JOURNAL OF NEW RESULTS IN SCIENCE (JNRS)
ISSN: 1304-7981
http://dergipark.gov.tr/jnrs
Research Article

Volume : 7
Issue: 3
Year: 2018
Pages: 28-43

Received: 14.11.2018
Accepted: 12.12.2018

Editors-in-Chief: **Ebubekir ALTUNTAS**
Area Editor: **Serkan DEMİRİZ / Bilge GÖZENER**

Comment on Soft Set Theory and uni-int Decision-Making [European Journal of Operational Research, (2010) 207, 848-855]

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 [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 [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, EMO18a and 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 Çağman, 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, $\mu := \{^{\mu(x)}x : x \in E\}$.

Definition 2. 1. (Çağman et al., 2010; Enginoğlu, 2012) *Let U be a universal set, $\mu \in F(E)$, and α be a function from μ to $F(U)$. Then the graphic of α, denoted by α, defined by*

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

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,*

$$\alpha = \{(^{0}x_1, \{^{0.5}u_1, ^{0.6}u_3\}), (^{0.8}x_2, \{^{0.9}u_2, ^{0.2}u_3, ^{0.1}u_5\}), (^{0.6}x_3, \{^{0.5}u_2, ^{0.7}u_4, ^{0.2}u_5\}), (^{1}x_4, \{^{1}u_3, ^{0.9}u_4\})\}$$

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 α (or briefly fpfs-matrix of α) and defined by*

$$[a_{ij}] = \begin{bmatrix} a_{01} & a_{02} & a_{03} & \cdots & a_{0n} & \cdots \\ a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \end{bmatrix}$$

such that $for\ i = \{0,1,2,\cdots\}$ *and* $j = \{1,2,\cdots\}$,

$$a_{ij} := \begin{cases} \mu(x_j), & i = 0 \\ \alpha({}^{\mu(x_j)}x_j)(u_i), & i \neq 0 \end{cases}$$

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 us consider the fpfs-set* α *provided in Example 2.1. Then, the fpfs-matrix of* α *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* $[a_{ij}] \wedge [b_{ik}]$.

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* $[a_{ij}] \vee [b_{ik}]$.

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* $[a_{ij}] \overline{\wedge} [b_{ik}]$.

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* $[a_{ij}] \underline{\vee} [b_{ik}]$.

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 $[b_{ik}]$

Step 2. Find and-product/or-product *fpfs*-matrix $[c_{ip}]$ of $[a_{ij}]$ and $[b_{ik}]$

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

$$s_{i1} \coloneqq \max\{\max_j \min_k (c_{ip}), \max_k \min_j (c_{ip})\}$$

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

$$\max_j \min_k (c_{ip}) \coloneqq \begin{cases} \max\limits_{j\in I_a}\left\{\min\limits_{k\in I_b} c_{0p}c_{ip}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

$$\max_k \min_j (c_{ip}) \coloneqq \begin{cases} \max\limits_{k\in I_b}\left\{\min\limits_{j\in I_a} c_{0p}c_{ip}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

Step 4. Obtain the set $\{u_k \in U \mid s_{k1} = \max\limits_i s_{i1}\}$

Preferably, the set $\{^{s_{i1}}u_i \mid u_i \in U\}$ or $\left\{^{\frac{s_{k1}}{\max s_{i1}}}u_k \middle| u_k \in U\right\}$ 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 $[b_{ik}]$

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

$$s_{i1} \coloneqq \max\{\max_j \min_k (a_{ij}, b_{ik}), \max_k \min_j (a_{ij}, b_{ik})\}$$

such that $i \in \{1,2,\dots,m-1\}$, $I_a \coloneqq \{j \mid a_{0j} \neq 0\}$, $I_b \coloneqq \{k \mid b_{0k} \neq 0\}$, and

$$\max_j \min_k (a_{ij}, b_{ik}) \coloneqq \begin{cases} \min\left\{\max\limits_{j\in I_a}\{a_{0j}a_{ij}\}, \min\limits_{k\in I_b}\{b_{0k}b_{ik}\}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

$$\max_k \min_j (a_{ij}, b_{ik}) \coloneqq \begin{cases} \min\left\{\max\limits_{k\in I_b}\{b_{0k}b_{ik}\}, \min\limits_{j\in I_a}\{a_{0j}a_{ij}\}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

Step 3. Obtain the set $\{u_k \in U \mid s_{k1} = \max\limits_i s_{i1}\}$

Preferably, the set $\{^{s_{i1}}u_i \mid u_i \in U\}$ or $\left\{^{\frac{s_{k1}}{\max s_{i1}}}u_k \middle| u_k \in U\right\}$ can be attained.

