

Informe Trabajo Práctico Integrador

Valderrama Miguel, Gartner Francisco Nehuen

Introducción a la Programación y Análisis Numérico, Depto. de Ciencias Básicas, Facultad de Ingeniería, UNLP, La Plata, Argentina.

Resumen—Se desea explorar el comportamiento de un circuito en diferentes casos y condiciones, con distintas fuentes de entradas, tan solo conociendo la información de las mediciones de las tensión de entrada, la tensión de salida y las corrientes de algunos elementos. También se buscarán estimar los valores de los elementos pasivos del circuito a través de los datos ya mencionados. Toda la resolución y cálculos será realizada con variados métodos numéricos de estimación

I. INTRODUCCIÓN

El problema abordar consiste en la resolución de un circuito, del cual se encuentra los valores de cada elemento eléctrico que lo compone, siendo estos una resistencia, inductor y un capacitor con una fuente de voltaje en régimen de transitorio (y luego sinusoidal), como se muestra en la Figura 1. Se conoce que el circuito se encuentra inicialmente descargado y que el régimen transitorio comienza en $t_0=0$. Se busca explorar el comportamiento de la tensión de salida $u_s(t)$ cambio la señal de entrada del circuito.

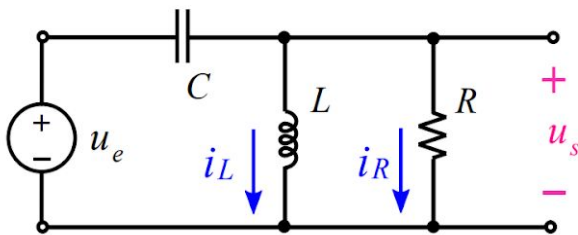


Fig 1. Circuito RLC

Para abordar el problema, se utilizaron las mediciones realizadas con un multímetro y sintetizadas por los archivos *mediciones_iL*, *mediciones_iR* y *mediciones_us*.

Principalmente se utiliza Octave como software de análisis numérico para efectuar cálculos y realizar gráficos. Todos los algoritmos y archivos utilizados para solucionar los problemas mostrados se encontraran adjuntos.

La motivación de resolver el circuito anterior, es la exploración de su comportamiento, en régimen transitorio y sinusoidal.

II. MARCO TEÓRICO

Octave será la herramienta principal a utilizar, este es un software de análisis numérico, escrito en lenguaje C++ usando la biblioteca STL¹.

Todos los algoritmos mencionados pueden ser escritos en un código de Octave y ser ejecutados en el mismo, cada algoritmo además de tener sus ventajas matemáticas, se diferenciarán en su costo computacional (cantidad de operaciones), a menor costo computacional, mejor funcionamiento del algoritmo.

A. Métodos numéricos:

A fin de observar el comportamiento en varios instantes y condiciones se utilizan diversos métodos numéricos, para el estudio y comparación de los casos.

Para los primeros casos mostrados en la próxima sección se utilizan diferentes métodos de ajuste discretos, que consisten en encontrar una curva que pasen por una serie de puntos dados y que pueda cumplir las condiciones dadas de cada método, con un error cuadrático medio (promedio de errores al cuadrado entre el punto y la curva obtenida) bajo.

Cuando a los ajustes discretos se le exigen que cumplan las condiciones dadas y que pasen por todos los puntos, se comienza a llamar interpolación, estos tienen mayor costo computacional pero muchas más ventajas comparados con los ajustes discretos convencionales, como por ejemplo, un error cuadrático medio (ECM) menor.

Se denomina interpolación a la obtención de un polinomio que tiene la restricción de que necesariamente tiene que pasar por todos los puntos dados. Se puede comprobar que este polinomio $p(x)$ existe por el Teorema de Existencia y Unicidad (TEU) como dice en la Ref. [1] “Para un dado conjunto de $N + 1$ puntos o datos, $\{(x_n, y_n)\}_{n=0}^N$, existe un polinomio de grado menor o igual a N que pasa por todos los puntos y el mismo es único.”

