Algoritmo de construção de matrizes de zeros e uns com soma das filas prescritas

Rosário Fernandes
Departamento de Matemática
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2829-516 Caparica, Portugal
e-mail: `mrff@fct.unl.pt`

Henrique F. da Cruz
Departamento de Matemática
Universidade da Beira Interior
Rua Marquês D'Avila e Bolama
6201-001 Covilhã, Portugal
e-mail: `hcruz@ubi.pt`

Resumo: Descrevemos um algoritmo para a construção de matrizes cujas posições são zeros e uns satisfazendo determinadas condições envolvendo partições de números inteiros positivos.

Abstract: We describe a simple process for the construction of matrices, with zeros and ones, and satisfying conditions related with partitions of positive integers.

palavras-chave: Matrizes de zeros e uns, quadros de Young; Partições.

keywords: (0,1)-matrices, Young tableaux, Partitions.

1 Matrizes simples ou complicadas?

Ainda que à primeira vista, matrizes cujas posições são apenas zeros e uns, as matrizes-$(0,1)$, possam dar a ideia de serem matrizes com uma estrutura muito simples e portanto mais fáceis de estudar, isto não é de todo verdade. Com efeito, a sua aparente simplicidade dá origem a uma grande variedade de problemas muitos dos quais ainda sem solução. As aplicações das matrizes-$(0,1)$ podem ser encontradas não apenas em muitas áreas da Matemática, nomeadamente numa grande variedade de problemas de Combinatória, mas também em muitas áreas da Física, Química e Biologia. As matrizes-$(0,1)$ em que a sequência das somas das linhas e a sequência das somas das colunas se encontram prescritos têm recebido a atenção de muitos autores, [9, 7, 1, 2, 3, 5, 4]. Pretendemos com este texto descrever um algoritmo de construção de matrizes-$(0,1)$, verificando certas condições.

2 Partições e quadros de Young

Uma *partição* de peso $t \geq 0$ é uma sequência não crescente de números inteiros, não negativos, cuja soma é igual a t. O número de elementos não nulos em λ é o *comprimento* de λ e é denotado por $l(\lambda)$. Quando λ é uma partição de peso t, representamos λ por uma sequência finita $(\lambda_1, \lambda_2, \ldots, \lambda_{l(\lambda)})$, onde $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{l(\lambda)} > 0$.

A *partição conjugada* de $(\lambda_1, \ldots, \lambda_{l(\lambda)})$ é a partição λ^* definida por

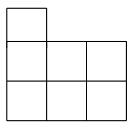
$$\lambda_j^* = |\{i : l(\lambda) \geq i \geq 1,\ \lambda_i \geq j\}|, \quad \text{para } 1 \leq j \leq \lambda_1.$$

Sejam α e β duas sequências (podem ser partições ou não) com o mesmo peso. Dizemos que α é *dominada* por β, $\alpha \preceq \beta$, quando

$$\alpha_1 + \ldots + \alpha_i \leq \beta_1 + \ldots + \beta_i, \qquad \text{para } i \geq 1.$$

Um *diagrama de Young* é uma representação visual do conceito de partição: Se $(\lambda_1, \lambda_2, \ldots, \lambda_{l(\lambda)})$ é uma partição de peso t, o diagrama de Young de λ é uma coleção de t caixas, dispostas em linhas todas com início numa mesma coluna, com λ_i caixas na linha i, para $i \geq 1$ (numerando as linhas de baixo para cima).

Exemplo 2.1 *O diagrama de Young da partição λ tal que $\lambda = (3, 3, 1)$ é*



A partição conjugada da partição λ é a partição $(\lambda_1^*, \lambda_2^*, \lambda_3^*)$ em que

$$\lambda_1^* = |\{i : 3 \geq i \geq 1,\ \lambda_i \geq 1\}| = 3,\ \lambda_2^* = 2,\ \lambda_3^* = 2.$$

Um *quadro de Young de formato* λ, [3, 6], é um preenchimento do diagrama de Young de λ, colocando um inteiro positivo, em cada uma das suas caixas, de acordo com a seguintes regras:

1. os inteiros de cada linha do quadro de Young aparecem por ordem não decrescente (da esquerda para a direita);

2. os inteiros de cada coluna do quadro de Young aparecem por ordem estritamente crescente (de baixo para cima).

