MAC0315 – Programação Linear
Marcos Kazuya Yamazaki, NUSP: 7577622
Victor de Queiros Mattoso Archela dos Santos, NUSP: 7557202

Relatório EP3

Neste exercício programa foi implementado o algoritmo simplex tabular onde a função $[ind\ v\ d] = simplex(A, b, c, m, n)$ calcula o problema de programação, dada a matriz A, o vetor b, o vetor da função objetivo c, o número de restrições m, o número de variáveis n.

Assim como o método simplex, este exercício programa ao chamar a funcao $simplex$, esta por sua vez chama outras duas funções que resolvem respectivamente a fase um e a fase dois do método simplex.

Em ambas as fases, a função $metodoSimplex(c, xb, cj, A, IB, m, n, fase)$ é chamada. Essa função é onde o tableau é manipulada em busca da solução ótima dado um problema.

Ela recebe como parametros de entrada os termos da função objetiva c:

- Que na primeira fase do simplex será composta por n zeros e m uns, pois nesse momento queremos minimizar a função $0 * \sum_{i=1}^{n} x_i + \sum_{i=1}^{m} y_i$, onde $y_1, \ldots, y_m$ são as variáveis auxiliares.
- Já na segunda fase, o vetor será a função objetiva original do problema.

O próximo parametro é o vetor xb que são os valores das variáveis básicas:

- Na primeira fase será os elementos de b_i onde $i = [1, m]$ ou $-b_k$ se o valor de b_k for negativo para algum $k = [1, m]$.
- Na segunda fase, esse vetor será dado como solução pela fase um.

O vetor cj' será calculado através da formula $cj(i) = c(i) - cb' * A(:, i)$, pois nesse programa a matriz A já é o tableau usado no simplex, pois logo no começo fazemos $A = B^{-1} * A$, que é a matriz básica invertida multiplicado pela matriz das restrições do problema de programação linear. Como a única vez em que achamos a matriz inversa básica, a matriz das colunas básicas é uma matriz identidade, já que estamos usando variáveis auxiliares no problema, o tempo gasto será constante.

A matriz A, como falado anteriormente, onde:

- Na primeira fase será composta por $n + m$ colunas e m linhas, onde as últimas m colunas é uma matriz identidade, e as n primeiras colunas é a matriz do problema original.
- Na segunda fase já é uma matriz resultante da fase um, pois pode ter havido a remoção de colunas e linhas redundantes. Como linhas que eram linearmente dependentes.

Também é recebido como parametro de entrada, o vetor IB, que indica os índices das variáveis básicas:

- Na fase um, os índices são $n+1, \dots, n+m$, que serão os índices das variáveis auxiliares.
- Na fase dois, como em outros caso, depende do resultado da fase um do problema.

Outros parametros de entradas, são as variáveis m, que indica o número de restrições que o problema tem, n, que indica o número de variáveis que o problema tem (sem contar com as variáveis auxiliares), e $fase$, que pode valer 1 ou 2, que indica qual a fase do simplex essa função irá resolver.

Dentro da função, é criada uma nova variável np que valerá n, e caso estamos resolvendo a fase um do problema, $n = n + m$. Fazemos isso, para diferenciar quais são as variáveis originais e quais são as auxiliares na hora da impressão. Pois enquanto essa função faz a troca de variáveis, não nos importaremos se ela é original ou auxiliar.

Após isso podemos dividir o que acontece nessa função em três principais etapas:

1. a primeira, é verificar o vetor dos custos reduzidos cj e pegar aquele o índice do primeiro elemento negativo (Regra de Bland), este índice é guardada na variável $entrou$, caso nenhum dos custos forem menores que zero, foi achada a solução ótima do problema.

2. Na coluna $entrou$ da matriz A, é verificada todos os valores, procurando por todos os elementos que forem positivos, se for positivo calculamos um theta temporário $t = xb(i)/A(i, entrou)$, como todos os elementos de xb são positivos, t também será positivo. Escolheremos pela definição o menor t de todas as linhas da coluna $entrou$ da matriz A, pois como cada coluna da matriz é o vetor negativo das direções, pegando o menor deles, temos o maior passo para minimização da função objetiva a partir da solução atual. Caso houver empate, pela regra de Bland, escolheremos aquele de menor índice da linha. E caso não houver nenhum elemento positivo, a o custo da função vai para $-\infty$, na direcão $-A(:, entrou) = d$ a partir da solução viável atual.

3. Após achar os índices quem entram na base e aquela que sai, fazemos as operações elementares de linhas para cada linha da matriz A e para o vetor xb chamando função $pivotar(A, xb, k, entrou, m, n)$ afim de tornar a coluna $entrou$ da matriz A num $k - ésimo$ vetor da base canônica (e^k), onde k é o índice da linha básica que sai da base. Também fazemos as alterações no vetor cj achando os novos custos reduzidos, e zerando os custo das variáveis básicas.

