**CirOp**

O programa CirOp é um *framework* que esta sendo desenvolvido para projetar e otimizar circuitos analógicos a partir de simulações e algoritmos metaheuristicos. Várias famílias de circuitos estão sendo analisadas para serem introduzidas, entre elas fontes de corrente e tensão, amplificadores operacionais, amplificadores de baixo ruído (LNA), osciladores e amplificadores neuronais. A Figura 1 mostra, para as metaheuristicas algoritmo Genético, *Simulated Annealing* e *Particle Swarm*, a estrutura geral do *framework*. Importante observar que

- a mesma função de *fitness*, que fornece a nota de um indivíduo, é aplicada para todos os algoritmos metaheuristicos utilizados. Isto simplifica a introdução de novos algoritmos

- a mesma função de *fitness* é aplicada para todas as topologias de circuito que pertence a uma mesma família. O que muda entre uma topologia e outra é apenas o arquivo que descreve o circuito.

O *framework* CirOp se encontra na pasta Otimizador\_v7. Nesta pasta há o programa CirOp que inicializa a execução abrindo o menu da Figura 2. Este menu é criado pelo programa Opt.m. No arquivo CirOp.m podem ser mudadas algumas configurações que são:

- variável “slash”: pode ter valor “\”, para sistemas Windows, ou “/”, para sistemas UNIX.

- variáveis “simulador” e “simuladorRF”: contem os endereços dos executáveis HSPICE e HSPICERF (nos casos em que esse também for usado). No sistema original temos que o HSPICE esta em ***C:\synopsys\Hspice\_A-2008.03\BIN\hspice\_mt.exe*** (simulador='START /Realtime/wait C:\synopsys\Hspice\_A-2008.03\BIN\hspice\_mt.exe ')e o HSPICERF, em ***C:\synopsys\Hspice\_A-2008.03\BIN\hspicerf.exe***(simuladorRF = 'START /Realtime/wait C:\synopsys\Hspice\_A-2008.03\BIN\hspicerf.exe ').

Atualização da temperatura e da solução corrente

Inicialização

Atualização da posição das partículas

GA

PS

SA

Continua

Continua

Continua

Definição dos parâmetros do problema

Geração da nova população

Seleção de indivíduos

Simulação e medidas

Geração de soluções na vizinhança da solução corrente

Inicialização

Escolha da metaheurística

Geração de população inicial

Reprodução

(*crossover* e mutação)

Verificação dos critérios de parada

Verificação dos critérios de parada

Escolha do circuito

Cálculo da função objetivo

**Geração de arquivos de simulação**

INÍCIO

Verificação dos critérios de parada

Função

de *fitnness*

Figura 1. Diagrama simplificado do CirOp

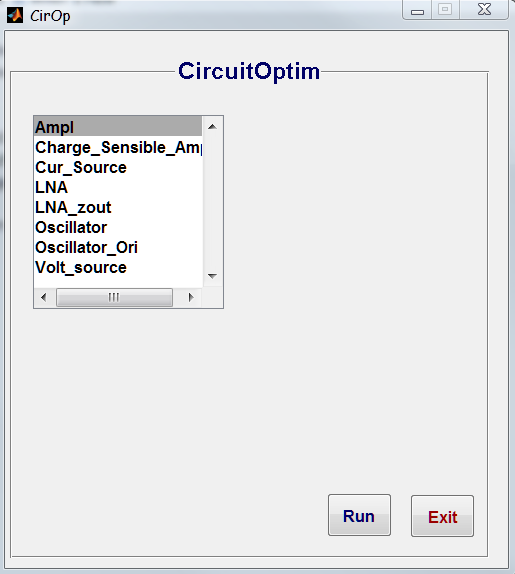


Figura 2. Menu inicial do programa CirOp

**1. Arquivos**

Três pastas importantes encontradas são Start, Models e Circuits.

**Start**

Nesta pasta estão os principais programas do circuito. Dois programas importantes são:

- MenuCir: programa do menu que é mostrado na Figura 3. Este programa é chamado a partir do comando Run, Figura 2.

- MetaHeu: programa que faz a chamada de todas as metaheristicas.

**Models**

Nesta pasta estão os modelos de simulação para os dispositivos.

**Circuits**

Nesta pasta estão informações necessárias para otimizar as topologias em cada família. Estas informações incluem a função de *fitness*, menus específicos e pastas com arquivos de descrição dos circuitos em cada uma das diferentes topologias configuradas.

