

UN AMBIENTE COMPUTACIONAL PARA LA MODELIZACIÓN DE SISTEMAS DINÁMICOS NO LINEALES CON BOND GRAPHS

Ernesto Kofman y Sergio Junco

Departamento de Electrónica

Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Universidad Nacional de Rosario

Riobamba 245 Bis - 2000 Rosario - Argentina

Tel.: 0341 - 480 8543 / 7Fax: 0341 - 480 2654

e-mail: ekofman@eie.fceia.unr.edu.ar; sjunco@fceia.unr.edu.ar

Palabras Clave: Computer-aided Modeling and Simulation, Bond Graphs, Nonlinear Dynamics.

Resumen

Se presenta un software de apoyo a la modelización de Sistemas Físicos Dinámicos con Diagramas de Enlaces (Bond Graphs). También admite combinaciones con Diagramas de Bloques, denominadas Diagramas Mixtos. Es una aplicación independiente que funciona en el entorno Windows, y brinda al usuario una interface totalmente gráfica para la edición de dichos diagramas y su almacenamiento en archivos. Tiene incorporadas herramientas de análisis causal y la posibilidad de conversión (totalmente automatizada) de los Diagramas de Enlaces a Diagramas de Bloques Simulink, incluyendo su parametrización completa para su simulación directa. El programa consta de librerías con los elementos más comunes de los Diagramas de Enlaces, de Bloques y Mixtos. Permite además la incorporación de nuevos elementos y/o librerías de manera relativamente sencilla. La aplicación incluye también una serie de modelos confeccionados a modo de ejemplos y un archivo de ayuda con una descripción exhaustiva de la utilización del mismo.

1. Introducción

La técnica de los diagramas de enlaces de energía (BG, por las siglas del inglés *Bond Graphs*) constituye un enfoque estructurado de modelización de sistemas dinámicos con notación unificada para fenómenos de los diferentes dominios de la Física (Karnopp and Rosenberg, 1983; Junco, 1986). Los BG en esencia capturan los fenómenos de intercambio de energía, cuantificándolos instantáneamente según la potencia en juego en el sistema. Entre sus principales características y ventajas, se cuentan las de ser un lenguaje gráfico, acausal en principio, y de formulación independiente de las alinealidades presentes en el sistema. Su característica gráfica permite articularlos directamente con Diagramas de Bloques (DB), permitiendo así capturar también fenómenos que no involucran intercambio de energía sino sólo interrelación de señales en el sistema, y de tal manera construir combinaciones de BG con DB, designadas aquí como Diagramas Mixtos (DM).

Siendo simple codificar en un BG las relaciones causales entre las variables que describen al sistema, se pueden derivar inmediatamente modelos causales tales como DB o Ecuaciones de Estado (EE), de una forma completamente algoritmizable. Por ende, en un BG (o DM) se tienen simultáneamente las estructuras física de intercambio de potencia y la matemática del sistema, facilitándose así tanto la ejecución de procedimientos formales (p. ej. simulación del modelo), como la concreción de análisis cualitativos (estabilidad, controlabilidad, etc) y el desarrollo de técnicas de diseño y/o control de los sistemas (Junco, 1993; Cellier and Granda, 1993 / 95 / 97 / 99).

La unicidad o genericidad de representación para distintos dominios físicos –derivadas de la propiedad de analogía y de la concepción energética en que se basan los BG–, y la posibilidad de encapsular modelos y conectarlos a otros (BG y/o DB) a través de puertos externos, permiten desarrollar una estrategia de modelado jerárquica y orientada a objetos (Cellier, 1992; Cellier et al., 1996).

Las propiedades mencionadas convierten al método BG en una herramienta valiosa para la representación de sistemas complejos, permitiendo la informatización de la modelización con BG, y la manipulación de los modelos computacionales resultantes.

En este trabajo se presenta un software interactivo de modelización de sistemas dinámicos con BG, desarrollado en base a las propiedades anteriores. El programa está desarrollado en Visual Basic con versiones para 16 y 32 bits. Funciona bajo entornos Windows, 3.1x y posteriores, lo que permite el aprovechamiento máximo de las principales características de las aplicaciones Windows afines a las necesidades de la visualización y manipulación de un formalismo gráfico (implementación multidocumento para trabajar simultáneamente con diferentes ventanas gráficas; uso de todos los métodos gráficos de Windows para la edición: cortar, copiar, pegar, mover, etc.). Funcionalmente consiste de una estructura abierta de librerías con componentes correspondientes a las representaciones BG, DB y DM, con los cuales el usuario puede construir su modelo. Todos los componentes del diagrama

resultante (BG o DM) pueden ser parametrizados, independientemente de la causalización, lo cual por una parte facilita el tratamiento de sistemas implícitos, y por otra confiere la propiedad de reusabilidad a los modelos generados con este software, aún como subsistemas en otros modelos que fueren situaciones causales distintas. Una vez formulado el BG básico acausal el usuario tiene las opciones de causalizar manual, automática, o semiautomáticamente, paso previo a la posibilidad de convertir automáticamente los modelos generados a modelos Simulink de todas las versiones, con lo cual se puede explotar toda la potencia de simulación y análisis del entorno Matlab.

El trabajo está organizado de la siguiente manera. En la Sección 2 se reseñan los principios de la modelización con BG. En la Sección 3 se hace una descripción funcional del software. En la Sección 4 se describen sus principales aspectos informáticos: estructura de datos, algoritmos, estructura del programa. En la Sección 5 se presentan ejemplos de uso y algunas aplicaciones que ilustran las cualidades del software. Por último, en la Sección 6 se extraen algunas conclusiones y se indican los próximos pasos de ampliación del entorno.

2. La modelización con bond graphs

Al sólo fin de facilitar la exposición y la comprensión de las características del software desarrollado se hace aquí un resumen ad-hoc de los fundamentos del método. Por teoría, métodos y aplicaciones más avanzados y generales ver las referencias citadas. La más extensa fuente de información sobre el tema se encuentra en la página web <http://www.ece.arizona.edu/~cellier/bg.html>.

