

---

## MODELIZACIÓN Y SIMULACIÓN DE UN SISTEMA SMIB UTILIZANDO SOFTWARE LIBRE

**J. C. VASCHETTI \***  
GECaP – UTN - FRC  
Argentina

**J. C. GOMEZ**  
GECaP – UTN - FRC  
Argentina

**C. IBARLUCEA**  
GECaP – UTN - FRC  
Argentina

**C. COYOS**  
GECaP – UTN - FRC  
Argentina

***Resumen** - En el presente trabajo se muestra la Modelización y Simulación de un Sistema conformado por un Generador Síncrono acoplado a una Barra Infinita (SMIB) utilizando el paquete Scilab / Xcos® de acceso libre y puesto a disposición de los usuarios por el consorcio Scilab®. Los resultados obtenidos son comparados con el ejemplo SMIB provisto en el paquete Matlab / Simulink®. Se muestra en detalle la programación de la función GENERIC la cual permite modelar y simular el Espacio de Estado del sistema dinámico bajo estudio. Se puede concluir que la funcionalidad entregada por la dupla Scilab/Xcos presenta idénticas características a las entregadas por Matlab/Simulink, existiendo otras consideraciones a tener en cuenta a la hora de comparar prestaciones, las cuales también son analizadas en la presente investigación.*

***Palabras clave:** Modelización y Simulación - Sistema de Potencia - Espacio de Estado - Software Libre.*

### 1. INTRODUCCIÓN

En Sistemas Eléctricos de Potencia se utilizan para su diseño y estudio paquetes de simulación con distintas características. Se pueden clasificar según sus prestaciones [1] en los siguientes grupos:

1. Paquetes comerciales con capacidad para simular transitorios electromagnéticos y electromecánicos, PSCAD/EMTDC®, ATP/EMTP®, DIGSILENT®, NEPLAN®.
2. Paquetes comerciales con capacidad para simular transitorios electromecánicos, PSS/E®, EUROSTAG®.
3. Paquetes comerciales de propósito general con Toolbox para sistemas de potencia, Matlab/Simulink/SimPowerSystems®.
4. Paquetes de libre acceso con o sin Toolbox para sistemas de potencia, Scilab/Xcos®, GNU Octave®.

Si bien esta clasificación es acotada tanto en su lista como en sus características [2] es bastante representativa de lo que hoy existe en el mercado. Los paquetes de propósitos específicos como los mencionados en (1.) y (2.) tienen aplicaciones muy dedicadas al problema en cuestión, tal es el caso del cálculo de cortocircuito, mientras que los de propósito general, (3.) y (4.) permiten implementar estrategias de control de diversa complejidad, simular en tiempo real, utilizar técnicas de inteligencia artificial, implementar control óptimo, etc; todo esto gracias a sus Toolbox específicos.

Contar con un paquete de libre disponibilidad es algo muy tentador, obviamente que algún precio se debe pagar por esto, generalmente en este tipo de software lo primero que se nota es lo poco amigable e insuficiente de la documentación con la que se cuenta, lo segundo que se percibe, es la carencia de

---

\* jvaschetti@scdt.frc.utn.edu.ar

herramientas para análisis y diseño de la problemática específica, no obstante ello, estas falencias se convierten en una oportunidad en el campo de la investigación y del desarrollo.

Un último aspecto a tener en cuenta y no menos importante que los anteriores es la institución que está avalando el desarrollo del producto, lo cual da garantías de continuidad y calidad, sobre todo la de permanecer como paquete de libre acceso, en este caso detrás de este proyecto se encuentra el Scilab Consortium ([www.scilab.org](http://www.scilab.org)) creado en el 2003 por el INRIA, Institut National de Recherche en Informatique et en Automatique ([www.inria.fr](http://www.inria.fr)) y que actualmente se encuentra vinculado a la Digiteo Foundation ([www.digiteo.fr](http://www.digiteo.fr)).

Con el objeto de evaluar las prestaciones de la dupla Scilab / Xcos se modeliza y se simula el comportamiento de una máquina síncrona vinculada a una barra infinita (Single Machine – Infinite Bus, SMIB), en particular se replica el ejemplo “power\_simplealt.mdl” que viene simulado como ejemplo en el Toolbox SimPowerSystems® de Matlab / Simulink® y contra el cual se comparará el resultado obtenido.

