Elaboration and Implantation of an Algorithm
Solving a Capacitated Four-Index Transportation Problem

R. Zitouni, A. Keraghel and D. Benterki

Département de Mathématiques  
Faculté des sciences  
Université Ferhat Abbas  
Sétif, 19000, Algérie

Abstract
A simplicial algorithm was introduced by R. Zitouni and A. Keraghel in 2003 for solving a capacitated transportation problem with four subscripts. In this paper, we are interested in the numerical study of this algorithm in order to realize a practical software to meet real needs. The tests carried out on different examples are significant.

Mathematics Subject Classification: 65K05, 90C05, 90C08

Keywords: Linear Programming, Transportation Problem with Four Subscripts, Capacitated Problem

1 Introduction
The theoretical and algorithmic bases of the classical two-index transportation problem have been formulated by several world-famous researchers like F. L. Hitchcock, G. B. Dantzig, L. V. Kantorovich and M. K. Savourine, ... during the period 1940-1960.

Most of these developments are based on linear programming technics.
In the decade 1960-1970, intensive works have been devoted to the three-, and more generally, to the multi-index transportation problem but without capacities, see for instance references [3] and [4].

In fact, the capacities of the transportation paths are mathematically modeled as additional constraints to express very important real needs. Obviously, this involves some theoretical and algorithmical complications which are often difficult to treat in a general context. This justify in part, an almost absence of significant studies related to capacitated transportation problems with an index greater than two.

In this paper, we focus our attention to the numerical implementation of the algorithm introduced by R. Zitouni and A. Keraghel in reference [15] for solving the previously transportation problem. We shall show via some tests that the results are significant.

2 Statement of the problem

The capacitated four-index transportation problem $T_C$ is formulated as follows:

\[
\text{Minimize } Z = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} \sum_{l=1}^{q} c_{ijkl} x_{ijkl}
\]

subject to the constraints:

\[
\sum_{j=1}^{n} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ijkl} = \alpha_i \quad \text{for all } i = 1, \ldots, m,
\]

\[
\sum_{i=1}^{m} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ijkl} = \beta_j \quad \text{for all } j = 1, \ldots, n,
\]

\[
\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{l=1}^{q} x_{ijkl} = \gamma_k \quad \text{for all } k = 1, \ldots, p,
\]

\[
\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} x_{ijkl} = \delta_l \quad \text{for all } l = 1, \ldots, q,
\]

\[
0 \leq x_{ijkl} \leq d_{ijkl} \quad \text{for all } (i, j, k, l).
\]

In this problem, $\alpha_i, \beta_j, \gamma_k, \delta_l, d_{ijkl}$ and $c_{ijkl}$ are given and are such that for all $i, j, k, l$, we have $\alpha_i > 0, \beta_j > 0, \gamma_k > 0, \delta_l > 0, d_{ijkl} > 0$ and $c_{ijkl} \geq 0$.

This problem can be equivalently formulated as the linear program

\[
\text{Min } [c^t x : \ Ax = b, \ 0 \leq x \leq d],
\]
where \( x, c, d \in \mathbb{R}^{mnpq} \) \( b \in \mathbb{R}^{m+n+p+q} \) and \( A \) is a \((m+n+p+q) \times (mnpq)\) matrix.

A feasible solution \( x = (x_{ijkl}) \) of \((T_C)\) is called a program.

A program \( x \) is called basic if the columns of the submatrix \( A_x \) obtained from \( A \) by keeping only the columns corresponding to the variables \( x_{ijkl} \) such that

\[
0 < x_{ijkl} < d_{ijkl}
\]

are linearly independent.

A basic program \( x \) is said to be non degenerate if

\[
\text{rank}(A_x) = \text{rank}(A).
\]

Given a basic program \( x \), the 4-tuple \((i, j, k, l)\) is called interesting if

\[
0 < x_{ijkl} < d_{ijkl}.
\]

We assume that the following feasibility assumption holds

\[
\sum_{i=1}^{m} \alpha_i = \sum_{j=1}^{n} \beta_j = \sum_{k=1}^{p} \gamma_k = \sum_{l=1}^{q} \delta_l = H.
\]

It results that

\[
\text{Rank}(A) = m + n + p + q - 3.
\]

