
Serdar Enginoğlu*, Samet Memiş*, Tutku Öngel*

*Department of Mathematics, Faculty of Arts and Sciences, Çanakkale Onsekiz Mart University, Çanakkale, Turkey
*Corresponding author: serdarenginoglu@gmail.com

ABSTRACT: The uni-int decision-making method constructed by and-product/or-product was defined and applied to a decision-making problem by Çağman and Enginoğlu [2010a. Soft Set Theory and Uni-Int Decision Making, European Journal of Operational Research. 207: 848-855]. The method has a potential for applications in several areas such as machine learning and image processing. Recently, this method has been configured by Enginoğlu and Memiş [2018. A Configuration of Some Soft Decision-Making Algorithms via fpfs-Matrices.Cumhuriyet Science Journal. 39(4): In Press] via fuzzy parameterized fuzzy soft matrices (fpfs-matrices), faithfully to the original, because a more general form is needed for the method in the event that the parameters or objects have uncertainties. However, in the case that a large amount of data is processed, the method has a disadvantage regarding time and complexity. To deal with this problem and to be able to use this configured method denoted by CE10 effectively, we suggest two algorithms in this paper, i.e. EMO18a and EMO18o, and prove that CE10 constructed by and-product (CE10a) and constructed by or-product (CE10o) are special cases of EMO18a and EMO18o, respectively, if first rows of the fpfs-matrices are binary. We then compare the running times of these algorithms. The results show that EMO18a and EMO18o outperform CE10a and CE10o, respectively. Particularly in problems containing a large amount of parameters, EMO18o offer up to 99.9966% and 99.9965% of time advantage, respectively. Afterwards, we apply EMO18o to a performance-based value assignment to the methods used in the noise removal, so that we can order them in terms of performance. Finally, we discuss the need for further research.

Keywords – Fuzzy sets, Soft sets, Soft decision-making, Soft matrices, fpfs-matrices

1. Introduction

The classical sets are inadequate to deal with some problems containing uncertainties. To that end, fuzzy set theory was put forward by Zadeh (1965). Similar to the fuzzy sets, the concept of soft sets too has been produced by Molodtsov (1999) due to difficulties in construction of fuzzy sets. In this respect, the soft set theory is a very useful mathematical tool to model some problems containing uncertainties and so far many theoretical and applied studies from algebra to decision-making problems (Atmaca, 2017; Atmaca and Zorlutuna, 2014; Bera et al. 2017; Çağman et al. 2010, 2011; Çağman and Enginoğlu, 2010a; Çağman and Enginoğlu, 2012; Çağman and Enginoğlu, 2010b; Çağman et al. 2011; Çıtak and Çağman, 2017; Çıtak and Çagman, 2015; Enginoğlu, 2012; Enginoğlu et al. 2015; Karaaslan, 2016; Maji et al. 2001, 2002, 2003; Muştuoğlu et al. 2016; Sezgin, 2016; Sezgin et al. 2019; Tunçay and Sezgin, 2016; Ullah et al. 2018; Zorlutuna and Atmaca, 2016) have been conducted on this concept.

Recently, some decision-making algorithms constructed by soft sets (Çağman and Enginoğlu, 2010a; Eraslan, 2015; Maji et al. 2002; Razak and Mohamad, 2011), fuzzy soft
sets (Çağman et al. 2011; Das and Borgohain, 2012; Eraslan and Karaaslan, 2015; Maji et al. 2001; Razak and Mohamad, 2013), fuzzy parameterized soft sets (Çağman et al. 2011; Çağman and Deli, 2012), fuzzy parameterized fuzzy soft sets (fpfs-sets) (Çağman et al. 2010; Zhu and Zhan, 2016), soft matrices (Çağman and Enginoğlu, 2010b; Vijayabalaji and Ramesh, 2013) and fuzzy soft matrices (Çağman and Enginoğlu, 2012; Khan et al. 2013) have been configured (Enginoğlu and Memiş, 2018) via fuzzy parameterized fuzzy soft matrices (fpfs-matrices) (Enginoğlu, 2012). One of the configured methods is CE10 (Çağman and Enginoğlu, 2010a; Enginoğlu and Memiş, 2018) constructed by and-product (CE10a) or constructed by or-product (CE10o). In the case that a large amount of data is processed, these two methods still have a disadvantage regarding time and complexity. To deal with this problem, it is worthwhile to study the simplification of the algorithms. In the event that first rows of the fpfs-matrices are binary, although there exist simplified versions of CE10a and CE10o, they no have in the other cases. Therefore, in this study, we aim to develop two algorithms which have the ability of CE10a and CE10o and are also faster than them.

In Section 2 of the present study, we introduce the concept of fpfs-matrices and present the soft decision-making method CE10. In Section 3, we propose two fast and simple algorithms, denoted by EMO18a and EMO18o, which accept CE10a and CE10o as special cases, respectively, provided that first rows of the fpfs-matrices are binary. A part of this section has been presented in (Enginoğlu et al. 2018). In Section 4, we compare the running times of these algorithms. In Section 5, we apply EMO18o to the decision-making problem in image denoising. Finally, we discuss the need for further research.

2. Preliminaries

In this section, firstly, we present the definition of fpfs-sets and fpfs-matrices. Throughout this paper, let $E$ be a parameter set, $F(E)$ be the set of all fuzzy sets over $E$, and $\mu \in F(E)$. Here, $\mu := \{\mu(x); x \in E\}$.

**Definition 2.1.** (Çağman et al. 2010; Enginoğlu, 2012) Let $U$ be a universal set, $\mu \in F(E)$, and $\alpha$ be a function from $\mu$ to $F(U)$. Then the graphic of $\alpha$, denoted by $\alpha$, defined by

$$
\alpha := \{\mu(x), \alpha(\mu(x)); x \in E\}
$$

that is called fuzzy parameterized fuzzy soft set (fpfs-set) parameterized via $E$ over $U$ (or briefly over $U$).