## 2. MODELO SMIB

En la Figura (1.a) se aprecia la vinculación generador – barra infinita para el estudio propuesto (Transferencia de potencia entre fuentes activas conectadas por una reactancia inductiva), las características de una barra infinita son su frecuencia y tensión constantes, el generador no puede variar ninguna de las dos [3].

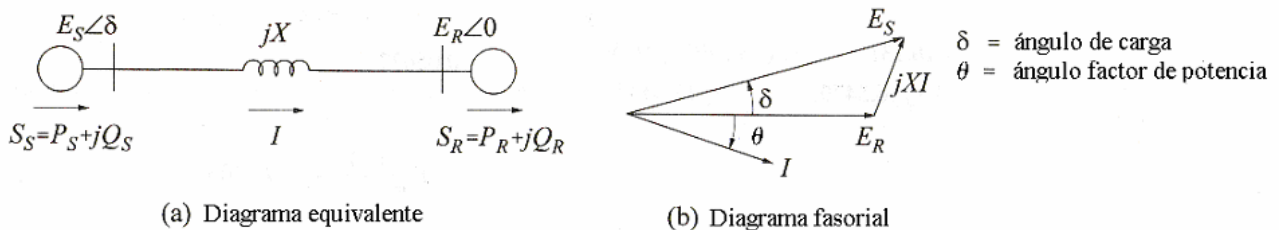


Fig. 1. Diagrama SMIB

Para el análisis se toma como referencia la tensión en la barra infinita,  $E_R$ , la Figura (1.b) muestra el diagrama vectorial de la vinculación [4]. La transferencia de potencia activa está vinculada al ángulo  $\delta$  entre la tensión interna de la máquina  $E_S$  y la tensión de barra  $E_R$  mediante la ecuación (1).

$$P_e = \frac{E_s E_R}{X} \sin \delta \quad (1)$$

## 3. MODELO ELÉCTRICO Y MECÁNICO SIMPLIFICADO DEL GENERADOR

La máquina síncrona se modela en forma simplificada, tanto en su parte eléctrica como en su parte mecánica [4]. El modelo mecánico simplificado se obtiene a partir de la dinámica del rotor, y se expresa en la ecuación (2), [5].

$$J \frac{d\omega}{dt} = T_m - T_e - K_a (\omega_m - \omega_{m0}) \quad (2)$$

Donde:

- $\omega_m$  Velocidad angular mecánica del rotor.
- $\omega_{m0}$  Velocidad mecánica de sincronismo.
- J Momento de Inercia del conjunto de masas acopladas.
- $K_a$  Coeficiente de par amortiguador.
- $T_m$  Par mecánico.
- $T_e$  Par Eléctrico (Reacción de inducido).

La vinculación entre la parte eléctrica y la mecánica se obtiene considerando al campo rotante del estator como referencia y expresando el desplazamiento del campo rotante del rotor en términos del ángulo  $\delta$ , [5]. Se está ahora en condiciones de obtener una representación en Espacio de Estado la cual se expresa mediante el sistema de ecuaciones diferenciales (3) y (4); se debe tener en cuenta la relación entre el par y la potencia.

$$\frac{d\delta}{dt} = \omega - \omega_0 \quad (3)$$

$$\frac{d\omega}{dt} = \frac{\omega_0}{2H} \left[ P_m - \frac{E_S E_R}{X} \sin \delta - \frac{D}{\omega_0} (\omega - \omega_0) \right] \quad (4)$$

Donde:

- $\alpha$  Ángulo mecánico del eje del rotor.
- $p$  Pares de polos de la máquina.
- $\omega$  Velocidad eléctrica del rotor.
- $\omega_0$  Velocidad eléctrica de sincronismo.
- $H$  Constante de Inercia en seg.
- $D$  Factor de Amortiguamiento.

#### 4. MÉTODO NUMÉRICO DE EULER

Con el objeto de comprender posteriormente la estructura del bloque GENERIC y su función asociada, se presenta a continuación un algoritmo numérico para resolver el problema de valor inicial, éste es solo uno de los tantos posibles métodos existentes, los cuales se encuentran expresados en los diferentes solvers que poseen los paquetes.

