

# IMPLANTACIÓN DEL SOFTWARE SCILAB/SCICOS EN LAS PRÁCTICAS DE ASIGNATURAS AFINES AL ÁREA DE INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

R. PUERTO, R. ÑECO, C. FERNÁNDEZ, M.A. VICENTE

*Departamento de Ingeniería de Sistemas Industriales.*

*Universidad Miguel Hernández de Elche.*

*España.*

*{r.puerto, ramon.neco, c.fernandez, suni}@umh.es*

*En el presente documento se describe la utilización del software SCILAB/SCICOS para la resolución de prácticas afines al Área de Ingeniería de Sistemas y Automática. El interés del uso de dicho software radica, por una parte, en el uso de un software libre frente a otros paquetes de software comerciales y por otra, por la validez y funcionalidad de dicho software para la realización de prácticas de ingeniería de control. El documento se estructura de la siguiente forma: una breve introducción sobre SCILAB/SCICOS, objetivos a alcanzar, enumeración de ventajas e inconvenientes y ejemplos.*

## 1. Introducción

Se presenta en este documento la experiencia realizada en la asignatura ‘Sistemas Electrónicos de Control’ de tercer curso de la titulación de Ingeniería Técnica de Telecomunicaciones (especialidad Sistemas Electrónicos) de la Universidad Miguel Hernández de Elche, en la cual se ha utilizado el software SCILAB/SCICOS para la realización de las prácticas de la asignatura. La principal motivación del uso de dicho software es el abaratamiento de costes en los equipos de laboratorio (software libre) así como la amplia difusión que éste está teniendo en la comunidad científico – técnica.

SCILAB [1,2] es un lenguaje de programación de alto nivel para cálculo científico – interactivo de libre uso desarrollado en INRIA (Institut National de Recherche en Informatique et Automatique) y la ENPC (École Nationale des Ponts et Chaussées) desde 1990. Es responsabilidad de *Scilab Consortium*, fundado en 2003, el cual está compuesto actualmente por 18 miembros, INRIA, DIGITEO, ECOLE CENTRALE DE PARIS, ECOLE POLYTECHNIQUE, ENGINSOFT France, PSA, PEUGEOT, CITROËN y RENAULT entre otros. SCILAB fue creado para ser un sistema abierto donde el usuario puede definir nuevos tipos de datos y operaciones entre los mismos. Resumiendo, se puede decir que SCILAB es un paquete de software científico para el cálculo numérico con un entorno amigable para el usuario.

Por otra parte, SCICOS (Connected Object Simulator) es la herramienta de software para modelado y simulación de Sistemas Dinámicos incluida en SCILAB bajo licencia GPL desarrollado por Ramine Nikoukhah en el *Meatalu Project* de INRIA en el centro Paris-Rocquencourt. SCICOS provee una interfaz gráfica de usuario para editar modelos e interconectar los bloques que representan las funciones fundamentales predefinidas por SCILAB (generador de ondas, ganancia, osciloscopio, etc....).

## 2. Características generales de SCILAB / SCICOS

SCILAB es un entorno de programación altamente flexible, cuyas principales características y prestaciones son:

- Potencia: Incluye cientos de funciones matemáticas, operadores y comandos que ofrecen al usuario la posibilidad de ser utilizados en gran cantidad de aplicaciones, además de permitir crear y definir funciones propias.
- Capacidad de generación de gráficos 2D y gráficos 3D.
- Permite operaciones matriciales, polinomiales y con funciones de transferencia.
- Permite la resolución de sistemas de ecuaciones lineales y ecuaciones diferenciales.
- Permite la creación y utilización de conjuntos de funciones destinadas a aplicaciones específicas denominadas *toolboxes*, por ejemplo Control, Optimización, Procesado de Señal, etc....
- Precio: SCILAB es significativamente más barato, especialmente si se considera el hecho de que este software es de código abierto e incluye la misma cantidad de funciones que se encuentran disponibles únicamente en las *toolboxes* adicionales de otros paquetes comerciales (control, bases de datos, estadística, etc....) ya que todas estas son realizadas por desarrolladores externos a la compañía y se distribuyen de manera gratuita.
- Funcionalidad: Debido a que SCILAB es un programa de código abierto presenta cierta ventaja frente a otros programas que aún adquiriéndolos bajo licencia poseen versiones gratuitas (LE Lite Edition) pero que limitan la capacidad del software como por ejemplo restringiendo funciones de adquisición de datos o funciones de comunicación que permitan obtener datos de otros programas.

