita si puede descomponerse desde un principio en un conjunto de subsistemas o bloques interconectados. Tanto para poder llegar a esta representación como para poder usarla para un primer estudio del sistema se hace necesario; por un lado obtener modelos de cada bloque componente y por otro lado poder interconectar los bloques de forma práctica. El simulador que se presenta en esta comunicación, BLOCS (Block Oriented Continuous Simulator)/12/, junto con un programa interactivo de diseño de bloques, objeto de otro trabajo /8/, pretende aportar una ayuda tanto al análisis como al diseño de sistemas dinámicos determinísticos complejos.

BLOCS es un simulador continuo de sistemas dinámicos determinísticos, orientado a diagramas de bloques y basado en técnicas de -- procesado digital de señales y programación estructurada. Las siguientes características lo diferencian netamente de otros simuladores:

1. Utiliza técnicas de procesado digital de señales /18,19/, en lugar de la reducción y resolución del sistema de ecuaciones de estado equivalente al diagrama de bloques.
2. Ha sido desarrollado con técnicas softwa-

re en la línea de la programación estructurada /25/, que dan mayor fiabilidad al producto, lo hacen más fácilmente modificable y transportable y le garantizan una vida más larga.

3. La comunicación usuario-ordenador se ha cuidado especialmente y se realiza interactivamente por medio del programa general MEM /12/.
4. Está preparado para abordar los diseños o análisis desde una perspectiva TOP-DOWN y está verdaderamente orientado a la descripción de sistemas por diagramas de bloques, ya que, a diferencia de otros lenguajes /1/, incluye algoritmos de ordenamiento automático de los bloques.

Las tres primeras características se desarrollan adecuadamente en los apartados siguientes. A continuación comentamos el diseño -- desde una perspectiva TOP-DOWN. Esta metodología se basa en trabajar de lo general a lo particular y no considerar ningún detalle -- particular hasta que se hayan considerado todos los aspectos generales. Gráficamente -- (Figura 1) podemos representar el método por una pirámide en dos dimensiones. Para el diseño de un sistema nos situamos en la cúspide de la pirámide, estudiamos la idea desde una perspectiva general, especificamos las características básicas y generales del sis-

- M. Bertran, E.T.S.E. Telecomunicació de la U.P. de Barcelona. Jordi Girona Salgado, s/n. Barcelona - 34.  
- J.M. Jover, E.T.S.E. Telecomunicació de la U.P. de Barcelona. Centre de Càlcul, U.P.B. Gregori Marañón, s/n. Barcelona, 28.  
- Article rebut el Juny del 1.980.

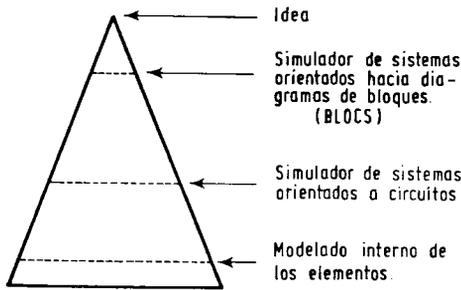


Fig. 1

Lugar que ocupa nuestro simulador, BLOCS, en el diseño de un sistema por la metodología "Top-Down".

tema pero sin entrar en detalles innecesarios por el momento. A continuación descendemos de nivel y dividimos el sistema en módulos. Cuando estamos seguros de que la especificación modular es correcta, volvemos a descender de nivel, para representar todo nuestro sistema por bloques directamente simulables por un simulador como BLOCS. El análisis de los resultados proporcionados por el simulador nos permitirá ver si se cumplen las especificaciones determinadas anteriormente (en niveles más altos de la pirámide).

Cuando el sistema orientado a bloques cumple todas las especificaciones podemos descender de nivel para, si lo deseamos, obtener una implementación hardware de cada bloque. A nivel de circuito podemos simular el sistema con muchos de los programas existentes -- /13/, por ejemplo: ECAP, SCEPTRE /6/ y SUPER SCEPTRE, NAP2 /20,21/, ANP3 /14,17/, MOFRAN /2/. En otros casos será mejor usar directamente electrónica digital /15/ para la implementación.

El nivel correspondiente a BLOCS en la metodología TOP-DOWN lo diferencia de otros simuladores. Al estar todavía lejos de muchos detalles de implementación, la utilidad de BLOCS no queda limitada al área electrónica o de comunicaciones.

