Circuitos elétricos II – primeiro semestre de 2010
Prof. Antonio Carlos Moreirão de Queiroz

Trabalho: Implementar um programa que analise um circuito linear invariante no tempo usando análise nodal modificada, e calcule as transformadas de Laplace das soluções usando o "algorítmo de eliminação", como descrito no livro "Basic Circuit Theory", de Desoer e Kuh, página 597. As páginas pertinentes do livro estão ao fim deste documento.

O programa deverá ler o circuito de um arquivo de texto contendo o "netlist", com o formato:
Primeira linha: É um título (ou o número de nós, se o netlist for gerado pelo programa Edfil). Ignorar.
Linhas seguintes: Um componente descrito por linha, em formato livre:

Resistor: **R**<nome> <nó1> <nó2> <Resistência>
Indutor: **L**<nome> <nó1> <nó2> <Indutância> [**IC=**<corrente inicial>]
Capacitor: **C**<nome> <nó1> <nó2> <Capacitância> [**IC=**<tensão inicial>]
Fonte de tensão controlada a tensão: **E**<nome> <nóV+> <nóV-> <nóv+> <nóv-> <Av>
Fonte de corrente controlada a corrente: **F**<nome> <nóI+> <nóI-> <nói+> <nói-> <Ai>
Fonte de corrente controlada a tensão: **G**<nome> <nóI+> <nóI-> <nóv+> <nóv-> <Gm>
Fonte de tensão controlada a corrente: **H**<nome> <nóV+> <nóV-> <nói+> <nói-> <Rm>
Fonte de corrente: **I**<nome> <nó+> <nó-> <Parâmetros>
Fonte de tensão: **V**<nome> <nó+> <nó-> <Parâmetros>
Acoplamento entre indutores: **K**<nome> <nome de indutor> <nome de indutor> <k>
Amplificador operacional ideal: **O**<nome> <nó saída> <nó saída> <nó entrada> <nó entrada>
Comentário: *<comentário>

Os nós podem ser nomes. Veja o exemplo MNA1 para ver como atribuir números aos nomes. Se o programa Edfil for usado para gerar o netlist, os nós serão números, mas o programa pode continuar a tratar como se fossem nomes.

Os parâmetros para as fontes devem ser:
<parâmetros> = <valor> <IMPULSO ou DEGRAU>

O programa deve montar um sistema nodal modificado onde os termos sejam polinômios (representados internamente por vetores de coeficientes numéricos), e resolver o sistema pelo método da eliminação, obtendo numeradores e denominadores das variáveis do sistema, na forma de polinômios de "*s*". A forma descrita no livro acha uma das variáveis, mas é simples adaptar o algorítmo para achar todas, resolvendo a triangularização para todas as saídas.

As saídas geradas devem ser salvas em arquivos, de forma a que possam ser processadas por algum outro programa que plote respostas em frequência ou inverta as transformadas para obter resposta no tempo.

É conveniente deixar os polinômios com coeficiente de grau mais alto 1, colocando constantes multiplicando os polinômios. A constante multiplicando o denominador deve ser ajustada para 1, com apenas o numerador ficando com constante diferente de 1.

Um trabalho similar foi feito no curso há algum tempo, mas com resolução pelo (ineficiente) método de Cramer. O programa LapMNA pode ser usado para verificar os resultados do programa e para plotar as respostas em frequência correspondentes às transformadas (o que só faz sentido no caso de condições iniciais nulas e entradas impulsionais). É possível que o algorítmo produza resultados diferentes, por cancelar termos comuns nos numeradores e denominadores.

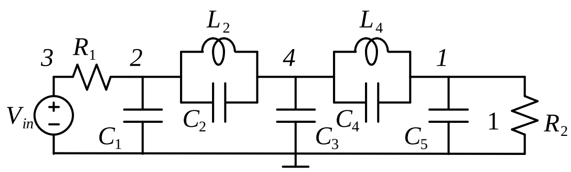
Uma possibilidade para o tratamento de fontes em degrau é derivar as entradas, tratando todas as fontes em impulso como "doublets" (*ks*) e as fontes em degrau como impulsos. Assim todos os polinômios do sistema são de primeira ordem e entradas em impulso e degrau são diretamente possíveis. Para obter a solução correta, basta deslocar os coeficientes do denominador obtido de 1 para maior grau, ou melhor, se possível, deslocar os coeficientes dos numeradores de 1 para menor grau, caso todos eles tenham o coeficiente constante nulo.

O programa LapElim, que implementa o algorítmo como pedido, está já disponível. Ele também plota respostas em frequência, como o LapMNA faz.

