Heuristic for Approximating an Inconsistent Pairwise Comparison Matrix

Marcin Anholcer

Al.Niepodległości 10, 61-875 Poznań

Abstract

In several multiobjective decision problems Pairwise Comparison Matrices (PCM) are applied to evaluate the decision variants. The problem that arises very often is inconsistency of given PCM. In such a situation it is important to approximate the PCM with a consistent one. The most common way is to minimize the Euclidean distance between the matrices. In the paper we consider minimization of the maximum distance.

Keywords: Heuristics, nonlinear programming, decision making, pairwise comparison

2010 MSC: 90C59, 90B50, 90C30, 91B08

1. Introduction

One of the popular tools of multiobjective decision making is the Analytic Hierarchy Process, introduced by Saaty (see e.g. Saaty (1980), Erkut and Tarimcilar (1991)) and studied by numerous authors. During the process, the Decision Maker provides pairwise comparisons between \( n \) given decision
variants. Usually the comparisons are defined by the pairwise comparison matrix $A = [a_{ij}]$, where the number $a_{ij}$ says how many times the variant $i$ is preferred to the variant $j$. The values of $a_{ij}$, $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n$ should satisfy the following conditions:

$$a_{ji} = \frac{1}{a_{ij}} \text{ for } i, j = 1, 2, \ldots, n, \quad (1)$$

$$a_{ij}a_{jk} = a_{ik} \text{ for } i, j, k = 1, 2, \ldots, n. \quad (2)$$

If the above conditions are satisfied, the pairwise comparison matrix $A$ is called consistent. The condition (1) is rather easy to fulfil in the practical use of the method (the decision maker may e.g. fill only the elements of $A$ above the diagonal and then the remaining ones are easily calculated). The condition (2) is much more difficult to satisfy and is the main source of the inconsistency.

It is easy to prove that the matrix $A$ is consistent if and only if there exist a vector $w = (w_1, w_2, \ldots, w_n)$ with positive entries such that

$$a_{ij} = \frac{w_i}{w_j}, \text{ for } i, j = 1, 2, \ldots, n. \quad (3)$$

The elements of $w$ are interpreted as the explicit values representing the priorities of the decision variants. Finding their values is then essential. Note that if some vector $w$ defines the matrix $A$ then also the vector $\lambda w$ does for every $\lambda > 0$. 
2. Problem formulation

As in the real life problems the matrix $A$ very often is not consistent, finding the vector $w$ is impossible (it does not exist). In such a situation the goal is to find the vector $v$ that defines the matrix $B$ which is as close as possible to the original pairwise comparison matrix $A$. The distance between matrices $A$ and $B$ may be calculated in various ways. One of the methods is to calculate the Saaty’s inconsistency index using the eigenvalues of the (relative) estimation error matrix, which can be approximated by the row-wise geometric means (see e.g. Saaty (1980), Mogi and Shinohara (2009)). Estimation errors are calculated as the quotients or differences of the respective elements of $A$ and $B$.

Another approach is to calculate some kind of average of the errors. The most popular measure is the square mean calculated according to the formula

$$G_2(A, v) = \left[ \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} (a_{ij} - \frac{v_i}{v_j})^2 \right]^{1/2} . \tag{4}$$

This method of the inconsistency measurement (called least square method, LSM) was introduced by Koczkojaj (1993) and used e.g. in Anholcer et al. (2010), Bozóki (2008), Fülöp et al. (2010), Fülöp (2008), Bozóki and Rapcsák (2008), Mogi and Shinohara (2009).

In two latter papers also other inconsistency measures were considered. Mogi and Shinohara analyzed the general mean which can be defined as

$$G_p(A, v) = \left[ \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \left| a_{ij} - \frac{v_i}{v_j} \right|^p \right]^{1/p} . \tag{5}$$
If $p = 2$, we obtain this way LSM. Other special cases, also considered
by Mogi and Shinohara are $p = -\infty$ (minimum), $p = -1$ (harmonic mean),
$p = 0$ (geometric mean), $p = 1$ (arithmetic mean) and $p = \infty$ (maximum).
In the remainder of this paper we will be interested in the last measure. To
be more precise, we want to solve the following problem:

$$\min G_{\infty}(A, v) = \max_{1 \leq i, j \leq n} \left| a_{ij} - \frac{v_i}{v_j} \right|,$$

$$v_1 = 1,$$

$$v_j > 0, \ j = 1, 2, \ldots, n.$$  \hspace{1cm} (6)-(8)

The condition $v_1 = 1$ has been introduced in order to normalize the vector
$v$ (if some vector $v$ is the solution to the above problem, then also every vector
$\lambda v$ is for every $\lambda > 0$). Of course other normalizing conditions may be used
(compare e.g. Anholcer et al. (2010), Bozóki (2008), Fülöp (2008)).

The problem (6)-(8) is a difficult optimization problem, as the objective
function is neither convex nor concave and thus no local search algorithm
may be applied in order to find the global optimum.