2.1 Estructura y componentes del sistema

Un sistema físico dinámico puede pensarse como una porción delimitada del Universo en la que a cada instante se verifica un intercambio (y balance) de potencia entre todos los elementos que lo componen. El enfoque *bondgraph* distingue clases elementales de componentes en función del procesamiento que éstos hacen de la energía. En su repertorio básico de elementos hay *fuentes* (modelan la incorporación al Sistema de los elementos del resto del Universo con acciones relevantes sobre aquel), *almacenadores* y *disipadores* de energía. Hay además un conjunto de elementos que modelan la distribución instantánea de la energía entre aquellos: los *enlaces* (*bonds*), encargados de transportar la potencia; y los *conservadores de potencia* (*vínculos*, *acopladores*, *convertidores*), que caracterizan la estructura en la que los elementos del primer grupo se reúnen para constituir el sistema.

Independientemente de su dominio físico, dos componentes son modelados por el mismo elemento *bondgraph* si procesan la energía de la misma forma.

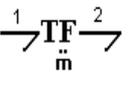
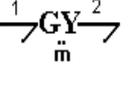
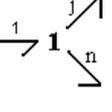
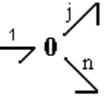
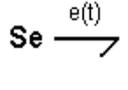
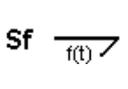
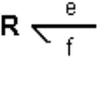
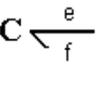
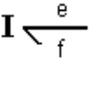
Este hecho motiva la generalización de las variables físicas a cuatro tipos: *esfuerzo e* y *flujo f* (*variables de potencia*, su producto da potencia), e *impulso p* y *desplazamiento q* (*variables de energía*, son las respectivas integrales de las dos anteriores; la energía en los almacenadores es una función de ellas). La *Tabla 1* ilustra una generalización posible, con las correspondientes variables en cada dominio físico.

Tabla 1: Variables Generalizadas

Dominio físico	Variables de Potencia		Variables de Energía	
	Esfuerzo e	Flujo f	Momento p	Desplaz. q
Traslación	Fuerza	velocidad	Impulso	Desplaz.
Rotación	Torque	velocidad angular	Momento angular	Ángulo
Fluido-dinámica	Presión	caudal	Impulso del fluido	Volumen
Electro-magnetismo	Tensión	corriente	Flujo magnético	Carga eléctrica
Química	Potencial químico	flujo molar		número de moles
Termo-dinámica	Temp. absoluta	flujo de entropía		entropía

Con *fuerza* y *tensión* como casos particulares del *esfuerzo generalizado* se corresponden las analogías *energía cinética ~ energía del campo magnético*, o *inercia ~ bobina*; y *energía potencial ~ energía del campo eléctrico*, o *resorte ~ capacitor*. Con otra homologación, p. ej., la de *fuerza* y *corriente*, hubiera resultado otra generalización, también admisible. La *Tabla 2* resume el conjunto de *elementos básicos* del lenguaje de modelización *bondgraph*. Hay dos tipos de *fuentes*, de *esfuerzo* y de *flujo*, según cual sea su variable independiente; y dos tipos de *almacenadores*, los *capacitores "C"* (generalización del almacenador de energía potencial) y las *inercias "I"* (generalización del almacenador de energía cinética). Hay dos tipos de *vínculos*, los *"0"* y los *"1"*, ambos multipuerta (admiten varios enlaces adyacentes); *representan* las *estructuras* mas simples: *esfuerzo común* y *flujo común*, respectivamente. Así, todos los enlaces unidos a un vínculo *"0"* (*"1"*) tienen asociada la misma variable de *esfuerzo* (*flujo*). En el dominio eléctrico, un vínculo *"1"* (corriente común) representa la estructura de un circuito serie, mientras que vínculo *"0"* (tensión común) representa la de un paralelo.

Tabla 2: Elementos básicos de los Diagramas de Enlace

Fenómeno energético	Clase	Elementos				Ejemplos técnicos	
		Símbolo	Nombre	Potencia Energía	Relación Constitutiva		
Conservación o Continuidad de Potencia	Acopladores		Enlace	$P_{transferida} = e \cdot f$		Cardán, Línea eléctrica.	
	Acopladores Convertidores	Multipuertas		Transformador	$P_{absorbida} =$ $P_1 - P_2 = 0$	$e_2 - m \cdot e_1 = 0$ $f_1 - m \cdot f_2 = 0$	Piñón-cremallera Pistón-cilindro
				Girador	$P_{absorbida} =$ $P_1 - P_2 = 0$	$f_2 - m \cdot e_1 = 0$ $f_1 - m \cdot e_2 = 0$	Giróscopo Conversión electro-mecánica
	Vínculos	Multipuertas		Vínculo uno	$P_{absorbida} =$ $\sum_{j=1}^n \pm P_j = 0$	$f_1 = \dots = f_n = f$ $\sum_{j=1}^n \pm e_j = 0$	Velocidad común Circuito serie
				Vínculo cero	$P_{absorbida} =$ $\sum_{j=1}^n \pm P_j = 0$	$e_1 = \dots = e_n = e$ $\sum_{j=1}^n \pm f_j = 0$	Serie mecánico Circuito paralelo
	Generación de Energía	Fuentes	Monopuertas		Fuente de esfuerzo	$P_{entregada} =$ $e(t) \cdot f$	$e(t)$ independiente
				Fuente de flujo	$P_{entregada} =$ $f \cdot e(t)$	$f(t)$ independiente	Fuente de corriente Bomba hidráulica
Disipación de energía	Disipadores	Monopuertas		Resistor	$P_{disipada} =$ $f \cdot e$	$f(e, f) = 0$	Rozamiento Electro-resistor
Conservación de Energía	Almacenadores			Capacitor	$e_{almac} =$ $e_0 + \int_{q_0}^q e(q) dq$	$f - \frac{dq}{dt} = 0$ $f(e, q) = 0$	Elasticidad materiales Tanque de agua
				Inercia	$e_{almac} =$ $e_0 + \int_{p_0}^p f(p) dp$	$e - \frac{dp}{dt} = 0$ $f(f, p) = 0$	Inductancia Inercia mecánica

Un vínculo "I", al conservar potencia, tiene una propiedad adicional a la del flujo común, y es que la sumatoria de los esfuerzos asociados al mismo a través de los enlaces es nula (al ser nula la sumatoria de potencias y el flujo común a todos los enlaces, es simple ver que la suma de los esfuerzos es también nula). Lo mismo pasa con un vínculo "O" respecto a la sumatoria de los flujos. Un vínculo "I" expresa la generalización de la Ley de Kirchhoff de Tensiones, y un vínculo "O", la generalización de la Ley de Kirchhoff de Corrientes.