Para testar o programa, coloque a princípio circuitos com solução conhecida, mas progressivamente mais complicados, como uma cadeia de resistores em série com uma fonte de corrente, um circuito com vários indutores em paralelo ou vários capacitores em série, etc. O circuito do exemplo abaixo pode ser tornado mais complexo com a substituição dos indutores por giradores ligados a capacitores, e os giradores podem ser montados com transcondutores ou transresistores. Modelos com amplificador operacional podem ser usados também, por exemplo na fonte de tensão, e mesmo para os indutores. Há vários exemplos no arquivo do programa LapElim.

Exemplo:

Seja o circuito, que é um filtro passa-baixas elíptico de 5ª. ordem:



Netlist como gerado no Edfil:

4
R2 1 0 1
R1 2 3 1
L2 4 2 5.86082142617873E-1
L4 1 4 8.81627694904598E-1
C1 2 0 1.41517470742647
C3 4 0 2.13067465696923
C5 1 0 1.84421824323888
C2 4 2 1.08537480612191
C4 1 4 3.64397675632798E-1
VIN 3 0 1 IMPULSO

Polinômios obtidos pelo LapMNA (grau, coeficientes e constante multiplicativa):

Denominador:
5
0.229849081204
0.788126922436
1.129227734851
1.847118770184
0.923399216741
1.000000000000
1.000000000000
Numerador para o nó 1:
4
4.893289967663
-0.000000000000
4.684747122751
-0.000000000000
1.000000000000
0.023486149679

Com o método da eliminação, o mesmo denominador deve ser obtido para todas as saídas, exceto a do nó 3, sobre a fonte de tensão, onde se vai obter 1/1.

O trabalho deve ser feito em grupos de até 3 alunos. O prazo de entrega, com código, relatório com exemplos e demonstração (não imprima código fonte), é até uma semana antes da segunda prova. Deve ser feito em linguagem compilada, como C ou Pascal, como um programa autônomo que rode, preferencialmente, em Windows. O programa, relatório, exemplos, e tudo o mais necessário para executá-lo em qualquer PC, como bibliotecas (evite) não pode resultar em um arquivo .zip com mais de 2 MB.

2.3 The Elimination Algorithm

We shall present an algorithm for obtaining an equivalent triangular system from a given system. The algorithm constitutes a proof that the transformation to triangular form can always be achieved by means of successive elementary row transformations.

For convenience let us redefine the two systems again. We denote the n variables by $x_1, x_2, \ldots, x_n$ and the n equations by $\mathcal{E}_k$, $k = 1, 2, \ldots, n$. The given system 1 of linear differential equations is

$$\begin{aligned}
&\mathcal{E}_1\colon\ p_{11}(D)x_1 + p_{12}(D)x_2 + \cdots + p_{1n}(D)x_n = f_1\\
&\mathcal{E}_2\colon\ p_{21}(D)x_1 + p_{22}(D)x_2 + \cdots + p_{2n}(D)x_n = f_2\\
&\ \vdots\quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\mathcal{E}_k\colon\ p_{k1}(D)x_1 + p_{k2}(D)x_k + \cdots + p_{kn}(D)x_n = f_k\\
&\ \vdots\quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\mathcal{E}_n\colon\ p_{n1}(D)x_1 + p_{n2}(D)x_2 + \cdots + p_{nn}(D)x_n = f_n
\end{aligned}$$

where D denotes d/dt, the p_{ij} are polynomials of degree 2 at most in D with constant coefficients, and the f_i include contributions from the independent sources and from some of the initial conditions. Let us denote the equations of the equivalent triangular system by $\widehat{\mathcal{E}}_k$, $k = 1, 2, \ldots, n$. The triangular system T of linear differential equations is

$$\begin{aligned}
&\widehat{\mathcal{E}}_1\colon\ \widehat{p}_{11}(D)x_1 + \widehat{p}_{12}(D)x_2 + \widehat{p}_{13}(D)x_3 + \cdots + \widehat{p}_{1n}(D)x_n = \widehat{f}_1\\
&\widehat{\mathcal{E}}_2\colon\ \qquad\qquad\quad \widehat{p}_{22}(D)x_2 + \widehat{p}_{23}(D)x_3 + \cdots + \widehat{p}_{2n}(D)x_n = \widehat{f}_2\\
&\ \vdots\qquad\qquad\qquad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\widehat{\mathcal{E}}_{n-1}\colon\ \qquad\qquad\qquad\qquad \widehat{p}_{n-1,n-1}(D)x_{n-1} + \widehat{p}_{n-1,n}(D)x_n = \widehat{f}_{n-1}\\
&\widehat{\mathcal{E}}_n\colon\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \widehat{p}_{nn}(D)x_n = \widehat{f}_n
\end{aligned}$$

ALGORITHM

Step 1 In case one needs to calculate only one network variable, reorder, if necessary, the unknown functions $x_1, \ldots, x_n$, so that x_n is the network variable of interest.