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

EMO18o's Algorithm Steps

Step 1. Construct two *fpfs*-matrices $[a_{ij}]$ and $[b_{ik}]$

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

$$s_{i1} \coloneqq \max\{\max_j \min_k (a_{ij}, b_{ik}), \max_k \min_j (a_{ij}, b_{ik})\}$$

such that $i \in \{1,2,\dots,m-1\}$, $I_a \coloneqq \{j \mid a_{0j} \neq 0\}$, $I_b \coloneqq \{k \mid b_{0k} \neq 0\}$, and

$$\max_j \min_k \left(a_{ij}, b_{ik}\right) \coloneqq \begin{cases} \max\left\{\max\limits_{j\in I_a}\{a_{0j}a_{ij}\}, \min\limits_{k\in I_b}\{b_{0k}b_{ik}\}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

$$\max_k \min_j \left(a_{ij}, b_{ik}\right) \coloneqq \begin{cases} \max\left\{\max\limits_{k\in I_b}\{b_{0k}b_{ik}\}, \min\limits_{j\in I_a}\{a_{0j}a_{ij}\}\right\}, & I_a \neq \emptyset \text{ and } I_b \neq \emptyset \\ 0, & \text{otherwise} \end{cases}$$

Step 3. Obtain the set $\{u_k \in U \mid s_{k1} = \max\limits_i s_{i1}\}$

Preferably, the set $\left\{{}^{s_{i1}}u_i \mid u_i \in U\right\}$ or $\left\{{}^{\frac{s_{k1}}{\max s_{i1}}}u_k | u_k \in U\right\}$ can be attained.

Theorem 3. 1. *CE10a is a special case of EMO18a provided that first rows of the fpfs-matrices are binary.*

PROOF. Suppose that first rows of the *fpfs*-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 $j \in I_a \coloneqq \{a_1, a_2, \dots, a_s\}$ and $k \in I_b \coloneqq \{b_1, b_2, \dots, b_t\}$,

$$\begin{aligned}
\max_j \min_k (c_{ip}) &= \max_{j\in I_a}\left\{\min_{k\in I_b} c_{0p}c_{ip}\right\} \\
&= \max_{j\in I_a}\left\{\min_{k\in I_b}\left\{\min\{a_{0j}, b_{0k}\}.\min\{a_{ij}, b_{ik}\}\right\}\right\} \\
&= \max_{j\in I_a}\left\{\min_{k\in I_b}\left\{\min\{a_{ij}, b_{ik}\}\right\}\right\} \\
&= \max\Big\{\min\left\{\min\{a_{ia_1}, b_{ib_1}\}, \min\{a_{ia_1}, b_{ib_2}\}, \dots, \min\{a_{ia_1}, b_{ib_t}\}\right\}, \\
&\qquad \min\left\{\min\{a_{ia_2}, b_{ib_1}\}, \min\{a_{ia_2}, b_{ib_2}\}, \dots, \min\{a_{ia_2}, b_{ib_t}\}\right\}, \\
&\qquad \dots, \min\left\{\min\{a_{ia_s}, b_{ib_1}\}, \min\{a_{ia_s}, b_{ib_2}\}, \dots, \min\{a_{ia_s}, b_{ib_t}\}\right\}\Big\} \\
&= \max\Big\{\min\{a_{ia_1}, \min\{b_{ib_1}, b_{ib_2}, \dots, b_{ib_t}\}\}, \min\{a_{ia_2}, \min\{b_{ib_1}, b_{ib_2}, \dots, b_{ib_t}\}\}, \\
&\qquad \dots, \min\{a_{ia_s}, \min\{b_{ib_1}, b_{ib_2}, \dots, b_{ib_t}\}\}\Big\} \\
&= \min\left\{\max\{a_{ia_1}, a_{ia_2}, \dots, a_{ia_s}\}, \min\{b_{ib_1}, b_{ib_2}, \dots, b_{ib_t}\}\right\} \\
&= \min\left\{\max_{j\in I_a}\{a_{ij}\}, \min_{k\in I_b}\{b_{ik}\}\right\}
\end{aligned}$$

$$= \min\left\{\max_{j\in I_a}\{a_{0j}a_{ij}\}, \min_{k\in I_b}\{b_{0k}b_{ik}\}\right\}$$

$$= \max_j \min_k (a_{ij}, b_{ik})$$

In a similar way, $\max_k \min_j (c_{ip}) = \max_k \min_j (a_{ij}, b_{ik})$. Consequently,

$$\text{max-min}(a_{ij}, b_{ik}) = \text{max-min}(c_{ip})$$

□

Theorem 3. 2. (Enginoğlu et al., 2018) *CE10o is a special case of EMO18o provided that first rows of the fpfs-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