Por otra parte SCICOS [3] permite:

- Modelar gráficamente, compilar y simular Sistemas Dinámicos.
- Combinar comportamiento continuo y discreto en el mismo modelo.
- Seleccionar elementos modelados desde la paleta de bloques estándar.
- Programar nuevos bloques en lenguaje C, Fortran o lenguaje SCILAB.
- Ejecutar simulaciones desde el entorno SCILAB.
- Generar código C de un modelo SCICOS con un *generador de código*.
- Generar ejecutables de control en tiempo real complicados por medio de las librerías de tarjetas de adquisición de datos SCICOS-HIL (para sistema operativo Linux y Windows) o SCICOS-RTAI [4,5] (para sistema operativo Linux) combinados con la librería SCICOS-FLEX.
- Utilizar bloques implícitos desarrollados en lenguaje Modelica, utilizando la librería correspondiente SCICOS/Modelica (Coselica toolbox).

Cabe mencionar que todo el software utilizado es multiplataforma (Windows, Mac, Linux), lo cual permite una difusión mayor que otros programas ceñidos a plataformas específicas.

Evidentemente existen varios paquetes de las características de SCILAB/SCICOS que ofrecen funcionalidades similares. El más importante de ellos (y el estándar de facto tanto en el ámbito académico como industrial) es sin duda Matlab/Simulink [7]. Este software es muy potente y con un entorno gráfico muy cuidado. Posee asimismo multitud de *toolboxes* para aplicaciones muy diversas (comunicaciones, control, aeroespacial, etc....) y es además también un sistema multiplataforma (Windows, Unix, MacOSX). Hasta la fecha las prácticas de las asignaturas asociadas al área de Ingeniería de control se han

venido realizando con dicho software. El único inconveniente de Matlab es su elevado coste, permisible para un Departamento/Universidad pero completamente inasequible para el alumno (ni siquiera, a veces, en su versión ‘Estudiante’). Este factor obligaba a los alumnos a realizar la mayoría de prácticas de forma presencial en el laboratorio a la par que impedía su utilización (al menos de forma legal) fuera del entorno universitario. Con SCILAB/SCICOS se ha querido dar una solución a este problema. Por una parte, como se ha comentado anteriormente, se ha dotado a los laboratorios de puestos de trabajo de un coste mucho menor que los puestos de trabajo ‘convencionales’ al introducir software libre, y por otra, se ha facilitado al alumno una herramienta de acceso libre mediante la cual puede realizar sus prácticas y desarrollos fuera del entorno universitario.

Tampoco sería justo comentar sólo estas dos alternativas de software ya que existen muchas más, algunos en desarrollo y otras en versiones ya muy estables como Octave[7] y R[8].

### 3. Implantación del software SCILAB/SCICOS en la asignatura ‘Sistemas Electrónicos de Control’.

Tan importante como la elaboración de los tutoriales y guiones de prácticas bajo esta nueva herramienta nos parece a los autores la implantación de la nueva metodología dentro de la asignatura.