It is useful to present the data of the problem via the following transportation table. It consists of an array of \((m+n+p+q)\) rows and \((mnpq)\) columns, three additional rows and an additional column. The entries of column \( P_{ijkl} \) of the first, second, and third additional rows are for the data of the quantities \( d_{ijkl}, c_{ijkl}, \) and \( x_{ijkl} \) respectively. The additional column is for the data of quantities \( \alpha_i, \beta_j, \gamma_k, \) and \( \delta_l \) respectively. Finally the entry of the array on the line corresponding to \( \alpha_i \) and the column \( P_{i'jkl} \) is 1 if \( i = i' \) and 0 if not. Same thing for \( \beta_j, \gamma_k, \) and \( \delta_l \). We give an example below.
3 Algorithm

The following algorithm shares with the simplex method and the potential methods a structure consisting in two phases, a finite convergence and the use of the pivot principle.

**Phase 1:** (It finds a basic program or says that \((T_C)\) is not solvable)

**Step 1:**

**Initialization:**

For all \((i, j, k, l)\), \(\hat{\alpha}_i = \alpha_i\), \(\hat{\beta}_j = \beta_j\), \(\hat{\gamma}_k = \gamma_k\), \(\hat{\delta}_l = \delta_l\) and \(b_{ijkl} = 0\), \((b_{ijkl}\) is a boolean variable equal to 1 if \(x_{ijkl}\) has already been determined and 0 if not yet),

\[ E = \{(i, j, k, l), \text{ such that } b_{ijkl} = 0\}. \]
Iteration:

While $E \neq \phi$ do

- Choose an 4-tuple $(\bar{i}, \bar{j}, \bar{k}, \bar{l}) \in E$, such that $c_{\bar{i}\bar{j}\bar{k}\bar{l}} = \min_{(i,j,k,l) \in E} c_{ijkl}$, (see Remark 1, below)

- Take $x_{\bar{i}\bar{j}\bar{k}\bar{l}} = \min(\hat{\alpha}_i, \hat{\beta}_j, \hat{\gamma}_k, \hat{\delta}_l, d_{\bar{i}\bar{j}\bar{k}\bar{l}})$, and $b_{\bar{i}\bar{j}\bar{k}\bar{l}} = 1$, (i.e., $x_{\bar{i}\bar{j}\bar{k}\bar{l}}$ is determined),

- Update $\hat{\alpha}_i, \hat{\beta}_j, \hat{\gamma}_k, \hat{\delta}_l$ as follows
  \begin{enumerate}
  \item $\hat{\alpha}_i = \hat{\alpha}_i - x_{\bar{i}\bar{j}\bar{k}\bar{l}}$.
    if $\hat{\alpha}_i = 0$ then take $x_{ijkl} = 0$ for all $(j, k, l) \neq (\bar{j}, \bar{k}, \bar{l})$ and $b_{ijkl} = 1$ for all $(j, k, l)$,
  \item $\hat{\beta}_j = \hat{\beta}_j - x_{\bar{i}\bar{j}\bar{k}\bar{l}}$.
    if $\hat{\beta}_j = 0$ then take $x_{ijkl} = 0$ for all $(i, k, l) \neq (\bar{i}, \bar{k}, \bar{l})$ and $b_{ijkl} = 1$ for all $(i, k, l)$,
  \item $\hat{\gamma}_k = \hat{\gamma}_k - x_{\bar{i}\bar{j}\bar{k}\bar{l}}$.
    if $\hat{\gamma}_k = 0$ then take $x_{ijkl} = 0$ for all $(i, j, l) \neq (\bar{i}, \bar{j}, \bar{l})$ and $b_{ijkl} = 1$ for all $(i, j, l)$,
  \item $\hat{\delta}_l = \hat{\delta}_l - x_{\bar{i}\bar{j}\bar{k}\bar{l}}$.
    if $\hat{\delta}_l = 0$ then take $x_{ijkl} = 0$ for all $(i, j, k) \neq (\bar{i}, \bar{j}, \bar{k})$ and $b_{ijkl} = 1$ for all $(i, j, k)$.
  \end{enumerate}
Step 2:

a) Take

\[ \varepsilon = \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j = \sum_{k=1}^{p} e_k = \sum_{l=1}^{q} f_l, \]  

such that:

\[ a_i = \alpha_i - \sum_{j=1}^{n} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ijkl} \quad \text{with} \quad i = 1, \ldots, m, \]
\[ b_j = \beta_j - \sum_{i=1}^{m} \sum_{k=1}^{p} \sum_{l=1}^{q} x_{ijkl} \quad \text{with} \quad j = 1, \ldots, n, \]
\[ e_k = \gamma_k - \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{l=1}^{q} x_{ijkl} \quad \text{with} \quad k = 1, \ldots, p, \]
\[ f_l = \delta_l - \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} x_{ijkl} \quad \text{with} \quad l = 1, \ldots, q. \]

i) If \( \varepsilon = 0 \), then \( x = (x_{ijkl}) \) is an initial basic program for the problem \((T_C)\), we denote it by \( x^{(0)} \). Go to Phase 2.

ii) Construct a problem \( T_C(\tilde{M}) \) by the procedure described in (P1) below, and find an initial basic program \( x^{(0)} \) for the problem \( T_C(\tilde{M}) \), as in step 1.

Then, according to Remark 1 \( x^{(0)}_{m+1,n+1,p+1,q+1} = 0 \)

\((\pi^{(0)} = (x_{ijkl}), \text{with } i = 1, \ldots, m+1, j = 1, \ldots, n+1, k = 1, \ldots, p+1 \text{ and } l = 1, \ldots, q+1)\).

If \( \pi^{(0)} \) is optimal then the problem \( (T_C) \) is not solvable. Stop.

b) Improvement of a basic program for \( T_C(\tilde{M}) \).

Initialization: \( r = 1, \varepsilon > 0 \) is given,

1) Determine \( \pi^{(r)} \) as in Phase 2.

2) If \( x^{(r)}_{m+1,n+1,p+1,q+1} = \varepsilon \), then \( x^{(r)} = (x^{(r)}_{ijkl}) \) with \( i = 1, \ldots, m \), \( j = 1, \ldots, n \), \( k = 1, \ldots, p \), and \( l = 1, \ldots, q \), is an initial basic program for the problem \((T_C)\). Go to Phase 2.

3) If \( \pi^{(r)} \) is optimal (Phase 2), then the problem \( (T_C) \) is not solvable. Stop.

4) Do \( r = r + 1 \) and repeat 1), to 3).
Next, we describe the second phase.

**Phase 2:** *(Research of an optimal program for \((T_C)\))*

When Phase 2 starts, we know an initial basic program \(x^{(0)}\). Take \(r = 0\).

a) Determine the set \(I^{(r)}\) of the interesting 4-tuples \((i, j, k, l)\), (see Remark 2 below).

b) For all \((i, j, k, l) \in I^{(r)}\), solve the linear system

\[
    u_i^{(r)} + v_j^{(r)} + w_k^{(r)} + t_l^{(r)} = c_{ijkl}.
\]

c) For all \((i, j, k, l) \notin I^{(r)}\) take

\[
    \Delta_{ijkl}^{(r)} = c_{ijkl} - (u_i^{(r)} + v_j^{(r)} + w_k^{(r)} + t_l^{(r)})
\]

and

\[
    \Gamma^{(r)}_0 = \{ \Delta_{ijkl}^{(r)} \text{ such that } x_{ijkl}^{(r)} = 0 \},
\]

\[
    \Gamma^{(r)}_d = \{ \Delta_{ijkl}^{(r)} \text{ such that } x_{ijkl}^{(r)} = d_{ijkl} \}.
\]

If the following optimality condition holds

\[
    \Delta_{ijkl}^{(r)} \geq 0 \quad \text{for all } \Delta_{ijkl}^{(r)} \in \Gamma^{(r)}_0
\]

and

\[
    \Delta_{ijkl}^{(r)} \leq 0 \quad \text{for all } \Delta_{ijkl}^{(r)} \in \Gamma^{(r)}_d,
\]

then the program \(x^{(r)}\) is optimal. Stop.