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

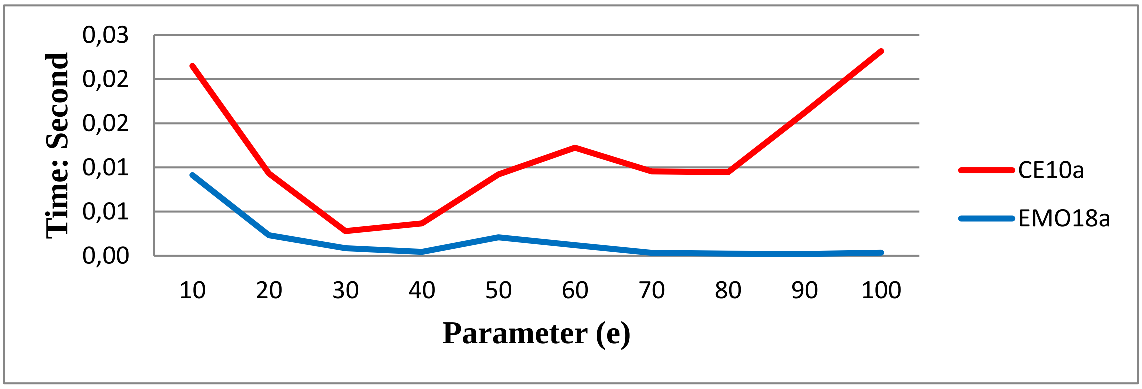

Fig. 1. The figure for Table 1

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

	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
CE10a	1.4126	4.9158	10.5733	18.2499	28.8022	39.2240	54.8238	71.3663	92.1878	117.1658
EMO18a	0.0072	0.0032	0.0017	0.0021	0.0041	0.0037	0.0030	0.0032	0.0036	0.0040
Difference	1.4054	4.9127	10.5716	18.2478	28.7981	39.2203	54.8208	71.3631	92.1842	117.1618
Advantage (%)	99.4908	99.9356	99.9838	99.9886	99.9858	99.9907	99.9946	99.9955	99.9961	99.9966

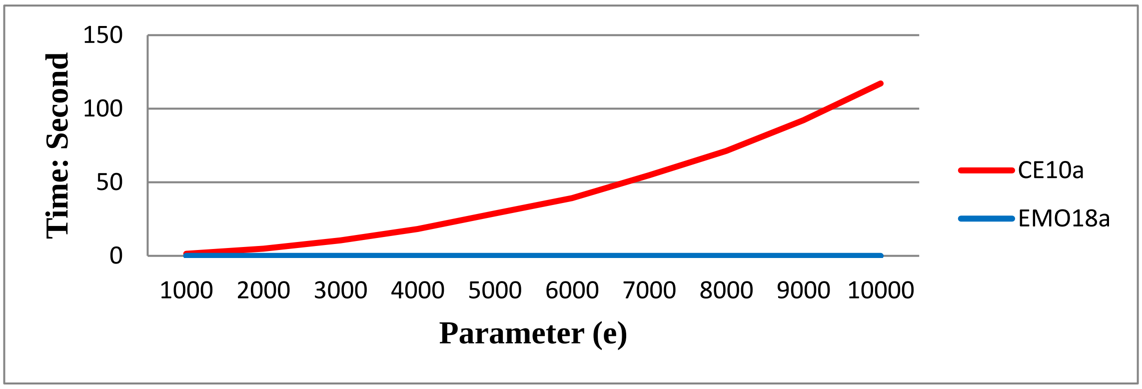

Fig. 2. The figure for Table 2

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

	10	20	30	40	50	60	70	80	90	100
CE10a	0.0174	0.0069	0.0019	0.0020	0.0057	0.0078	0.0023	0.0033	0.0057	0.0039
EMO18a	0.0063	0.0024	0.0006	0.0007	0.0024	0.0016	0.0007	0.0008	0.0008	0.0009
Difference	0.0110	0.0045	0.0013	0.0013	0.0032	0.0062	0.0016	0.0025	0.0050	0.0030
Advantage (%)	63.5402	65.6632	69.1512	65.4336	56.9135	79.9089	68.9759	77.1017	86.2588	77.6234

