ASADES

Avances en Energías Renovables y Medio Ambiente
Vol. 19, pp.08.13-08.24, 2015. Impreso en la Argentina
ISSN 2314-1433 - Trabajo selecionado de Actas ASADES2015

SIMUSOL: ECUACIONES IMPLÍCITAS CON INTERPRETACIÓN DINÁMICA, CORTES INDUCTIVOS Y LAZOS CAPACITIVOS.

Diego Saravia
Facultad de Ciencias Exactas – Universidad Nacional de Salta (U.N.Sa.)
Fax: 54-387-4255489, Av. Bolivia 5150. 4400 Salta, Argentina
e-mail: Diego.Saravia@gmail.com, dsa@unsa.edu.ar,

Recibido 17/09/15, aceptado 13/10/15

RESUMEN: Se presentan resultados en el desarrollo de software de simulación para uso conjunto con SIMUSOL, como alternativa al SCEPTRE. Se incorporan ecuaciones implícitas, mediante intérpretes dinámicos que no requieren de compilación. El simulador es capaz de detectar y modelar sistemas que representados como circuitos tienen: "conjuntos de corte", formados por "inductancias" y "fuentes de corriente" (en adelante cut-sets inductivos), y "lazos", formados por "capacitores" y "fuentes de tensión" (en adelante loops capacitivos). El software desarrollado puede reemplazar a SCEPTRE con ventajas, sin otro límite que la memoria de la computadora. Este software también reemplaza al software PSICRO.

Palabras claves: SIMUSOL, SCEPTRE, PSICRO, EXPRTK, SWIG, loops, cut-sets

INTRODUCCION

Se desarrolla un software libre que resuelve ecuaciones diferenciales y algebraicas combinadas e implícitas con el fin de simular sistemas representables como circuitos eléctricos o análogos. El objeto de este desarrollo es reemplazar el uso de software SCEPTRE en SIMUSOL. SIMUSOL simula sistemas complejos vinculados a las energías alternativas, sobre la base de analogías de sistemas térmicos con circuitos eléctricos. Permite introducir gráficamente los sistemas a simular. SIMUSOL tiene una importante comunidad de usuarios y continuamente se presentan trabajos de investigación y desarrollo que lo usan. SCEPTRE es llamado por SIMUSOL para resolver las ecuaciones diferenciales que deduce. Si bien SCEPTRE es muy útil y eficaz a la hora de resolver las ecuaciones necesarias, tiene limitaciones varias que se discuten más adelante y se hace conveniente y necesario desarrollar una alternativa. No existen y no conocemos alternativas o estudios similares en desarrollo en este momento. En el trabajo se presenta el desarrollo de un nuevo software junto con los estudios y evaluaciones de diferentes herramientas, y tecnologías al efecto de comprobar si el desarrollo es eficaz para resolver los problemas planteados y validar que los resultados de las simulaciones son correctos. El proceso de cálculo seguido tiene justificación teórica, y se valida para descartar errores de programación o análisis.

Antecedentes

Desde el 2000 se desarrolla una línea de programas informáticos para simular sistemas térmicos (Saravia. L. y Saravia, D. 2000). Y en especial SIMUSOL, desde el 2002. (Alía, Saravia, L. y Saravia, D. 2002, Alía, Saravia, L. y Saravia, D. 2012. Saravia, D. y Alía 2012).

La simulación se realiza invocando al SCEPTRE (Kuo, 1971; Magnuson, 1982) y se ha desarrollado un software alternativo para superar algunas de las dificultades y los límites que implica el uso de SCEPTRE.

En (Saravia, D. 2014) se presentaron avances relacionados con la metodología de los Bond Graph y la integración simpléctica usando GSL (Galassi, Davies, Theiler, Gough, Jungman, Booth, y Rossi, 1996).

Problemáticas a resolver en este trabajo

Ecuaciones implícitas en forma dinámica.
Se pretende encontrar los valores de las variables que resuelven simultáneamente todas las ecuaciones que definen un paso (habitualmente en el tiempo), de un sistema dado.

Al inicio de la ejecución del nuevo software, usando el software EXPRTK (Partow,2015) en combinación con SWIG (Volkman, 2005, Beazley, Fletcher y Dumont, 1998, Beazley, 1999, Proyecto SWIG 2015), se cargan las ecuaciones del archivo de texto, de extensión ".d", que define el problema, y se "interpretan". El archivo ".d", es producido habitualmente por SIMUSOL para ser interpretado por SCEPTRE.

Luego el software se encuentra en condiciones de calcular los resultados de las mismas, lo que se efectúa cada vez que es necesario sin repetir la interpretación inicial. Este proceso de dos etapas es lo que hace eficiente a EXPRTK.

EXPRTK, calcula el resultado residual de cada ecuación dada, y GSL propone y encuentra las variables que hacen simultáneamente todos los residuales cero, llamando reiteradamente a EXPRTK a tal fin.

Conjuntos especiales de elementos
En un circuito eléctrico cuando se encuentran un conjunto de elementos que sacándolos, separan al circuito en dos se habla de un **conjunto de corte (cut-set)**. Cuando se encuentra un "anillo" de elementos se habla de un **lazo (loop)**. Estos conjuntos nos interesan, ya que requieren un esfuerzo especial para simularlos, en dos casos, que describimos:

- **Cut-sets inductivos** (Figura 1 Der.:) Aparecen cuando se puede partir un circuito en dos partes, de forma que sólo queden conectados con elementos del tipo o inductivo o bien fuentes de corrientes. **S**on la generalización de los circuitos serie conteniendo bobinas**.** Si uno piensa el nodo intermedio de dos bobinas en serie (Figura 1: Izq.) como uno de los circuitos cortados y los extremos como parte del otro, se ve claramente el concepto de "corte" y la separación de dos circuitos. En este caso las corrientes en las dos bobinas son iguales (o suman cero dependiendo del sentido que le asignemos) y todos los futuros cambios también.
- **Loops capacitivos** (Figura 1 Centro:) Aparecen cuando en un circuito se forma un anillo de elementos que sólo son o del tipo capacitivo o bien fuentes de tensión. Son la generalización de los circuitos de condensadores en paralelo. En efecto dos condensadores puestos en paralelo forman un loop. El voltaje de ambos debe coincidir, y no sólo el voltaje sino sus derivadas y los valores futuros mientras estén conectados..