En forma compacta el sistema representado por las ecuaciones (3) y (4) puede ser expresado como en la ecuación (5), [3 - 5].

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \quad (5)$$

La ecuación (5) es una representación no lineal en Espacio de Estado del sistema SMIB donde  $\mathbf{x}$  es el vector de estado de las  $n$  variables dependientes como función del tiempo  $t$ , asociada a esta representación se encuentra el conjunto de valores iniciales  $\mathbf{x}_0$  en  $t_0$ . Con el objeto de simplificar la explicación se trata el problema como una ecuación diferencial de primer orden, ecuación (6).

$$\dot{x} = f(x, t) \quad (6)$$

Con  $x = x_0$  en  $t = t_0$ , la Figura (4) ilustra el principio del método de Euler, se trata de aproximar la ecuación (6) en el punto inicial, ecuación (7), mediante la tangente cuyo valor en dicho punto está dado por la ecuación (8).

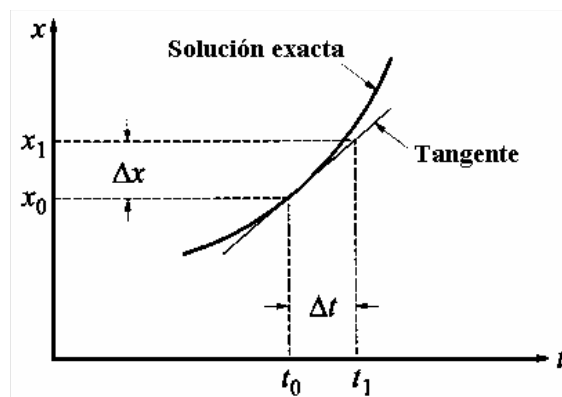


Fig. 4. Método de Euler

$$\left. \frac{dx}{dt} \right|_{x=x_0} = f(x_0, t_0) \quad (7)$$

$$\Delta x = \left. \frac{dx}{dt} \right|_{x=x_0} \Delta t \quad (8)$$

El valor de  $x$  en el instante  $t = t_1 = t_0 + \Delta t$ ,  $x_1$ , está dado por la ecuación (9), la cual se corresponde con los dos primeros términos de la serie de Taylor alrededor del punto  $(x_0, t_0)$ .

$$x_1 = x_0 + \Delta x = x_0 + \left. \frac{dx}{dt} \right|_{x=x_0} \Delta t \quad (9)$$

Si ahora se produce un incremento  $\Delta t$  se determina  $x_2$  en  $t_2 = t_1 + \Delta t$  como sigue:

$$x_2 = x_1 + \left. \frac{dx}{dt} \right|_{x=x_1} \Delta t \quad (10)$$

Usando la ecuación (10) en forma recursiva se puede obtener la solución de la ecuación (6), no se discutirá aquí por cuestiones de espacio problemas tales como la estabilidad de la solución o propagación del error. Se hace notar que para obtener la solución es necesario conocer la expresión funcional de la derivada y el valor inicial que es una solución de la misma, esto último es de fundamental importancia para comprender el bloque GENERIC.

## 5. BLOQUE XCOS GENERIC

La dupla Scilab / Xcos constituye al igual que Matlab / Simulink [6] un Engine Científico (Scilab), sobre el cual se asienta un Simulador con interface gráfica CAD (Computer Aided Desing) orientada a objetos (Xcos). Cada bloque dentro de Xcos representa mediante un ícono el comportamiento dinámico del sistema al cual hace referencia, Figura (5).

Para determinar el comportamiento de estos bloques se necesitan dos funciones asociadas, la computacional y la de interfase, esta última siempre está escrita en lenguaje Scilab, mientras que la primera puede estarlo en C, Fortran o Scilab; por razones de espacio no se tratará en el presente artículo la función de Interface.

En la Figura (5) vemos el ícono que representa el bloque GENERIC (Generic Block 3) tal cual aparece en el Palette de User-Defined Functions. Antes de tratar la estructura y funcionamiento de la función computacional asociada se verá el diagrama de operación del simulador Xcos, el cual permitirá entender aspectos de la función mencionada.



Fig. 5. Bloques Scope, Clock y GENERIC en Xcos.