In the present paper, the set of all fpfs-sets over $U$ is denoted by $FPFS_E(U)$.

**Example 2.1.** Let $E = \{x_1, x_2, x_3, x_4\}$ and $U = \{u_1, u_2, u_3, u_4, u_5\}$. Then,

$$
\alpha = \{(x_1, [0,5,0,6,0,6,u_3]), (0,8,x_2, [0,9,u_2,0,2,u_3,0,1,u_5]), (0,6,x_3, [0,5,u_2,0,7,u_4,0,2,u_5]), (1,x_4, [1,u_3,0,9,u_4])\}
$$

is a fpfs-set over $U$.

**Definition 2.2.** (Enginoğlu, 2012) Let $\alpha \in FPFS_E(U)$. Then, $[a_{ij}]$ is called the matrix representation of $\alpha$ (or briefly fpfs-matrix of $\alpha$) and defined by
\[ [a_{ij}] = \begin{bmatrix}
  a_{01} & a_{02} & a_{03} & \cdots & a_{0n} & \cdots \\
  a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & \cdots \\
  \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
  a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} & \cdots \\
  \vdots & \vdots & \vdots & \ddots & \vdots & \vdots 
\end{bmatrix} \text{ for } i = \{0,1,2,\cdots\} \text{ and } j = \{1,2,\cdots\} \]

such that
\[ a_{ij} := \begin{cases}
  \mu(x_j), & i = 0 \\
  \alpha(\mu(x_j))(u_i), & i \neq 0
\end{cases} \]

Here, if \(|U| = m - 1\) and \(|E| = n\), then \([a_{ij}]\) has order \(m \times n\).

From now on, the set of all \(fpfs\)-matrices parameterized via \(E\) over \(U\) is denoted by \(FPFS_{E}[U]\).

**Example 2.2.** Let’s consider the \(fpfs\)-set \(\alpha\) provided in Example 2.1. Then, the \(fpfs\)-matrix of \(\alpha\) is as follows:

\[ [a_{ij}] = \begin{bmatrix}
  0 & 0.8 & 0.6 & 1 \\
  0.5 & 0 & 0 & 0 \\
  0 & 0.9 & 0.5 & 0 \\
  0.6 & 0.2 & 0 & 1 \\
  0 & 0 & 0.7 & 0.9 \\
  0 & 0.1 & 0.2 & 0
\end{bmatrix} \]

**Definition 2.3.** (Enginoğlu, 2012) Let \([a_{ij}],[b_{ik}] \in FPFS_{E}[U]\) and \([c_{ip}] \in FPFS_{E^2}[U]\) such that \(p = n(j - 1) + k\). For all \(i\) and \(p\),

If \(c_{ip} = \min\{a_{ij}, b_{ik}\}\), then \([c_{ip}]\) is called and-product of \([a_{ij}]\) and \([b_{ik}]\) and is denoted by \([a_{ij}] \land [b_{ik}]\).

If \(c_{ip} = \max\{a_{ij}, b_{ik}\}\), then \([c_{ip}]\) is called or-product of \([a_{ij}]\) and \([b_{ik}]\) and is denoted by \([a_{ij}] \lor [b_{ik}]\).

If \(c_{ip} = \min\{a_{ij}, 1 - b_{ik}\}\), then \([c_{ip}]\) is called andnot-product of \([a_{ij}]\) and \([b_{ik}]\) and is denoted by \([a_{ij}] \lnot [b_{ik}]\).

If \(c_{ip} = \max\{a_{ij}, 1 - b_{ik}\}\), then \([c_{ip}]\) is called ornot-product of \([a_{ij}]\) and \([b_{ik}]\) and is denoted by \([a_{ij}] \lor [b_{ik}]\).

Secondly, we present the algorithm CE10 constructed by and-product/or-product (Çağman and Enginoğlu, 2010a; Enginoğlu and Memiş, 2018).
CE10’s Algorithm Steps

Step 1. Construct two \( fpfs \)-matrices \([a_{ij}]\) and \([b_{lk}]\)

Step 2. Find and-product/or-product \( fpfs \)-matrix \([c_{ip}]\) of \([a_{ij}]\) and \([b_{lk}]\)

Step 3. Obtain \([s_{i1}]\) denoted by max-min \((c_{ip})\) defined by

\[
s_{i1} := \max\{\max_j\min_k(c_{ip}), \max_k\min_j(c_{ip})\}
\]

such that \(i \in \{1,2,\ldots,m - 1\}\), \(I_a := \{j \mid a_{0j} \neq 0\}\), \(I_b := \{k \mid b_{0k} \neq 0\}\), \(p = n(j - 1) + k\), and

\[
\max_j\min_k(c_{ip}) := \begin{cases} \max_{j \in I_a}\left\{\min_{k \in I_b}c_{ip}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}
\]

\[
\max_k\min_j(c_{ip}) := \begin{cases} \max_{k \in I_b}\left\{\min_{j \in I_a}c_{ip}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}
\]

Step 4. Obtain the set \(\{u_k \in U \mid s_{k1} = \max_i s_{i1}\}\)

Preferably, the set \(\{s_{i1}u_i \mid u_i \in U\}\) or \(\left\{\frac{s_{k1}}{\max_i s_{i1}}u_k \mid u_k \in U\right\}\) can be attained.

3. The Soft Decision-Making Methods: EMO18a and EMO18o

In this section, firstly, we propose a fast and simple algorithm denoted by EMO18a.