Estructuras complejas requieren combinaciones de vínculos "I" y "O" y de otro tipo de elementos estructurales, los *acopladores* y los *convertidores*, entre los que se cuentan los *transformadores (TF)*, los *giradores (GY)*, y, por supuesto, los *enlaces*. Los *transformadores* y *giradores* son bipuertas, es decir, tienen conectados solamente dos enlaces, con lo que tienen asociados dos pares de variables de potencia. El transformador vincula entre sí las variables del mismo tipo (esfuerzo con esfuerzo y flujo con flujo) con una relación estática y de manera tal que se conserve la potencia (esto implica que los productos de las variables de esfuerzo y flujo sean iguales en los dos enlaces). Algo similar realiza el girador, sólo que vincula las variables de manera cruzada (esfuerzo con flujo).

La Fig. 1 muestra un *enlace* o *bond*, al cual se asocian las dos variables generalizadas de potencia e y f , cuyo producto expresa la potencia P transmitida instantáneamente en el sentido del arpón o semiflecha (cuando es positiva).

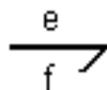


Fig. 1: Enlace. Transporta potencia $P = e \cdot f$ a la derecha

El BG de la Fig. 3, que modela al circuito de la Fig. 2a, provee una ilustración simple de las ideas vertidas de la Tabla 2. Con las analogías de la Tabla 1 es muy fácil mostrar que el mismo BG modela al sistema mecánico de la Fig. 2b. Las variables análogas que determinan la estructura son: en el vínculo "I", $f \sim I \sim V$; en el vínculo "O", $e \sim U_C \sim F_R$.

2.2 Causalidad

Hasta aquí los componentes del sistema han sido tratados como procesadores de potencia. Para tratarlos como procesadores de señal es necesario asignar *relaciones causales* entre las variables asociadas a todos los elementos de un BG. Dado un elemento de un BG, la *causalidad* discrimina a las variables de potencia de los enlaces adyacentes en dos grupos: las consideradas *entradas* al elemento (*datos, causas*) y las *salidas* (*resultados, efectos*). En el caso de

elementos con variables internas, esto a su vez induce relaciones causales de las variables externas (de potencia) con las variables internas del elemento.

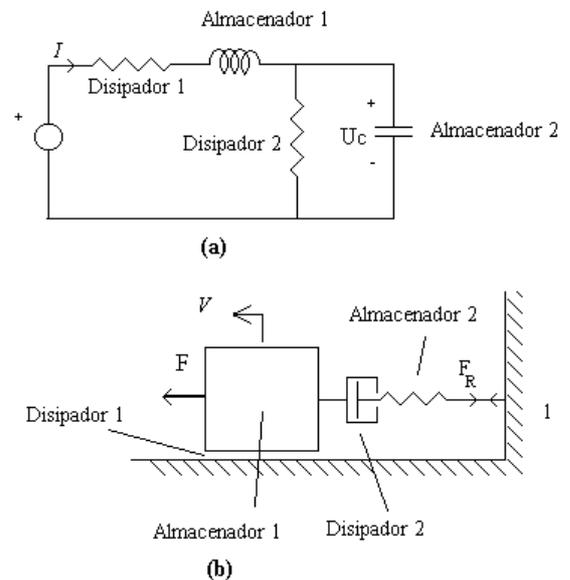


Fig. 2: Dos simples sistemas análogos

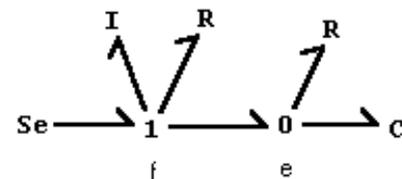


Fig. 3: BG de los sistemas de Fig. 2

En un enlace pueden darse las dos situaciones básicas de la Fig. 4, lo que permite codificarlas poniendo una *barra causal* en el extremo hacia el cual va dirigido el esfuerzo, en lugar de poner las flechas típicas de los DB.

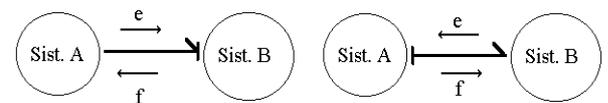


Fig. 4: Causalidad

El procedimiento de la Fig. 5 para asignar causalidad a un BG (SCAP: *Sequential Causality Assignment Procedure*, Karnopp and Rosenberg, 1983) respeta las condiciones causales detalladas en la Tabla 3: las *fuentes* sólo pueden fijar al sistema sus variables independientes; teniendo relaciones fijas entre sus variables de potencia, los *elementos estructurales* sólo pueden asumir un conjunto restringido de todas las configuraciones causales de sus enlaces adyacentes; los *almacenadores* pueden asumir cualquier situación causal, pero la *integral es preferencial*; los *disipadores* son de causalidad indiferente.

Tabla 3: Condiciones causales. Siendo la causalidad completamente independiente del flujo de potencia, la semiflecha o arpón no se dibujó en ningún enlace.