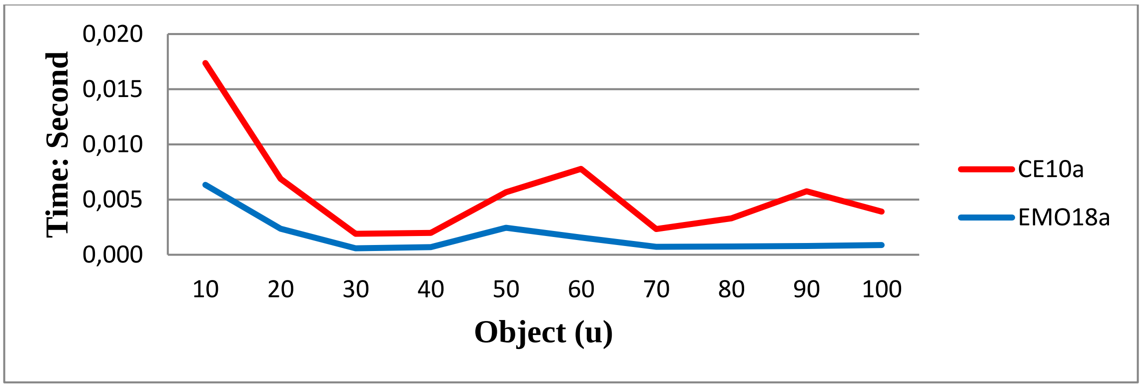

Fig. 3. The figure for Table 3

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

	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
CE10a	0.0596	0.1619	0.2678	0.4543	0.5862	0.8606	1.0839	1.3455	1.7614	2.2319
EMO18a	0.0148	0.0199	0.0276	0.0384	0.0499	0.0631	0.0752	0.0855	0.0994	0.1134
Difference	0.0448	0.1420	0.2402	0.4159	0.5363	0.7975	1.0087	1.2600	1.6620	2.1185
Advantage (%)	75.1099	87.6852	89.6930	91.5504	91.4930	92.6671	93.0655	93.6467	94.3553	94.9205

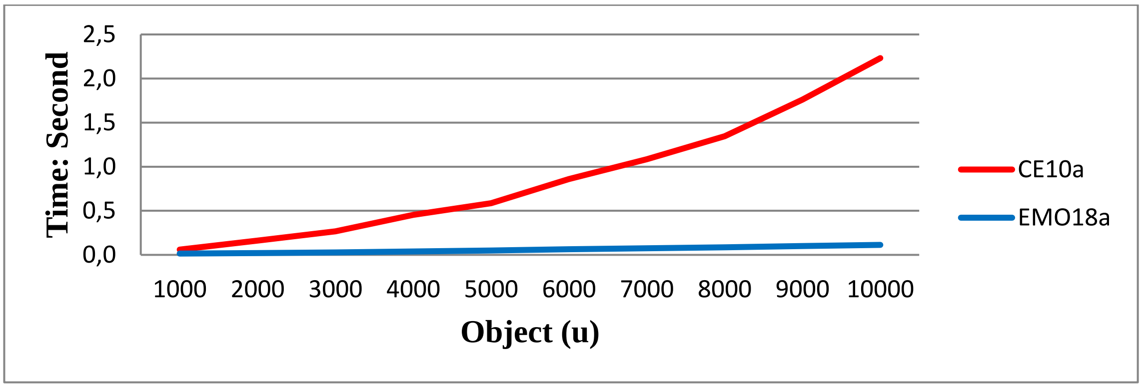

Fig. 4. The figure for Table 4

Table 5. The results for the parameters and the objects ranging from 10 to 100

	10	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
CE10a	0.0175	0.0078	0.0073	0.0141	0.0227	0.0500	0.0683	0.0936	0.1104	0.1536
EMO18a	0.0061	0.0023	0.0007	0.0007	0.0026	0.0020	0.0008	0.0010	0.0010	0.0011
Difference	0.0115	0.0055	0.0066	0.0134	0.0201	0.0480	0.0675	0.0926	0.1094	0.1525
Advantage (%)	65.3281	70.1501	90.6707	95.0966	88.6854	96.0033	98.8679	98.9627	99.1043	99.2978

