

## Circuitos elétricos II – Primeiro semestre de 2009 – Trabalho

Escrever um programa que analise no tempo circuitos não lineares variantes no tempo com os elementos descritos abaixo. A maior parte dos elementos são lineares invariantes no tempo, mas são incluídas chaves controladas por tensão (portanto não lineares) e capacitores lineares variantes com uma tensão de controle, que podem ser variantes no tempo de uma forma especificada pelo sinal de controle. O programa poderia servir para analisar máquinas eletrostáticas, usando o capacitor variante no tempo e as chaves. Pode analisar também circuitos com diodos, osciladores caóticos, fontes chaveadas idealizadas, etc.

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. As condições iniciais podem ser dadas ou não:

Resistor: **R**<nome> <nó+> <nó-> <Resistência>  
Indutor: **L**<nome> <nó+> <nó-> <Indutância> [**IC**=<corrente inicial>]  
Capacitor: **C**<nome> <nó+> <nó-> <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>  
Resistor não linear por partes: **N**<nome> <nó+> <nó-> <4 pontos  $v_i j_i$ >  
Chave controlada a tensão: **S**<nome> <nó+> <nó-> <controle+> <controle-> <Gligada> <Gdesligada>  
<Limiar>  
Capacitor controlado a tensão: **B**<nome> <nó+> <nó-> <controle+> <controle-> <Capacitância por volt>  
[**IC**=<carga inicial>]

Opcionalmente:

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>

E também:

Comentário: \*<comentário>  
Comando: .TRAN <tempo final> <passo> <passos internos>

Os passos internos servem para aumentar a precisão sem ter que gerar tabelas muito grandes. Entre cada passo listado na tabela é feito o número especificado de passos iguais, com resultados não listados.

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. O programa MNA1 também mostra como montar as estampas dos elementos resistivos, inclusive as quatro fontes controladas.

Os parâmetros para as fontes devem ser:

Para fonte CC: DC <valor>

Para fonte senoidal: SIN (<nível contínuo> <amplitude> <frequencia (Hz)> <atraso> <fator de atenuação> <ângulo> <número de ciclos>)

A fonte senoidal vale, considerando  $X_i$  = parâmetro #i:

Antes do tempo de atraso  $X_4$  ou depois do numero de ciclos  $X_7$ :

$$X_1 + X_2 \operatorname{sen} \left( \pi \frac{X_6}{180} \right)$$

Em outros tempos:

$$X_1 + X_2 e^{(t-X_4)X_5} \operatorname{sen}\left(2\pi X_3(t - X_4) + \pi \frac{X_6}{180}\right)$$

Para fonte pulsada: PULSE (<amplitude 1> <amplitude 2> <atraso> <tempo de subida> <tempo de descida> <tempo ligada> <período> <número de ciclos>)

A fonte começa na amplitude 1, e fica aí até o fim do tempo de atraso. Então muda para a amplitude 2 variando em rampa dentro do tempo de subida, fica na amplitude 2 durante o tempo ligada, volta à amplitude 1 em rampa dentro do tempo de descida, e repete tudo com o período e o número de ciclos especificados. Termina na amplitude 1.

O resistor não linear por partes é definido por 4 pares de valores de tensão e corrente, por onde sua curva passa. A curva é reta entre pontos, e portanto são 3 retas. Os valores das tensões tem que ser dados em ordem crescente. Antes do ponto com tensão mais baixa e depois do ponto com tensão mais alta a curva continua as retas das extremidades. É então possível modelar vários tipos de diodos e outros tipos de resistor não linear com controle por tensão.

A chave é modelada como uma condutância linear com o valor dependendo de se a tensão de controle é maior (ou igual) ou menor que a tensão de limiar dada.

O capacitor controlado por tensão é ligado entre os nós a e b e controlado pela tensão entre os nós c e d. Ele segue a lei:

$$q(t) = C v_{cd}(t) v_{ab}(t)$$

Usando o método “backward” de Euler, esta equação resulta em um dispositivo não linear seguindo a lei:

$$j(t_0 + \Delta t) = \frac{C}{\Delta t} (v_{cd}(t_0 + \Delta t) v_{ab}(t_0 + \Delta t) - v_{cd}(t_0) v_{ab}(t_0))$$

Isto define uma corrente controlada por duas tensões de forma não linear, e o circuito é resolvido pelo método de Newton-Raphson. Note que a condição inicial é a carga inicial  $C v_{cd}(0) v_{ab}(0)$ , já que não se sabe  $v_{cd}(0)$ .

O programa deve operar usando o método “backward” de Euler para integração numérica, acoplado ao método de Newton-Raphson para resolver circuitos não lineares. Deve primeiramente realizar uma análise para achar a solução em  $t=0$ , usando as condições iniciais dadas (ou assumir nulas as que não forem dadas). Esta análise pode ser feita assumindo um passo 1000 vezes menor que o que será usado nos cálculos (o passo interno). A partir daí deve usar o passo interno, e mostrar os resultados a cada número de passos internos de tempo. A cada passo, o método de Newton-Raphson deve ser executado para achar a solução, usando a solução do tempo anterior como aproximação inicial. O programa deve contar quantas vezes o ciclo de Newton-Raphson é executado a cada tempo, e se o número passar de um valor razoável, tentar outra aproximação inicial para a solução. Deve contar também quantas vezes faz isto, e se o número passar de um valor razoável, abortar a análise. Em qualquer caso, deve haver meio do usuário interromper a análise.

As saídas geradas devem ser salvas em arquivos, de forma a que possam plotadas por outro programa.

A tabela deve ficar em formato de texto, com a primeira coluna sendo o tempo e as colunas seguintes as tensões nodais e correntes calculadas. A primeira linha deve conter os nomes das variáveis nas colunas.

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 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 1 MB.