



IMPLEMENTAÇÃO DE UM ALGORITMO GENÉTICO PARA DETERMINAÇÃO DO PONTO DE MÁXIMO E/OU MÍNIMO DA FUNÇÃO DO 2º e 3º GRAU

Ana Cláudia M. SILVEIRA¹; Renato Machado PEREIRA²

RESUMO

A história da equação de segundo e terceiro grau tem vários aspectos interessantes, em virtude dos quais ela se constitui num tópico atraente para estudos entre professores e alunos de matemática.

A proposta da pesquisa foi construir uma linha de investigação que possibilitasse a construção de um algoritmo genético para a determinação do ponto de máximo e/ou mínimo da função de 2º e 3º grau. Para tanto, desenvolvemos um estudo aprofundado da linguagem de programação C++. Em seguida, estudamos os conceitos e teorias relacionados com Algoritmos Genéticos. E finalizamos a pesquisa com a construção de um algoritmo genético capaz de determinar o ponto que maximiza e/ou minimiza uma função de 2º e 3º grau.

INTRODUÇÃO

O algoritmo genético foi inventado por John Holland no final da década de 60, buscando inspiração nos princípios oriundos do “mundo biológico”, mais especificamente na teoria da evolução Darwiniana (BARBOSA, 1997). Darwin, no seu livro “Origem das Espécies”, expôs a sua teoria da evolução por seleção natural, tomando como ponto de partida duas observações:

¹ Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – Campus Muzambinho. Muzambinho/MG, email: ana-claudia01@live.com;

² Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – Campus Muzambinho. Muzambinho/MG, email: renato.pereira@muz.ifsulde Minas.edu.br;

1. Os organismos vivos produzem grande número de sementes ou ovos, mas o número de indivíduos nas populações normais é mais ou menos constante, o que só se pode explicar pela grande mortalidade natural.
2. Organismos de uma mesma espécie são diversificados na forma e comportamento, sendo a variabilidade fortemente influenciada pela hereditariedade.

Assim, existe grande variabilidade e grande mortalidade, uns organismos terão maior probabilidade de deixar descendentes do que outros, na qual, Darwin chamou de seleção natural. Como Darwin demonstrou, a seleção natural ou “luta pela vida com sobrevivência dos mais aptos” é o fator essencial da evolução.

Os algoritmos genéticos tentam abstrair e imitar esses mecanismos evolutivos à resolução de problemas que requerem adaptação, busca e otimização. Constituem uma classe de ferramentas muito versátil e robusta e quando usado como algoritmo de minimização e maximização se distingue das técnicas mais comuns de programação matemática (BARBOSA, 1997).

Em particular, o objetivo desse trabalho foi desenvolver um algoritmo genético que pudesse determinar o máximo e/ou o mínimo de uma função de 2º e 3º grau. Há mil e setecentos anos antes da era cristã já sabia resolver a equação do 2º grau, mas apenas três mil anos mais tarde que foram resolver a do 3º grau (LIMA, 1987).

Uma implementação de um algoritmo genético para determinar o máximo e/ou mínimo de uma função de 2º e 3º grau começa com uma população aleatória de pares ordenados (x,y) , chamados de cromossomos. Esses pares ordenados são avaliados e associados a novos cromossomos de tal forma que os melhores pontos são associados aos indivíduos que representam uma melhor solução para o problema de otimização. O processo de solução adotado nos algoritmos genéticos consiste em gerar, através de regras específicas, um grande número de indivíduos, população, de forma a promover uma varredura tão extensa quanto necessária do espaço de soluções.