En la Figura (6.a) se pueden apreciar las relaciones analíticas que existen entre los estados, las entradas, las salidas y las funciones derivadas, en la Figura (6.b) se aprecia el diagrama de procesamiento del simulador, luego de una fase de inicialización, se actualizan las salidas, continuas y/o discretas, posteriormente entra en

un lazo en el que se resuelven recursivamente las funciones derivadas, este lazo depende del paso  $\Delta t$  fijado en la ecuación (8).

Se puede ahora vincular la Sección (3) y (4) con la presente; a partir de un estado conocido (condición inicial), de la expresión analítica que describe la evolución de los estados y de la ecuación de actualización de los mismos, y haciendo uso de un proceso de cálculo iterativo se resuelve el sistema representado por la Ecuación (5).

La función computacional del bloque GENERIC Figura (7.a) es llamada con dos argumentos (Función del tipo 5), una estructura de datos, Block, la cual permite el intercambio de información entre el simulador y el bloque, y un flag Figura(7.b), el cual nos indica en qué parte del proceso iterativo se encuentra el simulador, la función devuelve la estructura Block.

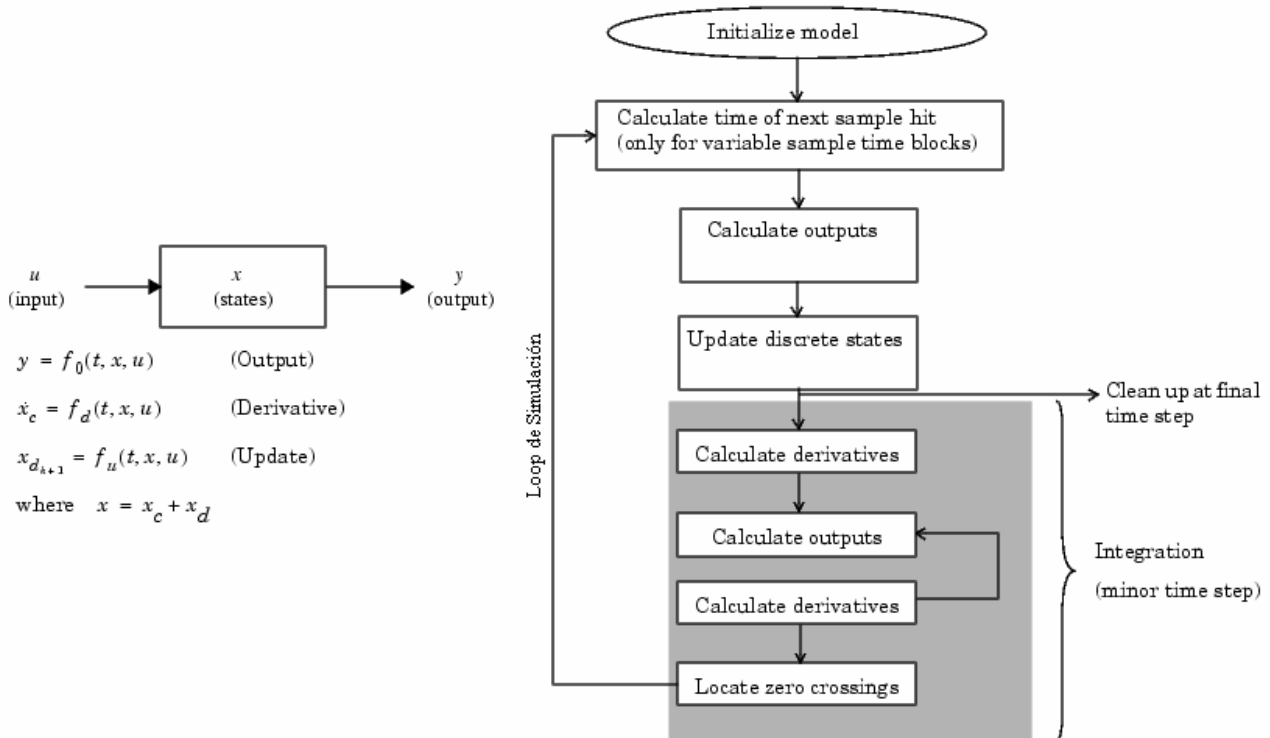


Fig. 6. Relaciones analíticas del bloque y diagrama de procesamiento del Simulador Xcos [6].

```
function block=myblock(block, flag)
...
//your simulation instructions
...
endfunction
```

Flag	Task
0	State derivative computation
1	Outputs update
2	States update
3	Output events timing
4	Initialization
5	Ending
6	Re-initialization for explicit blocks
7	Re-initialization for implicit blocks
9	zero-crossing functions and mode

Fig. 7. Estructura de la Función computacional tipo 5, Valores del flag y Tareas de la Función Computacional [6].

