TÓPICOS EM COMPUTAÇÃO EVOLUTIVA

Algoritmos Evolutivos Canônicos

ALGORITMO EVOLUTIVO SIMPLES – EV (M,N)

- Como representar os indivíduos
  - Vetor de comprimento fixo com L características escolhidas previamente. Ex.: <cor do cabelo,cor do olho, cor da pele, peso, altura>
- Definição das regras de ação, primeiro pseudocódigo

```
Criar uma população inicial de M
  indivíduos.
Faça para sempre
  Selecione um indivíduo da população
  atual para ser o pai
  Uso o pai selecionado para produzir um
  descendente parecido mas não uma cópia
  idêntica
  Selecione um membro da população para
  morrer.
Fim faça
```

ALGORITMO EVOLUTIVO SIMPLES – EV (M,N)

- Refinamento do pseudocódigo

```
Aleatoriamente construa a população inicial com M
  indivíduos (utilizando uma distribuição uniforme
  sobre o espaço geno/fenótipo) e calcule a função de
  aptidão
Faça para sempre
  Selecione um indivíduo da população atual para ser o
  pai
     – Selecione um para aleatoriamente utilizando
  probabilidade uniforme na população
  Uso o pai selecionado para produzir um descendente
     – faça uma cópia idêntica do pai, e então
  probabilisticamente mute-o para produzir o
  descendente
  Calcule a aptidão do novo indivíduo
  Selecione um membro da população para morrer
     – aleatoriamente selecione um candidato para
  apagar da população atual utilizando uma distribuição
  uniforme; e guarde o candidato ou o descendente
  dependendo de qual tem a maior aptidão
Fim faça
```

RESUMINDO EV(M,N)

- População de número fixo de indivíduos
- Cada indivíduo tem comprimento fixo de valores de ponto flutuante (real)
- A função de aptidão é definida por uma função de avaliação.
- Gera a população inicial aleatoriamente e avalia, após isso produz novos indivíduos copiando um pai aleatório e introduzindo alguma modificação por mutação.
- Sobrevive o novo indivíduo ou um indivíduo aleatório por meio da competição de aptidão
- O modelo onde um novo indivíduo é produzido por vez é chamado *steady-state*. Outro modelo alternativo é produzir vários indivíduos de um conjunto de pais.
- O modelo alternativo aplica a competição somente depois que todos os novos indivíduos são produzidos. Para isso utiliza-se dois conjuntos de indivíduos um para os M pais e um para os N descendentes.

EXERCÍCIO PARA PRÁTICA 1

- Implemente EV para computar a função de maximizar $f(x) = 50 - x^2$
- Cada indivíduo possui uma única posição <x> com valor inteiro
- Inicialize os indivíduos no intervalo [-100 100]
- População de 10 indivíduos
- Primeiro aplique mutação de passo 1,0, aleatoriamente defina se deve ser somado ou subtraído o passo.
- Calcule a média da população e o melhor indivíduo a cada iteração
- A cada 10 iterações imprima a população e os valores médios
- Altere o operador de mutação para uma distribuição normal de média 0(zero) e variância a ser especificada.
- Compare os resultados obtidos.

NUMERO ALETÓRIO COM DISTRIBUIÇÃO GAUSSIANA

```
Double ran_gaussian (double sigma) (sigma especifica a
  variância da distribuição)
{
  double x, y, r2;
  do {
       /* escolha x,y no quadrado (-1,-1) to (+1,+1) */
       /*(rand com distribuição uniforme [0,1])*/
       x = -1 + 2 * rand();
       y = -1 + 2 * rand();
       /* verifica se é um circulo unitário */
       r2 = x * x + y * y;
  }
  while (r2 > 1.0 || r2 == 0);
  /* transformada de Box-Muller */
   return sigma * y * sqrt (-2.0 * log (r2) / r2);
}
```

PROGRAMAÇÃO EVOLUTIVA (EP)

- Dado as características estocástica de EV(m,n) é possível que alguns membros da população atual nunca produzam descendentes e que pelo modelo de competição definido na "sorte" indivíduos fracos continuem na população.
- Assim é vantajoso tornar EV(m,n) mais determinístico se assumirmos que ambas as populações possuem igual tamanho (m=n) e modificar EV como segue:

```
Para I=1 até m faça
    Utilize pai I para produzir 1 filho e
    adicione a população de descendentes.
Fim para
Da combinação da população 2m de pais e
filhos, selecione somente os m
indivíduos com melhor aptidão para
sobreviver
```

PROGRAMAÇÃO EVOLUTIVA (EP)

- Essa alteração em EV(m,n) é precisamente a formulação inicial de programação evolutiva proposta por FOGEL.
- Cada pai em EP produz exatamente um filho.
- E indivíduos de aptidão baixa não possuem nenhuma chance de sobreviver.
- Em comparação com EV, EP converge mais rapidamente para uma região de alta aptidão.
- A razão para isso é que EP implemente um forte elitismo sobrevivendo somente os indivíduos de melhor aptidão.
- Essa é uma propriedade útil, mas também pode guiar os algoritmos para sub-ótimos locais.