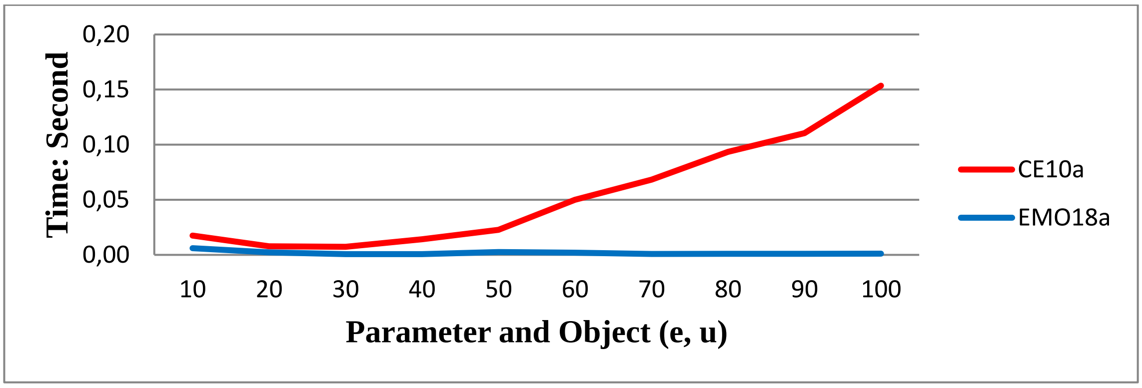

Fig. 5. The figure for Table 5

Table 6. The results for the parameters and the objects ranging from 100 to 1000

	100	**200**	**300**	**400**	**500**	**600**	**700**	**800**	**900**	**1000**
CE10a	0.2100	2.1134	8.4209	23.7731	56.3553	105.5166	188.7614	297.5683	485.1053	724.1639
EMO18a	0.0089	0.0048	0.0050	0.0075	0.0125	0.0155	0.0179	0.0225	0.0275	0.0331
Difference	0.2011	2.1086	8.4159	23.7656	56.3428	105.5011	188.7435	297.5458	485.0777	724.1308
Advantage (%)	95.7673	99.7739	99.9404	99.9683	99.9778	99.9853	99.9905	99.9924	99.9943	99.9954

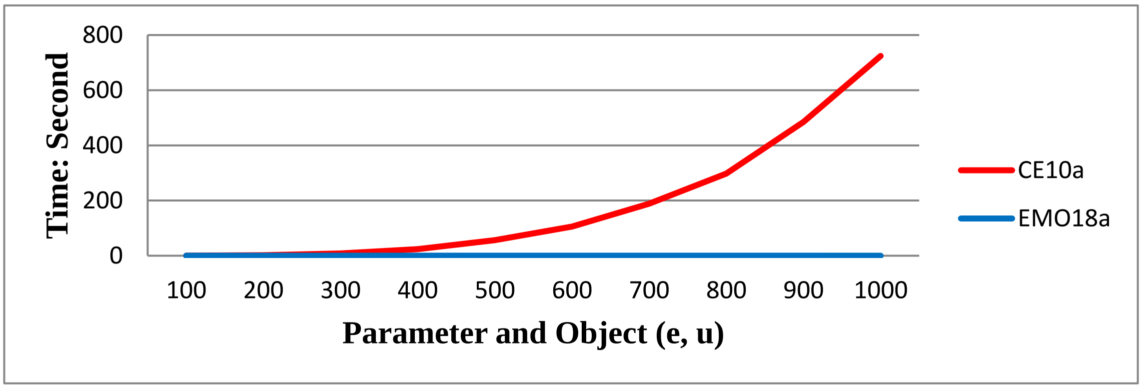

Fig. 6. The figure for Table 6

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

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

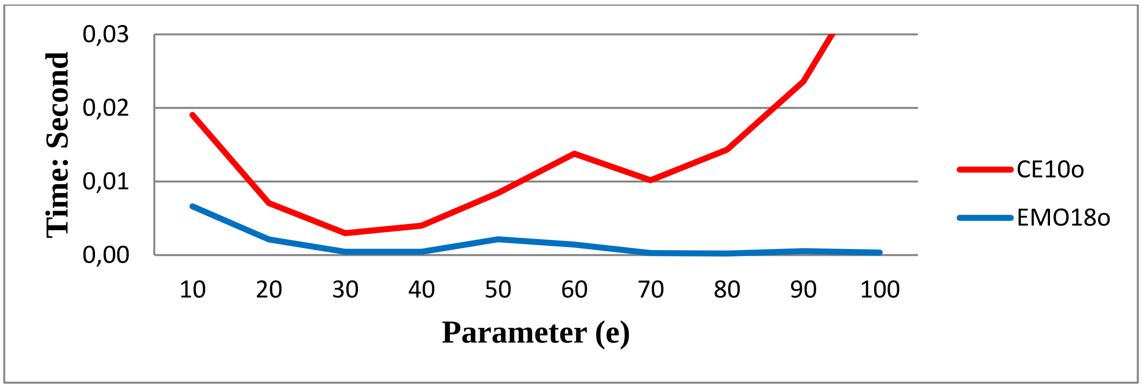

Fig. 7. The figure for Table 7

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

	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
CE10o	1.4542	5.1004	10.5845	18.3512	28.8547	39.7985	54.9162	73.4039	93.8602	117.4047
EMO18o	0.0075	0.0044	0.0018	0.0020	0.0041	0.0039	0.0031	0.0032	0.0044	0.0041
Difference	1.4468	5.0960	10.5828	18.3492	28.8505	39.7946	54.9131	73.4007	93.8557	117.4006
Advantage (%)	99.4859	99.9131	99.9834	99.9890	99.9857	99.9901	99.9944	99.9957	99.9953	99.9965