EMO18a’s Algorithm Steps

Step 1. Construct two \( fpfs \)-matrices \([a_{ij}]\) and \([b_{lk}]\)

Step 2. Obtain \([s_{i1}]\) denoted by max-min \((a_{ij}, b_{lk})\) defined by

\[
s_{i1} := \max\{\max_j\min_k(a_{ij}, b_{lk}), \max_k\min_j(a_{ij}, b_{lk})\}
\]

such that \(i \in \{1,2,\ldots,m - 1\}\), \(I_a := \{j \mid a_{0j} \neq 0\}\), \(I_b := \{k \mid b_{0k} \neq 0\}\), and

\[
\max_j\min_k(a_{ij}, b_{lk}) := \begin{cases} \max_{j \in I_a}\left\{\min_{k \in I_b}a_{ij}\right\}, \min_{k \in I_b}b_{lk}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}
\]

\[
\max_k\min_j(a_{ij}, b_{lk}) := \begin{cases} \max_{k \in I_b}\left\{\min_{j \in I_a}b_{lk}\right\}, \min_{j \in I_a}a_{ij}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}
\]

Step 3. Obtain the set \(\{u_k \in U \mid s_{k1} = \max_i s_{i1}\}\)

Preferably, the set \(\{s_{i1}u_i \mid u_i \in U\}\) or \(\left\{\frac{s_{k1}}{\max_i s_{i1}}u_k \mid u_k \in U\right\}\) can be attained.

Secondly, we present a fast and simple algorithm denoted by EMO18o given in (Enginoğlu et al. 2018).
EMO18a’s Algorithm Steps

**Step 1.** Construct two fpf s-matrices \([a_{ij}]\) and \([b_{ik}]\)

**Step 2.** Obtain \([s_{i1}]\) denoted by \(\max\min(a_{ij}, b_{ik})\) defined by

\[
s_{i1} := \max\{\max_j \min_k (a_{ij}, b_{ik}), \max_k \min_j (a_{ij}, b_{ik})\}
\]

such that \(i \in \{1, 2, \ldots, m - 1\}\), \(I_a := \{j \mid a_{oj} \neq 0\}\), \(I_b := \{k \mid b_{ok} \neq 0\}\), and

\[
\max_j \min_k (a_{ij}, b_{ik}) := \begin{cases} 
\max \left\{ \max_{j \in I_a} \{a_{0j} a_{ij}\}, \min_{k \in I_b} b_{0k} b_{ik} \right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\
0, & \text{otherwise}
\end{cases}
\]

\[
\max_k \min_j (a_{ij}, b_{ik}) := \begin{cases} 
\max \left\{ \max_{k \in I_b} \{b_{0k} b_{ik}\}, \min_{j \in I_a} a_{0j} a_{ij} \right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\
0, & \text{otherwise}
\end{cases}
\]

**Step 3.** Obtain the set \(\{u_k \in U \mid s_{k1} = \max_i s_{i1}\}\)

Preferably, the set \(\left\{\frac{s_{ki} u_i}{\max s_{i1}} u_k \mid u_k \in U\right\}\) can be attained.

**Theorem 3.1.** CE10a is a special case of EMO18a provided that first rows of the fpf s-matrices are binary.

**Proof.** Suppose that first rows of the fpf s-matrices are binary. The functions \(s_{i1}\) provided in CE10a and EMO18a are equal in the event that \(I_a = \emptyset\) or \(I_b = \emptyset\). Assume that \(I_a \neq \emptyset\) and \(I_b \neq \emptyset\). Since \(a_{0j} = 1\) and \(b_{0k} = 1\), for all \(j \in I_a := \{a_1, a_2, \ldots, a_s\}\) and \(k \in I_b := \{b_1, b_2, \ldots, b_t\}\),

\[
\max_k \min_j (c_{ip}) = \max_{j \in I_a} \min_{k \in I_b} c_{ij}
\]

\[
= \max_{j \in I_a} \min_{k \in I_b} \left\{ \min_j \left\{ \min_{a_{0j}} a_{ij}, \min_{b_{0k}} b_{ij}, \min_{a_{ij}} b_{ik} \right\} \right\}
\]

\[
= \max_{j \in I_a} \min_{k \in I_b} \left\{ \min_a a_{ij}, \min_{b_{0k}} b_{ik} \right\}
\]

\[
= \max \left\{ \min_{a_{ij}} a_{ij}, \min_{b_{0k}} b_{ik} \right\}
\]

\[
= \min \left\{ \max \min (a_{ia_1}, b_{ib_1}), \min (a_{ia_1}, b_{ib_2}), \ldots, \min (a_{ia_s}, b_{ibt}) \right\}
\]

\[
= \min \left\{ \max \min (a_{ia_1}, b_{ib_1}), \min (a_{ia_1}, b_{ib_2}), \ldots, \min (a_{ia_s}, b_{ibt}) \right\}
\]

\[
= \min \left\{ \max \min (a_{ia_1}, a_{ia_2}, \ldots, a_{ia_s}), \min (b_{ib_1}, b_{ib_2}, \ldots, b_{ibt}) \right\}
\]

\[
= \min \left\{ \max \min (a_{ij}), \min (b_{ik}) \right\}
\]
= \min_{j \in I_a} \{\max_{i \in I_b} \{a_{0j} a_{ij}\}, \min_{i \in I_a} \{b_{0k} b_{ik}\}\}
= \max_j \min_k (a_{ij}, b_{ik})

In a similar way, \(\max_k \min_j (c_{ip}) = \max_k \min_j (a_{ij}, b_{ik})\). Consequently,
\[
\max - \min (a_{ij}, b_{ik}) = \max - \min (c_{ip})
\]

\[\square\]

**Theorem 3.2.** (Enginoğlu et al. 2018.) CE10o is a special case of EMO18o provided that first rows of the ffp's-matrices are binary.