Tipo	Notación	
	Diagrama de Enlaces	Diagrama de Bloques
Necesario	$S_e \begin{array}{c} e(t) \\ \hline \end{array}$	$e(t) \longrightarrow$
	$S_f \begin{array}{c} \hline f(t) \end{array}$	$f(t) \longrightarrow$
Restringido	$\begin{array}{c} 1 \\ \hline TF \\ \hline m \\ \hline 2 \end{array}$	
	$\begin{array}{c} 1 \\ \hline TF \\ \hline m \\ \hline 2 \end{array}$	
	$\begin{array}{c} 1 \\ \hline GY \\ \hline m \\ \hline 2 \end{array}$	
	$\begin{array}{c} 1 \\ \hline GY \\ \hline m \\ \hline 2 \end{array}$	
	$\begin{array}{c} 1 \\ \hline 1 \\ \hline j \\ \hline n \end{array}$	
	$\begin{array}{c} 1 \\ \hline 0 \\ \hline j \\ \hline n \end{array}$	

Tabla 3: Condiciones causales (continuación)

Tipo	Notación		
	Diagrama de Enlaces	Diagrama de Bloques	
Preferencial	Integral	$C \begin{array}{c} e \\ \hline f \end{array}$	
		$I \begin{array}{c} e \\ \hline f \end{array}$	
	Derivativa	$C \begin{array}{c} e \\ \hline f \end{array}$	
		$I \begin{array}{c} e \\ \hline f \end{array}$	
Arbitrario	$R \begin{array}{c} e \\ \hline f \end{array}$		
	$R \begin{array}{c} e \\ \hline f \end{array}$		

En base a estas condiciones el SCAP se inicia fijando causalidad a una fuente, para luego propagarla por los elementos estructurales si es posible. Esto se repite hasta que se agotan las fuentes. Luego se retoma la misma iteración, pero comenzando por un almacenador de energía, asignándole la causalidad preferencial. Por último, si el BG no quedó plenamente causalizado, se repite la iteración empezando por algún disipador. Si se llega a esta última etapa con seguridad hay lazos algebraicos en el modelo. El método asegura el máximo número posible de almacenadores con causalidad integral en el modelo. Sus variables de energía son la típica elección de variables de estado, pero otras son posibles. Las ecuaciones de estado y el diagrama de bloques quedan automáticamente definidos. Se pueden dar procedimientos sistemáticos para leerlos del BG causalizado.

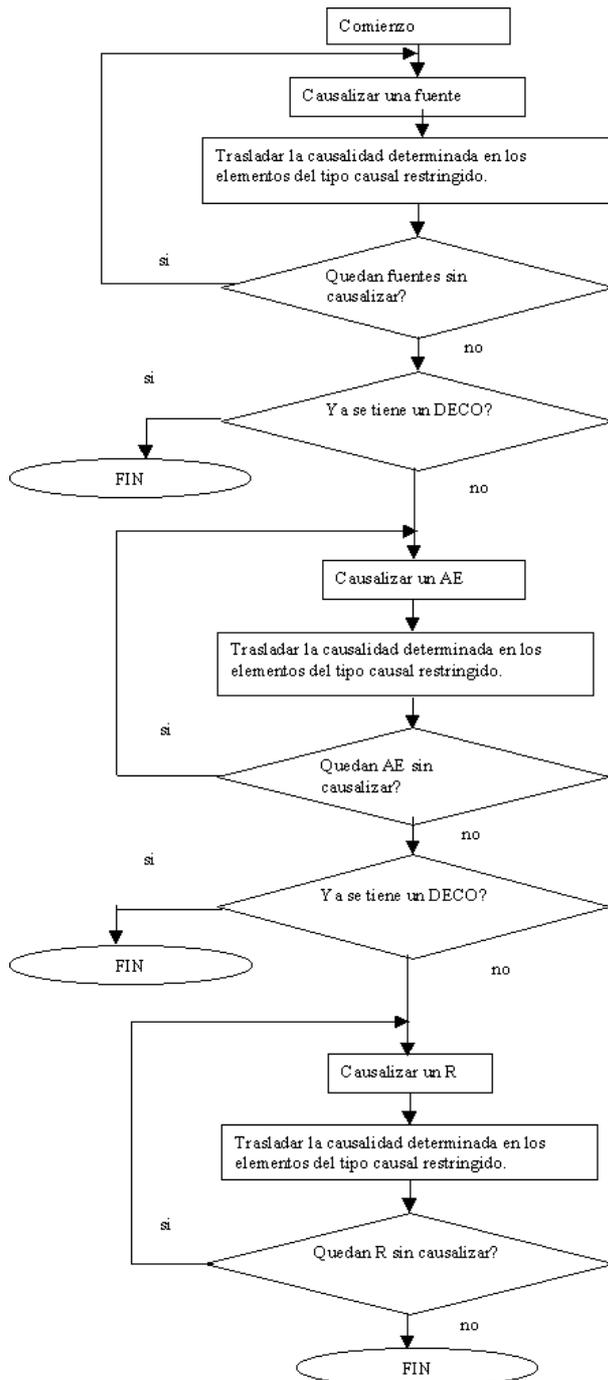
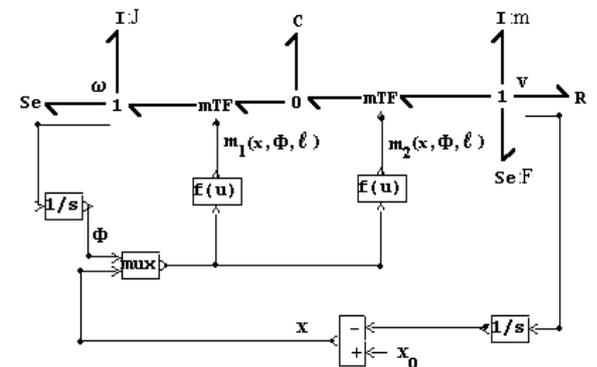
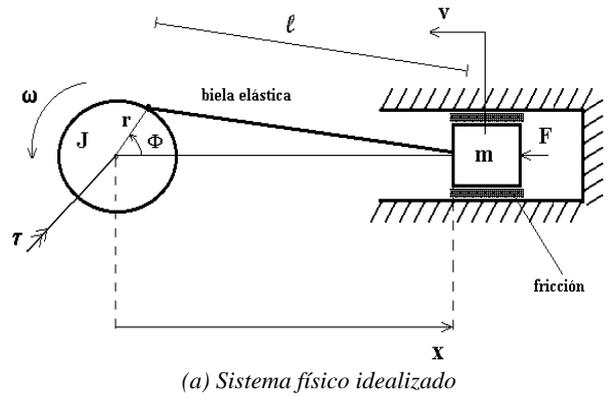


Fig.5: Procedimiento para asignar causalidad (SCAP)