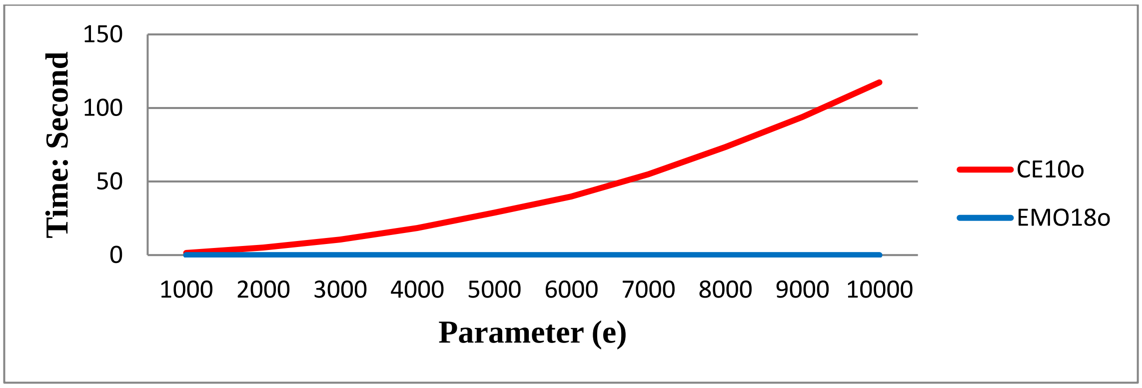

Fig. 8. The figure for Table 8

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

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

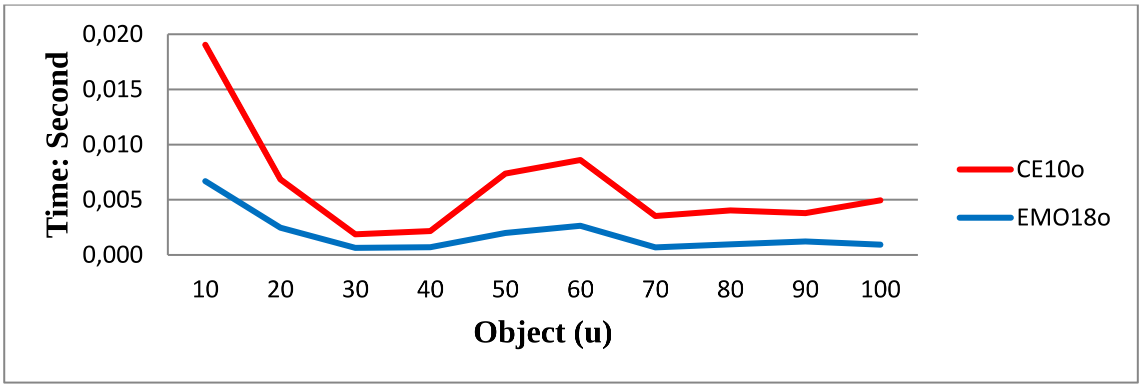

Fig. 9. The figure for Table 9

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

	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
CE10o	0.0455	0.1582	0.2538	0.4141	0.6190	0.9392	1.1141	1.3166	1.7072	2.1520
EMO18o	0.0105	0.0214	0.0282	0.0385	0.0504	0.0637	0.0735	0.0853	0.1006	0.1139
Difference	0.0350	0.1368	0.2256	0.3755	0.5686	0.8755	1.0406	1.2312	1.6066	2.0380
Advantage (%)	76.9072	86.4928	88.8727	90.6937	91.8561	93.2180	93.4048	93.5174	94.1086	94.7052