Ambos tipos de conjuntos de elementos, con sus particulares topologías, presentan problemas que requieren un tratamiento especial a la hora de resolverlos. Por lo que pudimos comprobar, la versión disponible del SCEPTRE no resuelve los mismos de una forma consistente, dependiendo la solución del orden de inclusión de los componentes en el archivo ".d". +

Notas: Las fuentes son elementos cuya tensión o corriente es determinada directamente por su "estado" o sea que son un "dato" del cálculo. El problema de los inductores en un cut-set es dual al de los condensadores en un loop.

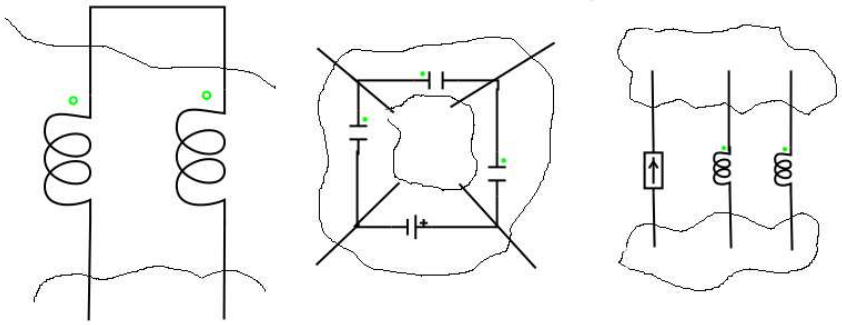


Figura 1. (Izq.:) Dos bobinas en serie son un cut-set. (Centro.:) Loop de tres condensadores y una fuente de tensión (Der.:) Cut-set de dos bobinas y una fuente de corriente.

Tenemos dos problemas relacionados al simular circuitos con estos conjuntos especiales:

- **Ajustes previos de cut-sets y loops.**

Por ejemplo, si tenemos condensadores en un loop, la suma de voltajes debe ser cero en el loop. Pero si los condensadores, antes de formar el loop, tienen una carga arbitraria, nada garantiza que la suma de sus tensiones sea cero.
Al conectarse podemos suponer que habrá un desplazamiento "instantáneo" de carga (éstos circuitos se describen no solo por la corriente que circula en ellos, sino por el desplazamiento de carga). Este desplazamiento debe en primer lugar hacer cero la suma de potenciales del loop. En segundo lugar debe respetar que ningún condensador quede con carga des-balanceada. Así la carga que sale de un condenador debe ser la misma que la que entra al próximo del loop, y la que sale del anterior, y así sucesivamente (para no involucrar al resto del circuito conectado al loop). Por lo que la carga que se desplaza es la misma en todo el loop. Con estas dos condiciones se puede calcular la distribución de voltajes en el loop luego del "ajuste", la que llamamos condición inicial. Es interesante notar que la carga total de un condensador es siempre cero. Cuando se carga una placa se descarga la otra. Estas consideraciones llevan a denominar "gorge" a lo que habitualmente se denomina carga (Denker, 2014).
De igual forma al abrirse los inductores y quedar conectados, la suma de corrientes deberá ser cero en el cut-set. Y en los dos circuitos que quedan vinculados entre sí, para no ser afectados, el potencial de cada uno de los dos sólo puede variar uniformemente en relación con el otro. La variación de tensión será la misma en todas las bobinas del cut-set. Entonces el cambio de flujo en cada inductor debe ser el mismo.

- **Cálculo del próximo estado, justificación teórica.**

En un circuito sin cut-sets ni loops, se puede determinar el próximo estado en dos etapas: (a) Se toma el valor instantáneo de todas las corrientes y voltajes considerando los condensadores como fuentes de tensión, y las bobinas como fuentes de corriente. Con estos valores de corrientes y voltajes en las bobinas y condensadores respectivamente (condiciones iniciales) más los voltajes y corrientes de las respectivas fuentes, calculamos las tensiones y corrientes en todo el circuito, en el que "sólo quedan resistencias". (b) Sabiendo que el cambio de voltaje en un condensador depende de la corriente que entra, y el cambio de corriente en una bobina del potencial que cae en la misma, podemos calcular el próximo estado (condiciones finales del paso).
Sin embargo cuando nos topamos con cut-sets capacitivos o loops inductivos, las ecuaciones de Kirchhoff quedan de entrada satisfechas en la distribución de corrientes en el cut-set o voltajes entre los componentes del loop. La condición de suma de voltajes cero en el loop y la de suma de corrientes cero en el cut-set, son redundantes porque las garantizan las condiciones iniciales (una vez calculadas de las previas), por lo que éstas ecuaciones resultantes de la aplicación de las leyes de Kirchhoff son redundantes y no aportan información pertinente para resolver el circuito. Sin embargo derivando las mismas obtenemos otras ecuaciones que sí nos permiten resolverlo. La condición de suma de voltajes cero y la de suma de corrientes cero reaparecen como suma de derivadas. Al reemplazar las derivadas de los voltajes de cada condensador y de las corrientes de cada bobina por su ecuación constitutiva, aparece la variable complementaria sin derivar (si son loops de condensadores: corrientes; para cut-sets de bobinas: voltajes). Nos queda así un sistema de ecuaciones diferente. Esta cuestión está vinculada al índice del sistema de ecuaciones diferenciales—algebraicas o DEA, En (Günther, M., Feldmann, U., y ter Maten, J. 2005. Thulasiraman, 2002. Mayer, 1971), se analiza con detalle.
Repasando lo anterior, en un caso de condensadores clásicos constantes: sabiendo que la suma de voltajes es cero, sus derivadas temporales lo serán, y como en éstos condensadores la derivada temporal de los voltajes es su corriente sobre su capacidad, queda:

$$\sum V_c + \sum V_F = 0$$

$$\sum \dot{V}_c + \sum \dot{V}_F = 0 = \sum \frac{I}{C} + \sum \dot{V}_F$$

Con lo cual la ecuación auto-satisfecha en términos de voltajes puede re-expresarse en términos de corrientes. La relación es similar en los cut-sets inductivos. Es importante notar que si existen fuentes de tensión en el loop, éstas quedan en la ecuación, no con su valor de tensión, sino si la hubiere, con su derivada temporal. En general el programa incorpora no sólo las ecuaciones de Kirchhoff sino también sus derivadas.

METODOLOGÍA

A los efectos de verificar el correcto funcionamiento del software desarrollado se elige para incorporar al presente trabajo, un conjunto de sistemas a simular, representativo de los problemas que se quieren solucionar. Los sistemas se listan en la sub sección "Ejecución y verificación". Se han seleccionado sistemas que plantean: a) elementos con ecuaciones constitutivas variables implícitas; b) cut-set inductivos c) loops capacitivos, y finalmente d) un sistema que no requiere ecuaciones diferenciales sino sólo algebraicas tendiente a demostrar que el presente software también sirve para reemplazar a PSICRO (Saravia, 2008).

Se tratan especialmente los archivos de entrada del SCEPTRE a los efectos de especificar valores que no puede calcular. En el caso de los cut-set y loops, se le provee en forma manual con los datos iniciales correctos; en el caso de ecuaciones implícitas en los elementos, se les provee definiciones equivalentes no implícitas eligiendo ejemplos en lo que ésto es posible.

RESULTADOS

Software que se incorpora
Se describe el uso de software de terceros, que se suman a otros programas presentados en trabajos previos.

EXPRTK
La biblioteca de programas "EXPRTK", escrita en C++, sirve para evaluar en forma dinámica (sin necesidad de usar compiladores, ni de escribir "programas"), expresiones matemáticas. Es muy eficiente. En una primera etapa el software realiza una interpretación de la expresión y luego el usuario puede evaluar la misma muchas veces. Ha sido desarrollada por Arash Partow (Partow,2015), y se perfecciona en forma continua.
Es software libre.

Tiene, entre otras, las siguientes capacidades:

- operadores matemáticos (+, -, *, /, %, ^)
- funciones predefinidas como (min, max, avg, sum, abs, ceil, floor, round, roundn, exp, log, log10, logn, root, sqrt, clamp, inrange, swap), trigonometría (sin, cos, tan, acos, asin, atan, atan2, cosh, cot, csc, sec, sinh, tanh, d2r, r2d, d2g, g2d, hyp)
- igualdades y desigualdades (=, ==, <>, !=, <, <=, >, >=)
- asignaciónes (:=, +=, -=, *=, /=, %=)
- lógica booleana (and, mand, mor, nand, nor, not, or, xor, xnor)
- estructuras de control (if-then-else, condicional ternario, switch case, return)
- ciclos (while loop, for loop, repeat until loop, break, continue)
- optimizaciones
- operaciones con cadenas de caracteres.
- variables locales, vectores, cadenas de caracteres, definidas por el usuario
- definición de funciones internas de hasta 6 variables.
- diferenciación e integración numérica.
- totalmente portable a diferentes arquitecturas y compiladores (gcc, intel, visual studio, etc.).

Su uso nos permite leer las ecuaciones de los mismos archivos de texto de entrada de datos, donde se definen los sistemas a simular, para ingresar sus propiedades y funciones características.
Al no ser necesario contar con compiladores fortran o C en la computadora, se simplificará la instalación y uso del SIMUSOL notoriamente. Las propiedades constitutivas de los elementos del circuito y otras pueden darse totalmente por ecuaciones implícitas y no solamente a través de una función que da el valor de un coeficiente, de por ejemplo, los componentes: R, L, o C, como se hace para el SCEPTRE.

SWIG

Es una herramienta de desarrollo que vincula programas o bibliotecas redactadas en C o C++ con interpretes como Perl.

En nuestro caso permite combinar las funciones en C de la librerías GSL que venimos usando desde PSICRO y las librerías en C++ de EXPRTK con programación Perl. Lo hace interpretando el código C/C++, generando código tipo "pegamento" requerido por, en nuestro caso Perl, para interactuar con C/C++. Con SWIG, el código Perl llama directamente a funciones escritas en C y C++.
Es software libre.

Desarrollo propio
El simulador desarrollado maneja listas de variables y de ecuaciones, de diferentes tipos.

Hay variables que representan el "estado del sistema", como por ejemplo la carga de un condensador (o su voltaje). Otras son relevantes para conocer algún aspecto del sistema simulado o porque son usadas en alguna ecuación que debe satisfacerse.

Si el sistema cambia, el cambio se manifiesta en las diferenciales de sus variables. En particular al calcular las "diferenciales" de las variables de estado, "sumándolas" a sus respectivas variables, el programa obtiene un nuevo estado del sistema.

Se trabaja con diferencias finitas. El programa reemplaza en la ecuación diferencial la derivada total de una variable en relación con, por ejemplo el tiempo, por el cociente de la diferencia de esa variable sobre el intervalo de tiempo, DT, cuyo valor se especifica expresamente, llamado "diferencial" del tiempo.