4. Simulation Results

In this section, we compare the running times of CE10a-EMO18a and CE10o-EMO18o by using MATLAB R2017b and a workstation with I(R) Xeon(R) CPU E5-1620 v4 @ 3.5 GHz and 64 GB RAM in this study.

We, firstly, present the running times of CE10a and EMO18a in Table 1 and Fig. 1 for 10 objects and the parameters ranging from 10 to 100. We then give their running times in Table 2 and Fig. 2 for 10 objects and the parameters ranging from 1000 to 10000, in Table 3 and Fig. 3 for 10 parameters and the objects ranging from 10 to 100, in Table 4 and Fig. 4 for 10 parameters and the objects ranging from 1000 to 10000, in Table 5 and Fig. 5 for the parameters and the objects ranging from 10 to 100, and in Table 6 and Fig. 6 for the parameters and the objects ranging from 100 to 1000. The results show that EMO18a outperforms CE10a in any number of data under the specified condition.

**Table 1.** The results for 10 objects and the parameters ranging from 10 to 100

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CE10a</strong></td>
<td>0.02150</td>
<td>0.00932</td>
<td>0.00279</td>
<td>0.00366</td>
<td>0.00922</td>
<td>0.01224</td>
<td>0.00956</td>
<td>0.00947</td>
<td>0.01622</td>
<td>0.02317</td>
</tr>
<tr>
<td><strong>EMO18a</strong></td>
<td>0.00913</td>
<td>0.00231</td>
<td>0.00085</td>
<td>0.00043</td>
<td>0.00209</td>
<td>0.00120</td>
<td>0.00033</td>
<td>0.00023</td>
<td>0.00020</td>
<td>0.00035</td>
</tr>
<tr>
<td><strong>Difference</strong></td>
<td>0.0124</td>
<td>0.0070</td>
<td>0.0019</td>
<td>0.0032</td>
<td>0.0071</td>
<td>0.0110</td>
<td>0.0092</td>
<td>0.0092</td>
<td>0.0160</td>
<td>0.0228</td>
</tr>
<tr>
<td><strong>Advantage (%)</strong></td>
<td>57.5357</td>
<td>75.1660</td>
<td>69.4789</td>
<td>88.1353</td>
<td>77.3262</td>
<td>90.2223</td>
<td>96.5866</td>
<td>97.5746</td>
<td>98.7938</td>
<td>98.4727</td>
</tr>
</tbody>
</table>

**Figure 1.** The figure for Table 1
**Table 2.** The results for 10 objects and the parameters ranging from 1000 to 10000

<table>
<thead>
<tr>
<th></th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
<th>6000</th>
<th>7000</th>
<th>8000</th>
<th>9000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMO18a</td>
<td>0.0072</td>
<td>0.0032</td>
<td>0.0017</td>
<td>0.0021</td>
<td>0.0041</td>
<td>0.0037</td>
<td>0.0030</td>
<td>0.0032</td>
<td>0.0036</td>
<td>0.0040</td>
</tr>
</tbody>
</table>

**Figure 2.** The figure for Table 2

**Table 3.** The results for 10 parameters and the objects ranging from 10 to 100

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10a</td>
<td>0.0174</td>
<td>0.0069</td>
<td>0.0019</td>
<td>0.0020</td>
<td>0.0057</td>
<td>0.0078</td>
<td>0.0023</td>
<td>0.0033</td>
<td>0.0057</td>
<td>0.0039</td>
</tr>
<tr>
<td>EMO18a</td>
<td>0.0063</td>
<td>0.0024</td>
<td>0.0006</td>
<td>0.0007</td>
<td>0.0024</td>
<td>0.0016</td>
<td>0.0007</td>
<td>0.0008</td>
<td>0.0008</td>
<td>0.0009</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0110</td>
<td>0.0045</td>
<td>0.0013</td>
<td>0.0013</td>
<td>0.0032</td>
<td>0.0016</td>
<td>0.0016</td>
<td>0.0025</td>
<td>0.0050</td>
<td>0.0030</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>63.5402</td>
<td>65.6632</td>
<td>69.1512</td>
<td>65.4336</td>
<td>56.9135</td>
<td>79.9089</td>
<td>68.9759</td>
<td>77.1017</td>
<td>86.2588</td>
<td>77.6234</td>
</tr>
</tbody>
</table>

**Figure 3.** The figure for Table 3

**Table 4.** The results for 10 parameters and the objects ranging from 1000 to 10000

<table>
<thead>
<tr>
<th></th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
<th>6000</th>
<th>7000</th>
<th>8000</th>
<th>9000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10a</td>
<td>0.0596</td>
<td>0.1619</td>
<td>0.2678</td>
<td>0.4543</td>
<td>0.5862</td>
<td>0.8606</td>
<td>1.0839</td>
<td>1.3455</td>
<td>1.7614</td>
<td>2.2319</td>
</tr>
<tr>
<td>EMO18a</td>
<td>0.0148</td>
<td>0.0199</td>
<td>0.0276</td>
<td>0.0384</td>
<td>0.0499</td>
<td>0.0631</td>
<td>0.0752</td>
<td>0.0855</td>
<td>0.0994</td>
<td>0.1134</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0448</td>
<td>0.1420</td>
<td>0.2402</td>
<td>0.4159</td>
<td>0.5363</td>
<td>0.7975</td>
<td>1.0087</td>
<td>1.2600</td>
<td>1.6620</td>
<td>2.1185</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>75.1099</td>
<td>87.6852</td>
<td>89.6930</td>
<td>91.5504</td>
<td>91.4930</td>
<td>92.6671</td>
<td>93.0655</td>
<td>93.6467</td>
<td>94.3553</td>
<td>94.9205</td>
</tr>
</tbody>
</table>
The results for the parameters and the objects ranging from 10 to 100