Considere a pasta Volt\_Source, por exemplo, que contem as informações para a otimização de fontes de tensão. Dos arquivos presentes, dois importantes são o fitness.m, que contem a função de fitness, e arquivo param\_mod.m que contém a função que prepara arquivos de comando para o simulador. Cada uma das pastas presentes aqui é esta relacionada com uma diferente topologia de fonte de tensão. Considere a pasta Cir\_Eder\_NMOS. Nela encontramos os seguintes arquivos de importância:

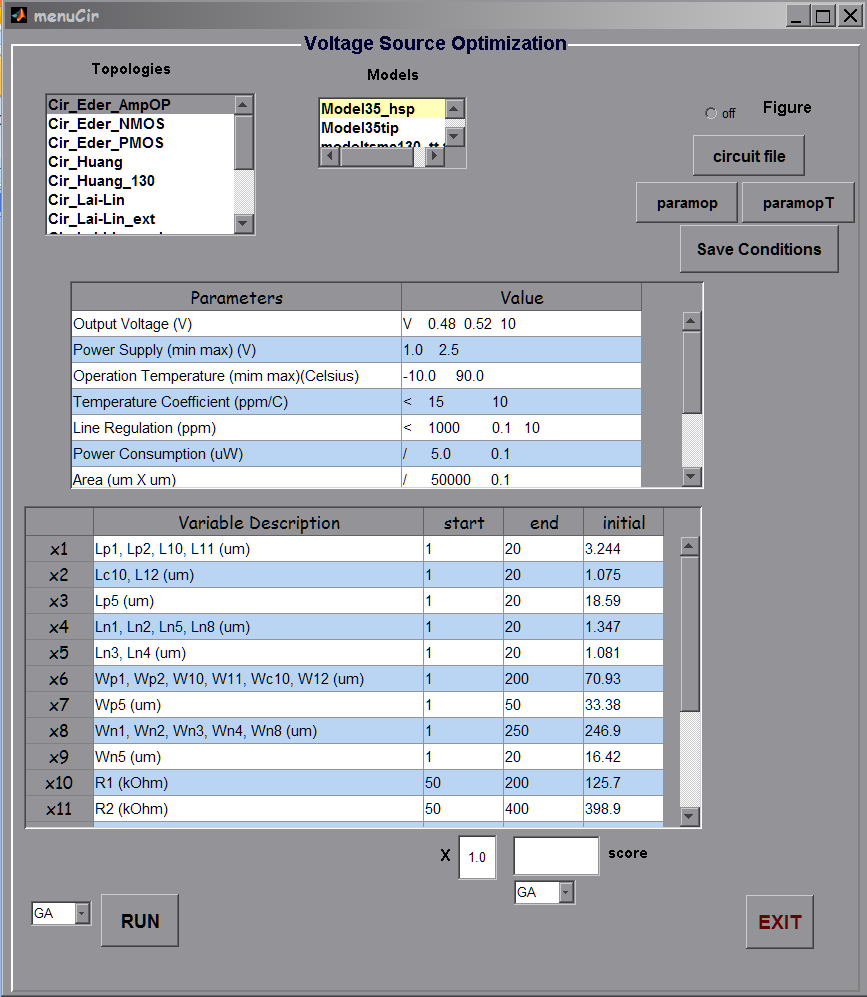
circuito.sp: descrição hspice da topologia.

circuito.png: esquemático do circuito,

param: parâmetros gerados pelo otimizador, com dimensões para o circuito, e comandos para simulação

paramop e paramopT: o param dos circuitos com melhores notas.

A pasta results guardará arquivos com resultados da nota , *score*, obtido ao longo das otimizações para se levantar estatísticas.



1

2

3

7

6

4

5

9

8

10

11

12

Figura 3. Menu para otimização de fontes de tensão

**2. Execução e menus**

Para executar o programa deve-se estar na pasta Otimizador\_v7 e executar o comando

* CirOp

Este vai então abrir o menu da Figura 2. Nele se escolhe uma das famílias para realizar a otimização e se da o comando de Run. Selecionando a familia Volt\_Source, por exemplo, será gerado o menu da Figura 3.