Exemplo 2.2 *Um quadro de Young de formato* $(3, 3, 1)$ *é, por exemplo,*

$$\begin{array}{|c|c|c|}\hline 4 & \multicolumn{2}{c}{} \\ \hline 2 & 2 & 3 \\ \hline 1 & 1 & 2 \\ \hline \end{array}$$

3 Algoritmo de inserção

O **algoritmo de inserção** por colunas, [3], é um algoritmo que agarra num quadro de Young T e num inteiro positivo x e constrói um novo quadro de Young $x \hookrightarrow T$. Este novo quadro tem mais uma caixa do que T, sendo as caixas de $x \hookrightarrow T$ preenchidas com os mesmos inteiros de T mais o inteiro x. O algoritmo funciona do seguinte modo:

1. Se x é maior do que todos os inteiros da primeira coluna de T, colocamos uma nova caixa no topo da primeira coluna de T e preenchemo-la com o inteiro x.

2. Caso contrário, procuramos na primeira coluna o maior inteiro k, que é menor ou igual a x. Substituímos, na caixa em que se encontra k, k por x. Inserimos, na coluna seguinte o inteiro k, que foi removido da sua caixa, pelo mesmo processo.

3. Continuamos com este processo até que, um inteiro seja inserido numa nova caixa, criada no topo de uma coluna ou numa nova coluna.

Exemplo 3.1 *Consideremos o seguinte quadro de Young* T *de formato* $(3, 3, 1)$.

$$\begin{array}{|c|c|c|}\hline 4 & \multicolumn{2}{c}{} \\ \hline 2 & 2 & 3 \\ \hline 1 & 1 & 2 \\ \hline \end{array}$$

Usando o algoritmo de inserção por colunas, vamos inserir neste quadro de Young, o inteiro 3. Ora, o maior inteiro da primeira coluna de T *é 4, superior ao nosso 3. Assim, temos que escolher o maior inteiro, da primeira* *coluna, que seja menor ou igual a 3. Neste caso é o 2. Segundo o algoritmo, substituímos o 2 pelo 3 e obtemos o seguinte quadro de Young*

$$\begin{array}{ccc} 4 & & \\ \mathbf{3} & 2 & 3 \\ 1 & 1 & 2 \end{array}$$

ficando com o inteiro 2, para inserir na segunda coluna, pelo mesmo processo. Nesta coluna, o maior inteiro é um 2, inteiro não inferior ao nosso 2. Mas este é o maior inteiro, da segunda coluna, que é menor ou igual ao nosso 2. Assim, substituímos o 2, da segunda coluna, pelo nosso 2 e obtemos o quadro de Young

$$\begin{array}{ccc} 4 & & \\ 3 & \mathbf{2} & 3 \\ 1 & 1 & 2 \end{array}$$

ficando com o inteiro 2, para inserir na terceira coluna, pelo mesmo processo. Nesta coluna, o maior inteiro é um 3, inteiro superior ao nosso 2. O maior inteiro, da terceira coluna que é menor ou igual ao nosso 2, é o 2. Assim, substituímos o 2, da terceira coluna, pelo nosso 2 e obtemos o quadro de Young

$$\begin{array}{ccc} 4 & & \\ 3 & 2 & 3 \\ 1 & 1 & \mathbf{2} \end{array}$$

ficando com o inteiro 2, para inserir na quarta coluna, pelo mesmo processo. Esta coluna não tem caixas, pelo que criamos uma nova caixa e preenchemo-la com o inteiro 2. O processo termina e obtemos o quadro de Young (de formato $(4,3,1)$*).*

$$\begin{array}{cccc} 4 & & & \\ 3 & 2 & 3 & \\ 1 & 1 & 2 & \mathbf{2} \end{array}$$

Ao longo deste texto não colocaremos as caixas nos quadros de Young.

4 Partições e matrizes

Dada uma matriz-$(0,1)$, A (também designada por $[a_{ij}]$), do tipo m por n (com m linhas e n colunas), podemos calcular a soma da linha i da matriz A, que designamos por R_i, i.e.,

$$\sum_{t=1}^{n} a_{it} = R_i,$$

e a soma da coluna j da matriz A, que designamos por S_j, i.e.,

$$\sum_{t=1}^{n} a_{tj} = S_j.$$

Obtemos assim a sequência das somas das linhas de A e a sequência das somas das colunas de A, i.e.,

$$(R_1, \dots, R_m) \ e \ (S_1, \dots, S_n).$$

Nem sempre estas sequências são partições.

