Projeto de Inovação com Algoritmos Genéticos

Relatório de Disciplina

José Geraldo de Carvalho Pereira

Análise do problema

▪ Tamanho da População

O primeiro passo no design de algoritmos genéticos competentes é o cálculo para estimar o tamanho da população. O tamanho da população N pode ser estimado usando a equação (Equação 1) desenvolvida por Harik et al.(1997), a qual considera dois fatores que influenciam na convergência para uma solução ótima: (1) o suprimento inicial de building blocks (BBs) e a seleção dos melhores BBs em detrimento de seus competidores.

A equação (1) é baseada em um modelo de random walk que representa o suprimento e a competição entre BBs como o Problema da Ruína do Jogador.

$$N \geq -2^{k-1} \ln(\alpha) \left(\sigma_{bb} \sqrt{\pi (m-1)} \Big/ d \right) \tag{1}$$

Onde k é o tamanho do BB, α é a probabilidade de falha, σ_{bb} é o desvio padrão do fitness dos BBs, d é a diferença entre o melhor e o segundo melhor BB e m representa o número máximo de BBs. O termo $\sigma_{bb} \sqrt{\pi (m-1)}$ representa a interferência do ruído na competiç ão entre BBs.

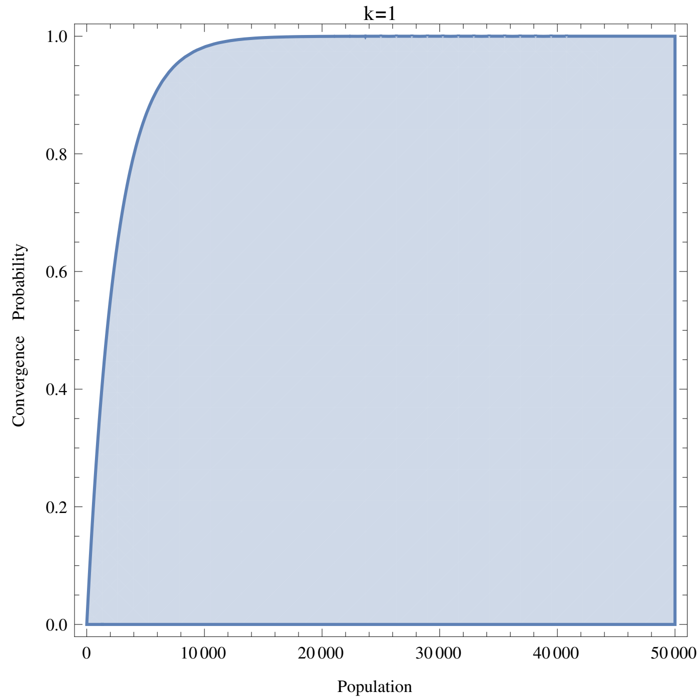
Para assegurar a convergência para uma solução próxima a ótima com uma probabilidade maior que 95%, o α deve ser menor que 5%.

Neste problema temos que k =1, pois cada BB corresponde a um estado dos elementos da regra de transição do autômato celular e m=13248 que são os números desses elementos variáveis na regra de transição. Os parâmetros σ_{bb} e d foram calculados de 500000 indivíduos gerados aleatoriamente e apresentaram os valores de 0,02399451 e 0,001972661.

```
In[48]:= sigma = 0.0241745599487;
dist = 0.001972661;
```

In[50]:=
```
RegionPlot[n ≥ -2^(1-1) Log[1 - x] (sigma Sqrt[Pi (13 248 - 1)] / dist), {n, 0, 50 000} , {x, 0, 1},
 FrameLabel → {"Population", "Convergence Probability"}, PlotLabel → "k=1"]
```

Out[50]=



Com um erro (α) de 0.01 (1%) o tamanho estimado da população é de aproximadamente 11.500 indivíduos.