Um pseudo-código (MEDINA & FERTIG, 2006) de algoritmo genético para otimização de funções do 2º e 3º grau seria:

Inicialize a população

Avalie indivíduo na população

Repita

Crie nova geração de indivíduos
Mutação
Cruzamento
Avalie indivíduos na nova população
Selecione indivíduos para sobreviver
Até critério de parada satisfeito
Fim

MATERIAL E MÉTODOS

Utilizamos como metodologia uma bibliografia de referência. A atenção inicial ficou voltada para os livros sobre programação na linguagem C++: “Algoritmos e Programação: Teoria e Prática” do autor Marco Medina e Cristina Fertig (MEDINA & FERTIG, 2006) e “C++: Como Programar” do autor Harvey Deitel e Paul Deitel (DEITEL & DEITEL, 2006); não se furtando à análise de outras obras que se mostraram necessárias para a pesquisa. Em seguida, trabalhamos com estudiosos e comentadores que tratam da teoria de Algoritmos Genéticos, tais como: “Introdução aos Algoritmos Genéticos” do autor Hélio Barbosa (BARBOSA, 1997) e “Fundamentos, Potencialidades e Aplicações de Algoritmos Evolutivos” do autor Leandro dos Santos Coelho (COELHO, 2003); e estabelecemos diálogos com outros clássicos pertinentes na discussão.

O desenvolvimento da pesquisa foi baseada na leitura dos livros, discussão com o orientador e implementação computacional dos algoritmos. As etapas da pesquisa se dividiram em:

- a) Estudo da linguagem C++: o objetivo desta etapa foi desenvolver o domínio da linguagem C++ e a técnica de programação exigida para implementação de algoritmos, que mais tarde fora estudado.
- b) Estudo da teoria de Algoritmos Genéticos: nesta etapa houve um estudo aprofundado dos conceitos e teorias relacionados aos algoritmos genéticos.
- c) Implementação de um algoritmo genético: esta etapa fechou toda a pesquisa com a implementação de um algoritmo genético para a determinação do ponto de máximo e/ou mínimo de uma função de 2° e 3° grau.

RESULTADOS E DISCUSSÃO

Todas as etapas do trabalho foram concluídas: estudo da linguagem C++, estudo da teoria de Algoritmos Genéticos e a implementação de um algoritmo genético. Um exemplo de programa computacional na linguagem C++ capaz de determinar o ponto de máximo ou mínimo de uma função de 2º grau seria o seguinte:

```
1: #include <cstdlib>
2: #include <iostream>
3: #include <math.h>
4: #include <conio.h>
5: #include <vector>
6: #include <stdio.h>
7: #include <stdlib.h>
8: using namespace std;
9:
10: int main()
11: {
12:     // declaração de variáveis
13:     int a, b, c, n, contador, contador1;
14:     float delta, x1, x2, Xv, Yv;
15:     double m, teste;
16:     vector <double> u, v, x, y;
17:
18:
19:     // leia a, b e c
20:     cout << "Digite os coeficientes da função do 2º grau: \n";
21:     cout << "a=";
22:     cin >> a;
23:     cout << "b=";
24:     cin >> b;
25:     cout << "c=";
26:     cin >> c;
27:
28:     cout << "A função do 2º grau é " << a << "x²" << b << "x" << c << "\n";
29:     // "\n" serve para pular a próxima linha.
30:
31:     // calcular as raízes
32:
33:     delta = pow(b,2) - (4 * a * c);
34:
35:     if (delta>=0)
36:     {
37:         x1 = (-b+sqrt(delta))/2*a;
38:         x2 = (-b-sqrt(delta))/2*a;
39:         cout << "Raiz 1: " << x1 << "\n";
40:         cout << "Raiz 2: " << x2 << "\n";
41:     }
42:     else
43:     {
44:         cout << "Não tem raízes reais \n";
45:     }
46:
47:     // Calcular o Xv e Yv
48:     Xv = - b/(2*a);
49:     Yv = - (delta)/(4 * a);
50:
51:     if ( a > 0)
52:     {
53:         cout << "Resultado: Ponto de mínimo com coordenadas \n";
54:         cout << "Xv: " << Xv << "\n";
55:         cout << "Yv: " << Yv << "\n";
56:     }
57:     else
58:     {
59:         cout << "Resultado: Ponto de máximo com coordenadas \n";
60:         cout << "Xv: " << Xv << "\n";
61:
62:         cout << "Yv: " << Yv << "\n";
63:     }
64:     // Inicia a população
65:
66:     cout << "Populacao inicial: \n";
67:
68:     n=20;
69:     u.resize(n);
70:     v.resize(n);
71:     x.resize(n);
72:     y.resize(n);
73:
74:     for(int i=0; i<n; i++)
75:     {
76:
77:         x[i] = rand() %10;
78:         y[i] = a*pow(x[i],2)+b*x[i]+c;
79:
80:         cout << "(" << x[i] << ", " << y[i] << ") \n";
81:     }
82:
83:
84:     contador=1;
85:     while (contador!=100)
86:     {
87:
88:
89:         // Ordenação dos pares ordenados:
90:         contador1=0;
91:         while (contador1!=19)
92:         {
93:             for(int j=0; j<n; j++)
94:             {
95:                 if ( a < 0 )
96:                 {
97:                     if (j+1=20)
98:                     {
99:                         if (y[j] < y[j+1])
100:                         {
101:                             x[j]=x[j+1];
102:                             y[j]=y[j+1];
103:                         }
104:                     }
105:                 }
106:                 else
107:                 {
108:                     if (j+1=20)
109:                     {
110:                         if (y[j] > y[j+1])
111:                         {
112:                             x[j]=x[j+1];
113:                             y[j]=y[j+1];
114:                         }
115:                     }
116:                 }
117:             }
118:             contador1 = contador1 + 1;
119:         }
120:
121:         // Mutação
122:         for(int j=10; j<n; j++)
123:         {
124:             x[j] = x[j] - rand() %20;
125:             y[j] = (a*pow(x[j],2)+ b*x[j]+c);
126:         }
127:
128:         // Cruzamento
129:         for(int j=1; j<10; j++)
130:         {
131:             x[j] = (x[j] /10000);
132:             y[j] = (a*pow(x[j],2)+ b*x[j]+c);
133:         }
134:
135:         // Impressão da Seleção das melhores pontas
136:
137:         cout << "Veja os pares selecionados: \n";
138:         for(int j=0; j<n; j++)
139:         {
140:             cout << "(" << x[j] << ", " << y[j] << ") \n";
141:         }
142:
143:         contador = contador + 1;
144:     }
145:
146:     // Impressão do melhor ponto de otimização
147:     cout << "O ponto de vertice e " << "(" << Xv << ", " << Yv << ") \n";
148:     if (a>0)
149:     {
150:         cout << "O ponto de mínimo da função pelo algoritmo genético e "
151:         << "(" << x[0] << ", " << y[0] << ") \n";
152:     }
153:     else
154:     {
155:         cout << "O ponto de máximo da função pelo algoritmo genético e "
156:         << "(" << x[0] << ", " << y[0] << ") \n";
157:     }
158:
159:     system("PAUSE");
160:     return EXIT_SUCCESS;
161: }
```