Como falado anteriormente, no ínicio da execução do programa duas funções são chamadas para resolver a fase um e a fase dois do simplex, a função $inicializacaoSimplex(A, b, m, n)$ resolve a primeira fase do simplex.

Primeiramente nesta função, multiplicamos cada linha i da matriz A por -1, caso o elemento b_i for menor que zero, afim de tornar $b_i > 0$ para todo $i = [1, m]$, após isso são criadas as m variáveis auxiliares e calculados os custos reduzidos das variáveis não básicas com a fórmula $\overrightarrow{cj_i} = cj_i - c'_B B^{-1} A^i$, porém como $B^{-1} = B = I_m$ que é a matriz identidade $m\ x\ m$, além disso $i \leq m => cj_i = 0$, e como no problema auxiliar queremos minimizar $\sum_{i=1}^{m} y_i$ o vetor c_B é composta apenas de uns. Logo o custo reduzido será o negativo da somatoria dos elementos de cada coluna da matriz A, e valerá zero para os custos reduzidos básicos que são de índices $> m$ (variáveis auxiliares).

Com isso podemos chamar a função para resolver o problema usando o metodo do simplex, que retornará a nova matriz A que será $B^{-1} * A$, e uma solução viável do problema. Se o valor da função criada for positiva nessa solução viável, isso quer dizer, que o problema original que foi proposto no início é inviável, porém caso seja valha zero, quer dizer que é possivel achar pontos viáveis no problema original. Caso nessa solução contém variáveis auxiliares, agora tentaremos remover-las, retirando a força e tentando colocar uma variável do problema original ou retirando a linha em que a variável auxliar estava, pois essa linha é linearmente dependente (que pode ser vista se a linha contém apenas zeros nas colunas das variáveis originais).

Após a retirada das variáveis auxiliares da solução viável, nessa tabela, o ponto são mandados para começar o metodo simplex novamente a partir dessa solução viável para o problema original passada no início da chamada sem as variáveis auxiliares, afim de achar agora a solução ótima caso houver.

Exemplo de um problema de programação linear com solução ótima

Entrada:

```
A =

   1   1   1
   1  -2   1
   2   2   2

m =  3
n =  3
b =

   2
   2
   4

c =

   1
  -1
  -1
```

Saida:

```
***** Simplex: Fase 1 *****

Iteracao 1:
                 |-> x1    |   x2    |   x3    |   y1    |   y2    |   y3    |
        -8.000 |  -4.000 |  -1.000 |  -4.000 |   0.000 |   0.000 |   0.000 |
   ------------+---------+---------+---------+---------+---------+---------+
<- y1    2.000 |   1.000*|   1.000 |   1.000 |   1.000 |   0.000 |   0.000 |
   y2    2.000 |   1.000 |  -2.000 |   1.000 |   0.000 |   1.000 |   0.000 |
   y3    4.000 |   2.000 |   2.000 |   2.000 |   0.000 |   0.000 |   1.000 |

Theta* = 2.000

Iteracao 2:
               |   x1    |   x2    |   x3    |   y1    |   y2    |   y3    |
         0.000 |   0.000 |   3.000 |   0.000 |   4.000 |   0.000 |   0.000 |
   ------------+---------+---------+---------+---------+---------+---------+
   x1    2.000 |   1.000 |   1.000 |   1.000 |   1.000 |   0.000 |   0.000 |
   y2    0.000 |   0.000 |  -3.000 |   0.000 |  -1.000 |   1.000 |   0.000 |
   y3    0.000 |   0.000 |   0.000 |   0.000 |  -2.000 |   0.000 |   1.000 |

Iteracao 2: Forcar a saida de uma variavel auxiliar
               |   x1    |-> x2    |   x3    |   y1    |   y2    |   y3    |
               |         |         |         |         |         |         |
   ------------+---------+---------+---------+---------+---------+---------+
   x1    2.000 |   1.000 |   1.000 |   1.000 |   1.000 |   0.000 |   0.000 |
<- y2    0.000 |   0.000 |  -3.000*|   0.000 |  -1.000 |   1.000 |   0.000 |
   y3    0.000 |   0.000 |   0.000 |   0.000 |  -2.000 |   0.000 |   1.000 |

Theta* = -0.000

Iteracao 3: Linha redundante, pois a matriz A possui linha(s) LD
               |   x1    |   x2    |   x3    |   y1    |   y2    |   y3    |
               |         |         |         |         |         |         |
   ------------+---------+---------+---------+---------+---------+---------+
   x1    2.000 |   1.000 |   0.000 |   1.000 |   0.667 |   0.333 |   0.000 |
   x2    0.000 |   0.000 |   1.000 |   0.000 |   0.333 |  -0.333 |   0.000 |
<- y3    0.000 |   0.000 |   0.000 |   0.000 |  -2.000 |   0.000 |   1.000 |

Iteracao 4:
               |   x1    |   x2    |   x3    |   y1    |   y2    |
               |         |         |         |         |         |
   ------------+---------+---------+---------+---------+---------+
   x1    2.000 |   1.000 |   0.000 |   1.000 |   0.667 |   0.333 |
   x2    0.000 |   0.000 |   1.000 |   0.000 |   0.333 |  -0.333 |

***** Simplex: Fase 2 *****

Iteracao 1:
               |   x1    |   x2    |-> x3    |
        -2.000 |   0.000 |   0.000 |  -2.000 |
   ------------+---------+---------+---------+
<- x1    2.000 |   1.000 |   0.000 |   1.000*|
   x2    0.000 |   0.000 |   1.000 |   0.000 |

Theta* = 2.000

Iteracao 2:
               |   x1    |   x2    |   x3    |
         2.000 |   2.000 |   0.000 |   0.000 |
   ------------+---------+---------+---------+
   x3    2.000 |   1.000 |   0.000 |   1.000 |
   x2    0.000 |   0.000 |   1.000 |   0.000 |

Solucao otima encontrada com custo: -2.000
x =

   0  -0   2
```