Así, el programa va avanzando en la integración de las ecuaciones diferenciales del sistema en estudio paso a paso (habitualmente en el tiempo), resolviendo un conjunto de ecuaciones implícitas en cada uno de ellos. En cada paso se tienen conocidas una serie de variables que definen el estado previo. Y un número igual de ecuaciones y variables adicionales que el sistema debe encontrar. Con las variables resueltas, entre ellas los "diferenciales" de las variables de estado, el integrador las acumula y define el estado final del paso y el valor de las variables de estado, que es a su vez el estado previo del próximo paso.
Casi todas las ecuaciones en el archivo ".d", se escriben sin igual, EXPRTK calcula su valor y las rutinas del GSL de búsqueda de ceros (ya usadas en PSICRO) se encargan de hacerlas cero encontrando el valor adecuado de las variable. Deben entonces pensarse como ecuaciones igualadas a cero. Las otras, del tipo terminación, (en RUN CONTROLS), cuando se cumplen (generalmente inecuaciones), valen uno y el programa termina su trabajo.

Cuando se trata de usar este programa para resolver un problema de ecuaciones implícitas de un paso, por ejemplo al reemplazar el PSICRO; no se incluyen diferenciales, o se imponen como 0, y se establece como condición de terminación el número uno. Entonces al cumplirse la condición de entrada, el programa sólo calcula un paso.

A los efectos de resolver situaciones como loops capacitivos, se piensa en el concepto de valores previos de las variables de estado, que pueden ser variables "integrales" de otras como el gorge (carga) de un condensador en relación con la corriente que le ingresa. Estos estados previos se "modifican" hasta obtener el "estado inicial". Así dos condensadores en paralelo con cargas previas que implican un voltaje diferente, transfieren sus cargas igualando el voltaje (operación que no conserva la energía eléctrica) y éste es el estado inicial. Toda la resolución del paso es simultánea y se satisfacen a la vez todas las ecuaciones. El programa puede modificar sus variables de estado en cada paso y en caso de requerirlo el integrador, puede aportar ecuaciones adicionales.

Fuentes de variables y ecuaciones

El programa tiene varias fuentes posibles de variables y ecuaciones:

a) Cada componente del circuito tiene sus ecuaciones "constitutivas" y sus variables, algunas de estado. El programa tiene ecuaciones predefinidas para componentes clásicos como R, L, C, V, J pero se pueden introducir componentes con ecuaciones arbitrarias, que vinculen sus

variables, variables de estado y variables diferenciales, de circuito o con cualquier otra del sistema, por ejemplo el tiempo.

b) El circuito y su topología vinculan a los componentes a través de las ecuaciones de Kirchhoff de voltajes y corrientes, "de redes" totalmente lineales. Incluso los cut-sets y loops aportan ecuaciones adicionales de redes, en términos de gorge y flujo.

c) Pueden ingresarse variables, además de los elementos de circuito, incluyendo nuevas de estado, y también las diferenciales de todas ellas, junto con ecuaciones que las relacionan entre sí y con las variables de redes o componentes, en forma directa.

Manejo de variables y ecuaciones

El software, maneja la incorporación de diferentes ecuaciones y la baja de otras, detectando además en forma automática la existencia de loops y cut-sets.

El software elije un conjunto (idealmente mínimo) de variables y ecuaciones resolubles, de todas las disponibles, manejando las ecuaciones de redes (lineales). Se las trabaja y simplifica siguiendo un esquema similar al de Gauss-Seidel. En este marco se detectan los cut-sets inductivos y loops capacitivos, y se plantean ecuaciones y variables específicas cuando aparecen.

Además, contiene rutinas que verifican que variables están en cada ecuación, calculando eventualmente las derivadas. Que una variable esté en una ecuación no necesariamente garantiza que, en las condiciones buscadas, tenga influencia en el valor de la misma.

Con esta información se descartan, por ejemplo, las ecuaciones de redes donde sólo aparezca el potencial absoluto de un nodo, junto con la variable que representa ese potencial.
También se descartan las ecuaciones auto-satisfechas con las condiciones iniciales.

Así se llega al conjunto deseado de ecuaciones y variables. Y se procede a resolverlo, para luego poder calcular el resto de las variables y avanzar al siguiente paso.

Sistemas con ecuaciones puramente algebraicas.

Un sistema de este tipo no contiene derivadas en las ecuaciones (ecuaciones diferenciales), sino que se resuelve algebraicamente, el conjunto de ecuaciones algebraicas, una sola vez (es constante). Vemos a continuación un ejemplo donde el software propuesto resuelve ecuaciones psicrométricas permitiendo el reemplazo de PSICRO, la calculadora psicrométrica (Saravia, 2008).

Los siguientes cuadros muestran por partes el archivo ".d", usado para resolver un caso típico de psicrometría, con ecuaciones aproximadas y unas pocas variables. El archivo ha sido simplificado para su mejor comprensión. El estilo general y los nombres pretenden ser lo más parecidos posibles al archivo de entrada del SCEPTRE, siendo similar a los archivos de todos los ejemplos presentados. En este caso se plantean 3 constantes, 3 funciones, 3 ecuaciones, 3 variables, 3 parámetros, una condición terminal y 6 variables de salida:

- La sección **CONSTANTS** permite introducir valores que el software se encarga de incorporar a todas las funciones y ecuaciones en forma automática apenas las va incorporando.

```
CONSTANTS
 TKO=273.15
 C_PMA =     0.0289645
 C_PMWG =    0.01801534
```