Aplicando nesse algoritmo a função $y = x^2 - 4x + 3$ temos como resultado o ponto (2,-1) que é o ponto de mínimo.

CONCLUSÕES

Procurando conhecer um pouco a Matemática Aplicada, que me interessei pelo estudo de Algoritmos Genéticos. Embora não seja assunto abordado no Ensino Médio, ele é de grande utilidade para o futuro nas áreas de Física, Química e Engenharia.

Enfim, tratou-se de uma teoria motivante que exigiu toda uma retomada de conhecimentos de assuntos da matemática do ensino médio para a construção de resultados interessantes da matemática aplicada.

REFERÊNCIAS BIBLIOGRÁFICAS

BARBOSA, H. J. C. **Introdução aos Algoritmos Genéticos**. SBMAC, 1997.

COELHO, L. S. **Fundamentos, Potencialidades e Aplicações de Algoritmos Evolutivos**. SBMAC (Sociedade Brasileira de Matemática Aplicada e Computacional), 2003. 103p.

DEITEL, H. M. & DEITEL, P. J. **C++: Como Programar**. 5 ed. São Paulo: Pearson Prentice Hall, 2006. 1164p.

LIMA, E. L. Equação do Terceiro Grau. **Matemática Universitária**, Rio de Janeiro, n. 5, p. 9-23, junho. 1987. Disponível em: <rmu.sbm.org.br/Conteudo/n05/n05_Artigo01.pdf>.

MEDINA, M. & FERTIG, C. **Algoritmos e Programação: Teoria e Prática**. 2 ed. São Paulo: Novatec Editora, 2006. 384p.