2.3 Diagramas mixtos

Se designará de este modo a la combinación de BG con DB. En un BG ésto puede ser necesario para representar la acción de ciertas señales sobre algunos de sus elementos, sin que en ella medie transferencia de potencia. Esto es típico en sistemas mecánicos con geometría variable (Fig. 6, a y b), y en la modulación por el flujo de excitación de la transferencia de potencia en la interconversión electromagnético-mecánica de energía en las máquinas eléctricas, entre muchos ejemplos citables. La señal se inyecta a algún elemento BG que se califica como *modulado* para

distinguir el hecho de que ahora tiene un parámetro variable; en la notación suele anteponerse una letra *M* (o *m*). El comportamiento energético es igual al de su par original. Se tienen así *fuentes moduladas* o *dependientes*, *transformadores* y *giradores modulados*, y *resistores*, *capacitores* e *inercias moduladas*. Un DM también puede resultar de la simple conexión de un BG con un DB ya existente, ambos modelando partes diferentes de un mismo sistema. Por convención cuando se toma información (señal) de un BG se lo hace de un vínculo, ya que en su versión en señal éstos contienen puntos de derivación (ver DBs correspondientes en la Tabla 2). De los vínculos "0" se toma la información del esfuerzo común, y de los "1", la del flujo común.



$$m_1 = \frac{x - r \cdot \cos(\Phi)}{\ell}; \quad m_2 = \frac{r \cdot x \cdot \sin(\Phi)}{\ell}$$

$$\ell^2 = r^2 + x^2 - 2 \cdot r \cdot x \cdot \cos(\Phi)$$

(b) Bond Graph

Fig.6: Interconversión rototraslacional de movimiento

3. Operación de PowerDynaMo

Power DynaMo es una aplicación que corre en ambientes Windows. Existen versiones de 16 bits (ejecutables bajo Windows 3.x y posteriores) y de 32 bits (para Windows 95 y posteriores). De manera interactiva el usuario puede editar Diagramas Mixtos, parametrizarlos completamente, asignarles causalidad, y convertir el código resultante en código Simulink a los fines de la simulación directa.

La Fig. 7 muestra la *Ventana Principal* de *Power DynaMo*, desde la cual se pueden abrir modelos existentes, modelos nuevos, y librerías. También puede accederse a la *Ayuda del Programa*, propiedad compartida por las ventanas de todos los modelos.

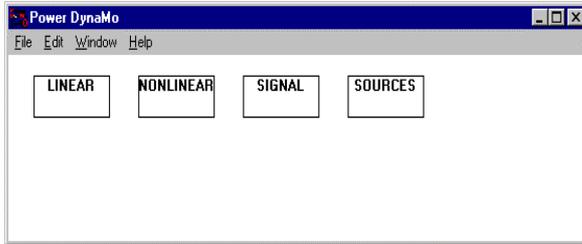


Fig. 7: La *Ventana Principal*

Cualquiera de las cuatro *librerías* disponibles se abre haciendo doble click sobre ella con el mouse (Fig. 8). Las librerías así abiertas no pueden modificarse. Sin embargo abriéndolas como si fueran un modelo pueden agregarse y quitarse elementos a voluntad. Pueden también agregarse nuevos elementos modificando el código de la librería (archivos con extensión *.lib* en el *directorio Library*). Estos son archivos de texto, que pueden abrirse con cualquier editor de texto. Los modelos se construyen arrastrando elementos desde las *Ventanas de Librería* hacia las *Ventanas de Modelo* (Fig.9), que pueden abrirse desde el Menú desplegable *File* de la *Ventana Principal*, copiando y luego pegando partes desde otros modelos, dibujando líneas de señal y enlaces de potencia, etc. *Power DynaMo* es una aplicación multidocumento, por lo que permite trabajar simultáneamente con varios modelos.

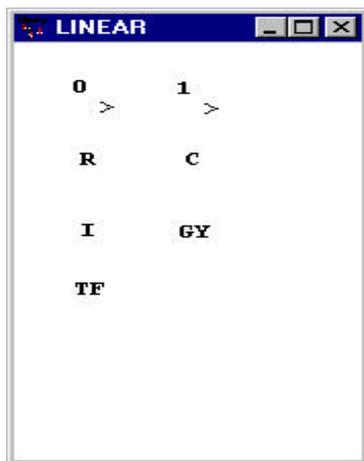


Fig. 8: *Ventana de Librería:* Elementos BG Lineales

Desde el menú *File* de una *Ventana de Modelo* se puede grabar, imprimir y cerrar el modelo. Desde el menú *Edit*, es posible seleccionar todo, copiar (al portapapeles), pegar (desde el portapapeles), eliminar la selección y cortar la selección. En el menú *Options* se puede elegir ver (o no) los parámetros de los

elementos de potencia, y la versión de Simulink hacia la cual se van a convertir los modelos (actualmente se puede elegir entre las versiones 1.3; 2.09 y 2.2). En la opción *Run* se puede generar el modelo Simulink, y asignar y chequear la causalidad del modelo en forma automática. Desde el menú *Help* se accede a la ayuda.

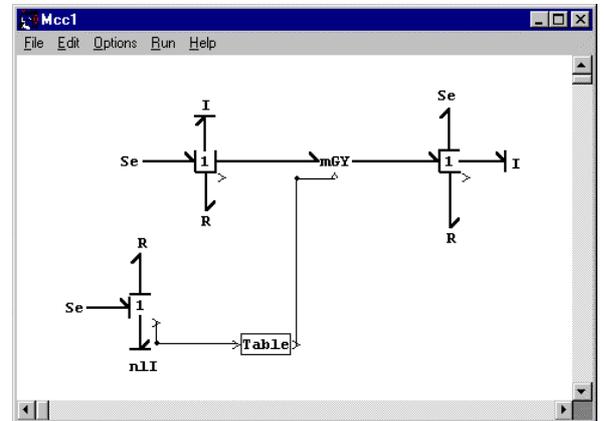


Fig. 9: Ejemplo de una *Ventana de Modelo*. MCC con excitación independiente.

