# **Estratégias Evolutivas**

# **Projeto de Filtros Passa Faixa**

1. **Introdução:**

A utilização de estratégias evolutivas é uma técnica de solução de sistemas baseada no princípio da evolução hereditária, tais sistemas contém uma população de possíveis soluções, essa população é submetida a um processo de seleção através de uma função de adequação (*fitness*) onde cada indivíduo é testado, e em que os melhores indivíduos passam por um processo de cruzamento e uma possível mutação.

1. **Procedimento:**

O algoritmo evolutivo é um processo probabilístico que contém um população de indivíduos, possíveis soluções para o sistema, à cada época essa população é submetida à uma seleção por uma função de *fitness*, através dessa pode-se escolher os indivíduos mais adequados, esses passam por um processo de cruzamento e mutação, os novos indivíduos são incluídos na população competindo com os indivíduos da geração anterior no próximo ciclo seletivo.

O sistema escolhido para se desenvolver o algoritmo foi o FILTRO PASSA FAIXAS ATIVO, como se ilustra na Figura 1.

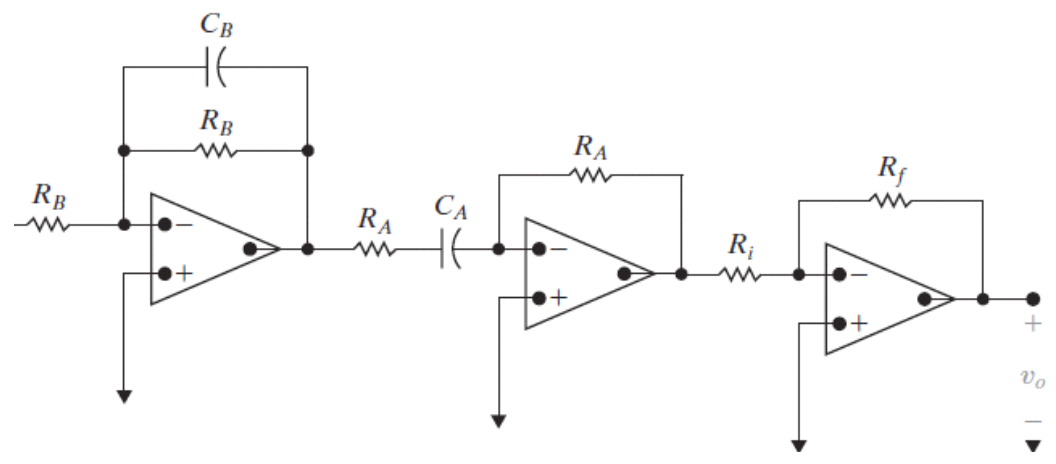
****

Figura 1: Filtro passa faixa ativo.

Pode-se observar, então que cada indivíduo solução do sistema deve conter seguinte estrutura: **[Ra Rb Rf Ri Ca Cb]** , esse capaz de definir por completo os componentes do filtro.

Dessa forma, gerou-se no MATLAB® um código capaz de desempenhar tal, função. A população inicial foi gerada aleatoriamente para os componentes citados anteriormente. Cada elemento do filtro tem restrições de forma que o sistema não encontre soluções utilizando componentes impraticáveis. Considerando-se uma população inicial de tamanho “*L*”, é definida uma matriz “*P0*” que contêm as “*L*” possíveis soluções iniciais (População Inicial):



Na Tabela 1, contêm o código em MATLAB® para a geração aleatória da população inicial, considerando-se as restrições dos componentes passivos do filtro passa-faixa.

Tabela 1. Geração da População Inicial

|  |
| --- |
| %Caracteristicas do filtro passa faixa    fc1 = 100; %Frequência de corte inferior  fc2 = 10000; %Frequência de corte superior  bK = 2; %Ganho do estágio de saída  bQ = sqrt(fc1\*fc2)/(fc2-fc1); % Fator de Qualidade      L=100; % Tamanho da população  epocas=1000; % Número de gerações (critério de parada)  mutProb=0.008; % Probabilidade de mutação  crossProb=0.65; % Probabilidade de cruzamento    %Restrições dos componentes  Rmin = 10;  Rmax = 10e3;  Cmin = 10e-9;  Cmax = 500e-6;  psense = 0.01; %margem de precisão das frequências de corte    %[Ra Rb Rf Ri Ca Cb]  P = [(Rmin+(Rmax-Rmin).\*rand(L,4)) (Cmin+(Cmax-Cmin).\*rand(L,2))]; |