Exemplo 4.1 *Sendo A a matriz do tipo 4 por 5, tal que*

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix},$$

então a soma das linhas de A é

$$R_1 = 2, \ R_2 = 4, \ R_3 = 1, \ R_4 = 3,$$

pelo que a sequência das somas das linhas de A é $(2,4,1,3)$, não sendo portanto uma partição. Se calcularmos a sequência das somas das colunas de A obtemos a sequência $(3,2,2,2,1)$, a qual já é uma partição.

Repare que se as sequências das somas das linhas e das somas das colunas de uma matriz-$(0,1)$, A forem duas partições, porque a soma dos inteiros de cada uma destas partições é o número de uns da matriz A, então podemos afirmar que as duas partições são do mesmo peso.

Mais complicado é o problema inverso. Sejam R e S duas partições com o mesmo peso, tais que $R = (R_1, \ldots, R_m)$ e $S = (S_1, \ldots, S_n)$. Denotamos por $\mathcal{A}(R, S)$ o conjunto das matrizes-$(0,1)$, $[a_{i,j}]$, do tipo m por n, que satisfazem

$$\sum_{t=1}^{n} a_{i,t} = R_i \text{ e } \sum_{t=1}^{n} a_{t,j} = S_j,$$

para todo $i = 1, \ldots, m$ e $j = 1, \ldots, n$.

Uma questão que naturalmente se coloca é a de saber em que condições o conjunto $\mathcal{A}(R, S)$ é não vazio. Esta questão foi resolvida, independentemente, por Gale e por Ryser na década de 50 do século XX, [7, 9].

Teorema 4.2 [3, 7, 9]*(Teorema de Gale-Ryser) Sejam R e S duas partições do mesmo peso. Então, $\mathcal{A}(R, S) \neq \emptyset$ se e só se $S \preceq R^*$.*

5 Das matrizes aos quadros de Young

Sejam l,m e n inteiros positivos. Uma *permutação generalizada* é uma matriz do tipo 2 por l cujas colunas formam um subconjunto do conjunto dos pares ordenados de $\{1, \ldots, m\} \times \{1, \ldots, n\}$, ordenado por ordem crescente, pela relação de ordem lexicográfica, i.e.,

$$\begin{pmatrix} i \\ j \end{pmatrix} < \begin{pmatrix} k \\ g \end{pmatrix} \text{ se, e só se, } i < k \text{ ou, } i = k \text{ e } j < g.$$

Existe uma bijeção natural entre o conjunto das matrizes-$(0,1)$ do tipo m por n cuja soma das entradas é igual a l e o conjunto das permutações generalizadas com l colunas, definida por:

$$A = [a_{ij}] \longmapsto \theta_A = \begin{pmatrix} i_1 & \ldots & i_l \\ j_1 & \ldots & j_l \end{pmatrix},$$

onde cada coluna de θ_A corresponde a uma posição de A ocupada por 1.

Com base no algoritmo de inserção por colunas, a correspondência de Robinson-Schensted-Knuth estabelece uma bijeção entre o conjunto das permutações generalizadas definidas no conjunto $\{1, \ldots, m\} \times \{1, \ldots, n\}$ e um conjunto de pares ordenados de quadros de Young, com formatos conjugados (o quadro de inserção P e o quadro de registo Q), [6, 8]. Estes quadros são construídos recursivamente. Começamos com os quadros vazios P_0 e Q_0. Para $k = 1, \ldots, l$, inserimos em P_{k-1} o elemento j_k da segunda linha de θ_A, construindo o quadro P_k, quadro este que tem mais uma caixa que P_{k-1}.

Para a construção do quadro Q_k colocamos, em Q_{k-1} uma nova caixa, na posição conjugada, da nova caixa de P_k. Essa caixa será preenchida com o inteiro i_k. P é o quadro P_l e Q é o quadro Q_l. Portanto, P é constuído com os inteiros de $\{1, \ldots, n\}$ enquanto que Q é construído com os inteiros de $\{1, \ldots, m\}$.

Chamamos formato de uma matriz-$(0,1)$ ao formato do quadro de inserção obtido pela correspondência de Robinson-Schensted-Knuth. O formato de uma matriz de $\mathcal{A}(R,S)$ não é uma partição arbitrária. Na realidade, se λ é o formato de uma matriz de $\mathcal{A}(R,S)$ então $S \preceq \lambda \preceq R^*$. Reciprocamente, se λ é uma partição com o mesmo peso de R e de S, satisfazendo $S \preceq \lambda \preceq R^*$, então existe uma matriz $A \in \mathcal{A}(R,S)$ cujo formato é igual a λ.