Step 2 Examine the first column of the matrix $\mathbf{P}(D)$. If there is only one polynomial that is not identically zero, say, p_{k1}, then interchange the first and the kth equation, and go to Step 7; if not, go to Step 3.

Step 3 Pick out of the first column the polynomial of least degree which is not identically zero. Call k the number of the row of this polynomial; then the name of the polynomial of least degree is p_{k1}.

Step 4 Divide $p_{k1}(D)$ into each polynomial $p_{i1}(D)$ for all $i \neq k$; call q_{i1} the quotient of the division of p_{i1} by p_{k1}, and call the remainder r_{i1}. Hence,

$$p_{i1}(D) - q_{i1}(D)p_{k1}(D) = r_{i1}(D) \qquad \text{for all } i \neq k$$

(Observe that the degree of each polynomial r_{i1} is at least 1 less than the degree of p_{k1}.)

Step 5 Write the following equivalent system:

$$\mathcal{E}_1' \triangleq \mathcal{E}_1 - q_{11}(D)\mathcal{E}_k$$
$$\mathcal{E}_2' \triangleq \mathcal{E}_2 - q_{21}(D)\mathcal{E}_k$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$\mathcal{E}_k' \triangleq \mathcal{E}_k$$
$$\mathcal{E}_{k+1}' \triangleq \mathcal{E}_{k+1} - q_{k+1,1}(D)\mathcal{E}_k$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$\mathcal{E}_n' \triangleq \mathcal{E}_n - q_{n1}(D)\mathcal{E}_k$$

The first column of coefficients of the system $(\mathcal{E}_1', \mathcal{E}_2', \ldots, \mathcal{E}_n')$ is $r_{11}(D)$, $r_{21}(D), \ldots, p_{k1}(D), \ldots, r_{n1}(D)$.

Step 6 Repeat Steps 2 through 5 for this equivalent system.

Step 7 After at most three such cycles, the system (now system 1′) is in the following form (we write p_{ij}^* because the present polynomial coefficients are possibly different from those of system 1):

$$\begin{aligned}
&\mathcal{E}_1^*: \; p_{11}^*(D)x_1 + p_{21}^*(D)x_2 + \cdots + p_{1n}^*(D)x_n = f_1^* \\
&\mathcal{E}_2^*: \qquad\qquad\quad\; p_{22}^*(D)x_2 + \cdots + p_{2n}^*(D)x_n = f_2^* \\
&\vdots \;\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
&\mathcal{E}_n^*: \qquad\qquad\quad\; p_{n2}^*(D)x_2 + \cdots + p_{nn}^*(D)x_n = f_3^*
\end{aligned}$$

(System 1′ is equivalent to system 1; that is, every solution of one is a solution of the other, and vice versa.) Now, disregard the *first* equation. If there is only one equation left, stop; if there is more than one equation left, go to Step 2, and repeat Steps 2 through 7 for the remaining equations.

This algorithm reduces system 1 step by step to the triangular form and can be readily programmed in a computer.

Example 3 Consider the linear time-invariant network shown in Fig. 2.2. The initial inductor currents $j_1(0)$ and $j_2(0)$ and the initial capacitor voltage $v_C(0)$ are given. We wish to find the minimal differential equation of the network variable j_2, the current in the inductor L_2. In view of the graph of the network under consideration we shall use mesh analysis. The mesh currents i_1 and i_2 are indicated in Fig. 2.2. Since there is a capacitor in mesh 1, we shall use the capacitor charge q_1 as the mesh variable instead of i_1. We have

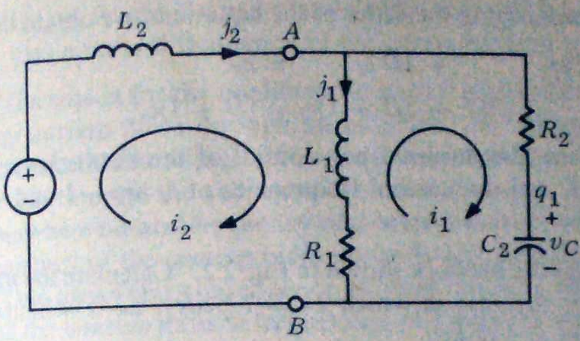


Fig. 2.2 With $R_1 = R_2 = 1$ ohm, $C_2 = 1$ farad, $L_1 = 1$ henry, and $L_2 =$ 2 henrys, the natural frequencies of j_2 are -1 and $-\frac{1}{2}$.

$$i_1(t) = \frac{dq_1}{dt} \qquad \text{and} \qquad q_1(0) = Cv_C(0)$$

The system of differential equations is

$$\mathcal{E}_1\colon \quad (D^2 + 2D + 1)q_1 - (D + 1)i_2 = 0$$
$$\mathcal{E}_2\colon \quad -D(D + 1)q_1 + (3D + 1)i_2 = e_s$$

Step 1 No reordering of the variables is necessary since the network variable of interest is already in the last column.