- La sección **FUNCTIONS** permite introducir funciones indicando su nombre y número de variables. Por convención las variables a usar serán siempre x, y, z, t, v, w, en ese orden. El EXPRTK usa funciones de hasta seis variables en la actualidad. Lo que no representa una limitación grave, dado que las funciones son sólo un instrumento de comodidad para escribir las ecuaciones y siempre se pueden colocar expresiones arbitrariamente largas en éstas últimas.

```
FUNCTIONS
  XS(2)=x/y

  PW(1)=exp(if ((x<=0), \
    -5674.5359 / (x+TKO) + 6.3925247 - 9.677843E-03 * (x+TKO)  \
   + .000000622115701 (x+TKO) (x+TKO) + .0000000020747825 (x+TKO) (x+TKO)
(x+TKO) \
   - 9.484024E-13 (x+TKO) (x+TKO) (x+TKO) (x+TKO) + 4.1635019 * log(x+TKO), \
\
    -5800.2206 / (x+TKO) +  1.3914993 - .048640239 (x+TKO) + .000041764768  *
(x+TKO) (x+TKO) \
    - .000000014452093 * (x+TKO) (x+TKO) (x+TKO)  + 6.5459673 * log(x+TKO) \
)) *101325/101418.71683

   WW(1)=x/(1-x)*C_PMWG/C_PMA;
```

- En la sección **EQUATIONS** se ingresan las ecuaciones a resolver, el programa busca las variables para que las expresiones ingresadas resulten cero.

```
EQUATIONS
X*(2501+1.805*T -4.186 * TBH)*(1-XS(PW(TBH),P))\
-(2501-2.381*TBH) *XS(PW(TBH),P)*(1-X)\
+(T-TBH)*C_PMA/C_PMWG*(1-X)*(1 – XS(PW(TBH),P))

PW(T)-PV

WW(X)-W
```

- En la sección **VARIABLES** se ingresan las incógnitas, y se puede introducir un valor semilla para facilitar el trabajo al sistema de resolución de las ecuaciones.

```
VARIABLES
X=.001
W=0.01
PV=1500
```

- **DEFINED PARAMETRS** es la sección que sirve para establecer cuáles son los datos y cuanto valen.

```
DEFINED PARAMETERS
T=20
TBH=15
P=101325
```

- **RUN CONTROLS** es la sección que incorpora las expresiones que al valer uno detendrán el cálculo, como aquí se puso uno el programa realizará una sola paso.

```
RUN CONTROLS
1
```

- La sección **OUTPUTS**, indica qué variables deben salvarse en el archivo de resultados.

```
OUTPUTS
T,TBH,P,X,W,PV
```

Ejecución del software y verificación

Se exhiben diversos resultados que en conjunto con los presentados en (Saravia, 2014), completan los ensayos tendientes a desarrollar un software de simulación que reemplace con beneficios al SCEPTRE (Kuo, 1971). Los sistemas que se incluyen en el trabajo, representativos de los evaluados con el software desarrollado son:

a) circuito tradicional presentado en (Saravia, D. 2014), pero con elementos variables, figuras 2, 3 y 4,
b) circuito con dos cut-sets inductivos, figura 5,
c) circuito con dos loops capacitivos, figura 6,
d) sistema psicrométrico estático (un paso) con ecuaciones algebraicas, tabla 1.

Para los tres primeros sistemas a), b) y c) se muestra su circuito en diagramas ".dia" tal como los representa el SIMUSOL. Se debe notar que el software presente no toma los datos todavía del SIMUSOL sino de archivos similares a los producidos por SIMUSOL del tipo ".d", editados a mano, por lo que los circuitos presentados no contienen las ecuaciones ni definiciones del programa desarrollado sino los usados por SCEPTRE para contrastarlos con el presente desarrollo.

Para el caso d) en la tabla 1 se presentan los resultados de ejecutar el programa propuesto sobre el archivo ".d" descripto anteriormente, los que coinciden con lo informado por la calculadora psicrométrica (Saravia, 2008).

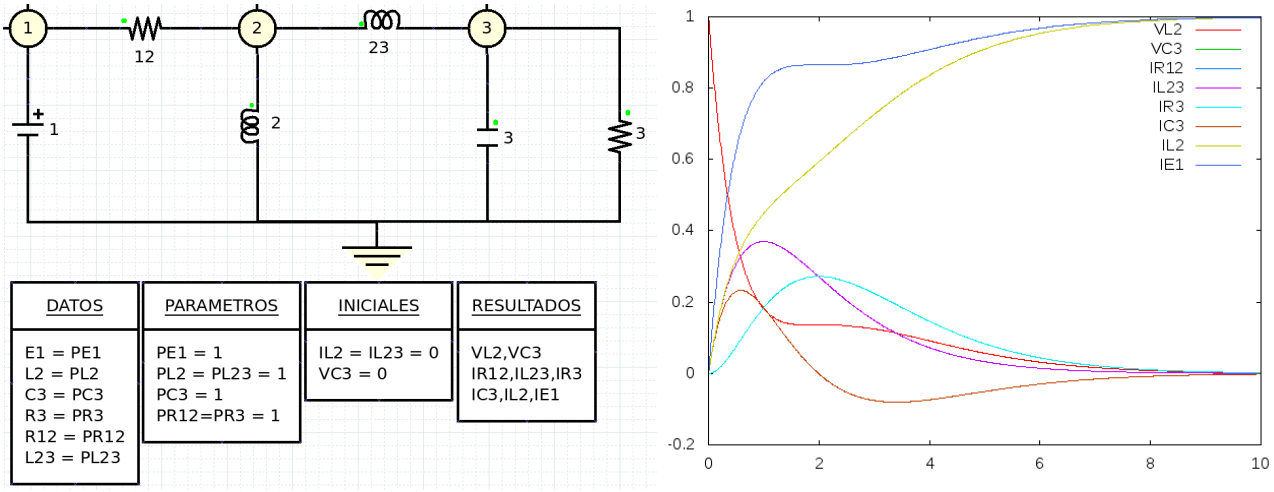


Figura 2: ***Sistema a.*** *Diagrama del SIMUSOL y resultados, coef. constantes, verificado en (*Saravia, 2014).