Estando ciente da configuração do filtro, pode-se definir as funções seletoras, regidas pelos parâmetros de projeto, esses são: frequência de corte inferior, frequência de corte superior, e ganho. Dessa forma, os indivíduos passam pelo processo de *fitness* definido por:



Em que:

* *fwc*1 é o erro da frequência de corte inferior entre o indivíduo e o valor teórico
* *fwc*2 é o erro da frequência de corte superior entre o indivíduo e o valor teórico
* *f*Q é o erro do fator de qualidade entre o indivíduo e o valor teórico
* *fk* é o erro do ganho entre o indivíduo e o valor teórico

Na Tabela 2, contem o código da avaliação de cada indivíduo da população “P”.

Tabela 2. Função de Avaliação (fitness)

|  |
| --- |
| Wc1 = (Ra.\*Ca).^(-1); %Frequência de corte inferior do indivíduo  Wc2 = (Rb.\*Cb).^(-1); %Frequência de corte superior do indivíduo  Q = sqrt(Wc1.\*Wc2)./(Wc2-Wc1); %Fator de qualidade do indivíduo  K = Rf./Ri; %Ganho do indivíduo  fWc1 = (Wc1 - 2\*pi\*fc1).^2; %Erro da freq de corte inferior do indivíduo  fWc2 = (Wc2 - 2\*pi\*fc2).^2; %Erro da freq de corte superior do indivíduo  fQ = (Q-bQ).^2; %Erro do fator de qualidade do indivíduo  fK = (K-bK).^2; %Erro do ganho do indivíduo  fit = (fWc1 + fWc2 + fQ + fK)/4; %Valor de aptidão do indivíduo |

Calculam-se as frequências de corte superior e inferior de cada indivíduo da população (Wc1 e Wc2), seu fator de qualidade (Q) e o ganho do estágio de saída (K). Esses valores são comparados com os valores desejados como requisitos de projeto (*fc*1, *fc*2, *bQ*, *bK*), através do cálculo do erro quadrático, esses erros são somados definindo um valor de adequação *fit* para o indivíduo.

Então depois da avaliação de *fitness* da primeira geração entra-se no processo cíclico o qual é apresentado na Tabela 3.

Tabela 3. Laço de evolução

|  |
| --- |
| for i=1:epocas  %Ordena o vetor fit de maneira crescente  [c,d]=sort(fit');    %Seleção  [S,fitS]=selec(P,fit);    %Cruzamento  [C]=cruzamento(S,fitS,crossProb);  %Mutação  [M]=mutacao(C,mutProb,Rmin,Rmax,Cmin,Cmax);    %Avaliação - Elitismo  [fitM] = fitness(M, fc1, fc2, bQ, bK);    %melhor indivíduo e substituição  [a,b]=sort(fitM');    P(d(L-4:L),:)=M(b(1:5),:);  fit(d(L-4:L)) = fitM(b(1:5));    R = P(b(1),:);  Prodigio(i) = fit(b(1));    end |

O procedimento será repetido por um valor limite de épocas selecionado anteriormente, à cada época verifica-se através da função “*sort”* os elementos mais aptos, escolhe-se então através da função “*selec”* os elementos que passarão pelo processo de reprodução e mutação.

O método de seleção implementado é o método da roleta, neste método, cada indivíduo da população é representado na roleta proporcionalmente ao seu índice de aptidão. Assim, aos indivíduos com alta aptidão é dada uma porção maior da roleta, enquanto aos de aptidão mais baixa é dada uma porção relativamente menor da roleta. Finalmente, a roleta é girada um determinado número de vezes, dependendo do tamanho da população, e são escolhidos, como indivíduos que participarão da próxima geração, aqueles sorteados na roleta, como ilustrado na Figura 2.

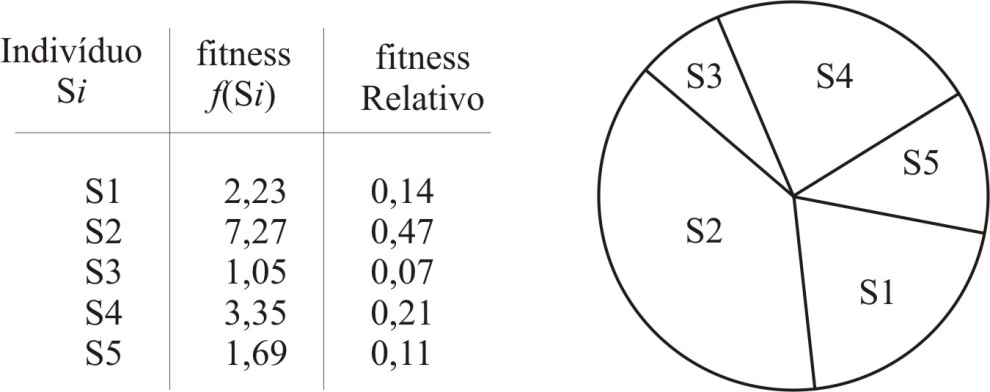


Figura 2: Roleta de aptidão.