Aunque solo existe un solo polinomio $p(x)$, hay diferentes formas de calcularlo. Uno de maneras de estimar el polinomio es a través del método de Newton², este se basa en hallar la fórmula de la interpolación utilizando medios recursivos, este método también se conoce como *Diferencias Divididas*. Para el cálculo de este método se toman n cantidad de puntos para obtener un polinomio $p(x)$ de grado $N-1$, para cada n se tiene un punto x y su imagen $f(x)$ y con estos se comienzan a calcular los coeficientes del polinomio $p(x)$ de la siguiente manera:

¹Standard Template Library

$f_0(x_i)$: término i-ésimo de la secuencia

$$f_1(x_0, x_1) = \frac{f_0(x_1) - f_0(x_0)}{x_1 - x_0}$$

$$f_2(x_0, x_1, x_2) = \frac{f_1(x_1, x_2) - f_1(x_0, x_1)}{x_2 - x_0}$$

$$f_i(x_0, x_1, \dots, x_{i-1}, x_i) = \frac{f_{i-1}(x_1, \dots, x_{i-1}, x_i) - f_{i-1}(x_0, x_1, \dots, x_{i-1})}{x_i - x_0},$$

Luego de obtener los coeficientes se colocan todos en el polinomio $p(x)$ de la siguiente forma:

$$p_i(x) = p_{i-1}(x) + f_i(x_0, x_1, \dots, x_{i-1}, x_i) \prod_{j=0}^{i-1} (x - x_j)$$

Así obtenemos el polinomio $p(x)$ de grado $N-1$ de interpolación por método de Newton. Esta forma es fácil de implementar en algoritmos, tiene un costo computacional moderado de $O(N^2)$, si se necesita agregar nuevos puntos este aprovecha los cálculos ya realizados anteriormente, sin embargo si se desea modificar un dato $(x_n, y_n)_{n=0}^N$ se deberá recalcular todo el sistema.

Una gran desventaja del método de Newton es la desestabilidad de este en sistemas mayores a 45x45, este problema viene dado por el costo computacional del método, que hace que no se cumplan restricciones dadas. A todo esto también se le suma el Fenómeno de Runge, este es un comportamiento indeseado propio de los polinomios. Un polinomio de alto grado presenta oscilaciones abruptas y por ello se producen errores grandes, aún en intervalos pequeños.

Una manera de evitar estos problemas es empleando otro método de interpolación llamado *Splines*, este reduce considerablemente la inestabilidad y el fenómeno de Runge.

Los Splines se suelen utilizar cuando la cantidad de puntos a interpolar es muy grande o se exigen una muy buena precisión numérica. Este procedimiento en vez de exigir a un solo polinomio $p(x)$ que cumpla todas las restricciones impuestas, este subdivide a $p(x)$ en varias trazas, polinomios o splines. El método se puede catalogar según su grado n empezando en $n=1$ (*Spline lineal*, *Spline cuadrático*, *Spline Cúbico*...). Cada Spline de grado n se calcula de una manera diferente, para $N+1$ puntos $\{(x_n, y_n)\}_{n=0}^N$, se definen N trazas o *Splines* de grado n .

Para un Spline lineal (grado 1), cada polinomio tendrá la forma de $P_n(x) = a_n + b_n(x - x_n)$, la condición que estos deben cumplir es que las trazas coincidan en los puntos dados, es decir, donde termina un polinomio $P_n(x)$ debe comenzar el siguiente polinomio $P_{n+1}(x)$. Un Spline cuadrático (grado 2) tiene la estructura de $P_n(x) = a_n + b_n(x - x_n) + c_n(x - x_n)^2$, este además de cumplir las condiciones del del Spline lineal, también debe cumplir que la pendientes en la unión de trazas, sean las mismas, es decir, $P'_n(x) = P'_{n+1}(x)$. Por último se explicará los Splines cuadráticos, (grado 3), este asimismo debe cumplir las condiciones de los anteriores agregando una nueva,

en la unión de las trazas, la concavidad debe tener la misma concavidad, es decir, $P''_n(x) = P''_{n+1}(x)$. Este tiene la forma de $P_n(x) = a_n + b_n(x - x_n) + c_n(x - x_n)^2 + d_n(x - x_n)^3$.

Se considera suficiente un Spline de grado 3 para interpolar cualquier conjunto de puntos, aunque existen splines de grados $n \geq 3$, estos son más difíciles de implementar en un algoritmo y requieren más costo computacional.

Todos los métodos numéricos explicados hasta ahora son para ajustes discretos e interpolación a través de polinomios, con estos se puede aproximar que sucede en el intermedio de los puntos, o fuera del intervalo, sin embargo en algunas ocasiones desea conocer también el área cubierta debajo de los puntos, esto se logra realiza con integrales, pero a su vez trae un problema, realmente no se conoce la expresión real del polinomio $p(x)$, por esto se utilizan métodos de integración numérica, que se originan en la interpolación pero hacen el cálculo que se necesita con un error menor.