## 6. SIMULACIÓN SISTEMA SMIB

En la Figura (8) se puede ver la pantalla del simulador con el diagrama completo del Sistema SMIB. En él se pueden apreciar los distintos bloques que lo componen:

1. Bloque de entrada,  $P_m$ , representa un escalón de potencia de 0.5 pu a 1.0 pu en el instante 0.5 seg.
2. Bloque Mecánico, modela la parte del sistema representado por la Ecuación (5).

3. Bloque Eléctrico, modela la Ecuación (1).
4. Bloque Barra Infinita, representa el sistema de potencia al cual se vincula la máquina.
5. Bloque Reactancia Interna, simula la Reactancia de la máquina sincrónica.
6. Wattímetro, calcula la potencia eléctrica desarrollada por la máquina.
7. Bloque de Fourier, calcula el ángulo  $\delta$  que existe entre la tensión a bornes de la máquina y la tensión en la barra infinita.

Exceptuando los bloques  $P_m$ , MScope y Clock, todos los otros han sido simulados utilizando la función GENERIC.

La Figura (9) muestra el código fuente para la función GENERIC asociada al modelo mecánico de la máquina.

En la Figura (10) se puede apreciar la respuesta del sistema ante un escalón de entrada en la potencia mecánica, de 0.5 pu a 1.0 pu. La gráfica superior corresponde a la velocidad mecánica de la máquina, como es una máquina de cuatro polos su velocidad es de 1800 rpm, el ángulo  $\delta$  tiene un valor inicial de  $5.65^\circ$  y luego del transitorio correspondiente llega a  $11.3^\circ$ . La última gráfica corresponde a la potencia eléctrica de la máquina la cual coincidentemente con la mecánica pasa de 0.5 pu a 1.0 pu. Los valores iniciales de los estados  $\mathbf{x}_0$  en  $t_0$  son calculados mediante una solución de flujo de potencia.

Utilizando la Figura (9) se puede efectuar el siguiente análisis:

1. Líneas 10, 14 y 19 detectan la etapa en la que se encuentra el proceso iterativo del simulador, si corresponde ejecuta el código pertinente.
2. Líneas 11 y 12, etapa de actualización de las salidas, el estado  $\mathbf{x}$  de la máquina es asignado directamente a las salidas, debe notarse que los datos son dejados a disposición del simulador en la estructura Block.
3. En la etapa de inicialización, línea 14,  $\text{flag} == 4$ , los valores iniciales de los estados son puestos en las salidas, esta parte del código se ejecuta una sola vez al comienzo de la simulación.
4. A partir de la línea 19 y hasta la 24 se efectúa el ciclo de resolución de las ecuaciones diferenciales que modelan la dinámica de la parte mecánica en este caso. Las líneas 20 y 21 asignan las variables de estado a variables intermedias, en las líneas 23 y 24 se calculan las derivadas.