d) Determine

\[
    \Delta_{i_0j_0k_0l_0}^{(r)} = \min_{(i,j,k,l)} \left[ \Delta_{ijkl}^{(r)} - \Delta_{ijkl}^{(r)} \right] \text{ such that:}
\]

\[
    \Delta_{ijkl}^{(r)} \in \Gamma^{(r)}_0, \quad \text{with } \Delta_{ijkl}^{(r)} < 0
\]

and

\[
    \Delta_{ijkl}^{(r)} \in \Gamma^{(r)}_d, \quad \text{with } \Delta_{ijkl}^{(r)} > 0,
\]

and specify if \(\Delta_{i_0j_0k_0l_0}^{(r)} \in \Gamma^{(r)}_0\) (or \(\in \Gamma^{(r)}_d\)).

e) Construct via the procedure described in (P2) below, a cycle \(\mu^{(r)}\) containing some interesting 4-tuples \((i, j, k, l)\) and the non interesting 4-tuple \((i_0, j_0, k_0, l_0)\) corresponding to \(\Delta_{i_0j_0k_0l_0}^{(r)}\).
Take
\[ \sigma^{(r)} = \{(i, j, k, l) \text{ such that } (i, j, k, l) \text{ is a } 4\text{-tuple forming the cycle } \mu^{(r)}\} , \]
\[ \sigma^{(r)-} = \{(i, j, k, l) \text{ such that } (i, j, k, l) \in \sigma^{(r)}, \text{ with } \alpha_{ijkl} < 0\} , \]
\[ \sigma^{(r)+} = \{(i, j, k, l) \text{ such that } (i, j, k, l) \in \sigma^{(r)}, \text{ with } \alpha_{ijkl} > 0\} . \]

\textbf{If} \( \Delta_{ijkl}^{(r)} \in \Gamma_0^{(r)} \), determine
\[ \theta_1^{(r)} = \min_{(i,j,k,l) \in \sigma^{(r)-}} \left( x_{ijkl}^{(r)} / -\alpha_{ijkl} \right) , \]
\[ \theta_2^{(r)} = \min_{(i,j,k,l) \in \sigma^{(r)+}} \left( (d_{ijkl} - x_{ijkl}^{(r)}) / \alpha_{ijkl} \right) , \]
\[ \theta^{(r)} = \min(\theta_1^{(r)}, \theta_2^{(r)}) . \]

Next, take
\[ x^{(r+1)} = \{x_{ijkl}^{(r)} + \alpha_{ijkl}\theta^{(r)}, \ (i, j, k, l) \in \sigma^{(r)}\} \cup \{x_{ijkl}^{(r)}, \ (i, j, k, l) \notin \sigma^{(r)}\} . \]

\textbf{Else} \( \Delta_{ijkl}^{(r)} \in \Gamma_d^{(r)} \), determine
\[ \theta_1^{(r)} = \min_{(i,j,k,l) \in \sigma^{(r)+}} \left( x_{ijkl}^{(r)} / \alpha_{ijkl} \right) , \]
\[ \theta_2^{(r)} = \min_{(i,j,k,l) \in \sigma^{(r)-}} \left( (d_{ijkl} - x_{ijkl}^{(r)}) / -\alpha_{ijkl} \right) , \]
\[ \theta^{(r)} = \min(\theta_1^{(r)}, \theta_2^{(r)}) . \]

Next, take
\[ x^{(r+1)} = \{x_{ijkl}^{(r)} - \alpha_{ijkl}\theta^{(r)}, \ (i, j, k, l) \in \sigma^{(r)}\} \cup \{x_{ijkl}^{(r)}, \ (i, j, k, l) \notin \sigma^{(r)}\} . \]

\textbf{f) Do } r = r + 1 \text{ and repeat a), to e) until the optimality condition holds.}

In the description of the above \textbf{Algorithm}, we have made reference to the two following remarks:

\textbf{Remark 1: If there are several elements corresponding to the minimum of } \ c_{ijkl}, \text{ we choose one, for instance the first found in the transportation table by going from the left to the right.}

\textbf{Remark 2: If the program is degenerate (i.e., the number of columns of } \ Ax, \text{ is strictly less than } \text{rank}(A), \text{ we complete } \ Ax \text{ with additional columns so that we obtain a matrix having } \text{rank}(A) \text{ linearly independent columns. Next } I^{(r)} \text{ can be determined.
Also, the above Algorithm makes appeal to the two following procedures:

(P1)- **Construction of a problem** $T_C(\tilde{M})$:

The problem $T_C(\tilde{M})$ is obtained from problem $(T_C)$ by adding four fictitious points with indices $m+1, n+1, p+1, and q+1$ such that:

$c_{m+1,n+1,p+1,q+1} = 0, c_{m+1,jkl} = c_{i,n+1,kl} = c_{ij,p+1,l} = c_{ijk,q+1} = \tilde{M}$ (where $\tilde{M}$ is a very large number) and there are no limitation on the capacities for the paths involving a fictitious point.