Usualmente la aproximación de la integrales numéricas suele realizarse utilizando trapecios para estimar el área debajo de una curva con un error de $O(h^3)$, no obstante existente un método mejor que aproxima la integral, dado que tiene un error de $O(h^5)$, este método es llamado la Regla de Simpson. El método se produce de un polinomio de interpolación de grado 2, este en vez de usar trapecios utiliza parábolas para aproximar el área de la integral. La demostración puede encontrarse en la Ref. [2].

El circuito a estudiar se rige con Ecuaciones Diferenciales Ordinarias (EDO), por lo tanto, se necesita una herramienta para resolver estas mismas.

Existen varios métodos de resolución dependiendo del orden n de la EDO, para una de orden 1, puede resolverse con el método de Euler, este se basa en estimar la solución de la ecuación a través de rectas tangentes; para orden 2, está el método de Taylor, con este se aproxima la solución mediante polinomios de Taylor; con orden 3, tenemos el método de Adams-Bashforth, estima el resultado de la ecuación diferencial usando el polinomio de interpolación de Newton, y por último el método de Runge-Kutta para orden 4, este último puede ser utilizado con EDO de varios órdenes, este basa en obtener aproximaciones integrando la EDO. Todas las demostraciones en muestran en la Ref. [3].

B. Leyes y teorías:

Con la finalidad de estudiar el circuito se utilizaran algunas leyes y teorías básicas que ayudan al simplificarlo, y hacer su estudio más sencillo.

Una ley básica pero bastante útil para el análisis del circuito es la ley de Ohm, esta establece como la tensión (V) en un elemento resistivo es proporcional a la corriente (I) que por él circula y a su valor de resistencia (R), siendo la fórmula general de la ley de Ohm $V = IR$.

Se utilizará el concepto de mediciones ideales, estas se definen como aquellas que poseen 100% de precisión y 100% de exactitud. Se tratara sólo como un concepto ya que estas no

tienen un valor real en el mundo físico, es decir, su exactitud y precisión no son del 100%.

III. SOLUCIONES NUMÉRICAS

Se quiere explorar el circuito para diferentes casos y condiciones, se comienza estudiando la tensión de la salida del circuito ($u_s(t)$) en instantes de tiempo $t \in [0, 3 \cdot 10^{-3}]$ con una tensión de entrada $u_e(t) = \{A = 10V \text{ si } t \geq t_0 = 0 \text{ y } A=0V \text{ caso contrario}\}$.

Se utilizará el archivo *mediciones_us.m* el cual sintetiza la tensión de salida del circuito para valores de $t \in [0, 3 \cdot 10^{-3}]$, para analizar y comparar el comportamiento del circuito en distintos instantes, indicando diferentes porcentajes del valor de lectura (*pvl*) y la cantidad de veces la cifra menos significativa (*ccms*).

En el primer caso, se suponen mediciones “ideales” (*pvl*=0% y *ccms*=0), es decir sin error, utilizando el archivo *mediciones_us.m* se consigue 22 puntos equidistantes en $t \in [0, 3 \cdot 10^{-3}]$, y con el archivo *Splines.m* se realizan tres interpolaciones con el método *Spline* (*lineal*, *cuadrático* y *cúbico*). Se elige este método por su alta estabilidad (reduce notablemente el efecto de *Runge*), poco costo computacional y gran ajuste. Los resultados obtenidos se muestran en la Fig 2.

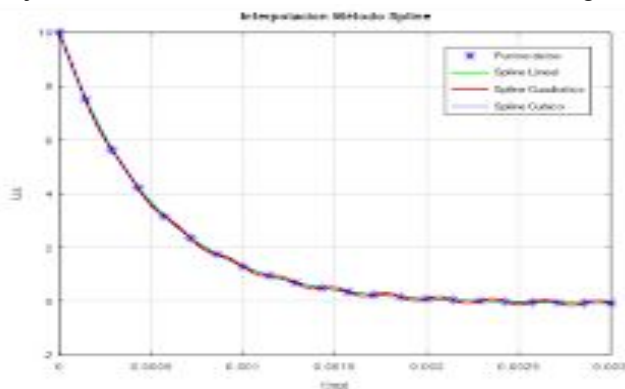


Fig 2. Interpolación (Spline), 22 puntos, *pvl*=0% y *ccms*=0.

Para el segundo caso se repite el procedimiento anterior, pero ahora con *pvl*=5% y *ccms*=10, los resultados se muestran en la Fig 3.