Como se ha comentado, tradicionalmente las prácticas de la asignatura ‘Sistemas Electrónicos de Control’ de 3º de Ingeniería Técnica de Telecomunicaciones (Especialidad Sistemas Electrónicos) han venido desarrollándose bajo el sistema operativo Windows y utilizando Matlab/Simulink (y algunos *toolboxes*: control y signal processing normalmente). Además tradicionalmente en asignaturas de cursos anteriores se ha utilizado también el mencionado software. Por esto se propuso que cada alumno eligiese el software con el que quería hacer las prácticas (evidentemente los resultados y las conclusiones obtenidas con ambos paquetes de software son idénticas). Se motivó a alumnado a utilizar el nuevo software aludiendo que éste es de uso libre, que no se infringe ninguna licencia al utilizarlo en cualquier tipo de máquina y que para lo que nos ocupa resulta tan potente como el anterior.

La distribución de alumnos que eligieron realizar la práctica con uno u otro software durante el año lectivo 2008 – 09 se muestra en la tabla 1:

**Tabla 1.** Distribución de alumnos por software

	% Alumnos
SCILAB/SCICOS	72%
MATLAB/SIMULINK	23%
AMBOS	5%

Como se puede observar, un pequeño porcentaje de alumnos optó por realizar las prácticas en ambos entornos con el fin de comparar los resultados obtenidos. Para aquellos que optaron por la plataforma SCILAB/SCICOS se comprobó que la curva de aprendizaje del nuevo entorno no era excesivamente elevada (ayudó mucho a ello los tutoriales y los ejemplos suministrados -bastante afines a las prácticas a realizar) y que dado un pequeño periodo de aprendizaje básico fueron capaces de resolver los mismos problemas y con igual diligencia que con el software comercial.

Durante el presente año ya se han propuesto las prácticas sólo bajo SCILAB/SCICOS. Es evidente por otra parte, que Matlab/Simulink posee facilidades y herramientas cuya utilidad y provecho está muy lejos todavía de ser ofrecidas por otras plataformas de software, por lo que se permite (e incluso se insta) a los alumnos a trabajar muchas veces con ambas herramientas. Por otra parte, el carácter libre del software

SCILAB/SCICOS hace que muchos más alumnos trabajen en casa con dicho software que con el de licencia comercial.

### 3. Ejemplos

Se presentan a continuación dos ejemplos [9][10] (uno de análisis y otro de diseño) que ilustran y avalan la validez y funcionalidad del software SCILAB/SCICOS.

#### 3.1 Análisis de Sistemas dinámicos de 2º orden discretos.

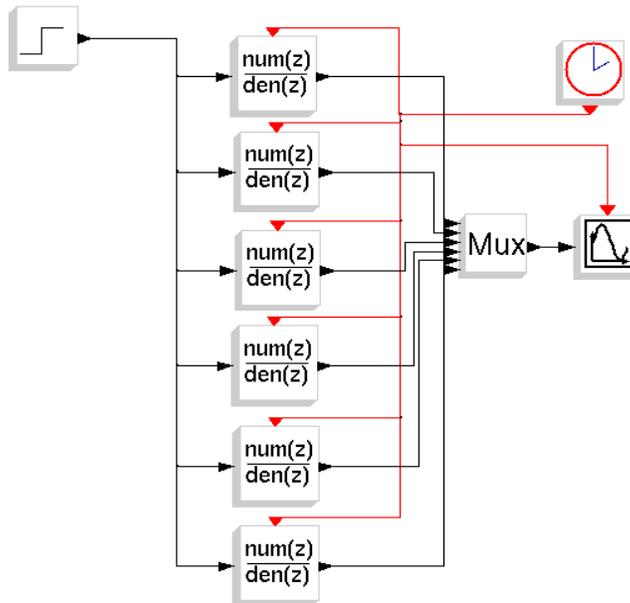
Se pretende en esta práctica estudiar y analizar el comportamiento temporal de distintos sistemas de 2º orden expresados éstos mediante su función de transferencia discreta a un periodo  $T = 0.1$  seg. Para ello, lo que se pide al alumno es que calcule los polos de dichos sistemas, que obtenga mediante simulación la salida del sistema ante entrada escalón unitario y que relacione la posición de los polos con las características de la señal de salida.

