Método Aditivo (enumeração) de BALAS

Um algoritmo enumerativo importante para resolver problemas de programação linear binária é o algoritmo aditivo desenvolvido por Egon Balas (1965). É chamado aditivo porque todas as operações matemáticas realizadas consistem em adicionar ou subtrair. O procedimento consiste em gerar uma sequência de soluções parciais, adicionando uma variável em cada iteração e considerando as soluções complementares (resto de soluções possíveis). Desta forma, podemos, por enumeração implícita, eliminar conjuntos de soluções sem a necessidade de avaliá-las exaustivamente. A seleção da variável adicionada é feita para reduzir ao máximo a inviabilidade na solução atual. Esse método é um procedimento de enumeração que encontra o ótimo mais rapidamente. No método de Balas, a eficácia consiste em avaliar apenas algumas soluções. O método começa colocando todas as variáveis iguais a zero e, em seguida, através de procedimento sistemático atribuindo 0 ou 1 para uma variável; então, substituídos em cada uma das restrições, a inviabilidade é determinada.

Para descrever o algoritmo, a seguinte forma geral de um problema de programação linear binária é considerada.

- A função objetivo deve ser do tipo de maximização, com todos os coeficientes não negativos.
- Todas as restrições devem ser do tipo $\leq$, com o rhs negativo, se necessário.

Seja o PLB (Programa Linear Binário)[1] na forma

$$\begin{aligned} \max \quad & Z = \sum_{j=1}^{n} c_j x_j \\ \text{sa} \quad & \sum_{j=1}^{n} a_{ij} x_j \leq b_i, i = 1, \dots, m \\ & x_j \in \{0,1\}, j = 1, \dots, n \end{aligned}$$

no qual $0 \leq c_n \leq \cdots \leq c_2 \leq c_1$. No caso do problema não se adaptar a esta forma padrão, o mesmo deverá sofrer as seguintes transformações.

a) $\forall c_j < 0$ faça $x_j = 1 - x'_j$ (tanto na função objetivo como nas restrições)

b) troque cada uma das restrições do tipo $\sum_{j=1}^{n} a_{ij} x_j = b_i$ por duas restrições: uma do tipo $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ e outra do tipo $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$.

c) troque cada restrição do tipo $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$ por $-\sum_{j=1}^{n} a_{ij} x_j \leq -b_i$.

[1] http://mayerle.deps.prof.ufsc.br/eps7005.htm
https://e-archivo.uc3m.es/bitstream/handle/10016/10290/PFC_Pablo_Martin_Perez.pdf?sequence=1

d) reordene as variáveis do PLB de modo que $c_n \leq \cdots \leq c_2 \leq c_1$.

O algoritmo proposto por Balas consiste em um processo de busca em uma árvore binária, na qual a cada nó da árvore corresponde uma solução parcial do PLB, cujos valores das k primeiras variáveis são conhecidos. A fim de se verificar a possibilidade de existência de uma solução viável e ótima, a partir desta solução parcial, dois testes deverão ser realizados: i) teste de otimalidade; e ii) teste de viabilidade.

i) Teste de Otimalidade

O teste de otimalidade considera o conhecimento prévio de uma solução ótima parcial, cujo valor da função objetivo é $Z_{ótimo}$. Considerando que na solução parcial as k primeiras variáveis são conhecidas, uma estimativa do valor máximo da função objetivo para uma solução completa derivada desta solução parcial, poderá ser obtida através de:

$$Z_{estimado} = \sum_{j=1}^{k} c_j x_j + \sum_{j=k+1}^{n} c_j$$

Na segunda parcela considerou-se $x_j = 1, j = k + 1, \ldots, n$. As k primeiras variáveis tem seus valores conhecidos; as demais variáveis são denominadas de livres.

Se o valor de $Z_{estimado}$ for superior a $Z_{ótimo}$, então é possível que exista uma solução 'melhor' gerada partir da solução parcial. Caso contrário, não há solução 'melhor' que a disponível (que é a parcial).

ii) Teste de Viabilidade

O teste de viabilidade, por sua vez, considera que para cada restrição deve ser satisfeita a seguinte condição:

$$\sum_{j=1}^{k} a_{ij} x_j + \sum_{j=k+1}^{n} a_{ij} x_j \leq b_i$$

portanto

$$\sum_{j=1}^{k} a_{ij} x_j \leq b_i - \sum_{j=k+1}^{n} a_{ij} x_j \leq b_i - \sum_{j=k+1}^{n} \min(0, a_{ij})$$

Então se para alguma restrição

$$\sum_{j=1}^{k} a_{ij} x_j > b_i - \sum_{j=k+1}^{n} \min(0, a_{ij})$$

conclui-se que não existirão valores para as variáveis ainda não conhecidas que permitam a obtenção de uma solução que satisfaça a restrição.