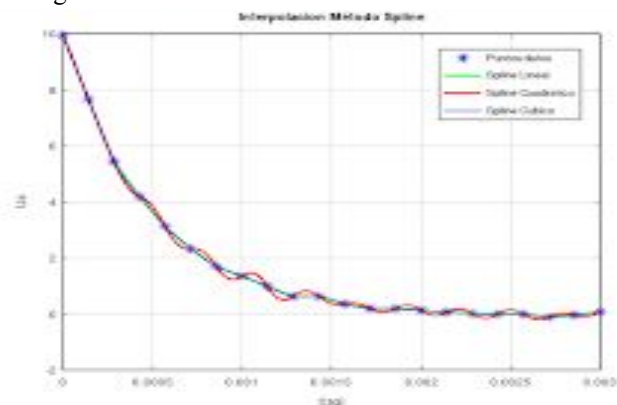


Fig 3. Interpolación (Spline), 22 puntos, *pvl*=5% y *ccms*=10.

Ahora se utiliza el mismo caso pero la interpolación será realizada con 60 puntos, los resultados se muestran en la Fig 4

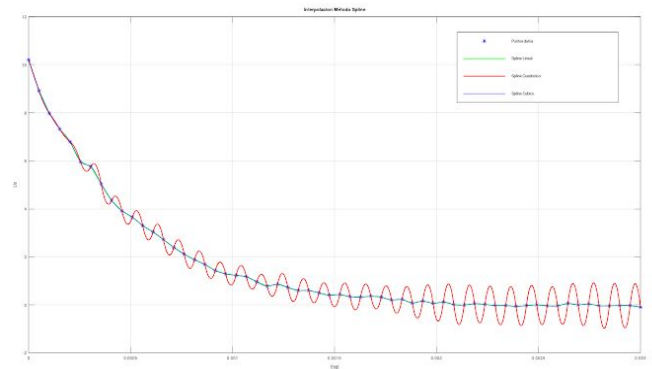


Fig 4. Interpolación (Spline), 60 puntos, *pvl*=5% y *ccms*=10.

Si se utilizara otro método de interpolación, como el método de Newton no se podría apreciar el comportamiento del circuito, por la inestabilidad de este mismo, se demuestra en la Fig 5. Se puede recrear utilizando el archivo *NewtonInterpolacionError.m*.

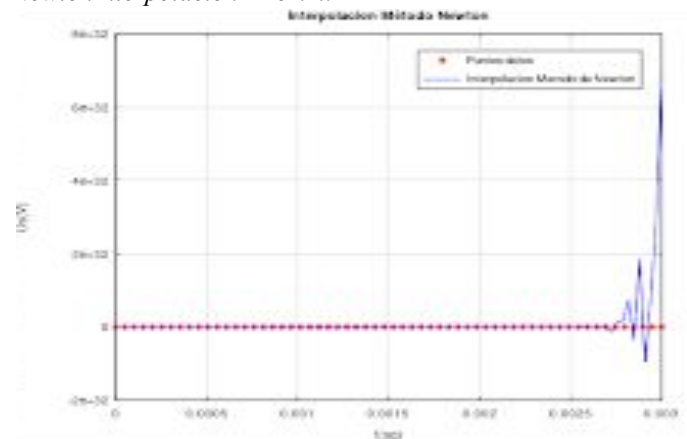


Fig 5. Interpolación (Newton), 60 puntos, *pvl*=5% y *ccms*=10.

Con los últimos datos dados por *mediciones_us.m* (60 puntos, *pvl*=5% y *ccms*=10), se realiza un ajuste polinomial de grado 2 usando el archivo *ComparacionAjustes.m*, este se basa en la interpolación de polinomios utilizando la matriz de Vandermonde, para realizar el ajuste del polinomio $p(x)$, que tiene la forma $p(x) = a + bx + cx^2$.

También se realiza un ajuste exponencial utilizando el mismo archivo, entregando un polinomio $g(x)$ exponencial. Un polinomio exponencial tiene forma $g(x) = Ae^{bx}$, y tiene una restricción importante, $g(x)$ no puede ser negativa si $A \geq 0$, para nuestro ajuste exponencial se dice que estas condiciones son estrictas, sin embargo el archivo *mediciones_us.m* del cual recibimos los puntos del ajuste, datos a ser tomados como $g(x)$, tiene puntos negativos, dado que son pocos, estos puntos son descartados para realizar el ajuste.

En la Fig 6 se muestra una comparación entre ambos ajustes.