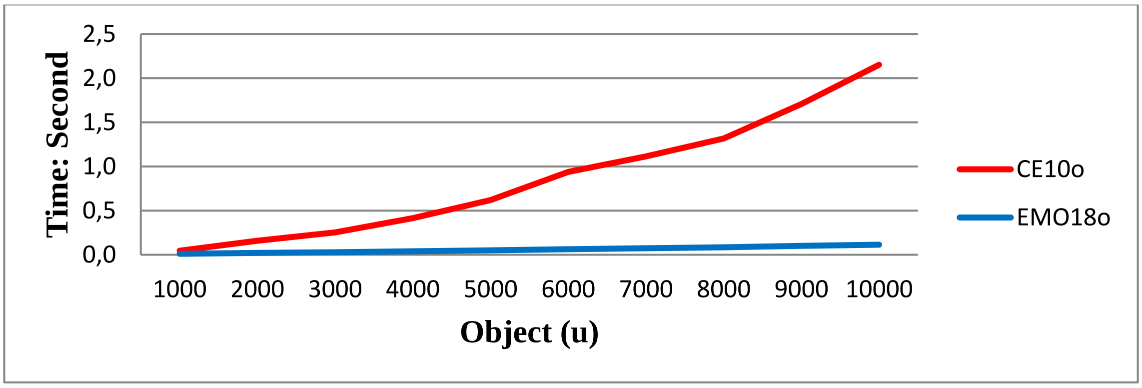

Fig. 10. The figure for Table 10

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

	10	20	30	40	50	60	70	80	90	100
CE10o	0.0128	0.0151	0.0109	0.0162	0.0425	0.0472	0.0682	0.0956	0.1185	0.2012
EMO18o	0.0048	0.0051	0.0023	0.0006	0.0034	0.0007	0.0008	0.0010	0.0011	0.0012
Difference	0.0080	0.0101	0.0086	0.0156	0.0391	0.0465	0.0674	0.0946	0.1174	0.1999
Advantage (%)	62.6342	66.5763	78.7540	96.2336	91.9715	98.5117	98.8262	99.0022	99.1040	99.3912

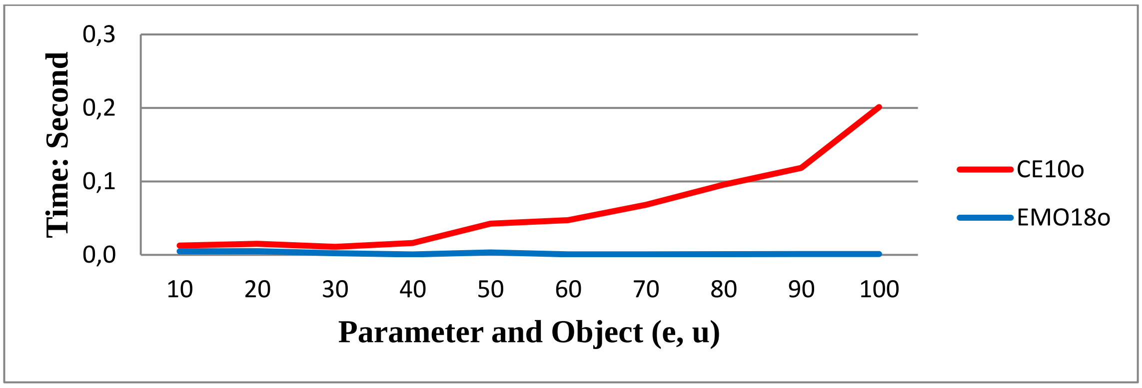

Fig. 11. The figure for Table 11

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

	100	200	300	400	500	600	700	800	900	1000
CE10o	0.1923	2.0348	8.3202	25.3170	58.2243	105.5598	190.4531	320.2249	503.3607	751.5572
EMO18o	0.0079	0.0048	0.0051	0.0080	0.0132	0.0149	0.0187	0.0228	0.0308	0.0336
Difference	0.1844	2.0300	8.3151	25.3090	58.2111	105.5450	190.4344	320.2021	503.3300	751.5236
Advantage (%)	95.8737	99.7642	99.9383	99.9684	99.9773	99.9859	99.9902	99.9929	99.9939	99.9955

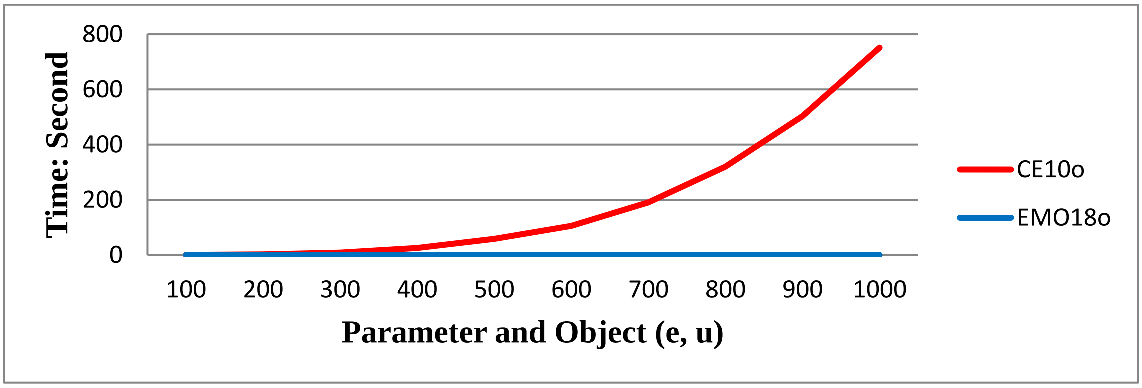

Fig. 12. The figure for Table 12

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 512×512 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