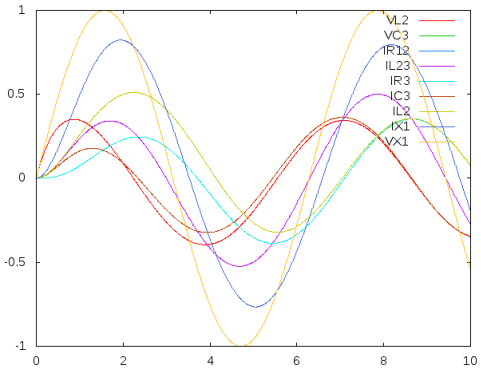


Figura 3: ***Sistema a****. resultados del con fuente de voltaje sinusoidal*

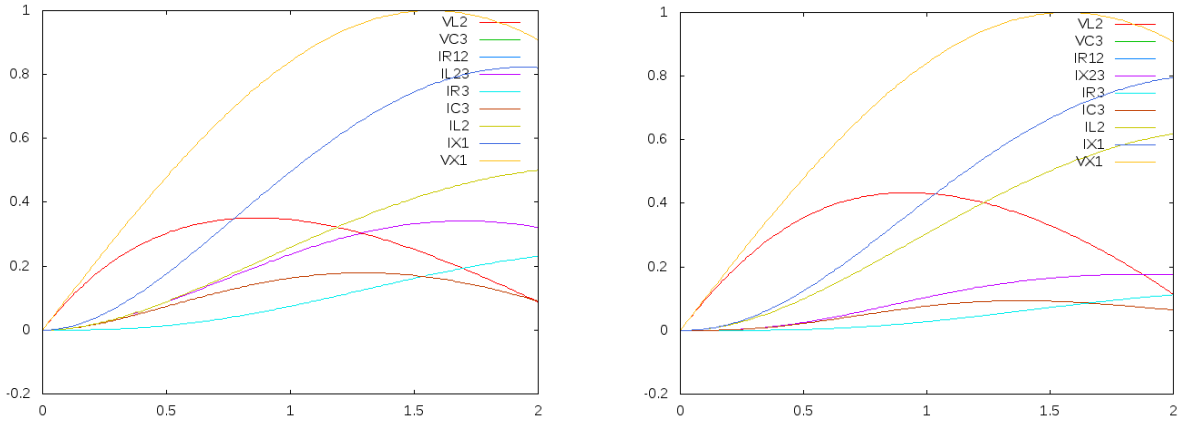
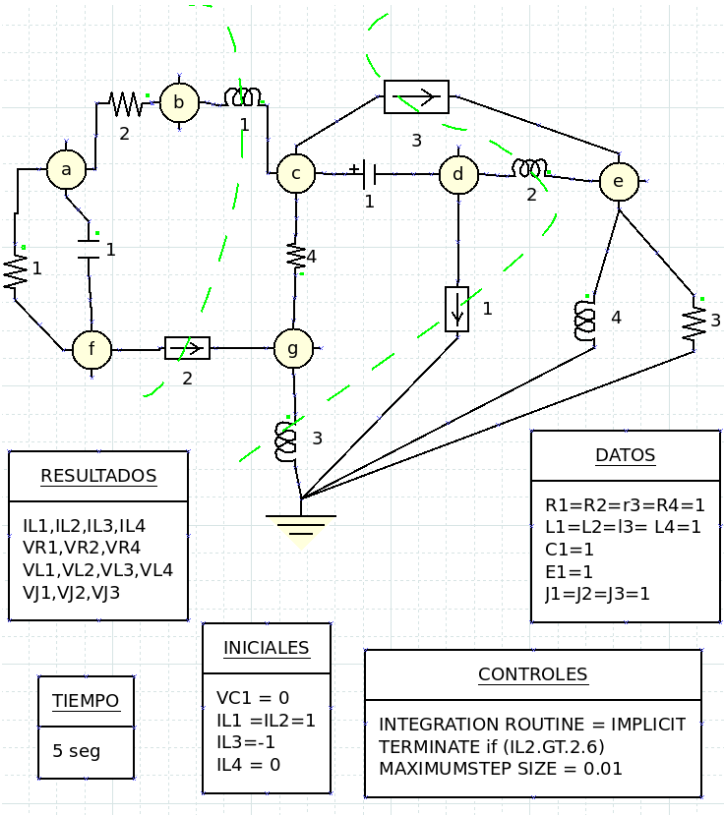
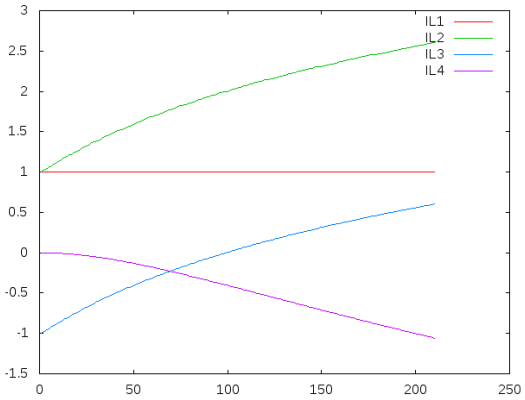


Figura 4: ***Sistema a****. resultados, hasta 2 seg. (Izq.:) fuente sinusoidal. (Der.:) fuente sinusoidal y bobina "cuadrática", donde la variación de corriente es proporcional al cuadrado del voltaje y no al mismo.*