No caso de falha de um destes dois testes, é necessário rever os valores atribuídos às k primeiras variáveis.

A ideia geral do algoritmo de Balas é começar com uma solução ótima não viável (porém dual viável), e desenvolver a enumeração em busca da solução viável, mantendo a otimalidade. A solução viável, conforme a enumeração avança, que produz o melhor objetivo (até aquele momento) é armazenada.

Passo 0

Escreva o PLB (em sua forma dada acima, neste texto). Faça $Z_{ótimo} = -\infty$ e $k = 1$.

Passo 1

Faça $x_k = 1$. Verifique se

$$Z_{estimado} = \sum_{j=1}^{k} c_j x_j + \sum_{j=k+1}^{n} c_j > Z_{ótimo}$$

e

$$\sum_{j=1}^{k} a_{ij} x_j \leq b_i - \sum_{j=k+1}^{n} \min\left(0, a_{ij}\right) \ , \forall i = 1,2, \dots, m$$

Em caso de sucesso vá ao **Passo 3**

Passo 2

Faça $x_k = 0$. Verifique se:

$$Z_{estimado} = \sum_{j=1}^{k} c_j x_j + \sum_{j=k+1}^{n} c_j > Z_{ótimo}$$

e

$$\sum_{j=1}^{k} a_{ij} x_j \leq b_i - \sum_{j=k+1}^{n} \min\left(0, a_{ij}\right) \ , \forall i = 1,2, \dots, m$$

Em caso de fracasso vá ao **Passo 4**

Passo 3

Faça $k = k + 1$. Se $k \leq n$, volte ao **Passo 1**. Senão uma solução completa melhor que a ótima temporária foi encontrada. Guarde esta como sendo a nova solução ótima temporária. Faça $Z_{ótimo} = Z_{estimado}$.

Passo 4 (*backtrack*)

Encontre $K = \left\{ j \middle| x_j = 1 \text{ e } 0 < j < k \right\}$. Se $K = \emptyset$, então PARE; a solução ótima temporária é a solução ótima do PLB. Em caso contrário, determine $k = \max\{j|j \in K\}$ e retorne ao **Passo 2**.

Quando a enumeração é completada o último valor de $Z_{ótimo}$ é associado à solução ótima do PLB. Se $Z_{ótimo} = -\infty$, então o PLB é inviável.

Segundo Stanley Zionts (1974), o algoritmo de Balas permite explorar de modo eficaz o conjunto de soluções viáveis do problema, através do exame de cada uma delas no máximo uma vez, obtendo a solução ótima (se ela existir) num número finito de iterações. Vale lembrar, no entanto, que o número de possíveis soluções é de 2^n, sendo n o número de variáveis de decisão do PLB.

Como desvantagem do método, pode-se citar o fato de que quando o mesmo é utilizado para resolução de problemas cujos coeficientes associados às variáveis na função objetivo têm valores muito próximos, pode ser necessário um número muito grande de iterações.

Exemplo 1 (Prof. Mayerle, UFSC)

$$\begin{array}{lll} \max & Z = & 10x_1 + 9x_2 + 7x_3 + 5x_4 + 2x_5 + 1x_6 \\ \text{sa} & & 1x_1 - 2x_2 + 2x_3 + 1x_4 - 2x_5 + 1x_6 \leq 1 \\ & & 2x_1 + 1x_2 - 2x_3 - 1x_4 + 1x_5 + 2x_6 \leq 2 \\ & & 3x_1 + 2x_2 - 1x_3 + 3x_4 - 4x_5 + 1x_6 \leq 2 \\ & & x_i \in \{0,1\} \end{array}$$

Passo #0

$$Z_{ótimo} = -\infty \text{ e } k = 1$$

Iteração #1 (K = 1)