(P2)- **Determination of cycles**:

A cycle $\mu^{(r)}$ is determined by solving the linear system

$$\sum_{(i,j,k,l) \in I^{(r)}} \alpha_{ijkl} P_{ijkl} = -P_{i0j0k0l0}$$

The non null solutions $\alpha_{ijkl}$ are called coefficients of the cycle $\mu^{(r)}$.

4 Numerical tests

Our tests are realized on a Pentium IV with a Microsoft Windows Environment, they are totally written with Borland Delphi 4.

Recall that our objective is the implementation of the algorithm. In fact, we have constructed some examples with different dimensions permitting to measure the stability and the robustness of the algorithm.

In the following examples, we present the optimal solutions only by their components non null.

**Example 1** Considering a transportation problem as $(T_C)$ with:

$$m = n = p = q = 2, \quad \alpha_1 = 15, \quad \alpha_2 = 7, \quad \beta_1 = 12, \quad \beta_2 = 10, \quad \gamma_1 = 6, \quad \gamma_2 = 16, \quad \delta_1 = 8, \quad \delta_2 = 14.$$  

The quantities $c_{ijkl}$ and $d_{ijkl}$ are given in the following table:

<table>
<thead>
<tr>
<th>$(i, j, k, l)$</th>
<th>1111</th>
<th>1112</th>
<th>1121</th>
<th>1122</th>
<th>1211</th>
<th>1212</th>
<th>1221</th>
<th>1222</th>
<th>2111</th>
<th>2112</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_{ijkl}$</td>
<td>5</td>
<td>7</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$c_{ijkl}$</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>$(i, j, k, l)$</td>
<td>2121</td>
<td>2122</td>
<td>2211</td>
<td>2212</td>
<td>2221</td>
<td>2222</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$d_{ijkl}$</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$c_{ijkl}$</td>
<td>9</td>
<td>4</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The optimal solution is: \( x^* = (x_{ijkl}) \) such that:

\[
\begin{align*}
  x_{1112} &= 6, & x_{1121} &= 3 = d_{1121}, \\
  x_{2221} &= 3, & x_{1222} &= 4 = d_{1222}, \\
  x_{2122} &= 3, & x_{1221} &= 2 = d_{1221}, \\
  x_{2222} &= 1.
\end{align*}
\]

Number of iterations = 07. Time of execution = 0, 22 sec.
The optimal value is \( z^* = 55 \).

**Example 2** Considering a transportation problem as \((T_C)\) with:

\[
m = n = p = q = 2, \quad \alpha_1 = 15, \quad \alpha_2 = 15, \quad \beta_1 = 10, \quad \beta_2 = 20, \quad \gamma_1 = 12, \\
\gamma_2 = 18, \quad \delta_1 = 9, \quad \delta_2 = 21.
\]

The quantities \(c_{ijkl}\) and \(d_{ijkl}\) are given in the following table:

<table>
<thead>
<tr>
<th>((i, j, k, l))</th>
<th>1111</th>
<th>1112</th>
<th>1121</th>
<th>1122</th>
<th>1211</th>
<th>1212</th>
<th>1221</th>
<th>1222</th>
<th>2111</th>
<th>2112</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>15</td>
<td>25</td>
<td>45</td>
<td>12</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>12</td>
<td>12</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i, j, k, l))</th>
<th>2121</th>
<th>2122</th>
<th>2211</th>
<th>2212</th>
<th>2221</th>
<th>2222</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>11</td>
<td>22</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>10</td>
</tr>
</tbody>
</table>

The optimal solution is: \( x^* = (x_{ijkl}) \) such that:

\[
\begin{align*}
  x_{1122} &= 9, 5, & x_{1211} &= 0, 5, \\
  x_{1212} &= 5, & x_{2111} &= 0, 5, \\
  x_{2221} &= 8 = d_{2221}, & x_{2212} &= 6 = d_{2212}, \\
  x_{2222} &= 0, 5.
\end{align*}
\]

Number of iterations = 03. Time of execution = 0, 16 sec.
The optimal value is \( z^* = 85 \).