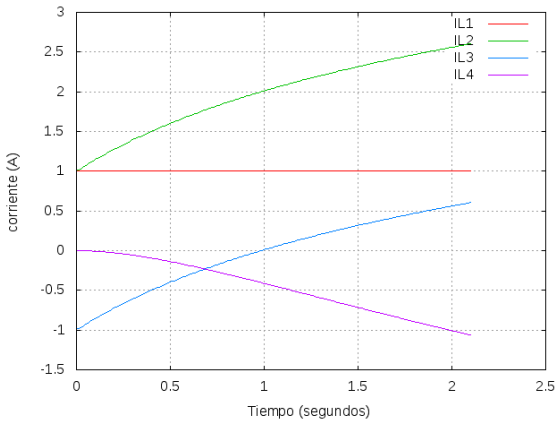
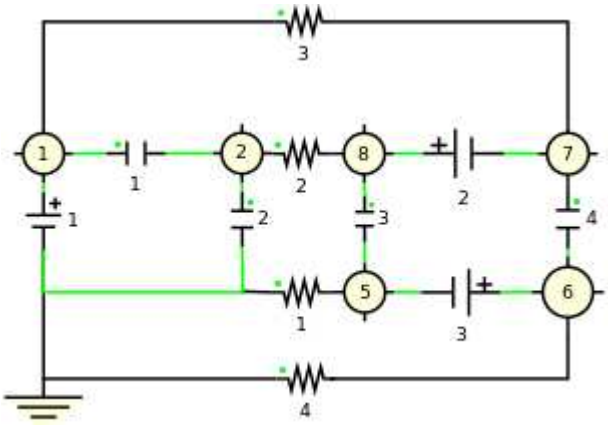
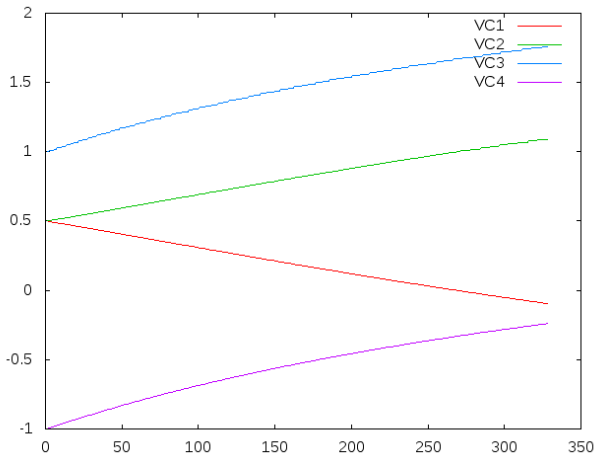
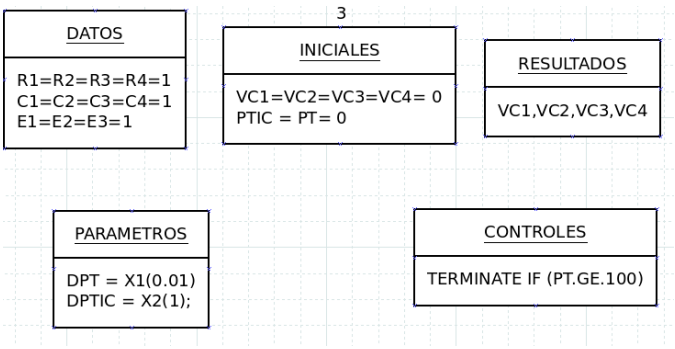


*Figura 5: **Sistema b**. (Izq.:) Diagrama y resultados : 2 cut-sets inductivos, en color verde (Der. Arr.:) programa desarrollado (Der. Aba.:) SCEPTRE.*







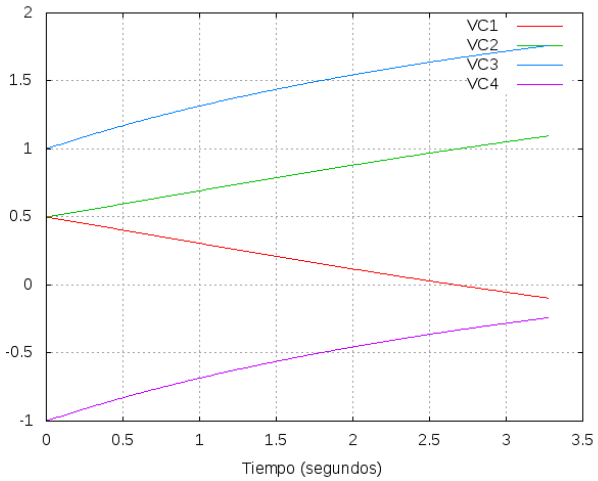


Figura 6: ***Sistema c.*** *(Izq.:) Diagrama y resultados : 2 loops capacitivos, en color verde. (Der. Arr.:) programa desarrollado (Der. Aba.:) SCEPTRE, con condiciones iniciales ajustadas.*

Punto	T	X	H	S	V	PV
central:	20.00	0.01360	1203.78	4.32	0.02405	2336.64
rocio:	11.75	0.01360	963.32	3.49	0.02338	1378.48
bulbo hum:	15.00	0.01682	1202.90	4.33	0.02364	1703.87
saturado:	20.00	0.02306	1630.64	5.80	0.02405	2336.64

#	T
2.00e+01	

Tabla 1: ***Sistema d.****, (Arriba:) resultados de PSICRO para el ejemplo. (Abajo:) programa desarrollado.T: temperatura, X: fracción molar de vapor, H: entalpía, S: entropía, V: volumen, PV: presión de vapor, TBH: temperatura de bulbo húmedo, P: presión, W: humedad. Coinciden los valores, se ven T con T (central, PSICRO), TBH con T (bulbo hum, PSICRO), X con X (central, PSICRO) y PV con PV(central, PSICRO)*

CONCLUSIONES

Se ha comprobado que las herramientas, técnicas, tecnologías y métodos planteados son eficaces para resolver el problema expuesto. Es decir, simular circuitos eléctricos y otros sistemas análogos.

La teoría descripta valida la forma de cálculo usada. El software desarrollado entrega resultados que no se diferencian de los entregados por el software usado de referencia para los circuitos y ejemplos de prueba, lo cuál indica que no se han cometido errores de programación que alteren los mismos.