The LSM problem (where $G_2$ was used instead of $G_{\infty}$) was studied e.g.
in Anholcer et al. (2010) (heuristic approach), Bozóki (2008) (systems of
nonlinear equations) and Fülöp (2008) (branch and bound algorithm). The
statistical and axiomatic approach leading to the geometric means solution
were used by Hovanov et al. (2008), while the simulation experiments com-
paring different inconsistency measures have been performed by Mogi and
Shinohara (2009). Our goal is to provide an effective method to derive
optimal weights using the function $G_{\infty}$ as the inconsistency measure.
3. New algorithm

The problem (6)-(8) may be reformulated as follows. Let us introduce additional variable \( z = G_\infty(A, v) \). Then we can rewrite the problem as

\[
\min z \quad \text{(9)}
\]

\[
|a_{ij} - \frac{v_i}{v_j}| \leq z, \quad i, j = 1, 2, \ldots, n \quad \text{(10)}
\]

\[
v_1 = 1, \quad \text{(11)}
\]

\[
v_j > 0, \quad j = 1, 2, \ldots, n. \quad \text{(12)}
\]

Note that the problems are not equal (the sets of feasible solutions are distinct), but they are equivalent (the optimal solutions to both problems are the same and they always exist). The problem (9)-(12) is difficult to solve as most constraints are neither convex nor concave nonlinear functions. Moreover, the set of feasible solutions is not closed and thus not compact (although the optimum exists). In order to find its approximate solution we are going to treat \( z \) as a parameter.

If we assume that the value of \( z \) is given, the problem (9)-(12) reduces to the following linear system:

\[
(a_{ij} - z)v_j \leq v_i \leq (a_{ij} + z)v_j, \quad i, j = 1, 2, \ldots, n \quad \text{(13)}
\]

\[
v_1 = 1, \quad \text{(14)}
\]

\[
v_j \geq 0, \quad j = 1, 2, \ldots, n. \quad \text{(15)}
\]

Note that the constraint (12) may be replaced with (15) as none of \( v_j \) may
be equal to 0, otherwise all of them would be according to the constraints (13). That would in turn contradict with the constraint (14).

The number of inequalities (13) may be reduced. Firstly, for every $i$, they are always satisfied as $a_{ii} = 1$ and $z \geq 0$.

The following operation in turn remove half of the remaining inequalities. Let us consider the inequalities in which the only variables are $v_i$ and $v_j$ for some $i \neq j$. They can be rewritten in the following form:

\begin{align*}
- v_i + (a_{ij} - z)v_j &\leq 0, \quad (16) \\
- v_i + \frac{v_j}{a_{ji} + z} &\leq 0, \quad (17) \\
\frac{v_i}{a_{ij} + z} - v_j &\leq 0, \quad (18) \\
(a_{ji} - z)v_i - v_j &\leq 0. \quad (19)
\end{align*}

One of the inequalities (16) and (17) implies the other so can be removed. More precisely, we choose the inequality

\begin{equation}
- v_i + \left( \max \left\{ a_{ij} - z, \frac{1}{a_{ji} + z} \right\} \right) v_j \leq 0. \quad (20)
\end{equation}

Analogously, we can eliminate one of the inequalities (18) and (19), by choosing the following

\begin{equation}
\left( \max \left\{ a_{ji} - z, \frac{1}{a_{ij} + z} \right\} \right) v_i - v_j \leq 0. \quad (21)
\end{equation}

Note that in both cases the chosen maxima have positive values. In order to solve the resulting linear system we formulate the following auxiliary linear programming problem.
We solve the above problem using the adapted version of the simplex method. The initial base variables are

\[
\begin{align*}
z_0 &= 1, \\
z_{ij}^1, z_{ij}^2 &= 0, \quad 1 \leq i < j \leq n.
\end{align*}
\]

The optimality criteria are equal to the coefficients in the constraint (25).

Also, we use additional stopping criterion: \( z_0 = 0 \). If this criterion is used, the initial inequalities system has feasible solution where the values of \( v_j \) are equal to those in the last solution of the problem (22)-(28). On the other hand, if the standard optimality condition is in use it means that the artificial variable \( z_0 = 1 \) remains in the base and the initial linear system is inconsistent. The entering variable is chosen as in the standard version of the algorithm, while in order to choose the leaving variable we do not have to calculate the quotients, as almost all the right sides are equal to 0. Thus the adaptation of the simplex method may be rewritten as follows.
Algorithm 1 Simplified simplex method

Step 1: The first base is formed by the variables $z_{ij}^1$, $z_{ij}^2 = 0$ and $z_0 = 1$. Go to step 2.

Step 2: Choose the maximal element of the row $r^*$ corresponding to the constraint (25). If it is nonpositive then STOP, the linear system (13)-(15) is inconsistent. Otherwise go to step 3.