Exemplo 5.1 *À matriz A, tal que $A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, corresponde a permutação generalizada θ_A, tal que $\theta_A = \begin{pmatrix} 1 & 2 & 2 & 3 & 3 \\ 2 & 1 & 3 & 1 & 2 \end{pmatrix}$. Pela correspondência de Robinson-Schensted-Knuth obtemos o par de quadros de Young (P,Q), onde*

$$2 \quad \rightarrow \quad 1\ 2 \quad \rightarrow \quad \begin{matrix} 3 & \\ 1 & 2 \end{matrix} \quad \rightarrow \quad \begin{matrix} 3 & & \\ 1 & 1 & 2 \end{matrix} \quad \rightarrow \quad \begin{matrix} 2 & 3 & \\ 1 & 1 & 2 \end{matrix} = P$$

$$1 \quad \rightarrow \quad \begin{matrix} 2 \\ 1 \end{matrix} \quad \rightarrow \quad \begin{matrix} 2 & \\ 1 & 2 \end{matrix} \quad \rightarrow \quad \begin{matrix} 3 & \\ 2 & \\ 1 & 2 \end{matrix} \quad \rightarrow \quad \begin{matrix} 3 & \\ 2 & 3 \\ 1 & 2 \end{matrix} = Q\ .$$

Portanto, a matriz A tem formato $(3,2)$ (formato do quadro P).

6 O algoritmo de Brualdi

Sejam R e S duas partições de peso t, tais que $S \preceq R^*$. Em 2006, Richard Brualdi, [2], apresentou dois algoritmos diretos, isto é, sem recorrer à correspondência de Robinson-Schensted-Knuth, para a construção de matrizes de $\mathcal{A}(R,S)$, cujos formatos sejam R^* e S, respetivamente.

Vamos descrever o algoritmo para a construção de matrizes de $\mathcal{A}(R,S)$, de formato R^*.

Suponhamos que $l(R) = m$ e $l(S) = n$ e seja T a sequência, $(T_1, \ldots, T_u)$, de inteiros positivos, tal que

$$\max_{1 \leq i \leq u} T_i \leq n.$$

Denotamos por $A(T;n)$ a matriz-$(0,1)$, $[a_{i,j}]$, do tipo u por m, tal que

$$\sum_{v=1}^{n} a_{i,v} = T_i, \quad i = 1, \ldots, u,$$

com a propriedade dos 1´s de cada linha ocorrerem nas posições iniciais.

Algoritmo 6.1 ([2])

1. *Consideremos a matriz* $A(R;n)$*;*

2. *Coloquemos* S_n *dos últimos* 1´*s de* S_n *linhas de* $A(R;n)$ *na coluna* n*, escolhendo-os das linhas com maior soma mas, em caso de linhas com igual soma, dando preferência às linhas com menor índice.*

3. *Seja* $A(R'';n-1)$ *a matriz do tipo* m *por* $n-1$ *construída com as colunas* $1, 2, \ldots, n-1$ *de* $A(R;n)$*, após a aplicação do passo 2.. Repetimos o passo 2. com* $A(R'';n-1)$ *em vez de* $A(R;n)$ *e* S_{n-1} *em vez de* S_n*. Continuamos deste modo, até obtermos uma matriz* $\widetilde{A'}$ *de* $\mathcal{A}(R,S)$*.*

Brualdi demonstrou que este algoritmo constrói uma matriz $\widetilde{A'}$ de $\mathcal{A}(R,S)$ com formato R^*, [2].

Exemplo 6.2 *Vamos aplicar este algoritmo no caso em que* $R = (7,5,3)$ *e* $S = (3,2,2,2,2,2,2)$*. Nesta situação temos* $R^* = (3,3,2,2,1,1)$*,* $S \preceq R^*$*. A matriz* $A(R;T)$ *é a matriz* $\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$.