## 2. PROBLEMAS SIMULABLES

En muchas ramas de la ingeniería es frecuente el uso de diagramas de bloques para una primera representación del modelo de un sistema físico, ya sea para diseñarlo o tan sólo analizarlo. Las comunicaciones y los servomecanismos no son los únicos ejemplos. La bioingeniería, la mecánica, y en general cualquier disciplina cuyos modelos estén basados en ecuaciones diferenciales, puede ser campo

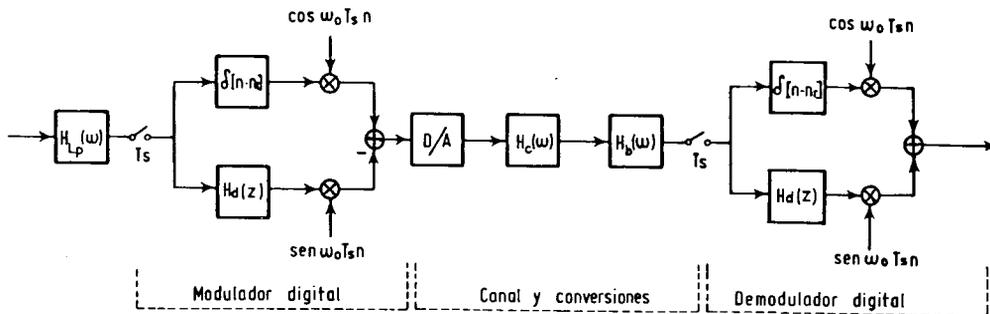


Fig. 2

Ejemplo de sistema simulable mediante BLOCS. Sistema digital de transmisión con Banda Lateral Unica, basado en un transformador de Hilbert, digital  $H_d(z)$ .

de aplicación de los esquemas de bloques.

Por limitaciones de espacio, ilustraremos el uso de BLOCS solamente en dos situaciones: un sistema modulador-demodulador de Banda Lateral Unica (BLU) /23/, y un sistema de regulación Frecuencia-Potencia de una Red de Transporte de Energía Eléctrica /3,22/. Por idénticas razones, los modelos de bloques correspondientes no representan el detalle que BLOCS podría simular, son simplemente ejemplos ilustrativos. No obstante, los resultados presentados aquí corresponden a simulaciones reales de los mismos efectuadas con BLOCS.

La Figura 2 representa el diagrama de bloques de un sistema de BLU digital. Se pretende usar BLOCS para tener una idea de la bondad de la transmisión de la banda de telefonía para diversos filtros digitales  $H_d(z)$ , que aproximan a un transformador de Hilbert ideal con retardo de  $n_x$  instantes de muestreo. El modulador y el demodulador digitales están conectados con sistemas analógicos: filtros de paso bajo  $H_{LP}(\omega)$ , y el canal de comunicaciones  $H_c(\omega)$  junto con el filtro de recepción  $H_b(\omega)$ . BLOCS permite pues la interconexión de sistemas digitales y analógicos en un mismo esquema de bloques. Asimismo, subesquemas de bloques operando a diferentes frecuencias de muestreo pueden interconectar

se. Las interconexiones deben efectuarse mediante bloques especiales: muestreadores, expansores y conversores.

Más adelante se comentará el uso de BLOCS para la observación de las bandas de frecuencia ocupadas a la salida del modulador, la transformada de Fourier de la señal modulada dejará ver claramente el grado de eliminación de la Banda Lateral conseguido con un transformador de Hilbert aproximado,  $H_d(z)$ , determinado.

Para su operación, el sistema necesita muestras de senoides. Una manera de probar la bondad de la recepción consistirá en excitar el sistema con una señal que ocupe uniformemente toda la banda de telefonía (pulso en frecuencia) y observar el espectro de frecuencias a la salida del sistema. Como puede comprobarse fácilmente, la señal requerida es una sinc (x) modulada a la frecuencia central de la banda de telefonía.

$$\frac{\sin w_a(t - t_d)}{\pi(t - t_d)} \cos w_c(t - t_d)$$

en donde

- $2w_a$  : ancho de banda de telefonía (rad/s)
- $w_c$  : pulsación central de la banda de telefonía.