Step 3: Chosen maximum indicates the pivot column. If it contains any positive element outside the row $r^*$, then choose the maximum over all such elements - it indicates the pivot row. Transform the constraint matrix and go to step 2. If the only positive element of pivot column is placed in row $r^*$, $z_0$ is the leaving variable. Transform the constraint matrix and STOP - new values of $v_j$ form the feasible solution of linear system (13)-(15).
Note that if the feasible solution exists for some \( z = z^* \), then it is also the solution for every \( z \geq z^* \). This means also that if the system (13)-(15) is inconsistent for some value of \( z = z^* \), then it is also inconsistent for every \( z \leq z^* \). This leads us to the following algorithm, where the starting point is generated by the geometric means of rows of \( A \).

**Algorithm 2 Main algorithm**

**Step 1:** Assume the accuracy level \( \varepsilon > 0 \). Let \( v_1^* = \left( \prod_{j=1}^{n} a_{ij} \right)^{1/n} \) and \( v_i = v_i^*/v_1^* \) for \( i = 1, 2, \ldots, n \). Let \( z = z_{\text{max}} = G_{\infty}(A, v) \) and \( z_{\text{min}} = 0 \). Proceed to step 2.

**Step 2:** If \( z - z_{\text{min}} < \varepsilon \) then STOP. Vector \( v \) is the desired approximation of the weight vector \( w \). Otherwise go to step 3.

**Step 3:** Set \( z := (z_{\text{max}} - z_{\text{min}})/2 \). Apply algorithm 1 to solve the problem (22)-(28). If \( z_0 = 0 \), save the new value of \( v \) and set \( z_{\text{max}} := z \). Otherwise do not change the value of \( v \) and set \( z_{\text{min}} := z \), then \( z := z_{\text{max}} \). Go back to step 2.

In every step of algorithm 2 the value of the difference \( z_{\text{max}} - z_{\text{min}} \) decreases twice, so in the finite number of iterations we obtain the approximation of the optimal solution. More precisely, if \( z_{\text{max}}^* \) denotes the initial value of \( z_{\text{max}} \), then the algorithm will stop after \( \log_2 \left[ \frac{z_{\text{max}}^*}{\varepsilon} \right] \) steps.

**4. Computational experiments**

The algorithm has been implemented in Java and and tested for a number of randomly generated problems. The assumed accuracy level was \( \varepsilon = 0.001 \). The application has been tested on the PC with Intel Core2 Duo CPU.
For every value of $n = 3, 4, \ldots, 10$ (10 is rather the maximal size of the comparison matrix in the real life problems) the elements of $A$ were chosen uniformly at random from the interval <1, $a_{max}$>, where $a_{max} \in \{3, 5, 10\}$. In every case 100 problems have been solved (that gives the total number of 2400 test problems). The average running times (in milliseconds) are given in the table 1.

Table 1: Average running times

<table>
<thead>
<tr>
<th>n</th>
<th>$a_{max} = 3$</th>
<th>$a_{max} = 5$</th>
<th>$a_{max} = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.0117</td>
<td>0.0127</td>
<td>0.0158</td>
</tr>
<tr>
<td>4</td>
<td>0.0237</td>
<td>0.0284</td>
<td>0.0297</td>
</tr>
<tr>
<td>5</td>
<td>0.0528</td>
<td>0.055</td>
<td>0.0605</td>
</tr>
<tr>
<td>6</td>
<td>0.0979</td>
<td>0.1240</td>
<td>0.1191</td>
</tr>
<tr>
<td>7</td>
<td>0.1720</td>
<td>0.1998</td>
<td>0.2047</td>
</tr>
<tr>
<td>8</td>
<td>0.2686</td>
<td>0.2934</td>
<td>0.3288</td>
</tr>
<tr>
<td>9</td>
<td>0.4215</td>
<td>0.4546</td>
<td>0.5110</td>
</tr>
<tr>
<td>10</td>
<td>0.6271</td>
<td>0.6607</td>
<td>0.7507</td>
</tr>
</tbody>
</table>

As we can see, in all the cases the running times are less than one millisecond, which is acceptable in real life applications.

5. Conclusion

The presented algorithm guarantees obtaining the solution for which the objective value is arbitrarily close to the optimal one. Of course it does not mean that the vector $v$ is arbitrarily close to its optimal value (distinct local
optima may be far from each other even if the objective values are very close).
However it is more that gives e.g. the heuristic for LSM given by Anholcer et al. (2010), which does not guarantee obtaining the objective value close to the optimal one. On the other hand the algorithm is fast and so very useful for finding the best consistent approximate of an inconsistent pairwise comparison matrix.

As far as the author knows given method is the first one for the inconsistency measured using the maximum distance $G_\infty$. Further research should focus on looking for the exact method of solving this problem and any methods for other measures (e.g. $G_p$ distance for arbitrary p, including Manhattan distance $G_1$).

References


Fülöp J, Koczkodaj W.W., Szarek S.J. (2010), A Different Perspective on a Scale for Pairwise Comparisons, Transactions on CCI I, LNCS 6220, pp. 71-84.