X1 = 1

$Z_{estimado} = \sum cx + \sum c = 10 + (9+7+5+2+1) = 34 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $1 \leq 1 - (-2+0+0-2+0) \therefore 1 \leq 5 \therefore ok_2$
- $\sum ax$: $2 \leq 2 - (0-2-1+0+0) \therefore 2 \leq 5 \therefore ok_2$
- $\sum ax$: $3 \leq 2 - (0-1+0-4+0) \therefore 3 \leq 7 \therefore ok_2$

$\Rightarrow$ **Passo 3**: Como ok_1 e ok_2 então K = 2. Ir ao **Passo 1.**

Iteração #2 (K = 2)

X1 = 1, X2 = 1

$Z_{estimado} = \sum cx + \sum c = (10+9) + (7+5+2+1) = 34 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $-1 \leq 1 - (0+0-2+0) \therefore -1 \leq 3 \therefore ok_2$
- $\sum ax$: $3 \leq 2 - (-2-1+0+0) \therefore 3 \leq 5 \therefore ok_2$
- $\sum ax$: $5 \leq 2 - (-1+0-4+0) \therefore 5 \leq 7 \therefore ok_2$

$\Rightarrow$ **Passo 3**: Como ok_1 e ok_2 então K = 3. Ir ao **Passo 1.**

Iteração #3 (K = 3)

$X1 = 1, X2 = 1, X3 = 1$

$Z_{estimado} = \sum cx + \sum c =(10+9+7) + (5+2+1) = 34 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $1 \leq 1 - (0\text{-}2+0) \therefore 1 \leq 3 \therefore ok_2$
- $\sum ax$: $1 \leq 2 - (-1+0+0) \therefore 1 \leq 3 \therefore ok_2$
- $\sum ax$: $4 \leq 2 - (0\text{-}4+0) \therefore 4 \leq 6 \therefore ok_2$

$\Rightarrow$ **Passo 3**: Como ok_1 e ok_2 então K = 4. Ir ao **Passo 1.**

Iteração #4 (K = 4)

$X1 = 1, X2 = 1, X3 = 1, X4 = 1$

$Z_{estimado} = \sum cx + \sum c =(10+9+7+5) + (2+1) = 34 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $2 \leq 1 - (-2+0) \therefore 2 \leq 3 \therefore ok_2$
- $\sum ax$: $0 \leq 2 - (0+0) \therefore 0 \leq 2 \therefore ok_2$
- $\sum ax$: $7 \leq 2 - (-4+0) \therefore 7 \leq 6 \therefore \boldsymbol{\sim ok_2}$

$\Rightarrow$ Como $\boldsymbol{\sim ok_2}$ então ir ao **Passo 2.** (X4 = 0)

$X1 = 1, X2 = 1, X3 = 1, X4 = 0$

$Z_{estimado} = \sum cx + \sum c = (10+9+7+0) + (2+1) = 29 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $1 \leq 1 - (-2+0) \therefore 1 \leq 3 \therefore ok_2$
- $\sum ax$: $1 \leq 2 - (0+0) \therefore 1 \leq 2 \therefore ok_2$
- $\sum ax$: $4 \leq 2 - (-4+0) \therefore 4 \leq 6 \therefore ok_2$

$\Rightarrow$ **Passo 3**: Como ok_1 e ok_2 então K = 5. Ir ao **Passo 1.**

Iteração #5 (K = 5)

$X1 = 1, X2 = 1, X3 = 1, X4 = 0, X5 = 1$

$Z_{estimado} = \sum cx + \sum c = (10+9+7+0+2) + (1) = 29 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $-1 \leq 1 - (0) \therefore -1 \leq 1 \therefore ok_2$
- $\sum ax$: $2 \leq 2 - (0) \therefore 2 \leq 2 \therefore ok_2$
- $\sum ax$: $0 \leq 2 - (0) \therefore 0 \leq 2 \therefore ok_2$

$\Rightarrow$ **Passo 3**: Como ok_1 e ok_2 então K = 6. Ir ao **Passo 1.**

Iteração #6 (K = 6)

$X1 = 1, X2 = 1, X3 = 1, X4 = 0, X5 = 1, X6 = 1$

$Z_{estimado} = \sum cx + \sum c = (10+9+7+0+2+1) \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $0 \leq 1 \therefore ok_2$
- $\sum ax$: $4 \leq 2 \therefore \boldsymbol{\sim ok_2}$
- $\sum ax$: $1 \leq 2 \therefore ok_2$