Las funciones de transferencia a analizar son las siguientes:

$$G_1(z) = \frac{z^2}{z^2 - z + 0.29} \quad G_2(z) = \frac{z^2}{z^2 - 1.6z + 0.89} \quad G_3(z) = \frac{z^2}{z^2 - 1.2z + 1}$$

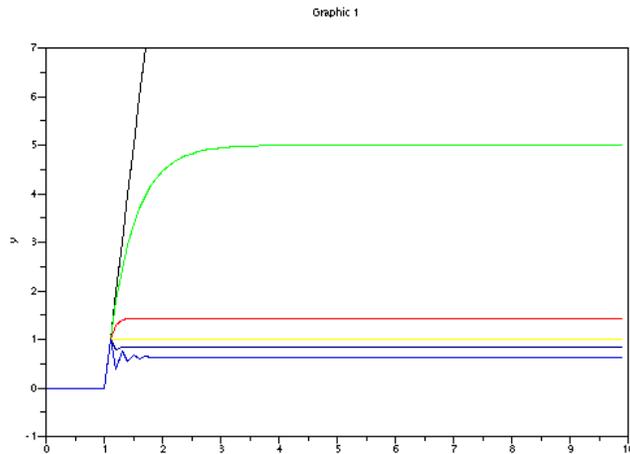
$$G_4(z) = \frac{z^2}{z^2 + z + 0.29} \quad G_5(z) = \frac{z^2}{z^2 + z + 0.89}$$

El esquema SCICOS para obtener la señal de salida correspondiente a cada función de transferencia se muestra en la figura siguiente (fig. 1):



**Figura 1.** Escenario SCICOS para la simulación.

Las respuestas obtenidas de dicha simulación se muestra en la figura 2:



**Figura 2.** Respuesta temporal de las salidas ante entrada escalón

Analizando los polos de la función de transferencia y comprobando la salida temporal de cada uno de los sistemas el alumno puede establecer la correspondencia entre la posición de los polos y características de la señal de salida.

### 3.2. Diseño de un regulador tipo PID discreto.

En el presente ejemplo se desea diseñar un regulador tipo PID discreto para un proceso modelado mediante la siguiente función de transferencia

$$G(z) = \frac{0.007(z+0.94)}{(z-0.95)(z-0.86)}$$

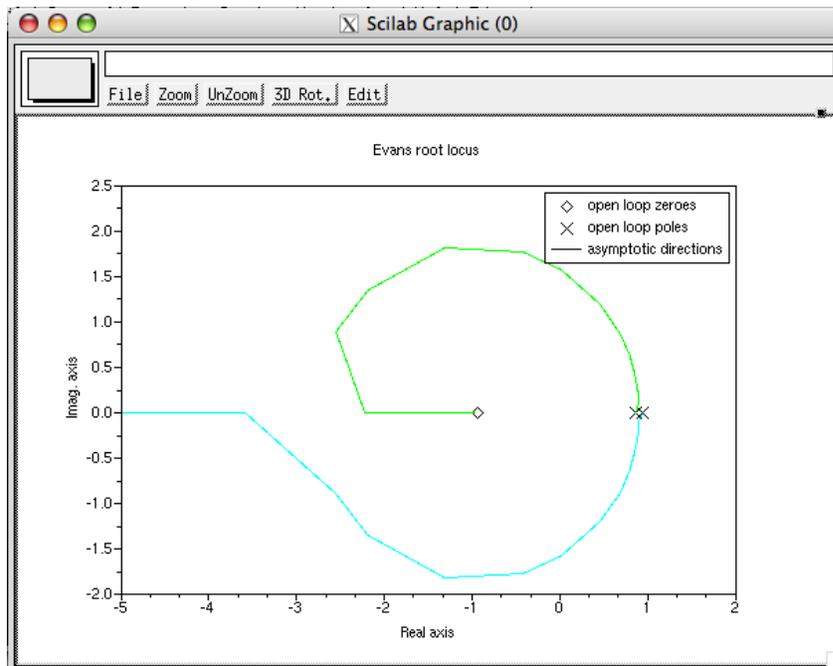
discretizada a un periodo de muestreo de  $T = 0.1$  seg. De forma que el comportamiento de la salida cumpla los siguientes requisitos: tiempo de establecimiento menor que 1.57 seg., sobreoscilación máxima menor del 15% y error de posición nulo. Con el fin de guiar al alumno en la elección del regulador, éste puede trazar con SCILAB el lugar geométrico de las raíces (Fig.3).