**Example 3** Considering a transportation problem as \((T_C)\) with:

\[
m = 3, \quad n = p = q = 2, \quad \alpha_1 = 6, \quad \alpha_2 = 8, \quad \alpha_3 = 12, \quad \beta_1 = 14, \quad \beta_2 = 12, \\
\gamma_1 = 9, \quad \gamma_2 = 17, \quad \delta_1 = 8, \quad \delta_2 = 18.
\]

The quantities \(c_{ijkl}\) and \(d_{ijkl}\) are given in the following table:
Capacitated four-index transportation problem

The optimal solution is: \( x^* = (x_{ijkl}) \) such that:

\[
\begin{align*}
x_{1121} &= 2, & x_{1221} &= 3, \\
x_{1222} &= 1, & x_{2222} &= 8, \\
x_{3112} &= 9, & x_{3121} &= 3.
\end{align*}
\]

Number of iterations = 0. Time of execution = 0.11 sec.

The optimal value is \( z^* = 55 \).

Example 4 Considering a transportation problem as \((T_C)\) with:

\[
m = p = 3, \quad n = q = 2, \quad \alpha_1 = 15, \quad \alpha_2 = 15, \quad \alpha_3 = 10, \quad \beta_1 = 15, \\
\beta_2 = 25, \quad \gamma_1 = 12, \quad \gamma_2 = 14, \quad \gamma_3 = 14, \quad \delta_1 = 18, \quad \delta_2 = 22.
\]

The quantities \( c_{ijkl} \) and \( d_{ijkl} \) are given in the following table:

\[
\begin{array}{cccccccccccc}
(i, j, k, l) & 1111 & 1112 & 1121 & 1122 & 1131 & 1132 & 1211 & 1212 & 1221 & 1222 \\
\hline
\(d_{ijkl}\) & 12 & 15 & 14 & 16 & 22 & 13 & 21 & 20 & 25 & 29 \\
\(c_{ijkl}\) & 5 & 2 & 3 & 7 & 8 & 5 & 4 & 2 & 6 & 9 \\
\hline
(i, j, k, l) & 2121 & 2122 & 2211 & 2212 & 2221 & 2222 & 3111 & 3112 & 3121 & 3122 \\
\hline
\(d_{ijkl}\) & 26 & 20 & 9 & 8 & 14 & 12 & 14 & 15 & 18 & 24 \\
\(c_{ijkl}\) & 8 & 5 & 4 & 7 & 4 & 1 & 5 & 2 & 3 & 6 \\
\hline
(i, j, k, l) & 3211 & 3212 & 3221 & 3222 \\
\hline
\(d_{ijkl}\) & 24 & 23 & 21 & 23 \\
\(c_{ijkl}\) & 8 & 9 & 8 & 7 \\
\end{array}
\]
The optimal solution is: \( x^* = (x_{ijkl}) \) such that:

\[
\begin{align*}
x_{1121} &= 1, & x_{1231} &= 2, & x_{2112} &= 7, \\
x_{2122} &= 3, & x_{2211} &= 5, & x_{3121} &= 4, \\
x_{3221} &= 6, & x_{1232} &= 12 = d_{1232}.
\end{align*}
\]

Number of iterations = 03. Time of execution = 0.17 sec.

The optimal value is \( z^* = 185 \).

**Example 5** Considering a transportation problem as \((T_C)\) with:

\[
m = n = p = 3, \quad q = 2, \quad \alpha_1 = 9, \quad \alpha_2 = 11, \quad \alpha_3 = 10, \quad \beta_1 = 8, \quad \beta_2 = 8, \quad \beta_3 = 14, \quad \gamma_1 = 7, \quad \gamma_2 = 8, \quad \gamma_3 = 15, \quad \delta_1 = 6, \quad \delta_2 = 24.
\]

The quantities \( c_{ijkl} \) and \( d_{ijkl} \) are given in the following table:

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>1111</th>
<th>1112</th>
<th>1121</th>
<th>1122</th>
<th>1131</th>
<th>1132</th>
<th>1211</th>
<th>1212</th>
<th>1221</th>
<th>1222</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>14</td>
<td>16</td>
<td>18</td>
<td>22</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>1231</th>
<th>1232</th>
<th>1311</th>
<th>1312</th>
<th>1321</th>
<th>1322</th>
<th>1331</th>
<th>1332</th>
<th>2111</th>
<th>2112</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>15</td>
<td>10</td>
<td>11</td>
<td>19</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>30</td>
<td>33</td>
<td>32</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>11</td>
<td>23</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>2121</th>
<th>2122</th>
<th>2131</th>
<th>2132</th>
<th>2211</th>
<th>2212</th>
<th>2221</th>
<th>2222</th>
<th>2231</th>
<th>2232</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>31</td>
<td>29</td>
<td>28</td>
<td>27</td>
<td>11</td>
<td>25</td>
<td>26</td>
<td>24</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>15</td>
<td>20</td>
<td>19</td>
<td>21</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>15</td>
<td>14</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>2311</th>
<th>2312</th>
<th>2321</th>
<th>2322</th>
<th>2331</th>
<th>2332</th>
<th>3111</th>
<th>3112</th>
<th>3121</th>
<th>3122</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>22</td>
<td>23</td>
<td>10</td>
<td>9</td>
<td>22</td>
<td>27</td>
<td>18</td>
<td>16</td>
<td>15</td>
<td>33</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>6</td>
<td>18</td>
<td>19</td>
<td>17</td>
<td>14</td>
<td>16</td>
<td>12</td>
<td>10</td>
<td>6</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>3131</th>
<th>3132</th>
<th>3211</th>
<th>3212</th>
<th>3221</th>
<th>3222</th>
<th>3231</th>
<th>3232</th>
<th>3311</th>
<th>3312</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>29</td>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>32</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>15</td>
<td>14</td>
<td>12</td>
<td>11</td>
<td>19</td>
<td>18</td>
<td>17</td>
<td>11</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>((i,j,k,l))</th>
<th>3321</th>
<th>3322</th>
<th>3331</th>
<th>3332</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_{ijkl})</td>
<td>31</td>
<td>31</td>
<td>30</td>
<td>27</td>
</tr>
<tr>
<td>(c_{ijkl})</td>
<td>11</td>
<td>12</td>
<td>20</td>
<td>14</td>
</tr>
</tbody>
</table>

The optimal solution is: \( x^* = (x_{ijkl}) \) such that:

\[
\begin{align*}
x_{1221} &= 1, & x_{1332} &= 8, & x_{2232} &= 5, \\
x_{2311} &= 1, & x_{3122} &= 8, & x_{3232} &= 2.
\end{align*}
\]
Number of iterations = 07. Time of execution = 0,11 sec. The optimal value is $z^* = 221$.

**Recapitulation:**
Take $M = m + n + p + q$ and $N = mnpq$.

<table>
<thead>
<tr>
<th>Example</th>
<th>Dimension of the Prob. $M \times N$</th>
<th>Number of iterations</th>
<th>Time of execution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8x16</td>
<td>7</td>
<td>0,22 sec</td>
</tr>
<tr>
<td>2</td>
<td>8x16</td>
<td>3</td>
<td>0,16 sec</td>
</tr>
<tr>
<td>3</td>
<td>9x24</td>
<td>1</td>
<td>0,11 sec</td>
</tr>
<tr>
<td>4</td>
<td>10x36</td>
<td>3</td>
<td>0,17 sec</td>
</tr>
<tr>
<td>5</td>
<td>11x54</td>
<td>7</td>
<td>0,11 sec</td>
</tr>
</tbody>
</table>

5 Conclusion

- Through the above numerical tests, we observe that the algorithm described in section requires few iterations and time to reach the optimal solution. Moreover, the solutions obtained show that the algorithm is stable and robust.

- Obviously, it is possible to think of linear programming methods (the simplex or the Karmarkar method, ...) to solve numerically the problem. Indeed it can be formulated as a linear programming problem but, such formulations induce a huge size and thereby a very bad computational behavior.

- Note that we have not treated the degenerateness phenomenon in this paper, but we have could avoided it through our tests by using a procedure for completing the base associated to the degenerate solution.

References


Received: April 21, 2007