⇒ Como **~ok_2** então ir ao **Passo 2.** (X6 = 0)

X1 = 1, X2 = 1, X3 = 1, X4 = 0, x5 = 1, X6 = 0

$Z_{estimado} = \sum cx + \sum c = (10+9+7+0+2+0) \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $-1 \leq 1 \therefore ok_2$
- $\sum ax$: $2 \leq 2 \therefore ok_2$
- $\sum ax$: $0 \leq 2 \therefore ok_2$

⇒ **Passo 3**: K = 7, então $Z_{ótimo} = Z_{estimado} = 28$. Ir ao **Passo 4.**

⇒ A resolução do exemplo não foi concluída (ir ao **Passo 4**).

Exemplo 2 (Problema da Mochila)

$$
\begin{array}{lll}
\max & Z = & 7x_1 + 5x_2 + 3x_3 \\
\text{sa} & & 5x_1 + 4x_2 + 2x_3 \leq 8 \\
 & & x_i \in \{0,1\}
\end{array}
$$

Passo #0

$$Z_{ótimo} = -\infty \text{ e } k = 1$$

Iteração #1 (K = 1)

X1 = 1

$Z_{estimado} = \sum cx + \sum c = (7) + (5+3) = 15 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $5 \leq 8 - (0+0) \therefore 5 \leq 8 \therefore ok_2$

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 2. Ir ao **Passo 1.**

Iteração #2 (K = 2)

X1 = 1, X2 = 1

$Z_{estimado} = \sum cx + \sum c = (7+5) + (3) = 15 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $9 \leq 8 - (0) \therefore 9 \leq 8 \therefore$ **~ok_2**

⇒ Como **~ok_2** então ir ao **Passo 2.** (X2 = 0)

X1 = 1, X2 = 0

$Z_{estimado} = \sum cx + \sum c = (7+0) + (3) = 10 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $5 \leq 8 - (0) \therefore 5 \leq 8 \therefore ok_2$

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 3. Ir ao **Passo 1.**

Iteração #3 (K = 3)

X1 = 1, X2 = 0, X3 = 1

$Z_{estimado} = \sum cx + \sum c = (7+0+3)=10 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: $7 \le 8 \therefore ok_2$

⇒ **Passo 3**: K = 4 e $Z_{ótimo} = Z_{estimado} = 10$. Ir ao **Passo 4**.

⇒ A resolução do exemplo não foi concluída (ir ao **Passo 4**).

Exemplo 3

$$
\begin{array}{lll}
\max & Z = & 19x_1 + 10x_2 + 7x_3 + 6x_4 + 5x_5 \\
\text{sa} & & -5x_1 - 1x_2 + 4x_3 - 3x_4 - 3x_5 \le -2 \\
& & 2x_1 + 5x_2 + 3x_3 + 2x_4 - 4x_5 \le 0 \\
& & -1x_1 + 2x_2 - 3x_3 + 5x_4 - 4x_5 \le -2 \\
& & x_i \in \{0,1\}
\end{array}
$$

Passo #0

$$Z_{ótimo} = -\infty \text{ e } k = 1$$

Iteração #1 (K = 1)

X1 = 1

$Z_{estimado} = \sum cx + \sum c = 19 + (10+7+6+5) = 47 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: $-5 \le -2 - (-1+0-3-3) \therefore -5 \le 2 \therefore ok_2$
- $\sum$ ax: $2 \le 0 - (0+0+0-4) \therefore 2 \le 4 \therefore ok_2$
- $\sum$ ax: $-1 \le -2 - (0-3+0-4) \therefore -1 \le 5 \therefore ok_2$

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 2. Ir ao **Passo 1**.

Iteração #2 (K = 2)

X1 = 1, X2 = 1

$Z_{estimado} = \sum cx + \sum c = (19+10) + (7+6+5) = 47 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: $-6 \le -2 - (0-3-3) \therefore -5 \le 4 \therefore ok_2$
- $\sum$ ax: $7 \le 0 - (0+0-4) \therefore 7 \le 3 \therefore$ **$\sim ok_2$**
- $\sum$ ax: $1 \le -2 - (-3+0-4) \therefore 1 \le 5 \therefore ok_2$

⇒ Como **$\sim ok_2$** então ir ao **Passo 2.** (X2 = 0)