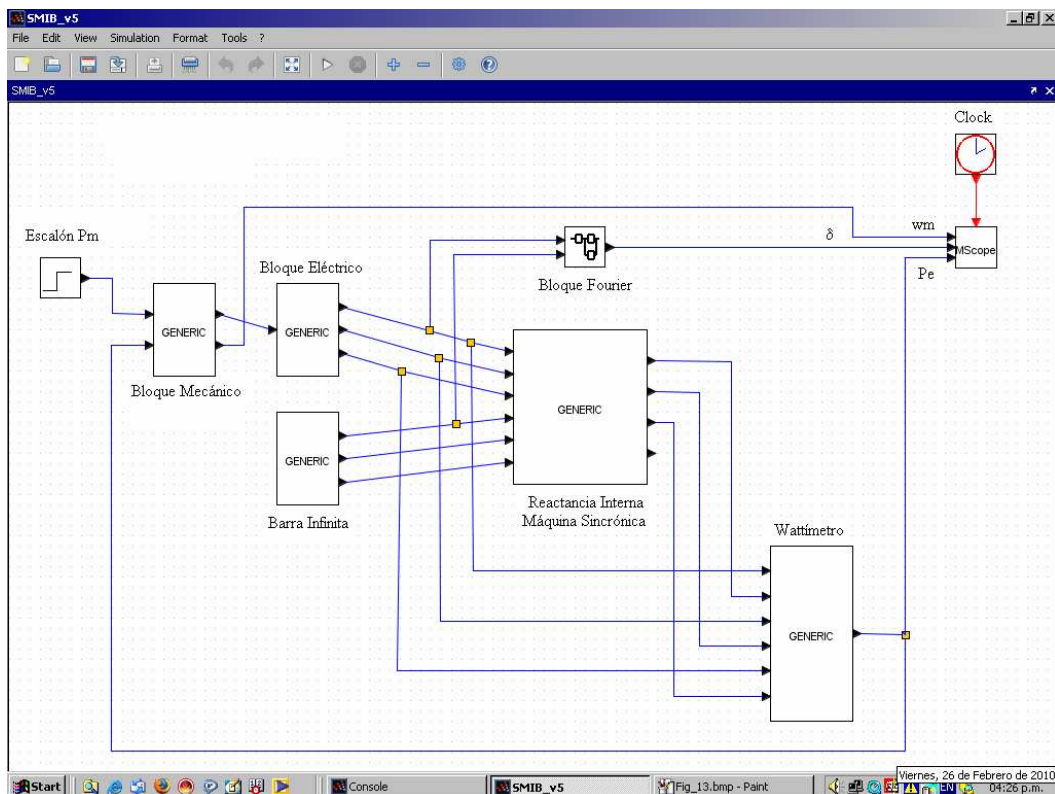


Fig. 8. Diagrama de simulación Xcos del Sistema SMIB.

```

SSM_mechanics.sci
1 // Función que simula la dinámica de la parte
2 // mecánica del rotor.
3 function [blocks] = SSM_mechanics(blocks,flag)
4     H = 3;
5     Kd = 64.3;
6     w0 = 1;
7     Zeta = 0.3;
8     Pm = 0.0;
9     /** UPDATE (Actualización de las salidas).
10    if flag==1 then
11        blocks.outptr(1)(1) = blocks.x(1);
12        blocks.outptr(2)(1) = w0 + blocks.x(2);
13    /** INIT (Inicialización de las salidas).
14    elseif flag==4 then
15        Pm = blocks.inptr(1)(1);
16        blocks.outptr(1)(1) = blocks.x(1); // Angulo mecánico del eje del rotor.
17        blocks.outptr(2)(1) = w0 + blocks.x(2); // Velocidad mecánica de la máquina.
18    /** DERIVATIVE (Cálculo de las funciones derivadas).
19    elseif flag==0 then
20        x1 = blocks.x(1);
21        x2 = blocks.x(2);
22        Pm = blocks.inptr(1)(1);
23        blocks.xd(1) = (2*pi*60)*(w0 + x2);
24        blocks.xd(2) = - (Kd/(2*H))*x2 + (1/(2*H*(w0+x2))) * Pm - (blocks.inptr(2)(1)/(2*H*(w0+x2)));
25    end
26 endfunction

```

Fig. 9. Código fuente del bloque mecánico.