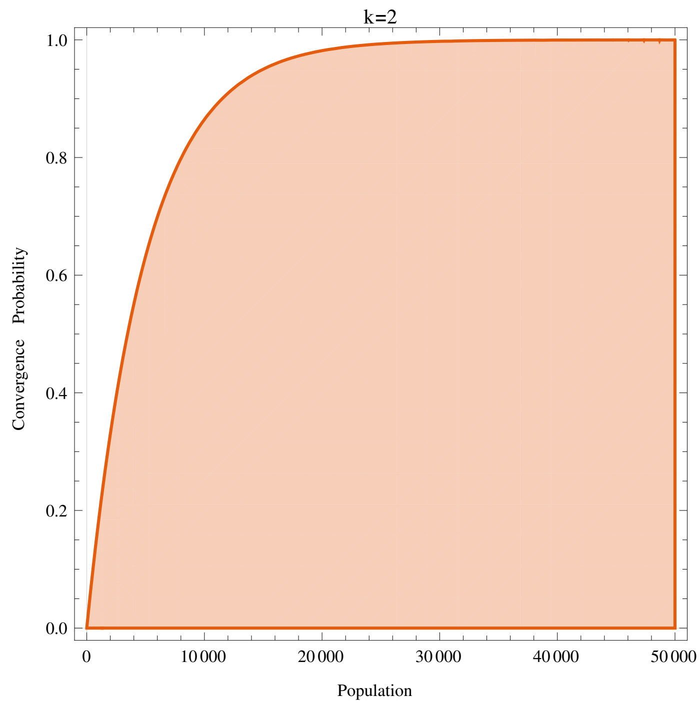
In[51]:=
```
-2^(1-1) Log[0.01] (sigma Sqrt[Pi (13 248 - 1)]) / dist
```

Out[51]= 11 512.9

No entanto, essa equação foi desenvolvida para algoritmos genéticos com representação binária e o nosso problema apresenta 4 estados possíveis para cada elemento da regra de transição dos autômatos. Dessa forma, talvez seja necessário considerar k = 2, o que no GA binário representa 4 possibilidades. Portanto, poderíamos considerar que no nosso problema, k=2 está implícito na codificação.

In[53]:=
```
RegionPlot[n ≥ - (2^(2-1) Log[1 - x] (sigma Sqrt[Pi (13 248 - 1)]))/dist,
  {n, 0, 50 000}, {x, 0, 1}, PlotTheme → "Scientific",
  FrameLabel → {"Population", "Convergence Probability"}, PlotLabel → "k=2"]
```

Out[53]=



Para k=2 e erro (α) de 0.01 (1%) o tamanho estimato da população é de aproximadamente 23000 indivíduos.

In[52]:=
```
-2^(2-1) Log[0.01] (sigma Sqrt[Pi (13 248 - 1)]) / dist
```

Out[52]= 23 025.8

▪ Complexidade computacional

Após a estimação do tamanho da população, podemos analisar a complexidade do GA.

In[59]:=
```
t = 2 * 13 248
```

Out[59]= 26 496

▪ Análise da deriva genética

A deriva genética ocorre em uma população quando a mutação e o crossover fazem os genes flutuarem e convergirem para um solução não ótima na ausência de uma pressão seletiva.

O número experado de gerações para os genes convergirem na ausência de pressão seletiva para uma população inicial de strings binárias geradas randomicamente com proporções iguais de 0s e 1s (Thierens et al., 1998) é estimado como:

$$t_{\text{drift}} = 1.4\,N \tag{2}$$

O que no nosso caso resulta em:

In[58]:= **tdrift = 1.4 × 11 500**

Out[58]= 16 100.

para k =1 e

In[57]:= `tdrift = 1.4 × 23 000`

Out[57]= `32 200.`

para k=2.

A equação 2 mostra que o tempo de convergência devido a deriva genética é uma função linear do tamanho da população. Para assegurar que uma convergência para um ótimo ao invés de uma convergência para por deriva genética podemos satisfazer a condiç ão

$$t < t_{\text{drift}} \tag{3}$$

No nosso problema temos que para k=2 a condição é satisfeita (t=26496, t_{drift} = 32200). No entanto, para k=1 temos que t_{drift} = 16100. Nesse caso, podemos aumentar N para aproximadamente 19000, ou mais, para satisfazer a equação.

In[61]:= `26 496 / 1.4`

Out[61]= `18 925.7`