ESTRATÉGIAS EVOLUTIVAS

- Uma observação sobre EV e EP é que eles produzem um pequeno número de descendentes de um pai, o que não é uma amostra potencial para produção útil.
- Na natureza indivíduos que deixam poucos descendentes são exceção e não regra, assim é interessante que o AE possa produzir mais filhos a partir de um conjunto de pais.
- Para isso é suficiente fazer com que n seja maior do que m.
- Essa idéia corresponde a primeira proposta de estratégias evolutivas.

ESTRATÉGIAS EVOLUTIVAS

- Os algoritmos tem a forma (μ+λ)-EE onde μ especifica o tamanho da população de pais e λ o tamanho da população de descendentes.
- O foco principal das EEs é o modelo (1+λ)-EE no qual toda a próxima geração é produzida por um único pai.
- Uma grande questão é quantos descendentes cada pai deve produzir.
- Comparando (1+10)-EE com EV e EP nota-se que os resultados de EE dependem muito do indivíduo produzido na população inicial.
- Além disso, EEs propõem uma coevolução da parâmetro de variância do operador de mutação Gaussiano.
- Essa coevolução utiliza a regra 1:5 para evoluir a variância, se 1 de 5 mutações é bem sucedida a taxa é aumentada, se a taxa de sucesso cai abaixo de 20% a taxa é diminuída.

ALGORITMOS GENÉTICOS (AG)

- Uma importante observação com relação a EV, EP e EE é que os indivíduos morrem somente se algum indivíduo mais novo possui aptidão melhor.
- Essa característica pode guiar o algoritmo para regiões de ótimos locais e impedir que o ótimo global seja alcançado.
- Uma solução é permitir que ocasionalmente novos indivíduos substituam antigos mesmo que não possuam aptidão maior.
- Um modo mais direto de diminuir o problema é utilizar um modelo geracional como o proposto pelos AGs.

ALGORITMOS GENÉTICOS

- No modelo geracional os pais sobrevivem apenas por uma geração e são substituídos por todos os descendentes
- AGs também utilizam a aptidão de forma diferente do que os demais algoritmos. A aptidão é utilizada para selecionar os indivíduos para reprodução nos AGs, enquanto que nos demais algoritmos a seleção para reprodução é feita de forma totalmente aleatória.

ALGORITMOS GENÉTICOS

GA

Aleatoriamente produza a população de m pais

Repita

Calcule e salve a aptidão u(i) de cada indivíduo na população de pais.

Defina probabilidades, p(i), para cada pai de forma que p(i) é proporcional a u(i).

Gere m descendentes pela seleção probabilística dos pais para produzir descendentes.

Selecione somente os descendentes para sobreviver.

Fim Repita

ALGORITMOS GENÉTICOS

- O método de seleção dos pais especificado no pseudocódigo é chamado de seleção proporcional a aptidão.
- Esse método permite que indivíduos na média produzem na média um descendente, indivíduos abaixo da média produzem menos de um e acima da média mais de um.

ALGORITMOS GENÉTICOS

- Reprodução de Múltiplos Pais
  - Fonte de variação nos descendentes produzidos
  - A idéia básica é que o descendente receba genes de mais de um pai
  - Essa recombinação de genes combinada com a mutação permite uma exploração mais agressiva do espaço de busca
  - A primeira proposta por Holland é conhecida como crossover de um ponto, onde um ponto divide o cromossomo em dois segmentos e o novo indivíduo é composto pelo primeiro segmento de um pai e o segundo de outro pai
  - Outras propostas são crossover de dois pontos e de n pontos.

ALGORITMOS GENÉTICOS

- Código genético universal
  - Holland enfatiza a necessidade de uma representação por string genética universal para representar o genoma de um indivíduo.
  - Por simplicidade é sugeria da representação binária.
  - Nesse caso a mutação é de bit-flip.
  - E a recombinação de crossover de ponto.