Una vez realizados los cálculos pertinentes, el regulador PID que se obtiene viene dado por la función de transferencia

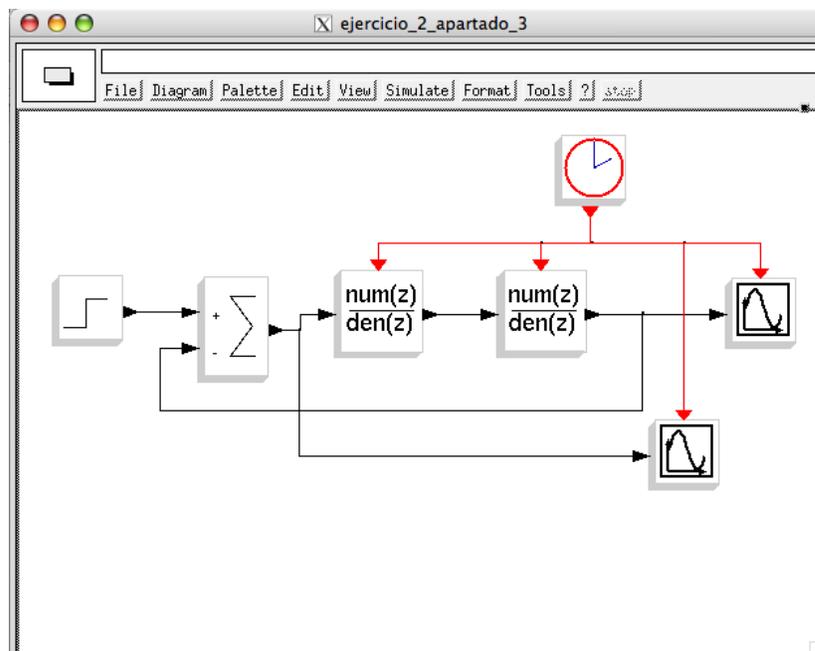
$$G_R(z) = 18.86 \frac{(z-0.967)(z-0.7)}{(z-1)z}$$

Ahora resta simular el comportamiento del sistema junto con el regulador obtenido mediante SCICOS y comprobar que la respuesta cumple los requisitos exigidos. Para ello se crea el esquema SCICOS que se muestra en la Fig. 4.

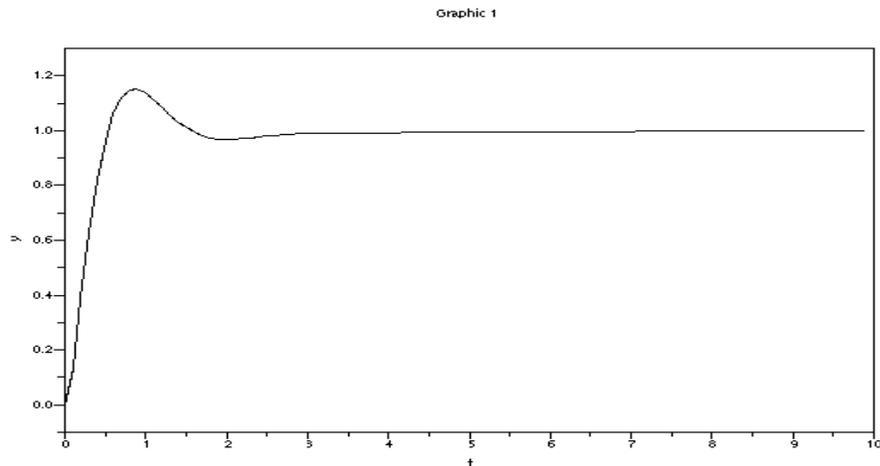
Una vez simulado el ejemplo, en la gráfica de la salida obtenida (Fig 5) se puede comprobar que se cumplen las especificaciones de diseño exigidas.