Os números que aparecem em cada matriz, depois dos dois traços verticais, indicam a sequência T*, que surge na matriz* $A(T;n)$*, a que se aplica o algoritmo (matriz que surge antes do traço vertical, em cada matriz).*

$$\left[\begin{array}{ccccccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 7 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 5 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 3 \end{array}\right] \longrightarrow \left[\begin{array}{cccccc|c||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 6 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 4 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 3 \end{array}\right]$$

$$\longrightarrow \left[\begin{array}{ccccc|cc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 5 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 3 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 3 \end{array}\right] \longrightarrow \left[\begin{array}{cccc|ccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 4 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 2 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 3 \end{array}\right]$$

$$\longrightarrow \left[\begin{array}{ccc|cccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 2 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 2 \end{array}\right] \longrightarrow \left[\begin{array}{cc|ccccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 2 \end{array}\right]$$

$$\longrightarrow \left[\begin{array}{c|cccccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right] \longrightarrow \left[\begin{array}{|ccccccc||c} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right]$$

Então, a matriz $\widetilde{A'}$ construída por este algoritmo é a matriz

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

que tem formato R^*, pois

$$\theta_{\widetilde{A'}} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 3 & 3 & 3 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 1 & 3 & 5 & 6 & 7 & 1 & 2 & 4 \end{pmatrix},$$

$$\begin{array}{c} 7 \\ 6 \\ 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{array} \longrightarrow \begin{array}{cc} 7 & \\ 6 & \\ 5 & 7 \\ 4 & 6 \\ 3 & 5 \\ 2 & 3 \\ 1 & 1 \end{array} \longrightarrow \begin{array}{ccc} 7 & & \\ 6 & & \\ 5 & 7 & \\ 4 & 6 & \\ 3 & 4 & 5 \\ 2 & 2 & 3 \\ 1 & 1 & 1 \end{array}$$

7 Generalização do algoritmo de Brualdi

Em [2], R. Brualdi considerou importante a descrição de um algoritmo para a construção de matrizes de $\mathcal{A}(R,S)$ com formato λ, tais que $S \preceq \lambda \preceq R^*$. Vamos em seguida apresentar uma solução para este problema, [4]. O algoritmo por nós proposto tem início com uma matriz construída pelo Algoritmo 6.1, e sem alterar o formato da matriz, vamos construir a matriz pretendida.

Definição 7.1 *Seja A uma matriz-$(0,1)$, $[a_{ij}]$, do tipo m por n. Uma* **sequência em escada** *de A é uma sequência $\mathcal{S}$ de pares ordenados,*

$$\mathcal{S} = ((x_1, y_1), \ldots, (x_p, x_p))$$

tal que:

1) $x_1 < m$;

2) *para todo* $s \in \{1, \ldots, p-1\}$, $x_s = x_{s+1} + 1$;

3) *para todo* $s \in \{1, \ldots, p-1\}$, $y_s < y_{s+1}$;

4) *para todo* $s \in \{1, \ldots, p\}$, $a_{x_s y_s} = 1$;

5) *para todo* $s \in \{1, \ldots, p\}$, $a_{x_s+1, y_s} = 0$.

Observação 7.2 *Uma sequência em escada é uma sequência de posições da matriz* A*, ocupadas por uns, dispostos em linhas consecutivas, que têm na posição abaixo delas, zeros, e que tem o aspeto de uma escada que sobe da esquerda para a direita.*

Exemplo 7.3 *Seja* A *a matriz, tal que* $A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$. *A sequência* S*, tal que* $\mathcal{S} = ((2,1),(1,4)) = (x_1, y_1),(x_2, y_2))$ *é uma sequência em escada porque*

1) $x_1 = 2 < m = 3$;

2) $x_1 = 2 = 1 + 1 = x_2 + 1$;

3) $y_1 = 1 < y_2 = 4$;

4) $a_{x_1 y_1} = a_{21} = 1$, $a_{x_2 y_2} = a_{14} = 1$;

5) $a_{x_1+1, y_1} = a_{31} = 0$, $a_{x_2+1, y_2} = a_{24} = 0$.