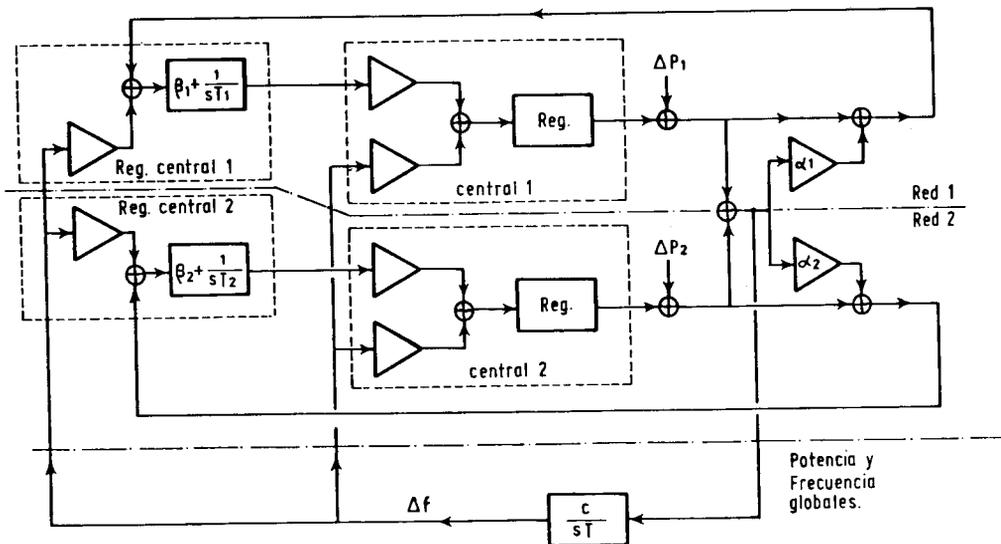


Fig. 3

Esquema simplificado de la regulación Frecuencia-Potencia de dos redes eléctricas interconectadas. Este tipo de sistemas con realimentaciones es también simulable con BLOCS.

$t_d$  : retardo para que el intervalo de simulación contenga suficientes lóbulos laterales a la izquierda del principal.

En la Figura 5, se indican los bloques requeridos por BLOCS para generar esta señal (Salida del bloque B3).

Para ajustarse a los programas de intercambio de energía entre compañías eléctricas, cada una de ellas dispone de un regulador central que, actuando sobre los reguladores locales de cada central, gobierna a distancia el grado de apertura del distribuidor de turbina de una o varias centrales de la compañía. El regulador se guía, en general, por dos señales de referencia: la frecuencia y el balance de potencia global con las compañías vecinas que compara con una potencia de consigna. La desconexión súbita, tanto de una tercera central como de una carga, produce perturbaciones sobre un estado de equilibrio que deben ser compensadas por la actuación de los reguladores centrales y locales. Un desequilibrio entre potencia global consumida y generada se traduce en aceleraciones de los alternadores que implican oscilaciones en la frecuencia global y quizás inestabilidad. BLOCS puede usarse para el estudio de estos fenómenos para diversos valores de los parámetros de los reguladores centrales.

Un esquema de bloques muy simplificado de la regulación para dos compañías vecinas puede verse en la Figura 3 /10/. Aparte de los esquemas de los reguladores centrales y locales de ambas compañías, puede observarse el bloque integrador del incremento de potencia global para traducirla en variación de frecuencia. Dos multiplicadores,  $\alpha_1$  y  $\alpha_2$ , simulan el reparto de la potencia global entre las dos compañías a través de las líneas de interconexión.

En un principio, las señales de perturbación ( $\Delta P_1, \Delta P_2$ ), serán funciones escalón y se simularán con los correspondientes bloques generadores de señales tal como se hacía para las senoides del ejemplo anterior. No obstante, una diferencia notable con el ejemplo de comunicaciones es la existencia de realimentaciones en el esquema. De hecho existen realimentaciones dentro de realimentaciones.

BLOCS admite redes de bloques con realimentaciones, no obstante, debido a la naturaleza del método que se usa para llevar a cabo la simulación existe un error implícito en cada bucle de realimentación; este fenómeno se comentará más adelante.

En el ejemplo del sistema BLU vimos que era interesante observar las transformadas de Fourier de las señales temporales; en cambio en el de Regulación es más conveniente ver las variaciones temporales. BLOCS dibuja, a demanda del usuario, tanto la representación frecuencial como la temporal de toda señal de salida de cualquier bloque del esquema.

### 3. MARCO DE UTILIZACION

Para la simulación con BLOCS del primer ejemplo es necesario disponer de un transformador de Hilbert digital aproximado,  $H_d(z)$ . Para la del segundo, de un modelo digital de un servomecanismo regulador de la apertura de los álabes de una turbina. En el primer caso debe diseñarse un filtro digital que aproxime la característica frecuencial analógica.

$$H(\omega) \begin{cases} -\pi/2 - n \frac{T_s}{T_r} \omega & , \omega \geq 0 \\ \pi/2 - n \frac{T_s}{T_r} \omega & , \omega < 0 \end{cases}$$

El diseño del filtro digital que aproxime el regulador del segundo ejemplo puede hacerse a base de medir la respuesta del servo a una entrada, un escalón por ejemplo, especificando de este modo el filtro en el dominio temporal.