A saida do programa é a tableau do método simplex, onde as variáveis que entram e saem da bases estão demarcadas com uma seta -> e <-, para indicar a variável que entra e sai respectivamente.

Também está indicado com um *, o elemento do vetor u escolhido para o $theta^*$ ser calculado.

As variáveis ausxiliares são impressas como $y1 \ldots ym$.

Na iteração dois, é quando acaba o método simplex da fase um, e como o valor ótimo vale zero, e na base ainda contém variaveis auxiliares, vamos retirar elas. Note que o valor dos custos reduzidos não são necessarias para essa operação.

Enquanto na segunda iteração houve a saida forçada de $y2$, colocaremos $x2$ na base, pois na dua direção u contém um elemento não nulo em uma variável auxiliar $y2$, por isso vamos tirar ela.

Já na terceira iteração a linha de $y3$, contém apenas elementos nulos nas colunas do problema original, ou seja, essa linha é redundante para o problema e pode ser excluida.

Note, também que houve apenas trocas degeneradas de base, ao eliminar essas variáveis auxiliares da base, mantendo o valor da função objetiva do problema auxiliar valendo zero (Iteração 2 e 3 da fase 1).

Exemplo de um problema de programação linear ILIMITADO

Entrada:

```
A =

  -1   2

m =  1
n =  2
c =

  -1
  -3

b = -10
```

Saida:

```
***** Simplex: Fase 1 *****

Iteracao 1:
               |-> x1    |    x2    |    y1    |
       -10.000 |  -1.000 |   2.000 |   0.000 |
   ------------+---------+---------+---------+
<- y1   10.000 |   1.000*|  -2.000 |   1.000 |

Theta* = 10.000


Iteracao 2:
               |    x1   |    x2   |    y1   |
         0.000 |   0.000 |   0.000 |   1.000 |
   ------------+---------+---------+---------+
   x1   10.000 |   1.000 |  -2.000 |   1.000 |

***** Simplex: Fase 2 *****

Iteracao 1:
               |    x1   |    x2   |
        10.000 |   0.000 |  -5.000 |
   ------------+---------+---------+
   x1   10.000 |   1.000 |  -2.000 |

O problema eh ilimitado e o custo vai para menos
infinito na direcao:
d =  2
A partir da solucao viavel:
x =

   10    0
```

Neste caso foi passado um problema em que o resultado é um problema ilimitado, isso quer dizer, que o custo da função objetiva do problema original vai para zero.

A fase do problema um é realizada normalmente, obtendo-se o valor ótimo zero no problema auxiliar, pois se o problema original é ilimitado, só conseguiremos saber isso na fase dois do simplex.

Neste caso, já na primeira iteração obtemos que o custo reduzido na variável não basica $x2$ é negativo. E ao ver os elementos dessa coluna, temos que ela é negativa, logo o custo da função vai para meno infinito a partir do ponto que está na base na direção $-u$.

Exemplo de um problema de programação linear INVIÁVEL

Entrada:

```
A =

  -1  -1

m =  1
n =  2
c =

  -1
  -3

b =  10
```

Saida:

```
***** Simplex: Fase 1 *****

Iteracao 1:
               |    x1    |    x2    |    y1    |
       -10.000 |   1.000  |   1.000  |   0.000  |
   ------------+----------+----------+----------+
   y1   10.000 |  -1.000  |  -1.000  |   1.000  |

O problema eh inviavel
```

Quando o problema passado para o algoritmo simplex, é um problema inviável, ou seja, não contém pontos viáveis que satisfaçam as restriçoes do PL. Isso é verificado logo na fase um do simplex, quando a variável básica do problema auxiliar contém variáveis auxiliares não nulas, e que com isso o valor da função objetiva desse problema é positiva.

Isso mostra que não é possivel achar um ponto viável só com as variáveis oiginais do problema, logo o problema é inviável.