X1 = 1, X2 = 0

$Z_{estimado} = \sum cx + \sum c = (19+0) + (7+6+5) = 37 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: -5 ≤ -2 – (0-3-3) ∴ -5 ≤ 1 ∴ ok_2
- $\sum$ ax: 2 ≤ 0 – (0+0-4) ∴ 2 ≤ 4 ∴ ok_2
- $\sum$ ax: -1 ≤ -2 – (-3+0-4) ∴ -1 ≤ 5 ∴ ok_2

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 3. Ir ao **Passo 1.**

Iteração #3 (K = 3)

X1 = 1, X2 = 0, X3 = 1

$Z_{estimado} = \sum cx + \sum c = (19+0+7) + (6+5) = 37 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: -1 ≤ -2 – (-3-3) ∴ -1 ≤ 4 ∴ ok_2
- $\sum$ ax: 5 ≤ 0 – (0-3) ∴ 5 ≤ 3 ∴ **~ok**$_2$
- $\sum$ ax: -4 ≤ -2 – (0-4) ∴ -4 ≤ 2 ∴ ok_2

⇒ Como **~ok**$_2$, então ir ao **Passo 2** (X3 = 0)

X1 = 1, X2 = 0, X3 = 0

$Z_{estimado} = \sum cx + \sum c = (19+0+0) + (6+5) = 30 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: -5 ≤ -2 – (-3-3) ∴ -5 ≤ 4 ∴ ok_2
- $\sum$ ax: 2 ≤ 0 – (0-4) ∴ 2 ≤ 4 ∴ ok_2
- $\sum$ ax: -1 ≤ -2 – (0-4) ∴ -1 ≤ 2 ∴ ok_2

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 4. Ir ao **Passo 1.**

Iteração #4 (K = 4)

X1 = 1, X2 = 0, X3 = 0, X4 = 1

$Z_{estimado} = \sum cx + \sum c = (19+0+0+6) + (5) = 30 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: -8 ≤ -2 – (-3) ∴ -8 ≤ 1 ∴ ok_2
- $\sum$ ax: 4 ≤ 0 – (-4) ∴ 4 ≤ 4 ∴ **~ok**$_2$
- $\sum$ ax: 4 ≤ -2 – (-4) ∴ 4 ≤ 2 ∴ **~ok**$_2$

⇒ Como **~ok**$_2$ então ir ao **Passo 2.** (X4 = 0)

X1 = 1, X2 = 0, X3 = 0, X4 = 0

$Z_{estimado} = \sum cx + \sum c = (19+0+0 +0) + (5) = 24 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum$ ax: -5 ≤ -2 – (-3) ∴ -5 ≤ 1 ∴ ok_2
- $\sum$ ax: 2 ≤ 0 – (-4) ∴ 2 ≤ 4 ∴ ok_2
- $\sum$ ax: -1 ≤ -2 – (-4) ∴ -1 ≤ 2 ∴ ok_2

⇒ **Passo 3**: Como ok_1 e ok_2 então K = 5. Ir ao **Passo 1.**

Iteração #4 (K = 5)

X1 = 1, X2 = 0, X3 = 0, X4 = 0

$Z_{estimado} = \sum cx + \sum c = (19+0+0+0+5)=24 \therefore Z_{estimado} > Z_{ótimo} \therefore ok_1$

- $\sum ax$: $-8 \leq -2 \therefore ok_2$
- $\sum ax$: $-2 \leq 0 \therefore ok_2$
- $\sum ax$: $-5 \leq -2 \therefore ok_2$

⇒ **Passo 3**: $K = 7$. $Z_{ótimo} = Z_{estimado} = 24$. Ir ao **Passo 4.**

⇒ A resolução do exemplo não foi concluída (ir ao **Passo 4**).

Exemplo 4 (exercício)

$$
\begin{array}{lll}
\max & Z = & 16x_1 + 8x_2 + 3x_3 + 1x_4 + 1x_5 \\
\text{sa} & & 2x_1 - 2x_2 + 1x_3 - 2x_4 + 3x_5 \leq 0 \\
 & & -2x_1 + 0x_2 + 1x_3 - 2x_4 + 2x_5 \leq -2 \\
 & & -1x_1 - 5x_2 + 1x_3 + 1x_4 - 2x_5 \leq -5 \\
 & & x_i \in \{0,1\}
\end{array}
$$

X1=X2=X4=1, X3=X5=0, Z=25