Para simular sistemas analógicos debe primero obtenerse su simulador discreto /19/ para aproximarlos seguidamente mediante un Filtro Digital.

Se ve pues que el primer problema a abordar para usar BLOCS es el de diseño de los filtros digitales que simularán los bloques dinámicos del esquema en cuestión. Existen métodos clásicos para el diseño de los filtros digitales más usuales: derivadores, integradores, paso bajo, paso alto, paso de banda, banda eliminada, etc... El diseño del Transformador de Hilbert o el del modelo del regulador debe abordarse con métodos más elaborados /4,7,16/, de hecho el problema de aproximación en filtrado digital está todavía abier-

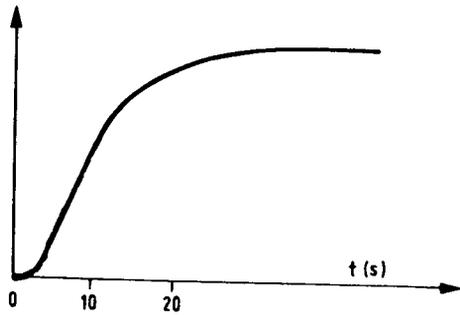


Fig. 4

Respuestas al escalón del servomecanismo regulador de apertura de un grupo generador hidráulico (Blo que Reg. en Fig. 3). Medida (—), modelo digital (---) para el simulador.

to. Existen, no obstante, métodos útiles para obtener diseños aceptables. El diseño de los dos filtros en los ejemplos se ha basado en la minimización del error cuadrático en el dominio temporal mediante el método de li nealización de Gauss /4,7,9/. El transformador de Hilbert se aproximó con un filtro pasa todo de orden ocho. El regulador con un filtro de orden cuatro. La Figura 4 muestra las respuestas al escalón medida sobre el regulador real /5/ y la aproximada correspondiente al modelo del regulador.

Para facilitar el uso de BLOCS se requiere poder diseñar, con el menor engorro posible, los filtros digitales para los bloques. En la actualidad, existen librerías de programas de diseño, que requieren programar para usarlas. A fin de facilitar la etapa de diseño de modelos, se procede en la actualidad a la construcción de un programa interactivo de diseño de filtros digitales que incluya varios métodos /8/. Su disponibilidad convertirá en trivial el diseño de modelos para la mayoría de bloques.

Una vez obtenido, con las técnicas mencionadas anteriormente, el modelo (filtro digital) de algunos bloques: transformadores de Hilbert, reguladores, derivadores, etc..., debemos dividir el sistema a simular en bloques standard del simulador:

- GENERADORES DE SECUENCIAS: delta, escalón, sinusoidal, seno de x sobre x (sinc), etc.
- BLOQUES LINEALES DE UNA ENTRADA: filtros digitales (FIR o IIR), amplificadores, retardos, etc.

- BLOQUES NO LINEALES DE UNA ENTRADA: rectificadores de onda completa o de media onda, leyes cuadráticas, raíces cuadradas, etc.
- BLOQUES ESPECIALES DE UNA ENTRADA: muestreadores, expansores, interruptores, etc.
- BLOQUES DE MAS DE UNA ENTRADA: sumadores, multiplicadores (el número de entradas es arbitrario), etc.

Así, por ejemplo, para generar el pulso de frecuencia necesario para excitar al modulador digital de BLU (sección anterior) podemos usar los bloques B1, B2 y B3 conectados según la Figura 5.

Para la simulación, debe asignarse uno de los bloques standard de BLOCS a cada elemento del diagrama de bloques original, que ha sido numerado con anterioridad. El orden asignado es irrelevante ya que BLOCS los reordena internamente como paso previo a la simulación.

Para introducir en el ordenador los datos que definen el sistema, BLOCS utiliza el programa MEM /11/ (Modifiable Expert Menu) que es totalmente general e independiente del simulador. MEM permite la entrada de datos interactivamente a cualquier programa de aplicación con las siguientes posibilidades:

- Respuestas numéricas o alfanuméricas a un menú definido por el usuario de MEM.
- Responder a varias preguntas de una vez, --

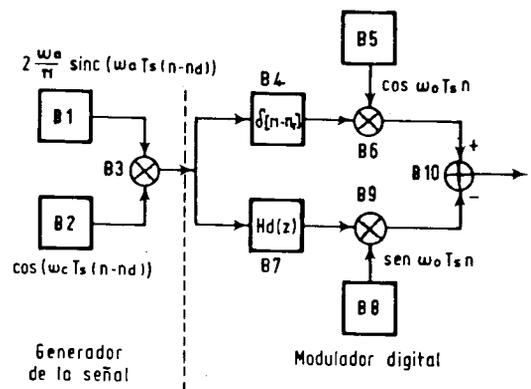


Fig. 5

Esquema de bloques, directamente representable en BLOCS correspondiente a la simulación del modulador digital de la Fig. 2 y a la generación de la señal de prueba.