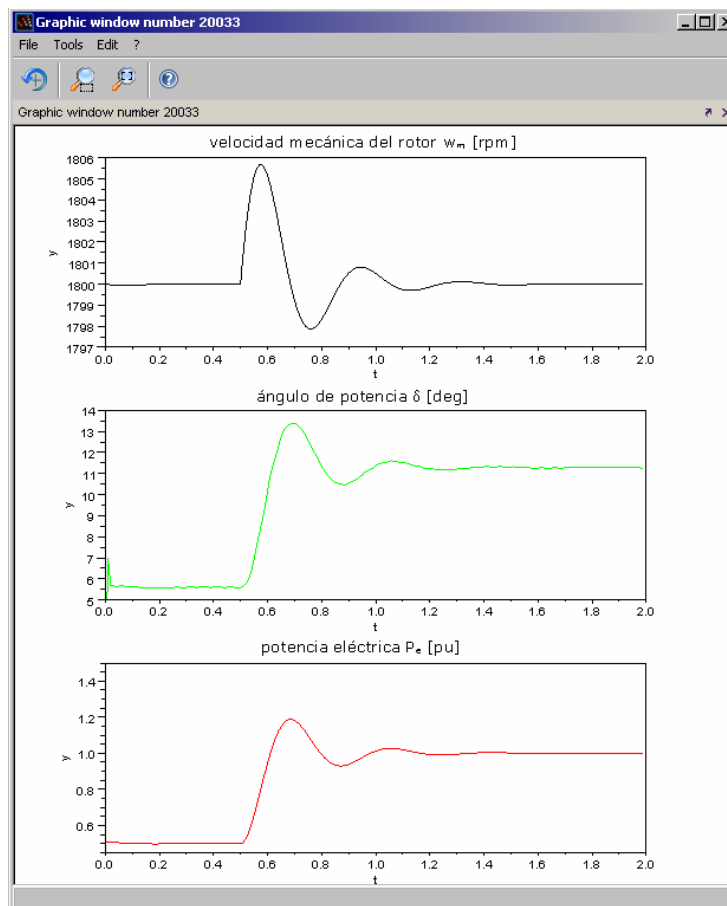


Fig. 10. Respuesta del Sistema SMIB ante un incremento tipo escalón en la  $P_m$  de entrada.

## 7. CONCLUSIONES

Si se compara la respuesta entregada por el ejemplo `power_simplealt.mdl` del Matlab/Simulink se podrá apreciar la coincidencia con las de la Figura (10), al ser iguales se omite su repetición. Se puede concluir que la funcionalidad entregada por la dupla Scilab/Xcos presenta idénticas características a las entregadas por Matlab/Simulink, existen otras consideraciones a tener en cuenta a la hora de comparar prestaciones [7], las cuales pueden ser resumidas en las siguientes:

1. Flexibilidad de la herramienta para permitir su interconexión con distintos paquetes.
2. Permitir la utilización de esquemas paralelos distribuidos, con el objeto de simular modelos complejos en tiempo real.
3. Soporte científico matemático lo suficientemente amplio como para desarrollar modelos ad-hoc.
4. Permitir estructuras del tipo HIL (Hardware in the Loop).
5. Contar con herramientas para resolver ecuaciones diferenciales del tipo ODE, DAE, con características stiff.
6. Permitir la simulación de sistemas en tiempo continuo y discreto.

Si se realiza un estudio detallado de las características exigidas se comprobará que cumple sobradamente con cada una de ellas, solo hay que estar dispuesto a pagar algún precio por la documentación ofrecida, todo lo cual permite concluir que el paquete propuesto cumple con las expectativas previas.

## REFERENCIAS

- [1] Milano, Federico (2005). An Open Source Power System Analysis Toolbox. *IEEE Trans. On Power Systems*. Vol. 20, 1199-1206.
- [2] Chuco, B. (2005). Electrical Software Tools Overview. *Proc. Of II Symposium Nacional de Tecnología SINATEC 2005*. IEEE Perú.
- [3] Kundur, Prabha (1994). *Power System Stability and Control*. The EPRI Power System Engineering Series. McGraw-Hill. ISBN 0-07-035958-X. New York. United States.
- [4] Grainger, John (1994). *Power System Analysis*. McGraw-Hill. ISBN 0-07-061293-5. New York. United States.
- [5] Gómez Expósito, Antonio (2002). *Análisis y operación de sistemas de energía eléctrica*. McGraw-Hill. ISBN 94-481-3592-X. Madrid. España.
- [6] Campbell, S. L. *et al.* (2006). *Modeling and Simulation in Scilab/Scicos*. Springer. ISBN 0-387-27802-8. New York. United States.
- [7] Vaschetti, Jorge *et al.* (2006). Simulador de entrenamiento y certificación, para operadores de centro de control. *Proc. Del Congreso Internacional de Distribución Eléctrica CIDEL 2006*. Buenos Aires. Argentina.