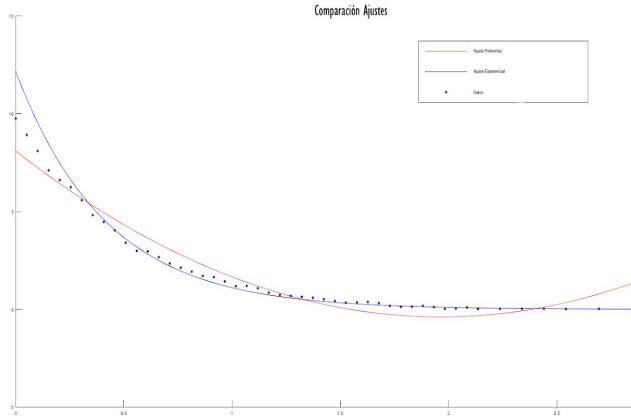


Fig 6. Comparación entre ajuste polinomial de grado 2 y ajuste exponencial (60 puntos, $pvl=5\%$ y $ccms=10$).

Ahora se quiere conocer el valor de los elementos pasivo del circuito (Bobina, Capacitor y Resistencia).

Se comenzará con el más sencillo que es la resistencia, para ello se utilizará el archivo *mediciones_us.m* y *mediciones_iR.m*, como ya mencionamos el primero mencionado nos expresa la tensión de salida $u_s(t)$, que para el circuito es la misma tensión que recibe la resistencia R ($[\Omega] = Ohmios$), para el segundo archivo *mediciones_iR.m*, este indicará la corriente que pasa a través de la resistencia en instantes de tiempo $t \in [0, 3 \cdot 10^{-3}]$, indicando también el pvl y $ccms$.

Primero se tendrá en cuenta una medición ideal, es decir, $pvl=0\%$ y $ccms=0$, con 22 puntos equidistantes. Los datos obtenidos serán dos vectores 22×1 ($[us]=tensión$, $[iR]=corriente$), con estos se aplica la ley de Ohm en el archivo *Rvalor.m*, el cual contiene un algoritmo que realizara todo esto en n veces, con estos se realiza una estimación de la resistencia R utilizando el promedio. Se asegura que es una buena estimación al ser el promedio un estimador insesgado del Error Cuadrático Medio (ECM). Sin embargo para un $pvl=0\%$ y $ccms=0$, no se tendrá problemas para la estimación, ya que estos datos no varían cada vez que se inicia un bucle en el algoritmo. Ahora se realiza la prueba para un $pvl=5\%$ y $ccms=10$. En esta instancia se puede notar que cada vez que se ejecuta el algoritmo, el promedio de cada n varía, y el algoritmo comienza a tener problemas, dado que al utilizar la ley de Ohm para $R = V/I$, los vectores que representan a V e I ($[us]$ y $[iR]$ respectivamente) tienen “malas” mediciones al final del intervalo (la precisión es muy baja al querer medir décimas o centésimas de Volts o Amperes sin cambiar la escala del instrumento), al ser estos números negativos (no válidos para una ser el valor de una resistencia) o muy cercanos a 0, lo cual hace que nuestros cálculos se vean distorsionados, algunas resistencias del promedio pueden hasta tener un valor Infinito (valor nada razonable). Se tienen dos casos para solucionar esto, el primer es hacer que en cada bucle, el algoritmo revise los vectores $[us]$ y $[iR]$, e ignore los valores menores a 0.01 en

ambos vectores, si esto sucede en un vector y en el otro no, esté descarta el instante t_0 en ambos vectores, y en el intervalo $t \in [0, 3 \cdot 10^{-3}]$, resultando así vectores del mismo tamaño $n \times m$.

No obstante en este caso se estaría ignorando puntos, que podrían ser mejor utilizados, así que se utilizará de modo conveniente el segundo caso, que consistirá en utilizar un intervalo $t \in [a, b]$, con un b menor a $3 \cdot 10^{-3}$ para así no tener los puntos problemáticos y mantener la cantidad de puntos. Se utiliza el intervalo $t \in [0, 3 \cdot 10^{-4}]$, con el que se logra estimar de manera exitosa la resistencia R .

Luego se desea conocer el valor de la bobina L para esto se recordará las ecuación constitutiva:

$$L = \left(\int_{t_0}^t u_L(t') dt' \right) / i_L(t) = L(t_n) \quad (1)$$

Con esta ecuación se puede saber el valor de la inductor L ($[H] = Henrys$), se comienza con un intervalo de tiempo $t \in [0, 3 \cdot 10^{-3}]$, 22 puntos equidistantes, un $pvl=5\%$ y $ccms=10$. Se conoce que la tensión de salida $u_s(t)$ es la misma que la tensión en el inductor $u_L(t)$ por lo tanto se utilizará el archivo *mediciones_us.m* para expresar $u_L(t)$. Para sintetizar la la corriente que pasa por el inductor en instantes t_0 se utilizará el archivo *mediciones_iL.m*.