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10a</td>
<td>0.0175</td>
<td>0.0078</td>
<td>0.0073</td>
<td>0.0141</td>
<td>0.0227</td>
<td>0.0500</td>
<td>0.0683</td>
<td>0.0936</td>
<td>0.1104</td>
<td>0.1536</td>
</tr>
<tr>
<td>EMO18a</td>
<td>0.0061</td>
<td>0.0023</td>
<td>0.0007</td>
<td>0.0007</td>
<td>0.0026</td>
<td>0.0020</td>
<td>0.0008</td>
<td>0.0010</td>
<td>0.0010</td>
<td>0.0011</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0115</td>
<td>0.0055</td>
<td>0.0066</td>
<td>0.0134</td>
<td>0.0201</td>
<td>0.0480</td>
<td>0.0675</td>
<td>0.0926</td>
<td>0.1094</td>
<td>0.1525</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>65.3281</td>
<td>70.1501</td>
<td>90.6707</td>
<td>95.0966</td>
<td>88.6854</td>
<td>96.0033</td>
<td>98.8679</td>
<td>99.9627</td>
<td>99.1043</td>
<td>99.2978</td>
</tr>
</tbody>
</table>

The results for the parameters and the objects ranging from 100 to 1000

<table>
<thead>
<tr>
<th></th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
<th>6000</th>
<th>7000</th>
<th>8000</th>
<th>9000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10a</td>
<td>0.2100</td>
<td>2.1134</td>
<td>8.4209</td>
<td>23.7731</td>
<td>56.3553</td>
<td>105.5166</td>
<td>188.7614</td>
<td>297.5683</td>
<td>485.1053</td>
<td>724.1639</td>
</tr>
<tr>
<td>EMO18a</td>
<td>0.0089</td>
<td>0.0048</td>
<td>0.0050</td>
<td>0.0075</td>
<td>0.0125</td>
<td>0.0155</td>
<td>0.0179</td>
<td>0.0225</td>
<td>0.0275</td>
<td>0.0331</td>
</tr>
<tr>
<td>Difference</td>
<td>0.2011</td>
<td>2.1086</td>
<td>8.4159</td>
<td>23.7656</td>
<td>56.3428</td>
<td>105.5011</td>
<td>188.7435</td>
<td>297.5458</td>
<td>485.0777</td>
<td>724.1308</td>
</tr>
</tbody>
</table>

The results for the parameters and the objects ranging from 1000 to 10000

<table>
<thead>
<tr>
<th></th>
<th>10000</th>
<th>20000</th>
<th>30000</th>
<th>40000</th>
<th>50000</th>
<th>60000</th>
<th>70000</th>
<th>80000</th>
<th>90000</th>
<th>100000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10a</td>
<td>3.485</td>
<td>7.276</td>
<td>11.668</td>
<td>20.238</td>
<td>30.511</td>
<td>42.003</td>
<td>54.606</td>
<td>68.209</td>
<td>81.812</td>
<td>95.415</td>
</tr>
<tr>
<td>EMO18a</td>
<td>0.251</td>
<td>0.502</td>
<td>0.753</td>
<td>1.255</td>
<td>1.756</td>
<td>2.257</td>
<td>2.758</td>
<td>3.259</td>
<td>3.760</td>
<td>4.261</td>
</tr>
</tbody>
</table>
Secondly, we present the running times of CE10o and EMO18o in Table 7 and Fig. 7 for 10 objects and the parameters ranging from 10 to 100. We then give their running times in Table 8 and Fig. 8 for 10 objects and the parameters ranging from 1000 to 10000, in Table 9 and Fig. 9 for 10 parameters and the objects ranging from 10 to 100, in Table 10 and Fig. 10 for 10 parameters and the objects ranging from 1000 to 10000, in Table 11 and Fig. 11 for the parameters and the objects ranging from 10 to 100, and in Table 12 and Fig. 12 for the parameters and the objects ranging from 100 to 1000. The results show that EMO18o outperforms CE10o in any number of data under the specified condition.

**Table 7.** The results for 10 objects and the parameters ranging from 10 to 100

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>0.01904</td>
<td>0.00705</td>
<td>0.00297</td>
<td>0.00400</td>
<td>0.00843</td>
<td>0.01377</td>
<td>0.01018</td>
<td>0.01433</td>
<td>0.02359</td>
<td>0.04069</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.00662</td>
<td>0.00213</td>
<td>0.00044</td>
<td>0.00045</td>
<td>0.00215</td>
<td>0.00144</td>
<td>0.00028</td>
<td>0.00021</td>
<td>0.00054</td>
<td>0.00034</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0124</td>
<td>0.0049</td>
<td>0.0025</td>
<td>0.0035</td>
<td>0.0063</td>
<td>0.0123</td>
<td>0.0099</td>
<td>0.0141</td>
<td>0.0231</td>
<td>0.0403</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>65.2227</td>
<td>69.7850</td>
<td>85.0606</td>
<td>88.7429</td>
<td>74.4344</td>
<td>89.5674</td>
<td>97.2488</td>
<td>98.5327</td>
<td>97.7243</td>
<td>99.1575</td>
</tr>
</tbody>
</table>