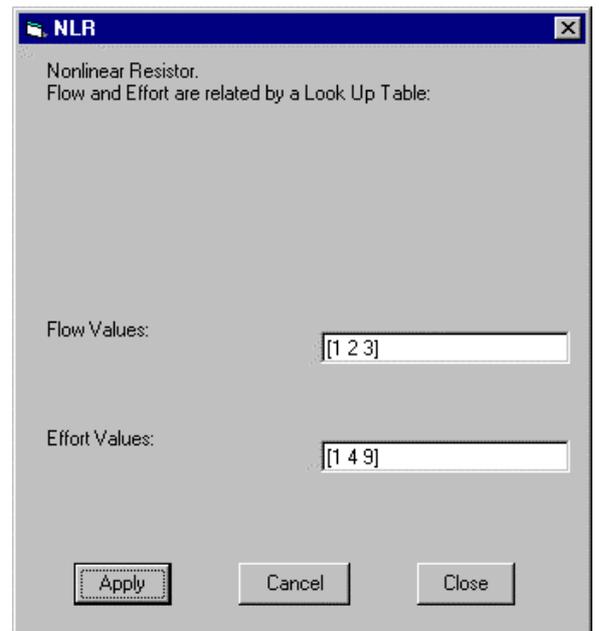


Fig. 10: Una *Ventana de Parámetros*.

Los enlaces (bonds) pueden dibujarse clickeando cerca de un elemento y arrastrando hasta otro. La causalidad también puede asignarse en forma manual haciendo doble click sobre el enlace en cuestión. Las líneas de señal pueden dibujarse “copiando” los conectores de señal (“>”, “<”, “v”, “^” o “•”) con el botón derecho y arrastrando hacia un conector de entrada. Una porción de la ventana se selecciona clickeando sobre un punto y arrastrando; resultarán seleccionados todos los elementos (en rojo) que queden en el interior del rectángulo punteado. Una

selección puede copiarse clickeando con el botón derecho sobre un elemento de la misma y arrastrando. Los parámetros de un elemento se cambian desde su *Ventana de Parámetros* (Fig. 10), que se abre con doble click sobre el elemento.

4. Descripción interna de PowerDynaMo

4.1 Representación interna de los modelos

Los modelos son representados internamente por una colección de objetos. Hay básicamente dos tipos de objetos: *elementos* y *líneas*. Los *elementos* se visualizan a través de un control “*Label*” asociado. Están ordenados en un arreglo, de modo que cada objeto puede ser unívocamente identificado por un número (numero de objeto), correspondiente al índice del arreglo. Sus principales propiedades son:

- El tipo (*type*), una cadena de 3 caracteres que identifica si es un uno, cero, inercia, etc.
- La posición en la ventana, dada por dos variables (*left*, *top*)
- El número máximo de enlaces que pueden ser conectados. (1 para los monpuertas, 8 para los vínculos “*I*” y “*O*”; 2 para los transformadores y giradores y ninguno para los elementos que manejan exclusivamente señal).
- El valor de los parámetros asociados.
- El “*Caption*”, i.e., las siglas visibles del elemento (**I** para una inercia, **GY** para un girador, etc.)
- Los números de conectores de señal de entrada y de salida asociados.
- Una variable booleana que indica si está o no actualmente seleccionado.
- Para los conectores de señal, se utiliza también el objeto al cual pertenece y la posición dentro de él.

Las *líneas* se visualizan a través de uno o más controles “*Line*” asociados. Sus propiedades son:

- Los 2 *elementos* que vincula (*obj1* y *obj2*).
- Una variable denominada “*causa*” que vale 0 si el enlace no está causalizado, 1 si se impone esfuerzo en el sentido de transmisión de potencia, 2 en el caso contrario, y 4 si se trata de una línea de señal.
- Una variable booleana que indica si está o no actualmente seleccionado.

Los *archivos de almacenamiento de los modelos* (extensión *.bgf*) están por lo tanto constituidos por un conjunto de órdenes que indican si se va a insertar un *elemento* o una *línea*, y cuales son las propiedades de dicho objeto. Las *librerías* son simplemente archivos de modelo con extensión *.lib*, ubicados en el subdirectorio *Library* de la aplicación.

4.2. Las rutinas de asignación de causalidad

Las rutinas de asignación automática de causalidad se basan casi totalmente en el procedimiento de asignación de causalidad descripto

en la Fig. 4. La rutina nunca cambia la causalidad asignada previamente a un enlace. En cambio, detecta errores causales debidos tanto a un posible conflicto estructural del modelo como a la mala asignación previa de causalidad por parte del usuario. En el algoritmo SCAP, hay básicamente dos acciones: una de asignar causalidad a un elemento acorde a su prioridad causal, y otra de extender esa causalidad a través de los vínculos, transformadores y giradores cuando sea posible. Esta segunda acción puede ejecutarse directamente mediante la opción *Extend Causality* del menú *Run*.

4.3. Las rutinas de conversión

Las rutinas de conversión a diagramas de bloques Simulink se basan en la existencia de un DB para cada elemento *bondgraph* causalizado, y por lo tanto de un DB para cada Diagrama de Enlaces Causalizado o DM. En base a esta propiedad se generaron archivos de texto (con extensión *.COD*) que tienen el código del DB Simulink correspondiente a cada elemento incluido en las librerías de la aplicación. Estos archivos se encuentran en diferentes directorios acorde a la versión de Simulink seleccionada para la conversión.

Los archivos *.COD* (archivos de código) contienen algunos parámetros del DB reemplazados por una expresión entre numerales (“#”). Esta expresión es el nombre de una variable que será reemplazada por el programa al generar el código de modo que los parámetros del objeto correspondiente pasen al código. Al final de cada archivo de código hay una serie de órdenes que indican al programa cuales son los nombres de los bloques desde los cuales sale o entra esfuerzo, flujo o señal. Esta información es utilizada cuando se adicionan las líneas al DB.

El procedimiento de conversión comienza leyendo el archivo *STA.COD* del directorio correspondiente a la versión de Simulink. Luego, para cada objeto de tipo *elemento* del modelo, lee el archivo de código correspondiente a su causalidad. (Hay 2 diferentes archivos para cada elemento según la causalidad). Una vez leído un archivo, se reemplazan las expresiones entre numerales y el código resultante se agrega al código completo.