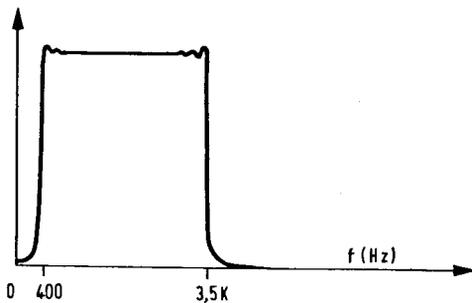


Fig. 6

Módulo de la transformada de Fourier de la señal de prueba del modulador BLU digital.

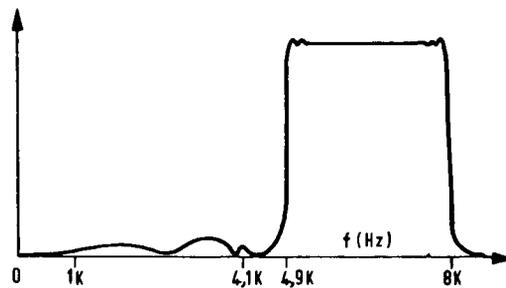


Fig. 7

Módulo de la transformada de Fourier de la salida del modulador BLU digital. Obsérvese la atenuación de la banda lateral inferior.

cuando el usuario ya conoce el diálogo -- (modo experto).

- Modificar la respuesta a preguntas formuladas con anterioridad.
- Intercomunicación de aplicaciones distintas por medio de ficheros.

Una vez introducido el diagrama de bloques, MEM permite el uso de BLOCS para la simulación de varios casos correspondientes a variaciones en algunos de sus parámetros. Después de cada simulación pueden obtenerse gráficamente, en representación frecuencial o temporal, las salidas de los bloques deseados.

Como ejemplo, la Figura 6 representa la transformada de la señal de entrada del modulador que corresponde a la ocupación uniforme de la banda de telefonía. El rizado es debido al intervalo temporal finito de simulación. La Figura 7 muestra la transformada de Fourier de la señal de salida del modulador digital. Obsérvese la atenuación de la banda lateral inferior.

#### 4. METODO DE SIMULACION

Al trabajar en el dominio temporal, el esquema de simulación consiste en un bucle. Cada pasada por el bucle equivale a recorrer, en el orden conveniente, todos los bloques del esquema de simulación, calculando el punto siguiente de la salida de cada uno de ellos. La primera pasada por el bucle equivale al instante inicial. Ello es posible ya que -- los bloques dinámicos están representados -- por Filtros Digitales que equivalen a recurrencias que permiten el cálculo del siguiente

te punto de su salida partiendo de un conjunto de puntos previos de su entrada y de su salida.

El esquema anterior exige una ordenación interna de los bloques para garantizar que cuando se calcule el siguiente punto de salida de un bloque, su entrada, que es la salida de otro bloque, ya haya sido calculada con anterioridad.

#### 5. PROBLEMAS PRINCIPALES

La simulación de sistemas por procesamiento digital de señales conlleva dos errores inherentes:

- Errores introducidos en las realimentaciones.
- Errores numéricos.

En este apartado se discuten estos errores, y se analiza la problemática de ordenamiento de bloques. En el apartado siguiente se compara para la simulación por procesamiento digital con otras alternativas, que también presentan errores inherentes.

Toda realimentación introduce un error en el cálculo de la salida del bloque que tiene como entradas la señal directa y la realimentada. Este error es debido a que al calcular la salida de dicho bloque en un instante discreto  $nT_s$  (con  $T_s$  el tiempo de muestreo), la entrada de la realimentación está calculada para el instante anterior  $(n-1)T_s$  lo cual provoca que la salida de este bloque no sea correcta. Este error --una de las entradas de

un bloque calculada en un instante anterior al considerado- se produce una vez por cada lazo de realimentación del sistema.