**Figure 7.** The figure for Table 7

**Table 8.** The results for 10 objects and the parameters ranging from 1000 to 10000

<table>
<thead>
<tr>
<th></th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
<th>6000</th>
<th>7000</th>
<th>8000</th>
<th>9000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>1.4542</td>
<td>5.1004</td>
<td>10.5845</td>
<td>18.3512</td>
<td>28.8547</td>
<td>39.7985</td>
<td>54.9162</td>
<td>73.4039</td>
<td>93.8602</td>
<td>117.4047</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.0075</td>
<td>0.0044</td>
<td>0.0018</td>
<td>0.0020</td>
<td>0.0041</td>
<td>0.0039</td>
<td>0.0031</td>
<td>0.0032</td>
<td>0.0044</td>
<td>0.0041</td>
</tr>
<tr>
<td>Difference</td>
<td>1.4468</td>
<td>5.0960</td>
<td>10.5828</td>
<td>18.3492</td>
<td>28.8505</td>
<td>39.7946</td>
<td>54.9131</td>
<td>73.4007</td>
<td>93.8557</td>
<td>117.4006</td>
</tr>
</tbody>
</table>

**Figure 8.** The figure for Table 8

**Table 9.** The results for 10 parameters and the objects ranging from 10 to 100

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>0.0190</td>
<td>0.0068</td>
<td>0.0019</td>
<td>0.0022</td>
<td>0.0074</td>
<td>0.0086</td>
<td>0.0035</td>
<td>0.0040</td>
<td>0.0038</td>
<td>0.0049</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.0067</td>
<td>0.0025</td>
<td>0.0006</td>
<td>0.0007</td>
<td>0.0020</td>
<td>0.0026</td>
<td>0.0007</td>
<td>0.0010</td>
<td>0.0012</td>
<td>0.0009</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0124</td>
<td>0.0044</td>
<td>0.0012</td>
<td>0.0015</td>
<td>0.0054</td>
<td>0.0060</td>
<td>0.0028</td>
<td>0.0031</td>
<td>0.0026</td>
<td>0.0040</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>64.8977</td>
<td>63.9534</td>
<td>65.7049</td>
<td>67.5994</td>
<td>73.0544</td>
<td>69.3178</td>
<td>80.7455</td>
<td>76.2758</td>
<td>67.9499</td>
<td>81.0780</td>
</tr>
</tbody>
</table>

Figure 9. The figure for Table 9

Table 10. The results for 10 parameters and the objects ranging from 1000 to 10000

<table>
<thead>
<tr>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
<th>6000</th>
<th>7000</th>
<th>8000</th>
<th>9000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>0.0455</td>
<td>0.1582</td>
<td>0.2538</td>
<td>0.4141</td>
<td>0.6190</td>
<td>0.9392</td>
<td>1.1141</td>
<td>1.3166</td>
<td>1.7072</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.0105</td>
<td>0.0214</td>
<td>0.0282</td>
<td>0.0385</td>
<td>0.0504</td>
<td>0.0637</td>
<td>0.0735</td>
<td>0.0853</td>
<td>0.1006</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0350</td>
<td>0.1368</td>
<td>0.2256</td>
<td>0.3755</td>
<td>0.5686</td>
<td>0.8755</td>
<td>1.0406</td>
<td>1.2312</td>
<td>1.6066</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>76.9072</td>
<td>86.4928</td>
<td>88.8727</td>
<td>90.6957</td>
<td>91.8561</td>
<td>93.2180</td>
<td>95.4048</td>
<td>93.5174</td>
<td>94.1086</td>
</tr>
</tbody>
</table>

Figure 10. The figure for Table 10

Table 11. The results for the parameters and the objects ranging from 10 to 100

<table>
<thead>
<tr>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>0.0128</td>
<td>0.0151</td>
<td>0.0109</td>
<td>0.0162</td>
<td>0.0425</td>
<td>0.0472</td>
<td>0.0682</td>
<td>0.0956</td>
<td>0.1185</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.0048</td>
<td>0.0051</td>
<td>0.0023</td>
<td>0.0006</td>
<td>0.0034</td>
<td>0.0007</td>
<td>0.0008</td>
<td>0.0010</td>
<td>0.0011</td>
</tr>
<tr>
<td>Difference</td>
<td>0.0080</td>
<td>0.0101</td>
<td>0.0086</td>
<td>0.0156</td>
<td>0.0391</td>
<td>0.0465</td>
<td>0.0674</td>
<td>0.0946</td>
<td>0.1174</td>
</tr>
<tr>
<td>Advantage (%)</td>
<td>62.6342</td>
<td>66.5763</td>
<td>78.7540</td>
<td>96.2336</td>
<td>91.9715</td>
<td>98.5117</td>
<td>98.8262</td>
<td>99.0022</td>
<td>99.1040</td>
</tr>
</tbody>
</table>
Figure 11. The figure for Table 11

Table 12. The results for the parameters and the objects ranging from 100 to 1000

<table>
<thead>
<tr>
<th></th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
<th>600</th>
<th>700</th>
<th>800</th>
<th>900</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CE10o</td>
<td>0.1923</td>
<td>2.0348</td>
<td>8.3202</td>
<td>25.3170</td>
<td>58.2243</td>
<td>105.5598</td>
<td>190.4531</td>
<td>320.2249</td>
<td>503.3607</td>
<td>751.5572</td>
</tr>
<tr>
<td>EMO18o</td>
<td>0.0079</td>
<td>0.0048</td>
<td>0.0051</td>
<td>0.0080</td>
<td>0.0132</td>
<td>0.0149</td>
<td>0.0187</td>
<td>0.0228</td>
<td>0.0308</td>
<td>0.0336</td>
</tr>
<tr>
<td>Difference</td>
<td>0.1844</td>
<td>2.0300</td>
<td>8.3151</td>
<td>25.3090</td>
<td>58.2111</td>
<td>105.5450</td>
<td>190.4344</td>
<td>320.2021</td>
<td>503.3300</td>
<td>751.5236</td>
</tr>
</tbody>
</table>

Figure 12. The figure for Table 12

5. An Application of EMO18o

In this section, we apply EMO18o to a performance-based value assignment to the methods used in the noise removal, so that we can order them in terms of performance.