Luego de leer los archivos de todos los elementos se agregan al código general las líneas asociadas a las señales *e* y *f* de cada enlace, según la información de sus archivos de código (para las fuentes se agrega sólo una). Para la generación de cada línea, se utiliza el archivo de código *LIN.COD*. En las versiones 2.09 y 2.2 de Simulink a cada línea generada se le asigna una trayectoria de modo tal que el DB Simulink sea relativamente ordenado (la versión 1.3 de Simulink lo ordena sola).

Por último, se agrega al código el archivo *END.COD* que contiene el final de código del DB Simulink. Una vez generado, el código se guarda en un archivo Simulink (extensión .m) seleccionado por el usuario.

5. Ejemplo de Aplicación

Un aplicación técnica que demuestra la eficacia de la utilización de los Bond Graphs y verifica el correcto funcionamiento del programa lo constituye el estudio de la dinámica del movimiento transversal de un *Puente Grúa* (Fig. 11). Se supuso que de un cuerpo de masa "M" (carro) pende un punto material de masa "m" (carga), sostenido por una barra elástica, de masa despreciable. Un MCC con excitación constante actúa sobre el carro. Hay fricción asociada al movimiento de ambas masas. Se supone que hay viento actuando sobre la carga. Sobre este modelo se implementa un simple control proporcional de posición. El modelo DM completo se muestra en la Fig. 12.

Tras la ejecución de las rutinas de conversión de Power DynaMo se obtuvo el DB Simulink de la Fig. 13. La simple inspección de las Figuras 12 y 13 permite comprobar la mayor simplicidad del DM frente al DB.

Habiéndose parametrizado completamente el DM, se simuló el DB resultante sin necesidad de ninguna otra acción más que la simple definición del tiempo total de simulación y del algoritmo a utilizar.

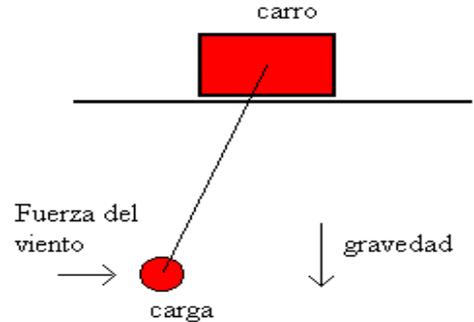


Fig. 11: Esquema del Puente Grúa

Se simuló un cambio de 5 metros en la posición de la carga a partir de una situación de reposo, con una referencia en escalón, bajo la presencia de una fuerza constante del viento. Los resultados se muestran en la Fig. 14, (a), (b) y (c).

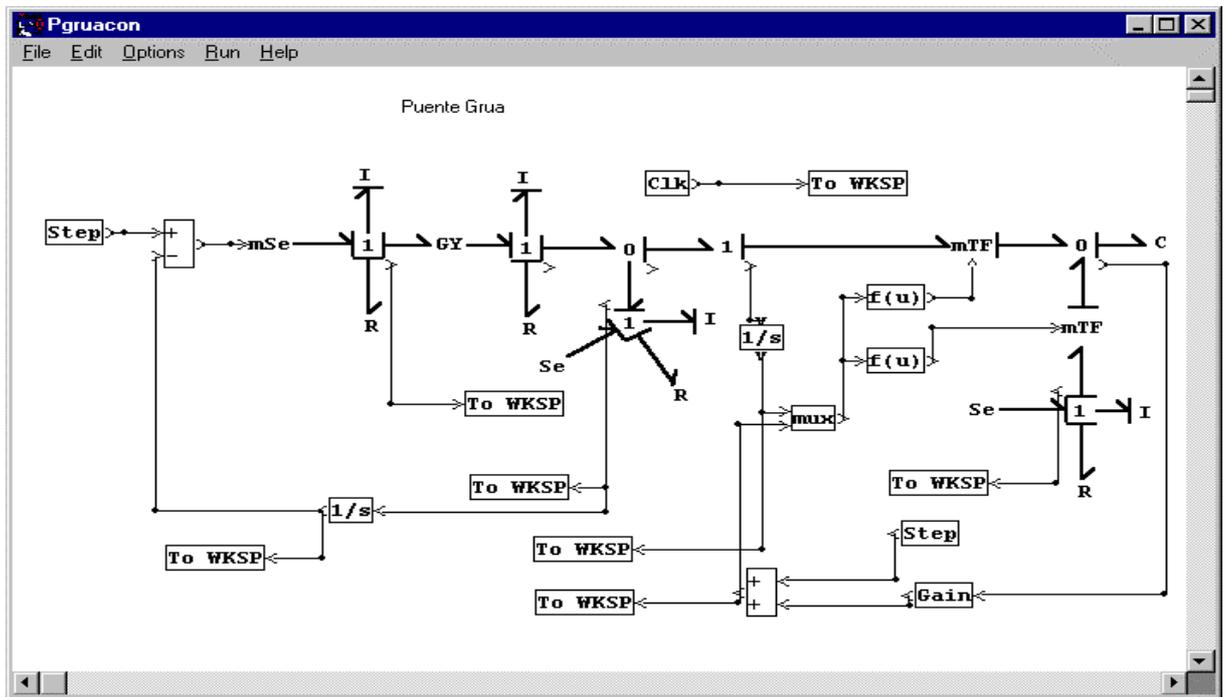


Fig.12: Modelo Power DynaMo del Puente Grúa con control (elemental) de posición

6. Conclusiones

Se presentó el software *PowerDynaMo*, una aplicación bajo Windows, independiente, programado en Visual Basic, que sirve de apoyo al modelado gráfico interactivo de sistemas físicos con DE y DM. Permite la edición, parametrización, causalización,

almacenamiento y conversión a DB Simulink de Diagramas de Enlaces o Mixtos –todo de manera interactiva. Los procedimientos de *PowerDynaMo* se basan en los métodos sistemáticos y estructurados de modelado disponibles para Diagramas de Enlaces. El programa posee una amigabilidad y facilidad de

operación superior en muchos aspectos a otros programas actualmente existentes, inclusive de nivel comercial. A fin de convertirlo en una herramienta más potente y sintética se trabaja actualmente en su complementación con librerías de elementos compactos correspondientes a Diagramas de Enlaces vectoriales y de potencia instantánea compleja, y de módulos estructurales de aparición recurrente en sistemas de distintos dominios físico-técnicos. Se trabaja también en la incorporación a *PowerDynaMo* de librerías de modelos de sistemas comunes en la técnica (máquinas eléctricas, actuadores hidráulicos, convertidores electrónicos de potencia, etc.), de programas que permitan la manipulación simbólica de la información contenida en sus archivos de modelos, y de rutinas numéricas propias de simulación para hacer autocontenido el ciclo de modelado y simulación.