**Figura 3.** Lugar geométrico de las raíces



**Figura 4.** Escenario SCICOS para la simulación



**Figura 5.** Salida del sistema regulado

#### 4. Conclusiones

Como conclusiones más importantes cabe destacar:

- Es un tipo de software que no se distribuye bajo licencia, sino que es de acceso libre evitando así cualquier desembolso de tipo económico por su utilización, ya sea para uso personal, laboral o docente. Debido a que este es un programa de código abierto, es decir, no se ha de pagar ninguna licencia para su utilización, es la herramienta perfecta para que grandes y pequeñas áreas académicas como universidades y centros de investigación lo implanten como uno de sus paquetes de software básicos para el aprendizaje
- Es capaz de realizar bloques y/o funciones utilizando los lenguajes de programación más extendidos en el mundo informático como son el lenguaje de programación en C o Fortran. Hay una actualización continua por parte de los creadores incluyendo nuevas funciones, librerías y mejoras constantes, e incluso aportes adicionales de desarrolladores externos que utilizan este software.
- Tiene la ventaja de ser un software multiplataforma, se puede usar en sistemas operativos Windows, Macintosh y Linux. Ventaja que puede ser aprovechada para crear puntos de investigación y desarrollo completamente gratuitos utilizando la combinación de sistema operativo Linux junto a SCILAB/SCICOS.
- Como se ha puesto de manifiesto en el ejemplo del apartado anterior, la combinación de uso de SCILAB/SCICOS permite la resolución óptima de problemas de ingeniería de control de la misma forma que se realizaría con otro tipo de software comercial.
- Dentro de las limitaciones que presenta cabe señalar que en comparación con otros programas del mismo ámbito posee un interfaz gráfico algo inferior pudiendo mejorar la calidad de éste e incrementar las opciones de representaciones gráficas que incluye. Otro inconveniente importante es el hecho de que el lenguaje de programación de funciones en SCILAB (archivos .sci) es algo primitivo, limitando así la capacidad de programación por parte del usuario. Es previsible que este último punto se resuelva en versiones posteriores.

Como comentario final cabe destacar que los autores han experimentado en el curso académico 2008/2009 con la implantación de este software en los laboratorios de control y automática obteniendo resultados igual de satisfactorios que los obtenidos con otros paquetes de software comercial.

## Referencias

- [1] L. Campbell, Jean-Philippe Chancelier & Ramine Nikoukhah. “Modeling and Simulation in Scilab/Scicos” Ed. Springer (ISBN 0-387-27802-8).
- [2] Página oficial SCILAB: [www.scilab.org](http://www.scilab.org)
- [3] Página oficial SCICOS: [www.scicos.org](http://www.scicos.org)
- [4] Página oficial RTAI: [www.rtai.org](http://www.rtai.org)
- [5] Página oficial Comedi: [www.comedi.org](http://www.comedi.org)
- [6] Página oficial de Matlab: [www.mathworks.com](http://www.mathworks.com)
- [7] Página oficial de Octave: [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
- [8] Página oficial de R: [cran.r-project.org/](http://cran.r-project.org/)
- [9] K. Ogata. “Sistemas de control en tiempo discreto”. Ed. Prentice Hall. 1996.
- [10] Karl J. Aström, T. Hägglund. “Control PID avanzado”. Ed. Pearson Education. 2009.