Meaning to be removed the noises which occur during the acquisition or transfer of an image, the image denoising is a necessary pre-process for image processing. One of the most common tools in this subfield is non-linear filters. However, since the filters outperform in different noise densities and have different running times, to be sorted by the performances of these filters has made difficult. To overcome this problem, in this section, we use the soft decision-making method EMO18o. For this reason, we evaluated the results of some salt-and-pepper noise removal methods, well-known in the literature, Progressive Switching Median Filter (PSMF) (Wang and Zhang, 1999), Decision Based Algorithm (DBA) (Pattnaik et al. 2012), Modified Decision Based Unsymmetrical Trimmed Median Filter (MDBUTMF) (Esakkirajan et al. 2011), Noise Adaptive Fuzzy Switching Median Filter (NAFSMF) (Toh and Isa, 2010), Different Applied Median Filter (DAMF) (Erkan et al. 2018) by using 2 traditional images Cameraman and Lena with 512 x 512 pixels, ranging in noise densities from 10% to 90%, and an image quality metric Structural
Similarity (SSIM) (Wang et al. 2004), commonly used in literature. These simulation results are as follows:

**Table 13. The SSIM results for the Cameraman image**

<table>
<thead>
<tr>
<th>Filters</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSMF</td>
<td>0.9722</td>
<td>0.9454</td>
<td>0.9044</td>
<td>0.8036</td>
<td>0.6215</td>
<td>0.1178</td>
<td>0.0576</td>
<td>0.0290</td>
<td>0.0129</td>
</tr>
<tr>
<td>DBA</td>
<td>0.9883</td>
<td>0.9664</td>
<td>0.9324</td>
<td>0.8795</td>
<td>0.8167</td>
<td>0.7413</td>
<td>0.6650</td>
<td>0.5841</td>
<td>0.4858</td>
</tr>
<tr>
<td>MDBUTMF</td>
<td>0.9501</td>
<td>0.8388</td>
<td>0.7740</td>
<td>0.8249</td>
<td>0.9014</td>
<td>0.9178</td>
<td>0.8954</td>
<td>0.7864</td>
<td>0.4062</td>
</tr>
<tr>
<td>NAFSM</td>
<td>0.9797</td>
<td>0.9642</td>
<td>0.9494</td>
<td>0.9340</td>
<td>0.9198</td>
<td>0.8975</td>
<td>0.8745</td>
<td>0.8344</td>
<td>0.7246</td>
</tr>
<tr>
<td>DAMF</td>
<td>0.9963</td>
<td>0.9911</td>
<td>0.9844</td>
<td>0.9760</td>
<td>0.9659</td>
<td>0.9511</td>
<td>0.9323</td>
<td>0.9008</td>
<td>0.8373</td>
</tr>
</tbody>
</table>

**Table 14. The SSIM results for the Lena image**

<table>
<thead>
<tr>
<th>Filters</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSMF</td>
<td>0.9840</td>
<td>0.9631</td>
<td>0.9163</td>
<td>0.7854</td>
<td>0.5640</td>
<td>0.1115</td>
<td>0.0542</td>
<td>0.0263</td>
<td>0.0123</td>
</tr>
<tr>
<td>DBA</td>
<td>0.9758</td>
<td>0.9422</td>
<td>0.8952</td>
<td>0.8308</td>
<td>0.7549</td>
<td>0.6651</td>
<td>0.5673</td>
<td>0.4442</td>
<td>0.3458</td>
</tr>
<tr>
<td>MDBUTMF</td>
<td>0.9542</td>
<td>0.8686</td>
<td>0.8137</td>
<td>0.8449</td>
<td>0.8841</td>
<td>0.8835</td>
<td>0.8521</td>
<td>0.7392</td>
<td>0.3395</td>
</tr>
<tr>
<td>NAFSM</td>
<td>0.9838</td>
<td>0.9667</td>
<td>0.9481</td>
<td>0.9293</td>
<td>0.9055</td>
<td>0.8809</td>
<td>0.8495</td>
<td>0.8043</td>
<td>0.6868</td>
</tr>
<tr>
<td>DAMF</td>
<td>0.9902</td>
<td>0.9792</td>
<td>0.9652</td>
<td>0.9503</td>
<td>0.9303</td>
<td>0.9090</td>
<td>0.8788</td>
<td>0.8382</td>
<td>0.7697</td>
</tr>
</tbody>
</table>

Let’s suppose that the success in high noise densities is more important than in the others. In that case, the values given in Table 13 and 14 can be represented with two \(fpfs\)-matrices as follows:

\[
[a_{ij}] = \begin{bmatrix}
0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\
0.9722 & 0.9454 & 0.9044 & 0.8036 & 0.6215 & 0.1178 & 0.0576 & 0.0290 & 0.0129 \\
0.9883 & 0.9642 & 0.9324 & 0.8795 & 0.8167 & 0.7413 & 0.6650 & 0.5841 & 0.4858 \\
0.9501 & 0.8388 & 0.7740 & 0.8249 & 0.9014 & 0.9178 & 0.8954 & 0.7864 & 0.4062 \\
0.9797 & 0.9642 & 0.9494 & 0.9340 & 0.9198 & 0.8975 & 0.8745 & 0.8344 & 0.7246 \\
0.9963 & 0.9911 & 0.9844 & 0.9760 & 0.9659 & 0.9511 & 0.9323 & 0.9008 & 0.8373
\end{bmatrix}
\]

and

\[
[b_{ij}] = \begin{bmatrix}
0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\
0.9840 & 0.9631 & 0.9163 & 0.7854 & 0.5640 & 0.1115 & 0.0542 & 0.0263 & 0.0123 \\
0.9758 & 0.9422 & 0.8952 & 0.8308 & 0.7549 & 0.6651 & 0.5673 & 0.4442 & 0.3458 \\
0.9542 & 0.8686 & 0.8137 & 0.8449 & 0.8841 & 0.8835 & 0.8521 & 0.7392 & 0.3395 \\
0.9838 & 0.9667 & 0.9481 & 0.9293 & 0.9055 & 0.8809 & 0.8495 & 0.8043 & 0.6868 \\
0.9902 & 0.9792 & 0.9652 & 0.9503 & 0.9303 & 0.9090 & 0.8788 & 0.8382 & 0.7697
\end{bmatrix}
\]