Here $k = 1$, and $p_{11}(D) = D^2 + 2D + 1$.

Step 4 $p_{21}(D) - (-1)p_{11}(D) = D + 1$

Steps 2 and 3 Here $q_{21}(D) = -1$ and $r_{21}(D) = D + 1$.

Step 5
$$\mathcal{E}_1' = \mathcal{E}_1\colon \qquad (D^2 + 2D + 1)q_1 - (D + 1)i_2 = 0$$
$$\mathcal{E}_2' = \mathcal{E}_2 - q_{21}(D)\mathcal{E}_1 = \mathcal{E}_2 + \mathcal{E}_1\colon \quad (D + 1)q_1 + \quad 2Di_2 = e_s$$

Since the first column is not yet in the desired form, we go back to Step 2. We obtain successively [we call $p'_{ij}(D)$ the coefficients of the system $\mathcal{E}_1', \mathcal{E}_2', \ldots$]

$$k = 2 \qquad p'_{21}(D) = D + 1$$

$$p'_{11}(D) - (D + 1)(D + 1) = 0 \qquad \text{or} \qquad \begin{aligned} q'_{11}(D) &= D + 1 \\ r'_{11}(D) &= 0 \end{aligned}$$

The new system is

$$\mathcal{E}_1'' = \mathcal{E}_1' - (D + 1)\mathcal{E}_2'\colon \qquad 0 \quad - (D + 1)(2D + 1)i_2 = -(D + 1)e_s$$
$$\mathcal{E}_2'' = \mathcal{E}_2'\colon \qquad (D + 1)q_1 + 2D\, i_2 \qquad = e_s$$

By interchanging the order of the equations, we obtain the triangular form

$$(D + 1)q_1 \qquad + 2D\, i_2 \qquad = e_s$$

$$(2D^2 + 3D + 1)i_2 = (D + 1)e_s$$

Therefore, the minimal polynomial of the network variable i_2 is $2D^2 + 3D + 1$, and the natural frequencies of i_2 are -1 and $-\frac{1}{2}$.

Exercise Consider the network shown in Fig. 2.2. Calculate the natural frequencies of j_2 for the case in which $L_2 = 1$ henry, $L_1 = 4$ henrys, $C_1 = 1$ farad, $R_1 = R_2 = 2$ ohms.

In conclusion, the elimination algorithm provides us with a systematic technique for obtaining the minimal differential equation and the natural frequencies of any network variable that we wish. Indeed, the zeros of the polynomial $\hat{p}_{nn}(s)$ are the natural frequencies of x_n.

3 Natural Frequencies of a Network

We now broaden our point of view and consider the linear time-invariant network $\mathfrak{N}$ as a whole. The main concept we wish to consider is that of natural frequency of a network. We say that a number s_k is a **natural frequency of the network** $\mathfrak{N}$ if s_k is a natural frequency of some *voltage* or a natural frequency of some *current* in the network $\mathfrak{N}$. It will turn out that in order to find the natural frequencies of a network we need not find all the natural frequencies of each voltage and of each current in the network.

Example The natural frequencies of the network of Example 5 in Sec. 1 are the natural frequencies of the voltage v_1, namely $j(1/\sqrt{L_1C_1})$ and $-j(1/\sqrt{L_1C_1})$, and the natural frequency of the voltage v_2, namely $-1/R_2C_2$.

Let us start by making two observations.

1. If $s_1 \neq 0$ and if s_1 is a natural frequency of a branch current, then it is also a natural frequency of the corresponding branch voltage. The reason is as follows: by assumption, for some initial state, the branch current j will include the term $K_1\epsilon^{s_1t}$, where $K_1 \neq 0$. Depending on the nature of the branches, the corresponding branch voltages will behave as follows:

   a. For a resistor, $v = Rj$, and v includes the term $RK_1\epsilon^{s_1t}$.

   b. For an inductor, $v = L(dj/dt)$, and v includes the term $LK_1s_1\epsilon^{s_1t}$.

   c. For a capacitor, $v = v(0-) + (1/C)\int_0^t j(t')\,dt'$, and v includes the term $CK_1(1/s_1)\epsilon^{s_1t}$.

Atualizado em 23/6/2010