Definição 7.4 *Seja* F *uma partição de peso* t *e seja* E*, tal que* $E = (E_1, \ldots, E_m)$*, uma sequência de inteiros não negativos, cuja soma é igual a* t *e tal que* $F \preceq E$*,* $F \neq E$*. Seja* $A = [a_{ij}]$ *uma matriz -*$(0,1)$ *do tipo* $l(F)$ *por* n *tal que*

$$\sum_{j=1}^{n} a_{i,j} = E_i, \quad \text{para } i = 1, \ldots, m.$$

Uma F**-sequência** $\mathcal{S}$ *de* A *é uma sequência em escada*

$$\mathcal{S} = ((x_1, y_1), \ldots, (x_p, y_p)),$$

tal que:

1) *Se* $(x,y) \in \mathcal{S}$ *e* $x = x_i,\ i \neq 1$ *então, para* $0 \leq l < y$,

$$\sum_{s=1}^{l} a_{x,s} + \sum_{\substack{s=l+1,\\ s\neq y_{i-1}}}^{y} a_{x+1,s} < \sum_{s=1}^{y} a_{x,s}$$

e se $x = x_1$ *então para* $0 \leq l < y_1$

$$\sum_{s=1}^{l} a_{x,s} + \sum_{s=l+1}^{y_1} a_{x+1,s} < \sum_{s=1}^{y_1} a_{x,s};$$

2) $\sum_{s=1}^{r} E_s > \sum_{s=1}^{r} F_s$, *para* $x_p \leq r \leq x_1$.

Observação 7.5 *Uma* F*-sequência* $\mathcal{S}$ *de* A *é uma sequência em escada em que se substituirmos os uns, da matriz* A*, que aparecem na sequência em escada, por zeros, e se substituirmos os zeros, que aparecem por baixo destes uns, por uns, obtemos uma matriz* B *em que a sequência* F *é dominada pela sequência das somas das linhas de* B*.*

Exemplo 7.6 *Seja* A *a matriz, tal que*

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

e seja F *a partição, tal que* $F = (2,1,1,1)$*. É fácil verificar que*

1. $F \preceq E$ *(*E *é a sequência da soma das linhas da matriz* A*, isto é,* $E = (2,2,1,0)$*);*

2. *a sequência* $((3,1),(2,3))$ *é uma sequência em escada de* A*;*

3. $(2,3)$ *e* $(3,1)$ *satisfazem a primeira condição da definição anterior;*

4. $\sum_{s=1}^{3} E_s = 5 > \sum_{s=1}^{3} F_s = 4$*;*

5. $\sum_{s=1}^{2} E_s = 4 > \sum_{s=1}^{2} F_s = 3$*.*

Então $((3,1),(2,3))$ *é uma* F*-sequência de* A*.*

Algoritmo 7.7 *Sejam* R*,* S *e* λ *partições de peso* t *tais que* $S \preceq \lambda \preceq R^*$*.*

1. *Usando o algoritmo de Brualdi* (Algoritmo 6.1) *construímos a matriz* $\widetilde{A'} \in \mathcal{A}(\lambda^*, S)$ *(matriz com formato* λ*);*

2. *Seja* B *a matriz do tipo* $l(R) = m$ *por* $l(S) = n$*, tal que a submatriz de* B *formada pelas linhas* $1, \ldots, l(\lambda^*)$ *é igual a* $\widetilde{A'}$ *e as outras linhas de* B *são linhas nulas.*

3. *Seja* K *a matriz tal que* $[k_{ij}] = K = B$*;*

4. *Seja* $\mathcal{S} = ((x_1, y_1), \ldots, (x_p, y_p))$ *uma* F*-sequência de* K*, em que* F *é a partição* S*, que satisfaz:*

   (a) *o inteiro* x_1 *é o maior possível;*

   (b) *os inteiros* $y_1, \ldots, y_p$ *são os maiores possíveis;*

   (c) *o inteiro* p *é o maior possível.*

5. *Seja* D *a matriz tal que* $D = [d_{xy}]$*, onde*

$$d_{xy} = \begin{cases} 0 & \text{se } (x, y) \in \mathcal{S} \\ 1 & \text{se } (x-1, y) \in \mathcal{S} \\ k_{xy} & \text{caso contrário} \end{cases}.$$

   Se $D \notin \mathcal{A}(R, S)$ *repetimos o passo 3. com* $K = D$*. Caso contrário, o algoritmo termina.*

Em [4] demonstrou-se que

Proposição 7.8 *O Algoritmo* 7.7 *constrói uma matriz* $D \in \mathcal{A}(R, S)$ *com formato* λ*.*

