

Universidade Federal do Rio de Janeiro  
Escola Politécnica  
Engenharia Eletrônica e de Computação  
Circuitos Elétricos II  
Alunos: Marcelle de Souza Campos  
Pedro Angelo Medeiros Fonini

## **Programa de análise de circuitos no tempo para estudo dos Métodos de Gear**

Nosso trabalho foi desenvolvido na linguagem C++, sendo ele composto por 4 arquivos: Makefile, trabalhoCircuitosEletricos.cpp, myFuntions.cpp e circuitAnalysis.h.

Constituído por classes, cada qual contendo sua respectiva função ao longo do escopo do programa, o código analisa o circuito da seguinte forma:

1-) Após a leitura do nome do arquivo contenedor da netlist a ser analisada, o arquivo é aberto para leitura. Conforme cada linha é lida, cada elemento, assim como suas características e especificações, são armazenados num objeto da classe ElementsList. Após a leitura ser finalizada, o arquivo é fechado;

2-) Neste passo, fazemos um loop sobre a lista inteira analisando cada elemento individualmente, até seu término. Conforme a leitura de cada elemento, é feita a estampa dele é formada, e assim montamos as matrizes que representam o sistema linear que deveremos resolver. As matrizes são implementadas como instâncias da classe CppMatrix. O método desta classe que resolve o sistema linear usa o algoritmo da decomposição QR, com reflexões de Householder.

### **Observações:**

- Na presença de cada curto, fonte de tensão, amplificador de tensão, transresistor ou indutor, uma variável  $j$  (de corrente) a mais é acrescentada às matrizes da análise modificada;

- Na presença de indutores e capacitores, para que ocorra a montagem de estampas, a função gearMethod é chamada. Esta função terá como responsabilidade apresentar o modelo destes elementos, que será constituído por uma resistência e uma fonte de corrente/tensão, de acordo com as respectivas especificações fornecidas ao programa, tais como a ordem do Método de Gear e o passo interno e o passo; Usamos fontes de tensão para o indutor (com o objetivo de medir a corrente que passa nele) e fontes de corrente para o capacitor.

3-) Um arquivo no formato txt com o nome do arquivo fornecido pelo usuário acrescentado “\_answer.m” é gerado e aberto para escrita;

4-) Terminada a construção das matrizes  $A$  e  $B$ , o sistema  $Ax = B$  é resolvido pela função solveMatrixSystem. O resultado, a matriz  $x$ , será impressa no arquivo de acordo com as especificações fornecidas;

5-) O 4o. Passo se repetira por uma certa quantidade de vezes, que será determinada a partir dos valores do passo interno, do tempo final e do passo;

6-) Terminado este processo, o arquivo finalmente será fechado e estará pronto para análise d usuário. Ele pode ser executado diretamente pelo MATLAB para gerar variáveis de nomes t, e1, e2, e3, ..., j<nome1>, j<nome2>, etc. Elas podem ser plotadas, por exemplo, com plot(t,e1, t,e2);

#### Especificações dos arquivos do programa:

-circuitAnalysis.h:

Nele estão definidas as seguintes classes:

- element;
- elementsList;
- cppmatrix;
- capacitor\_indutor;
- split.
- 

- TrabalhoCircuitosEletricos.cpp:

Arquivo contenedor do main, é o corpo do programa.

- MyFunctions:

Arquivo contenedor das funções do programa, chamadas pelo main ou por próprias funções internas. São elas:

- cppmatrix::initialize;
- cppmatrix::operator+;
- cppmatrix::operator\*;
- cppmatrix::submatrix;
- cppmatrix::make\_id;
- cppmatrix::subassign;
- cppmatrix::solveMatrixSystem;
- cppmatrix::fill\_out\_with\_zeros;
- elementsList::getElement;
- elementsList::NumberOfNodes;
- elementsList::buildModifiedNodalMatrix;
- elementsList::gearMethod;
- elementsList::printResult.

#### Exemplos:

capacitor descarregando:

circuito simples com um capacitor e um resistor, o capacitor com condição inicial.

capacitor carregando:

Idem, mas sem condição inicial, e com uma fonte de tensão

oscilador:

Um capacitor e um indutor, com condição inicial.

Passa altas:

Um filtro com topologia sallen-key passa altas

Passa baixas:

Idêntico, mas passa baixas

resistencia negativa:

igual ao oscilador, mas com uma resistência negativa em paralelo. A tensão cresce exponencialmente.

resistivo:

Circuito com 7 nós, e várias casos degenerados, para testar a estabilidade numérica. Foi constatado, por exemplo, que uma resistência de  $10e+9$  ou mais ligada a um nó flutuando (ou seja, sem corrente) apresenta uma certa queda de potencial.

teste\_sin:

Somente um nó e uma fonte de tensão com atraso, número de ciclos, atenuação, etc, etc.