La necesidad de trabajar con modelos discretos y cuantizados, tanto de sistemas como de señales, implícita en la simulación por procesado digital de señales, se traduce en errores /18,19/. Damos aquí tan sólo la lista de sus fuentes principales:

- Modelado con un Filtro Digital.
- Cuantificación de sus coeficientes.
- Cuantificación de las muestras de las señales analógicas.

Con independencia de los errores de modelado y representación finita, existe el error implícito en las sumas y multiplicaciones necesarias para llevar a cabo una simulación que también se realizan con precisión finita /18,24/.

Característica fundamental de todos los errores discutidos hasta aquí es que pueden en principio paliarse mediante el aumento del orden de los modelos, la disminución del tiempo de muestreo, o bien el aumento de la precisión de la representación numérica. Traduciéndose todo ello en aumento tanto del tamaño de memoria requerida como del tiempo de simulación.

Aparte de considerar los dos tipos de error anteriores, para conseguir una simulación fiable, es imprescindible ordenar adecuadamente los bloques antes de proceder a la si-

mulación. Esta ordenación no es trivial a causa de:

- La existencia de realimentaciones.
- La ordenación puede no ser única.

La existencia de realimentaciones complica el algoritmo de ordenación ya que el criterio que sigue es que un bloque puede ser -- ejecutado cuando todas sus entradas han sido calculadas para ese instante discreto de tiempo. En una realimentación siempre hay por lo menos una entrada que no cumple el requisito anterior, como puede verse en la Figura 8(a) En consecuencia, el algoritmo de ordenación debe ser capaz de distinguir los lazos de realimentación, que incluso pueden estar incluidos los unos en los otros o entremezclados.

La Figura 8 muestra un sistema cuya ordenación de bloques no es única, dependiendo de cómo tomemos el sistema. Estos problemas de unicidad sólo aparecen en realimentaciones; por lo tanto, debido al error inherente a toda realimentación, los resultados serán distintos según se adopte una ordenación u otra. El algoritmo de ordenación debe ser capaz de distinguir estas situaciones, para preguntar al usuario qué bloque desea que sea simulado primero, cuando hay más de un camino.

BLOCS utiliza un algoritmo basado en un stack y que hemos creado especialmente para solucionar estos problemas. Ocupa unas 700 instrucciones FORTRAN, si bien resuelve tam--

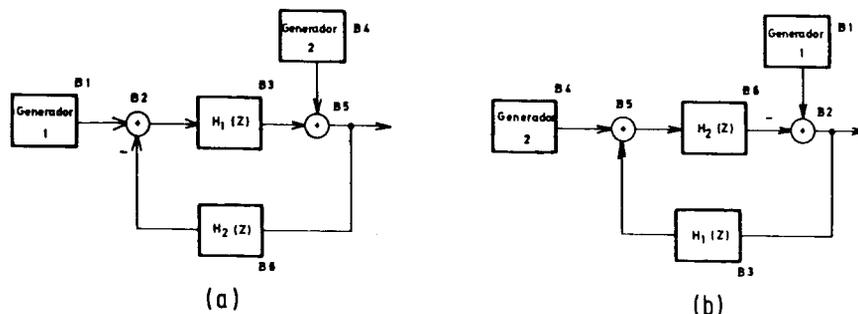


Fig. 8

Ilustración de la no unicidad de ordenación de bloques. Dos representaciones de un mismo esquema de bloques. En (a) se calcularía la salida de B2 antes que la de B5, en (b) se procedería a la inversa.

bién unos problemas originados por los bloques especiales (muestreadores, expansores, interruptores).

## 6. COMPARACION CON OTROS ENFOQUES

Esquemas de bloques pueden programarse en calculadores analógicos. Al representar sistemas de ecuaciones diferenciales, los esquemas de bloques pueden resolverse reduciendo el esquema a un sistema compacto de ecuaciones diferenciales y empleando métodos numéricos para obtener la solución para las condiciones iniciales dadas.

Dejando aparte el proceso de cablear un calculador analógico, el número de integradores requeridos constituye normalmente un límite para la representación con el debido detalle de muchos casos. Tan sólo la introducción de un modelo analógico del transformador de Hilbert digital que se ha mencionado requeriría ocho integradores. Los dos transformadores junto con los demás filtros del primer caso requerirían probablemente unos 32 integradores. La memoria disponible es la limitación equivalente en el campo discreto, pero aquí una recurrencia de orden  $n$ , equivalente a  $n$  integradores, requiere tan sólo  $2n+1$  constantes de coma flotante para los coeficientes y  $n$  para el estado del filtro, limitación irrisoria si se tiene en cuenta la tecnología actual de memorias digitales. Debido al uso de aritmética de coma flotante, los problemas de escalado implícitos en el uso de calculadores analógicos no existen en los simuladores digitales. Asimismo, el error de deriva, presente en los calculadores analógicos, tiene como contrapartida el error debido a la aritmética finita, que puede paliarse usando mayor precisión en la representación digital.