Exemplo 7.9 *Sendo* $R = (3^4, 2^2)$*,* $S = (4, 3^4)$ *e* $\lambda = (5^2, 4, 2)$*, então* $\lambda^* = (4^2, 3^2, 2)$ *e* $S \preceq \lambda \preceq R^*$*. Queremos construir uma matriz* $A \in \mathcal{A}(R, S)$ *com formato* λ*. Usando o* Algoritmo 6.1 *obtemos a matriz* $\widetilde{A'}$ *tal que*

$$\widetilde{A'} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Como $l(\lambda)=5$ *e* $l(R)=6$*, vamos juntar uma linha de zeros a* $\widetilde{A'}$ *e obtemos a matriz* B_0 *tal que*

$$B_0=\begin{bmatrix} 0&1&1&1&1\\ 1&0&1&1&1\\ 1&0&1&0&1\\ 1&1&0&1&0\\ 1&1&0&0&0\\ 0&0&0&0&0 \end{bmatrix}.$$

Neste caso, $E_0=(4,4,3,3,2,0)$*, sequência da soma das linhas de* B_0*. Como* $((5,2),(4,4),(3,5))$ *é uma* F*-sequência de* B_0*, em que* F *é a partição* S*, aplicando o Algoritmo 7.7, obtemos a matriz* B_1*, sendo*

$$B_1=\begin{bmatrix} 0&1&1&1&1\\ 1&0&1&1&1\\ 1&0&1&0&0\\ 1&1&0&0&1\\ 1&0&0&1&0\\ 0&1&0&0&0 \end{bmatrix}.$$

B_1 *tem a sequência de soma das linhas* $E_1=(4,4,2,3,2,1)$*. Esta matriz tem a* F*-sequência, em que* F *é a partição* S*,* $((5,1),(4,2),(3,3),(2,5))$*, e através do algoritmo obtemos a matriz* B_2 *onde*

$$B_2=\begin{bmatrix} 0&1&1&1&1\\ 1&0&1&1&0\\ 1&0&0&0&1\\ 1&0&1&0&1\\ 0&1&0&1&0\\ 1&1&0&0&0 \end{bmatrix}.$$

B_2 *tem a sequência de soma das linhas* $E_2=(4,3,2,3,2,2)$*. A sequência* $((2,4),(1,5))$ *é uma* F*-sequência de* B_2*, em que* F *é a partição* S*. Aplicando o algoritmo obtemos* B_3 *em que*

$$B_3=\begin{bmatrix} 0&1&1&1&0\\ 1&0&1&0&1\\ 1&0&0&1&1\\ 1&0&1&0&1\\ 0&1&0&1&0\\ 1&1&0&0&0 \end{bmatrix}.$$

Esta matriz pertence a $\mathcal{A}(R,S)$ *e portanto o algoritmo termina. A matriz* B_3 *tem formato* λ*, como podemos comprovar construindo o quadro de inserção, que se obtém aplicando a correspondência de Robinson-Schensted-Knuth à permutação generalizada* θ_{B_3}*.*

$$\begin{array}{lllll} 5 & 5 & & & \\ 4 & 4 & 4 & 5 & \\ 2 & 2 & 3 & 3 & 3 \\ 1 & 1 & 1 & 1 & 2 \end{array}$$

Referências

[1] R.A. Brualdi, Matrices of zeros and ones with fixed row and column-sum vectors, *Linear Algebra Appl.*, 33: 159–231 (1980).

[2] R.A. Brualdi, Algorithms for constructing $(0,1)$-matrices with prescribed row and column vectors, *Discrete Math.*, 306: 3054–3062 (2006).

[3] R.A. Brualdi, Combinatorial classes of matrices, *Encyclopedia of Mathematics and its Applications*, vol. 108: Cambridge University Press, Cambridge (2006).

[4] R. Fernandes e H.F.da Cruz, An extension of Brualdi′s algorithm for the construction of $(0,1)$-matrices with prescribed row and column sum vectors, *Discrete Math.*, 313: 2365–2379 (2013).

[5] C.M. da Fonseca e R. Mamede, On $(0,1)$-matrices with prescribed row and column vectors, *Discrete Math.*, 309: 2519–2527 (2009).

[6] W. Fulton, Young tableaux, *London Mathematical Society Student Texts*, vol. 35: Cambdrige University Press, Cambridge (1997).

[7] D. Gale, A theorem on flows in networks, *Pacific J. Math*, 7: 1073–1082 (1957).

[8] D.E. Knuth, Permutation matrices and generalized Young tableaux, *Pacific J. Math.*, 34: 709–727 (1970).

[9] H. J. Ryser, Combinatorial properties of matrices of zeros and ones, *Canad. J. Math*, 9: 371–377 (1957).