Referencias

- Cellier, F. 1992. "Hierarchical non-linear bond graph: A unified methodology for modelling complex physical systems", *Simulation*, Vol 58, No. 4, pp. 230-248.
- Cellier, F., H. Elmqvist, M. Otter, and W. Levine. 1995: "Determining Models", in *The Control Handbook (Levine et al., Eds.)*, Ch. 7, pp. 99-112, IEEE-CRC Press.
- Cellier, F. and J. Granda (Eds.). 1993/95/97/99. *Proceedings of the International Conferences on Bond Graph Modeling and Simulation (ICBGM)*, Society for Computer Simulation International Simulation Series.
- Junco, Sergio. 1986: *Los Diagramas de Enlaces y un Eficaz Método Estructurado de Modelado Analítico de Sistemas Dinámicos*. Anales del 2do. Congreso Latinoamericano de Control Automático, Buenos Aires, Octubre 1986.
- Junco, Sergio. 1993. *Stability Analysis and Stabilizing Control Synthesis via Lyapunov's Second Method Directly on Bond Graphs on Nonlinear Systems*. Proceedings of *IECON'93*, (Maui, HII, Nov. 17-20), pp. 2065-2069.
- Karnopp, D. and R. Rosenberg. 1983. *Introduction to Physical System Dynamics*. N.Y: McGraw-Hill.

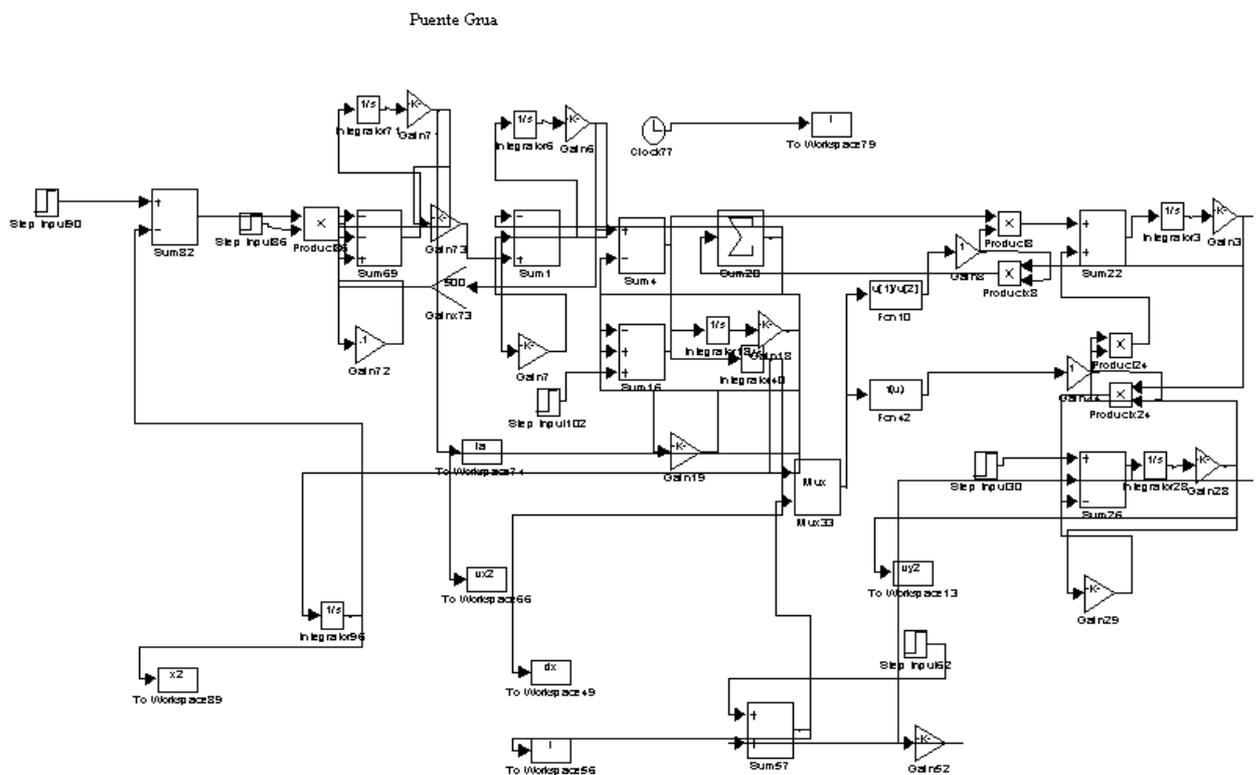


Fig. 13: Modelo Simulink del Puente Grúa con control de posición

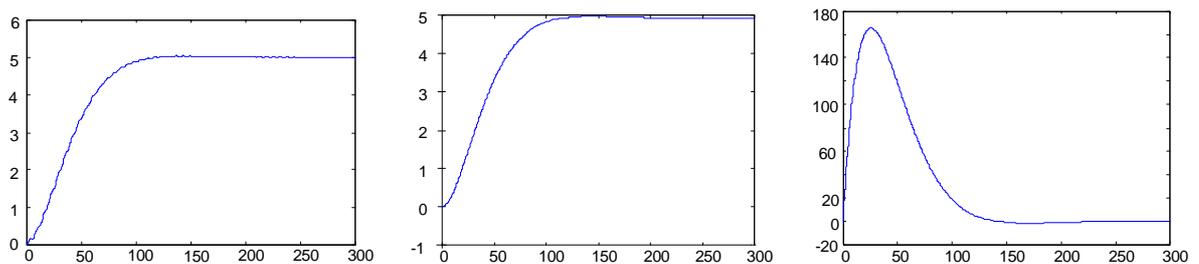


Fig. 14: Posición horizontal (a) de la carga: x_2 vs. t , (b) del carro: x vs. t . (c) Corriente de armadura: ia vs. t .