Os comandos/funcionalidades do menu da Figura 3 são:

1. Topologias que podem ser otimizadas dentro da família;

2. Modelos disponíveis (colocados na pasta Models);

3. Caso on, aparece o esquemático da topologia (circuito.png);

4. Apresenta o arquivo hspice que descreve topologia (circuito.sp);

5. Apresenta os arquivos paramop e paramopT. Quando são realizadas diversas otimizações, o paramop é o melhor resultado para a otimização atual e o paramopT, o melhor resultado entre todas. Durante a otimização esses arquivos podem ser observados, permitindo que se acompanhe os resultados;

6. Salva os resultados e configurações colocadas no menu;

7. Parâmetros para a otimização. Os parâmetros que devem ser otimizados são descritos por uma entrada que será utilizada em uma função *score*. Os formatos possíveis e seus significados são dados na Figura 4. Para a *Output Voltage*, por exemplo, foi escolhida a entrada ‘v 0.48 0.52 10’; para o *Temperature Coeficient*, a função ‘/ 15 10’. Dessa forma pode-se utilizar a função do *score* mais conveniente para a otimização realizada. Observe que ‘v’e ‘V’são funções diferentes.

Podem ser também indicados os transistores que se deseja forçar em fraca inversão, weak inv., ou em forte inversão, strong inv. Também é possível se colocar o peso de cada um dos parâmetros na função de *fitness*. Constantes também podem ser colocadas e elas aparecerão no arquivo .sp como M1, M2, etc.;

8. Tabela com as variáveis que desejamos determinar (X1, X2, etc.). Nessa tabela esta a descrição, o valor inferior, o valor superior e uma condição inicial. Podem-se editar as linhas e colunas da tabela, todas, acrescentar linhas ao fim da tabela, tecla insert, para novas variáveis, e tirar linhas ao fim da tabela, tecla F1.

As variáveis que apresentarem o valor inferior igual ao valor superior não serão otimizadas (reduz o numero de variáveis de otimização). Assim, podemos fixar o valor de quaisquer variáveis para facilitar as buscas;

9. Lista dos vários algoritmos disponíveis: GA (genetic algorithm - Matlab), SA (simulated annealing - Matlab), OS (Pattern Search – Matlab), MM (Minimax opt.- Matlab), SAM (Simulated annealing Modified), SCE, PSO (Particle Swarm), DE (Differential Evolution), EvN (Evolution with Neuralnetworks), PSON (Particle Swarm e redes neural), PSw (PSwarn Minho);

10. Executar a simulação. Gera o menu da Figura 5;

11. Carrega como condição inicial o melhor resultado gerado por qualquer um dos algoritmos;

12. Sai do menu. Não é possível sair do menu de outra forma. Com isso se garante que certos resultados são armazenados. Caso este menu não seja fechado, não será possível fechar o anterior nem o próprio Matlab.











Figura 4. Possíveis entradas da função *score* e seus significados (valor do *score* X medida (meas))

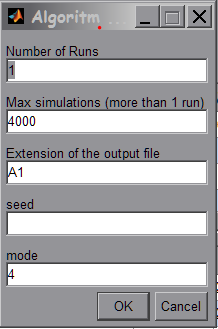


Figura 5. Menu para execução.

O menu da Figura 5 é o de execução do algoritmo. Nele se insere

- o numero de otimizações que se deseja realizar. Para GA, SA e PS, que executam metaheuristicas do MATLAB, quando tiver valor 0, a ferramenta optimtool é chamada;

- quantas simulações são executadas em cada otimização;

- o nome da extensão do arquivo de resultados. Por exemplo, o arquivo de resultados de uma otimização com GA fica na pasta de results e se chamara optimos.GA**A1**;

- numero inteiro que servira de semente para gerar soluções iniciais para as otimizações. Se não for colocada uma semente será utilizada a semente usada no sistema no momento;

- o modo indica se uma solução inicial será utilizada na otimização. Cinco modos são possíveis e estão mostradas na tabela A1.

Tabela A1.1 Modos de otimização. Xo é a solução inicial (Figura 3, entrada 8).

|  |  |  |
| --- | --- | --- |
| **mode** | **Primeira otimização** | **Outras otimizações**  **(quando for múltipla)** |
| **0** | utiliza Xo como uma das soluções iniciais | utiliza Xo como uma das soluções iniciais |
| **1** | utiliza Xo como uma das soluções iniciais | todas soluções iniciais aleatórias |
| **2** | utiliza Xo como uma das soluções iniciais | utiliza o ótimo da otimização anterior como uma solução inicial |
| **3** | todas soluções iniciais aleatórias | utiliza o ótimo da otimização anterior como uma solução inicial |
| **4** | todas soluções iniciais aleatórias | todas soluções iniciais aleatórias |