Os indivíduos selecionados passam, então por um processo de cruzamento, onde, de acordo com a probabilidade de cruzamento escolhida. Na Tabela 4 ilustra-se o código do operador de cruzamento. A variável “alfa” é um valor aleatório que cruza um elemento do vetor de indivíduos aptos com outro elemento do mesmo vetor, de índice definido pelo vetor de cruzamento *cruze*. Esse cruzamento é feito por uma soma ponderada pelo índice aleatório alfa e posicionada no vetor de elementos cruzados C.

Tabela 4. Operador de Cruzamento

|  |
| --- |
| alfa = rand;  C(2\*i-1,:)= alfa\*S(cruze(2\*i-1),:)+(1-alfa)\*S(cruze(2\*i),:);  C(2\*i,:) = alfa\*S(cruze(2\*i),:) +(1-alfa)\*S(cruze(2\*i-1),:); |

Em seguida essa população é submetida a um processo de mutação, ditada pela probabilidade de mutação selecionada anteriormente e restrições de componentes. Na Tabela 5 contem o código para a realização do operador de mutação.

Tabela 5. Operador de Mutação

|  |
| --- |
| if j<=4 %Resistores  M(i,j)=C(i,j)+1000\*randn;  if M(i,j)<Rmin & M(i,j)>Rmax  M(i,j)=C(i,j);  end  elseif j>4 %Capacitores  M(i,j)=C(i,j)+1e-6\*randn;  if M(i,j)<Cmin & M(i,j)>Cmax  M(i,j)=C(i,j);  end  end |

De acordo com a probabilidade de mutação, se o elemento for selecionado, o código mostrado anteriormente irá somar ao elemento um dado número aleatório, o resultado passa então por uma verificação, onde o elemento gerado deve se enquadrar nos limites definidos para os componentes.

Em seguida a matriz criada pelo cruzamento e mutação será avaliada utilizando a função de *fitness* e os cinco melhores elementos são incluídos na população anterior realizando-se assim a substituição dos melhores indivíduos. Na Tabela 6, é mostrado o processo de substituição, também conhecido como operador de elitismo.

Tabela 6. Operador de Elitismo

|  |
| --- |
| %Avaliação - Elitismo  [fitM] = fitness(M, fc1, fc2, bQ, bK);    %melhor indivíduo  [a,b]=sort(fitM');    P(d(L-4:L),:)=M(b(1:5),:);  fit(d(L-4:L)) = fitM(b(1:5)); |

Assim, o processo se reinicia, até a finalização das épocas, escolhendo-se por fim o melhor indivíduo, esse será verificado e caso não tenha atendido as requisitos de projeto ocorrerá mais um processo evolutivo.

1. **Resultados do Algoritmo Evolutivo**

A partir do processo de evolução, o indivíduo com melhor fitness representa a melhor solução do processo de otimização. Neste caso para a frequência de corte inferior (fc1), para a frequência de corte superior (fc2), fator de qualidade (Q) e ganho de saída (k), foi obtido o indivíduo solução contido na Tabela 7.

Tabela 7. Valores do melhor indivíduo.

|  |
| --- |
| [Ra Rb Rf Ri] = 1.0e+03 \*[ 0.0108 0.0232 9.7144 4.8572]  [Ca Cb] = 1.0e-03 \*[0.1475 0.0007]  fc1 = 99.9998  fc2 = 9.9992e+03  Q = 0.1010  k= 2.0000 |

Na Figura 3 ilustra-se o diagrama de bode do filtro passa-faixa com os valores encontrados pelo algoritmo evolutivo.



Figura 3: Diagrama de Bode do filtro passa-faixa.

Na Figura 4 ilustra-se o comportamento do fitness do melhor indivíduo para cada iteração do algoritmo evolutivo. Observa-se o processo de minimização do fitness (otimização direcionada).



Figura 4: Evolução do fitness do melhor indivíduo.