Se ha completado una segunda etapa en el desarrollo de un software libre capaz de superar el SCEPTRE en varios sentidos. La primera descripta en (Saravia, D., 2014). Se continúa el desarrollo optimizando y fortaleciendo el software para obtener versiones aptas para terceros integradas y distribuidas con SIMUSOL.

AGRADECIMIENTOS

Se agradece a Dolores Alía la preparación de modelos con el SIMUSOL/SCEPTRE para contrastar con los resultados de los desarrollos presentados.

REFERENCIAS

Alía, D., Saravia, L. y Saravia, D. (2002). Avances introducidos en la capacidad del simulador de sistemas solares térmicos SIMUTERM (SIMUSOL). AVERMA, 6, 2, 8.31-8.36. http://www.cricyt.edu.ar/asades/modulos/averma/trabajos/2002/2002-t008-a006.pdf

Alía, D., Saravia, L. y Saravia, D. (2012). SIMUSOL: Simulating thermal systems using SCEPTRE and Dia. JFSFK 1,1, 30-34. http://www.icfoss.org/ojs/index.php/foss/article/view/14

Beazley, D. M., Fletcher, D. y Dumont, D. (1998). Perl Extension Building with SWIG. O'Reilly Perl Conference. O'Reilly. EEUU. http://www.swig.org/papers/Perl98/swigperl.pdf

Beazley, D. M. (1999) Perl-C/C++ Integration with SWIG. Dep. Computer Science. U. Chicago. EEUU. http://es.slideshare.net/dabeaz/perlcc-integration-with-swig

Denker, J. (2014). Capacitor-to-Capacitor Transfer of Energy and Gorge. Sitio web personal. http://www.av8n.com/physics/capacitor-transfer.htm

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., y Rossi, F. (1996). Gnu Scientific Library. Página web del proyecto GSL. http://www.gnu.org/software/gsl/

Günther, M., Feldmann, U. y ter Maten, J. (2005). Modelling and discretization of circuit problems.. CASA (RANA) reports. Centre for Analysis, Scientific computing and Applications. Department of Mathematics and Computer Science of Eindhoven University of Technology (TU/e). Países Bajos. http://www.win.tue.nl/analysis/reports/rana05-37.pdf

Kuo, F. (1971). SCEPTRE: Computer program for circuit analysis. Prentice Hall. EEUU.

Magnuson, W. G. (1982). Electronic Circuit Analysis Using Computer Programs. Computer Program Newsletter. No.7. Lawrence Livermore Laboratory. EEUU. http://www.osti.gov/scitech/servlets/purl/6754836

Mayer, D. (1971). Electric Network analysis by the generalized cut-set matrix method. Aplikace matematiky, 16 ,1, 24–45. Academy of Sciences of the Czech Republic. http://dml.cz/bitstream/handle/10338.dmlcz/103324/AplMat_16-1971-1_3.pdf

Partow, A. (2015). C++ Mathematical Expression Toolkit Library (ExprTk). Sitio web personal. http://www.partow.net/programming/exprtk/

Proyecto SWIG (2015) Sitio web del proyecto. http://www.swig.org/

Saravia, D. (2008). Avances en Psicro, la calculadora psicrométrica. Presentación de Puyuspa y Calcula. AVERMA 12, 8.79-8.84 http://www.cricyt.edu.ar/asades/modulos/averma/trabajos/2008/2008-t008-a011.pdf La calculadora: http://www.psicro.org

Saravia, L. y Saravia, D. (2000). Simulación de sistemas solares térmicos con un Programa de cálculo de circuitos eléctricos de libre disponibilidad. AVERMA 4, 4, 8.17-8.22. http://www.cricyt.edu.ar/asades/modulos/averma/trabajos/2000/2000-t008-a004.pdf

Saravia, D. y Alía, D. (2012). Esándar de documentos empaquetados de SIMUSOL, repositorio CSSAN y otros avances. ERMA 30,21-28. http://www.cricyt.edu.ar/asades/modulos/averma/trabajos/2012/2012-t008-a023.pdf

Saravia, D. (2014). SIMUSOL: simulación de diagramas de enlace o Bond-Graphs. AVERMA 18, 08.43-08.50. http://www.cricyt.edu.ar/asades/modulos/averma/trabajos/2014/2014-t008-a006.pdf

Thulasiraman, K. (2002). Circuit Theory. Encyclopaedia of Physical Science and Technology. Volume 2. Academic Press. EEUU. http://www.cs.ou.edu/~thulasi/Misc/circuit_theory.pdf

Volkman, V. (2005) Expose Your C/C++ Program's Internal API with a Quick SWIG, Sitio web Codeguru. http://www.codeguru.com/csharp/.net/net_asp/scripting/article.php/c11103/Expose-Your-CC-Programs-Internal-API-with-a-Quick-SWIG.htm

Todos los URLs han sido accedidos y verificados el 26-09-2015.

AVERMA: Avances en energías renovables y medio ambiente, ISSN 2314-1433, ERMA: Energías renovables y medio ambiente, ISSN 0328-932X. JFSFK Journal of Free Software and Free Knowledge, 2277-7709.

ABSTRACT

This paper presents advances in a simulation software alternative to SCEPTRE. Both are used in conjunction with SIMUSOL. The new functionality includes fully implicit equations, that use dynamic performers and do not require compilation, and modeling capabilities to detect and calculate circuit systems with inductive "cut-sets" and capacitive “loops”. The first ones made up of "chokes" and "current sources", and the second one of "capacitors" and "voltage sources". This software could replace SCEPTRE with advantages, having no other limit than the computer memory. PSICRO could be replaced by this approach as well.

Keywords: SIMUSOL, SCEPTRE, PSICRO, EXPRTK, SWIG, loop, cut-set-