COMPUTAÇÃO EVOLUCIONÁRIA (2020) – Exercício #1 – Prof. Heitor S. Lopes

## Modelagem do Problema de Otimização da Fabricação de Garrafas Plásticas

**1- Natureza do problema de otimização**

Trata-se claramente de problema de maximização e tem apenas um critério (objetivo) a ser otimizado (lucro).

**2- Conjunto de variáveis do problema e codificação**

As variáveis manipuláveis que afetam a função objetivo são: o número de garrafas tipo leite (L) e o número de garrafas tipo suco (S) fabricadas semanalmente. Portanto, o cromossomo será formado por dois genes, correspondendo ao número de L e S, respectivamente.

Cada variável só pode ter valores inteiros e o limite inferior para ambas é 0. É necessário conhecer o limite superior para as variáveis para determinar o número de bits para representá-las. O limite superior para L é dado pelo enunciado do problema, sendo *700*. O limite superior para S pode ser estimado de diversas maneiras. Uma delas é através da suposição que a fábrica só produzisse S, então produziria *1200* unidades semanalmente. No entanto, se fosse colocado no depósito somente garrafas tipo S, a capacidade seria excedida *(1200*20=24000>14000)*, logo, o limite superior para S também seria 700 *(14000/20)*. Observe que há outras formas de estabelecer este limite superior para S. Considerando o exposto, o cromossomo pode ser codificado com dois genes de 10 bits cada ($L=800<2^{10}$ e $S=700<2^{10}$). Logo, o espaço de busca será $2^{20}=1048576$, que é um valor computacionalmente pequeno e de fácil avaliação com busca exaustiva.

**3- Restrições do problema**

Pela leitura do enunciado é fácil identificar as quatro restrições do problema:

- Tempo máximo de uso da extrusora: $h_1(L,S)=(6/100)*L+(5/100)*S \leq 60$
- Espaço máximo disponível no depósito: $h_2(L,S)=10*L+20*S \leq 14000$
- Demanda máxima de garrafas tipo leite: $h_3(L,S)=L \leq 700$
- Demanda máxima de garrafas tipo suco: $h_4(L,S)=S \leq 700$

Existem diversas alternativas para satisfazer as restrições do problema, sendo a mais simples (e ineficiente) simplesmente eliminar soluções que não satisfaçam todas as restrições. Devido ao caráter didático deste problema e, considerando a codificação em binário natural adotada, é mais interessante utilizar a técnica de penalidades sugerida por Goldberg (p. 85):

$$maximizar\ g(\vec{x}) - r\sum_{i=1}^{n} \Phi[h_i(\vec{x})]$$

O primeiro passo é definir a função de penalidade *Φ*, aplicada ao somatório das restrições. Goldberg sugere a função quadrática, mas aqui não será adotada devido à simplicidade do problema. Em seguida, deve-se garantir que as quatro restrições do problema sempre retornem um valor maior ou igual a zero, isto é: $\forall i \in [1..4], h_i \geq 0$) sempre retornando um valor positivo proporcional à violação dos seus limites (e um valor nulo quando não ocorrer violação). Há um claro problema de escala entre as restrições, pois cada uma se refere a uma grandeza com diferente faixa de valores. Por exemplo, se $h_1(L,S)$ for violado em *10* unidades, isto seria muito mais significativo do que se $h_2(L,S)$ fosse violado nas mesmas *10* unidades, pois o valor máximo para cada um seria *60* e *14000*, respectivamente. Para corrigir isto e normalizar todas as penalidades na faixa *[0..1]*, deve-se dividir cada função de penalização pelo seu valor máximo, obtendo-se as funções normalizadas:

$$\hat{h}_1 = \frac{max\left\{0, \left(\frac{6*L+5*S}{100}\right) - 60\right\}}{60}$$

$$\hat{h}_2 = \frac{max\{0, (10*L+20*S) - 14000\}}{140000}$$

$$\hat{h}_3 = \frac{max\{0, (L-700)\}}{700}$$

$$\hat{h}_4 = \frac{max\{0, (S-700)\}}{700}$$

Como a função *Φ* é o somatório dos $\hat{h}_i$ e estes estão normalizados, no pior caso (obviamente hipotético), *Φ* poderia atingir o valor *4*. Logo, para restringir *Φ* ao mesmo intervalo, divide-se por *4*.

Finalmente, deve-se definir o coeficiente de penalidade *r* que pondera o quanto as restrições afetam o valor da função de *fitness*, no caso mais óbvio é igual a *1*. Observe que quanto menor este valor, menos importante serão as violações das restrições para o valor final da função de *fitness*.

**4- Função objetivo e função de *fitness***

Deseja-se maximizar o lucro, portanto, a função objetivo a ser otimizada é $lucro(L,S)=5*L+4,5*S$. Uma vez que $L, S \in \mathbb{Z}^+$, esta função não gera valores negativos. Desta maneira é garantido o princípio da não-negatividade da função de *fitness*.

É desejável que o valor da função de *fitness* esteja dentro de um intervalo conhecido, por exemplo *[0..1]*, de modo a facilitar a avaliação visual da evolução do algoritmo. Para tanto, é necessário uma normalização desta função e isto é feito dividindo-se a função objetivo por uma constante que represente o máximo que ela possa atingir. Porém, este valor máximo é exatamente o que se deseja encontrar através do algoritmo genético. Assim, este valor deve ser estimado, mesmo que grosseiramente. Uma das maneiras de fazer isto é assumir os valores máximos de

$L$ e $S$ (ambos *700*). Desta forma, obtém-se: *lucro(700,700)=5*700+4,5*700=6650*. Outras formas mais elaboradas seria estimar valores de L e S tais que levassem à ocupação máxima da máquina extrusora ou do depósito.

Considerando a função objetivo, as restrições aos valores das variáveis e penalidades, a função de *fitness* final para este problema é:

$$fitness(L,S) = \frac{5 * L + 4,5 * S}{6650} - \frac{1}{4}\sum_{i=1}^{4} \hat{h}_i$$

**5- Solução ótima**

$(L,S) = (700,350)$

$lucro(L,S) = 5075$

Ocupação da extrusora = 59,5

Ocupação do depósito = 14000