El problema de diseño de filtros digitales puede ser quizás un inconveniente de nuestro enfoque, no obstante, su solución reduce a una simple recurrencia un sistema de orden elevado, eliminando la necesidad de integradores analógicos. En muchos casos, como paso previo al uso de un simulador analógico deben diseñarse filtros analógicos, un proceso de complejidad análoga a la de diseñar un filtro digital.

Al reducir los procesos de integración a recurrencias, nuestro enfoque no usa otros métodos numéricos no lineales e iterativos, por lo general más precisos, para la integración del sistema de ecuaciones diferenciales correspondientes al esquema de bloques. Ello hace que en situaciones en donde la precisión de cálculo es un factor preponderante, incluso sobre la comodidad y facilidad de obtener una primera solución, sea aconsejable el uso de aquellos métodos numéricos de integración de sistemas de ecuaciones diferenciales. En la mayoría de los casos, no obstante, el método de diagramas de bloques dará siempre una primera aproximación por lo general suficiente.

## 7. ESTADO DE LA IMPLEMENTACION

La versión actual del simulador BLOCS es completamente operativa. Está realizada en ANSI STANDARD FORTRAN comprendiendo unas 4000 líneas de código. El programa es fácilmente transportable.

Actualmente BLOCS permite comandos para

- Ordenar los bloques y proceder a la simulación.
- Representación temporal o frecuencial de la salida de cualquier bloque.
- Listado de los parámetros del sistema.
- Modificación de parámetros.
- Copia sobre impresora de los resultados producidos sobre la pantalla.
- Copia en fichero de los datos necesarios para poder reanudar la simulación en otra ocasión.
- Repetición del último comando.

BLOCS tiene bloques para la generación de las siguientes señales: delta digital, escalón, seno, coseno, sinc. Combinando con sumadores y multiplicadores estos bloques elementales, pueden generarse muchas más señales (ver Figura 5).

Naturalmente, los bloques generadores no admiten entrada. BLOCS tiene además los siguientes tipos de bloque con entradas:

- Filtro Digital Recurrente.
- Filtro Digital de respuesta impulsional finita.

- Amplificador.
- Retardo puro.

Los bloques no lineales soportados por BLOCS en la actualidad son:

- Rectificador de media onda.
- Rectificador de onda completa.
- Ley cuadrática.
- Raíz cuadrada del módulo.
- Aproximador lineal por tramos de una función característica.
- Aproximador lineal por tramos de una función característica con histéresis.
- Truncadores a una representación binaria - con una precisión determinada.

Bloques de funciones lógicas:

- AND/NAND
- OR/NOR
- NOT

Existen además tres bloques especiales:

- Muestreador.
- Expansor.
- Interruptor.

Otros bloques permiten incorporar funciones FORTRAN directamente definidas por el usuario. Las señales de entrada también pueden ser utilizadas en forma de tabla almacenadas en ficheros. Otros bloques pueden ser añadidos con relativa facilidad. Actualmente, por requerimientos de memoria, se pueden simular hasta 100 bloques con 2.048 muestras. Existe una versión segmentada que permite simular sistemas mucho mayores. También está previsto incorporar los bloques necesarios para el tratamiento de señales aleatorias.

## 8. CONCLUSIONES

Se ha descrito, y puesto en perspectiva con otros simuladores, un simulador, BLOCS, --- orientado a esquemas de bloques, basado en el procesado digital de señales y en técnicas de programación estructurada.

La ilustración con dos ejemplos, sistema BLU y regulación de una red eléctrica, del uso del simulador ha puesto de manifiesto la --- aplicabilidad del mismo a otras áreas en don

de se usen también diagramas de bloques para la construcción de modelos.

Los errores implícitos en el método pueden reducirse aumentando la memoria requerida y el tiempo de simulación. El continuo avance y proliferación del uso de tecnología electrónica digital traerá consigo la reducción del coste de la unidad de memoria y el aumento de la velocidad de procesado, por lo que BLOCS, junto con programas de diseño de Filtros Digitales, representará una alternativa de simulación de sistemas determinísticos cada vez más aconsejable.