Se comienza el algoritmo obteniendo los datos de *mediciones_us.m* y *mediciones_iL.m*, dando como resultados dos vectores 22×1 ($[us]$ y $[iL]$), luego se procede a realizar la integral numérica de $[us]$ a través del método de Simpson, con este dato, se utiliza la ec. (1), así se adquiere el valor del inductor L en instantes de tiempo $t \in [0, 3 \cdot 10^{-3}]$. Se le hace realizar al algoritmo en procedimiento en bucles varias veces nuevamente para así tener un promedio de estos y este sea una mejor aproximación de L . Otra vez se presenta una oscilación muy grande, en este caso, para el valor del inductor L . Revisando los datos cuidadosamente se nota como esta vez el vector $[iL]$ tiene “malas” mediciones al principio de intervalo, dado esto, se cambia el intervalo a $t \in [10^{-3}, 3 \cdot 10^{-3}]$, así se arregla el problema y se consigue un resultado satisfactorio. Todo el algoritmo dicho se ejecuta con el archivo *Lvalor.m*

Para el capacitor C ($[F] = Faradios$), se calcula utilizando la ecuación constitutiva:

$$C = \left(\int_{t_0}^t i_C(t') dt' \right) / u_C(t) = C(t_n) \quad (2)$$

Se empieza con un intervalo $t \in [0, 3 \cdot 10^{-3}]$, 22 equidistantes, un $pvl=5\%$ y $ccms=10$. Se inicia el algoritmo ejecutando los archivos *mediciones_us.m*, *mediciones_iL.m* y *mediciones_iR.m*, así obtenemos nuestro datos iniciales, tres vectores 22×1 ($[us]$, $[iL]$ y $[iR]$ respectivamente). Se conoce que la corriente del capacitor $i_C(t)$ es la suma de la corriente que pasa por el inductor $i_L(t)$ y la corriente de la resistencia

R $i_R(t)$. Se realiza la integral numérica de $i_C(t)$ utilizando la propiedad de linealidad como se muestra en Ec. (3), dado que no tenemos datos de la corriente $i_L(t)$:

$$I = \int_{t_0}^t i_C(t')dt' = \int_{t_0}^t i_R(t')dt' + \int_{t_0}^t i(t')dt' \quad (3)$$

Estas integrales son resueltas numéricamente por método de Simpson, el vector [iR] representa el valor de $i_R(t)$ y [iL] representa el valor de $i_L(t)$, en instantes de tiempo $t \in [0, 3.10^{-3}]$.

Para el cálculo de la tensión en el capacitor $u_C(t)$ se conoce que esta es la diferencia entre la tensión de entrada $u_e(t)$ y la tensión de salida $u_s(t)$. La tensión $u_e(t)$ se dice que es fija pues vale lo mismo para cada instante de tiempo $t \geq t_0 = 0$. Como $u_s(t)$ es representado por vector 22x1 [us] y $u_e(t)$ es un escalar, para calcular $u_C(t)$, se restan $u_e(t)$ menos $u_s(t)$ punto a punto, lo cual sigue generando un vector 22x1 (ahora llamado [uc]). Con todos los datos obtenidos, se pasa a realizar la Ec. (2). En este caso la problematica esta en el vector [uc], que tiene valores muy cercanos 0 en algunos puntos, esto se resuelve implementando al algoritmo la condición de que si el valor de [uc], es menor a 0.01 será descartado el punto en este vector y los otros vectores de prueba para este caso. Nuevamente esto se realiza en bucle n cantidad de veces para hacer un promedio del valor del capacitor C. Todo el algoritmo propuesto se encuentra en el archivo *Cvalor.m*.