If we apply EMO18o to the \(fpfs\)-matrices \([a_{ij}]\) and \([b_{ij}]\), then the score matrix and the decision set are as follows:

\[
[s_{11}] = [0.3214 \ 0.4673 \ 0.6291 \ 0.6675 \ 0.7536]^T
\]

and

\[\{0.4266^{PSMF}, 0.6201^{DBA}, 0.8349^{MDBUTMF}, 0.8858^{NAFSM}, 1^{DAMF}\}\]

The scores show that DAMF outperforms the other methods and the order DAMF, NAFSM, MDBUTMF, DBA, and PSMF is valid.
Let’s suppose that the success in low noise densities is more important than in the others. In that case, the values given in Table 13 and 14 can be represented with two \( fpfs \)-matrices as follows:

\[
[c_{ij}] = \begin{bmatrix}
0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\
0.9722 & 0.9454 & 0.9044 & 0.8036 & 0.6215 & 0.1178 & 0.0576 & 0.0290 & 0.0129 \\
0.9883 & 0.9664 & 0.9324 & 0.8795 & 0.8167 & 0.7413 & 0.6650 & 0.5941 & 0.4858 \\
0.9501 & 0.8388 & 0.7740 & 0.8249 & 0.9014 & 0.9178 & 0.8954 & 0.7864 & 0.4062 \\
0.9797 & 0.9642 & 0.9494 & 0.9340 & 0.9198 & 0.8975 & 0.8745 & 0.8344 & 0.7246 \\
0.9963 & 0.9911 & 0.9844 & 0.9760 & 0.9659 & 0.9511 & 0.9323 & 0.9008 & 0.8373 
\end{bmatrix}
\]

and

\[
[d_{ij}] = \begin{bmatrix}
0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\
0.9840 & 0.9631 & 0.9163 & 0.7854 & 0.5640 & 0.1115 & 0.0542 & 0.0263 & 0.0123 \\
0.9758 & 0.9422 & 0.8952 & 0.8308 & 0.7549 & 0.6651 & 0.5673 & 0.4442 & 0.3458 \\
0.9542 & 0.8686 & 0.8137 & 0.8449 & 0.8841 & 0.8835 & 0.8521 & 0.7392 & 0.3935 \\
0.9838 & 0.9667 & 0.9481 & 0.9293 & 0.9055 & 0.8809 & 0.8495 & 0.8043 & 0.6868 \\
0.9902 & 0.9792 & 0.9652 & 0.9503 & 0.9303 & 0.9090 & 0.8788 & 0.8382 & 0.7697 
\end{bmatrix}
\]

If we apply EMO18o to the \( fpfs \)-matrices \([c_{ij}]\) and \([d_{ij}]\), then the score matrix and the decision set are as follows:

\[
[s_{11}] = [0.8856 \ 0.8895 \ 0.8588 \ 0.8854 \ 0.8967]^T
\]

and

\[
\{0.9877 \text{PSMF}, \ 0.9920 \text{DBA}, \ 0.9577 \text{MDBUTMF}, \ 0.9875 \text{NAFSM}, \ 1 \text{DAMF}\}
\]

The scores show that DAMF performs better than the other methods and the order DAMF, DBA, PSMF, NAFSM, and MDBUTMF is valid.

6. Conclusion

The uni-int decision-making method was defined in 2010 (Çağman and Enginoğlu, 2010a). Afterwards, this method has been configured (Enginoğlu and Memiş, 2018) via \( fpfs \)-matrices (Enginoğlu, 2012). However, the method suffers from a drawback, i.e. its incapability of processing a large amount of parameters on a standard computer, e.g. with 2.6 GHz i5 Dual Core CPU and 4GB RAM. For this reason, simplification of such methods is significant for a wide range of scientific and industrial processes. In this study, firstly, we have proposed two fast and simple soft decision-making methods EMO18a and EMO18o which one of them has first been presented in (Enginoğlu et al. 2018). Moreover, we have proved that these two methods accept CE10 as a special case, under the condition that the first rows of the \( fpfs \)-matrices are binary. It is also possible to investigate the simplifications of the other products such as andnot-product and ornot-product (see Definition 2.3).

We then have compared the running times of these algorithms. In addition to the results in Section 4, the results in Table 15 and 16 too show that EMO18a and EMO18o outperform CE10a and CE10o, respectively, in any number of data under the specified condition. Finally, we have applied EMO18o to the determination of the performance of the methods used in (Erkan et al. 2018). It is clear that EMO18o, which is a fast and simple method, can be successfully applied to the decision-making problems in various areas such as machine learning and image enhancement.
Furthermore, other decision-making methods constructed by a different decision function such as minimum-maximum (min-max), max-max, and min-min can also be studied by the similar way.

7. Acknowledgements

We thank Dr. Uğur Erkan for technical support. This work was supported by the Office of Scientific Research Projects Coordination at Çanakkale Onsekiz Mart University, Grant number: FBA-2018-1367.

8. References