## 9. REFERENCIAS

- /1/ ACSL: "User Guide/Reference Manual". MITchell and Gauthier Assoc. Inc., Concord, USA, 1975.
- /2/ ANDERSEN, L.: "Preliminary Users Manual for the Program MOFRAN". Report LKT 0026. Institute of Circuit Theory and Telecommunication. Technical University of Denmark, 1974.
- /3/ BASAÑEZ, L.: "Estudio de la Regulación - Frecuencia-Potencia de Redes Interconectadas por medio de un Calculador Analógico". Instituto de Cibernética, Barcelona 1968.
- /4/ BERTRAN, M.: "Approximation of Digital - Filters in one and in two Dimensions". - IEEE Trans. Ac. Sp. and Signal Processing Vol. ASSP-23, No. 5, pp. 438-443, 1975.
- /5/ BLASI, A.: "Estudio Experimental de las Características de los Grupos...". Instituto de Cibernética, Barcelona, 1976.
- /6/ BOWERS, J.C. and SEDORF, S.R.: "SCEPTRE: A Computer Program for Circuit and System Analysis". Prentice-Hall Inc., USA, 1971.
- /7/ CADZOW, J.A.: "Recursive Digital Filter Synthesis via Gradient Based Algorithms" IEEE Trans. Ac. Sp. and Signal Processing Vol. ASSP-24, No. 5, 1976.
- /8/ FRAU, J.A.: "Diseño Interactivo de Filtros digitales". Proyecto de Fin de Carrera, ETSE Telecomunicació, UPB, 1980.

- /9/ HARTLEY, H.O.: "The Modified Gauss-Newton Method for the Fitting of Non-Linear -- Regression Functions by Least Squares". Technometrics, Vol. 3, pp. 269-280, 1961
- /10/ HUBER, R.: "Estudio de la Regulación F/P...". Instituto de Cibernética, Barcelona, 1974.
- /11/ JOVER, J.M.: "MEM User's Guide". ETSE Telecomunicació, Barcelona, 1980.
- /12/ JOVER, J.M.: "Simulador de Sistemas dinámicos orientado a diagramas de bloques". Proyecto de Fin de Carrera, ETSE Telecomunicació, UPB, 1980.
- /13/ LACROIX, M.: "Comparaison des programmes d'analyse générale des circuits électriques". Revue Technique Thompson, Vol. 6, No 1, pp. 17-24, 1974.
- /14/ LINDBERG, E.: "Systems simulation by-- means of ANP3 and NAP2 computer programs". Presented at the Summer School on Circuit Theory, PRAHA, CSSR, 1977.
- /15/ MANDADO, E.: "Sistemas electrónicos digitales". Marcombo, Barcelona, 1973.
- /16/ MARIÑO, J.B., MAYER, A., FIGUEIRAS, A.R.: "A Direct Approximation Technique for Designing Digital Equalizers with Simultaneous Specification of Magnitude and Phase". Proceed. 1978 European Conf. on Circuit-Thry. and Design. Lusanne, pp. 602-606, 1978.
- /17/ MOLICH-PEDERSEN, R.: "Users Manual for the ANP3 circuit analysis program".-- Institute of Circuit Theory and Telecommunication. Technical University of Denmark, 1975.
- /18/ OPPENHEIM, A.V., SCHAFER, R.W.: "Digital Signal Processing". Prentice-Hall- Inc. New Jersey, 1975.
- /19/ PAPOULIS, A.: "Sistemas Digitales y Analógicos...". Marcombo, Barcelona, 1978.
- /20/ RUBNER-PETERSEN, T.: "NAP2 A non-linear analysis program for electronic-- circuits. Users manual". Institute of Circuit Theory and Telecommunication. Technical University of Denmark, 1973.
- /21/ RUBNER-PETERSEN, T.: "NAP2 Application examples". Institute of Circuit Theory and Telecommunication. Technical University of Denmark, 1974.
- /22/ STERLING, M.J.H.: "Power System Control". Peregrinus, 1978.
- /23/ TAUB, H., SCHILLING, D.L.: "Principles of Communication Systems". Mc Graw-Hill, 1971.
- /24/ WILKINSON, J.H.: "Rounding errors in algebraic processes". Prentice-Hall Inc., New Jersey, 1963.
- /25/ WITTY, R.W.: "Dimensional Flowcharting". Software-Practice and Experience, Vol.7, No 5, pp. 553-584, 1977.

#### 10. NOTA

Este artículo es una versión actualizada de un trabajo de igual título que los autores presentaron al I Simposio de Modelado y Simulación organizado por el Comité Español de la IFAC, celebrado en Sevilla en Mayo de 1980.