Se desea conocer el comportamiento de la tensión de salida $u_s(t)$ en diferentes casos e instantes, se sabe que la ecuación diferencial que rige el comportamiento es:

$$\frac{\partial^2 u_s(t)}{\partial t^2} + \frac{1}{RC} \frac{\partial u_s(t)}{\partial t} + \frac{u_s(t)}{LC} = \frac{\partial^2 u_e(t)}{\partial t^2} \quad (4)$$

La solución analítica de esta ecuación es:

$$u_s(t) = A \frac{s_1 e^{s_1 t} - s_2 e^{s_2 t}}{s_1 - s_2}, \quad t \geq t_0 = 0;$$

$$\text{con } \begin{cases} s_1 = -\frac{1}{2RC} - \sqrt{\left(\frac{1}{2RC}\right)^2 - \frac{1}{LC}} \\ s_2 = -\frac{1}{2RC} + \sqrt{\left(\frac{1}{2RC}\right)^2 - \frac{1}{LC}} \end{cases}$$

Se realizará la comprobación numérica de la solución de esta ecuación utilizando los métodos de Euler, Taylor, Adams-Bashforth y Runge-Kutta. Se planteó un algoritmo de resolución de EDO de primer orden para cada método, sin embargo la EDO que rige la tensión $u_s(t)$ es de segundo orden, dado esto, se realiza un cambio de variables $\frac{du_s(t)}{dt} = u_1(t)$ y $\frac{du_1(t)}{dt} = u_2(t)$, siendo $u'_2(t) = u_1(t)$, con esto se plantea un

sistema de ecuaciones diferenciales, con todas siendo de primer orden y se puede utilizar el archivo *ComparacionEDO.m* sin ningún problema, este muestra una comparación entre los diferentes métodos mencionados.

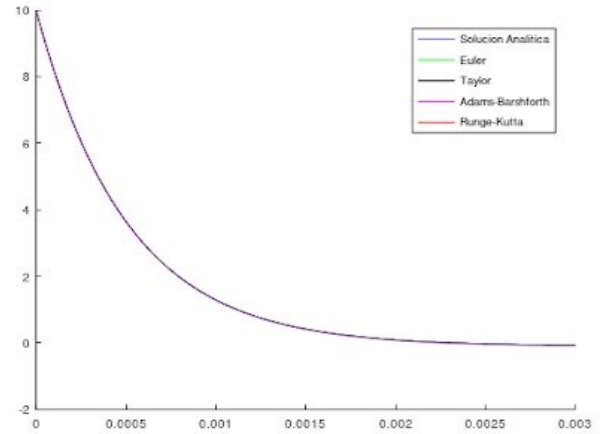


Fig 7. Comparación entre métodos de resolución de ecuaciones diferenciales

Todos los cálculos realizados hasta ahora, han sido en torno a que la tensión de entrada $u_e(t)$ es una fuente de tensión continua, pero ahora se desea saber la respuesta de la tensión $u_s(t)$ si la tensión $u_e(t)$ es igual a $\text{sen}(\omega t)$, es decir una entrada de tipo sinusoidal con $\omega = 2\pi f$, en este caso se estudian tres frecuencias diferentes $f = 1 \text{ Hz}$, $f = 100 \text{ Hz}$ y $f = 300 \text{ Hz}$. Dado que conocemos que el algoritmo para resolver EDO funciona correctamente, y también se sabe una estimación de los errores de cada método, se procede a replantear la ecuación con la cambiando la entrada $u_e(t)$ por $\text{sen}(\omega t)$. Conociendo la solución analítica de la EDO homogénea $y_h(t)$ que rige el circuito, se pudo encontrar fácilmente la solución particular $y_p(t)$ y se sabe que la solución general de la EDO es $y_g(t) = y_h(t) + y_p(t)$, esta ecuación es resuelta a través del método de Runge-Kutta. El algoritmo propuesto se encuentra en el archivo *RK5.m*.

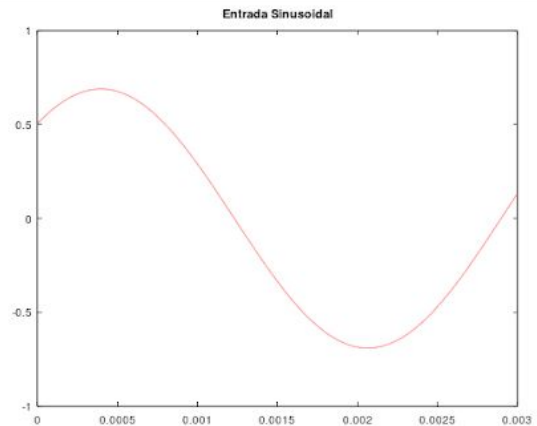


Fig 7. $u_s(t)$ con fuente de entrada sinusoidal $f=300 \text{ Hz}$.

Ahora se quiere conocer el comportamiento de $u_s(t)$ si la señal de entrada $u_e(t)$ es un fragmento de audio (*audio.mat*). Si se escucha el audio, en primer instancia, solo se escucha ruido blanco.

Al cargar el archivo *audio.mat* obtenemos un vector *sig* y un paso *h*, con esto datos se comienza a trabajar. Primero el vector *sig* es utilizado como puntos, se utiliza una variable llamada *diff* para guardar la diferencias entre un punto *n* y su siguiente *n+1*. Por consiguiente la entrada $u_e(t)$ se plantea como $u_e(t) = \frac{diff}{T}$.

Utilizando el método de resolución de ecuaciones diferenciales Runge-Kutta de orden 4, se llega a una solución del sistema y se procesa el audio. El algoritmo que resuelve este problema se encuentra con el nombre *audio.m*

IV. RESULTADOS Y DISCUSIÓN

Se pudo observar la tensión $u_s(t)$ en diferentes casos. Cambio de comportamiento cada vez que $u_e(t)$ se alteró la tensión $u_e(t)$.

En el primer caso siendo $u_e(t)$ una fuente de voltaje continua, $u_s(t)$ se comportó como una caída de tensión de manera exponencial. El método de splines reveló esto con una gran exactitud (también dependiendo del *pvl ccms*) pero se puede notar como el spline cuadrático tiene muchas oscilaciones, esto se puede deber a que quizás las consideraba cambios bruscos en las derivadas rápidamente.

Cuando se realiza el ajuste de los puntos en este caso con un polinomio de grado 2 y un otro ajuste de manera exponencial, se puede comprobar lo anteriormente dicho, es decir, la tensión $u_s(t)$ se comporta en este caso de forma exponencial.

A partir de los archivos sintetizadores de las corrientes y de las tensiones también se pudo estimar los elementos pasivos del circuito, es decir, la resistencia, el capacitor y la bobina. Se estimaron a través de sus promedios, realizando 1000 pruebas gracias a los algoritmos, que permitió reducir el ECM. Los resultados son presentados en la Tabla 1.

TABLA I: Valores de los Elementos Pasivos

Elemento	<i>pvl=0%</i> y <i>ccms=0</i>	<i>pvl=5%</i> y <i>ccms=10</i>
Resistencia [Ω]	2.0113	2.0259
Capacitor [<i>F</i>]	245,79 μ	246,59 μ
Inductor [<i>H</i>]	0.10164	0.10171

Para el caso *pvl=0%* y *ccms=0* con una sola prueba era suficiente para estimar el promedio, dado que este, en este caso, no tiene variaciones.

Conociendo los valores de cada elemento, se pudo comprobar la EDO que define el comportamiento de este

circuito, utilizando métodos numéricos.

Cuando la fuente de entrada $u_e(t)$ fue sustituida por una fuente de entrada de tipo sinusoidal $u_e(t) = \sin(\omega t)$, siendo $\omega = 2\pi f$, la tensión de salida se comportaba de manera diferente dependiendo de la frecuencia *f*, a una frecuencia muy baja como $f = 1 \text{ Hz}$, la tensión $u_s(t)$ en los instantes $t \in [0, 3.10^{-3}]$, tenía una conducta, que a plena vista se podría ser que la entrada era una fuente de continuas, sin embargo a frecuencias mayores, aproximadamente a $f = 100 \text{ Hz}$ se empezaba a comportar de forma sinusoidal, apenas apreciable. Si subimos la frecuencia a $f = 300 \text{ Hz}$, se aprecia totalmente el comportamiento sinusoidal de $u_s(t)$.

Por último se cambio la entrada por una señal del fragmento de *audio.mat*, como se dijo, en un principio este audio era solo

un ruido blanco, pero después de ser procesado con el método de Runge-Kutta se puede apreciar el canto “Aleluya”.

CONCLUSIONES

Cada alteración realizada a la tensión de entrada cambia el comportamiento del circuito, principalmente se observó la tensión de salida. Este circuito tan sencillo puede ser utilizado en, por ejemplo, sistemas de sonidos que reciben señales de entradas mecánicas, se convierten en señales eléctricas, para luego ser decodificadas y convertidas en sonido de nuevo.

REFERENCIAS

- [1] E. Maximenko, “Existencia y unicidad del polinomio interpolante,” 2012, pp.1-3.
- [2] “Integración Numérica,” en Cálculo, tema IV, Universidad del Salvador, 2016, pp. 88-97.
- [3] Jose S. Canóvas Peña, “Métodos numéricos para las ecuaciones diferenciales” 2009 (Departamento de matemática aplicada y estadística, España).