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

Table 14. The SSIM results for the Lena image

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

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:

$$[a_{ij}] := \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\ 0.9722 & 0.9454 & 0.9044 & 0.8036 & 0.6215 & 0.1178 & 0.0576 & 0.0290 & 0.0129 \\ 0.9883 & 0.9664 & 0.9324 & 0.8795 & 0.8167 & 0.7413 & 0.6650 & 0.5841 & 0.4858 \\ 0.9501 & 0.8388 & 0.7740 & 0.8249 & 0.9014 & 0.9178 & 0.8954 & 0.7864 & 0.4062 \\ 0.9797 & 0.9642 & 0.9494 & 0.9340 & 0.9198 & 0.8975 & 0.8745 & 0.8344 & 0.7246 \\ 0.9963 & 0.9911 & 0.9844 & 0.9760 & 0.9659 & 0.9511 & 0.9323 & 0.9008 & 0.8373 \end{bmatrix}$$

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:

$$[s_{i1}] = [0.3214 \;\; 0.4673 \;\; 0.6291 \quad 0.6675 \quad 0.7536]^T$$

and

$$\{{}^{0.4266}\text{PSMF},\ {}^{0.6201}\text{DBA},\ {}^{0.8349}\text{MDBUTMF},\ {}^{0.8858}\text{NAFSMF},\ {}^{1}\text{DAMF}\}$$

The scores show that DAMF outperforms the other methods and the order DAMF, NAFSMF, MDBUTMF, DBA, and PSMF is valid.

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:

$$[c_{ij}] := \begin{bmatrix} 0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\ 0.9722 & 0.9454 & 0.9044 & 0.8036 & 0.6215 & 0.1178 & 0.0576 & 0.0290 & 0.0129 \\ 0.9883 & 0.9664 & 0.9324 & 0.8795 & 0.8167 & 0.7413 & 0.6650 & 0.5841 & 0.4858 \\ 0.9501 & 0.8388 & 0.7740 & 0.8249 & 0.9014 & 0.9178 & 0.8954 & 0.7864 & 0.4062 \\ 0.9797 & 0.9642 & 0.9494 & 0.9340 & 0.9198 & 0.8975 & 0.8745 & 0.8344 & 0.7246 \\ 0.9963 & 0.9911 & 0.9844 & 0.9760 & 0.9659 & 0.9511 & 0.9323 & 0.9008 & 0.8373 \end{bmatrix}$$

and

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

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

$$[s_{i1}] = [0.8856 \;\; 0.8895 \;\; 0.8588 \quad 0.8854 \quad 0.8967]^T$$

and

$$\{^{0.9877}\text{PSMF},\ ^{0.9920}\text{DBA},\ ^{0.9577}\text{MDBUTMF},\ ^{0.9875}\text{NAFSMF},\ ^{1}\text{DAMF}\}$$

The scores show that DAMF performs better than the other methods and the order DAMF, DBA, PSMF, NAFSMF, 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.

Table 15. The mean/max advantage and max difference values of EMO18a over CE10a

Location	Objects	Parameters	Mean Advantage %	Max Advantage %	Max Difference
Table 1	10	10-100	84.9292	98.7938	0.0228
Table 2	10	1000-10000	99.9358	99.9966	117.1618
Table 3	10-100	10	71.0570	86.2588	0.0110
Table 4	1000-10000	10	90.4186	94.9205	2.1185
Table 5	10-100	10-100	90.2167	99.2978	0.1525
Table 6	100-1000	100-1000	99.5386	99.9954	724.1308

Table 16. The mean/max advantage and max difference values of EMO18o over CE10o

Location	Objects	Parameters	Mean Advantage %	Max Advantage %	Max Difference
Table 1	10	10-100	86.5476	99.1575	0.0403
Table 2	10	1000-10000	99.9329	99.9965	117.4006
Table 3	10-100	10	71.0577	81.0780	0.0124
Table 4	1000-10000	10	90.3777	94.7052	2.0380
Table 5	10-100	10-100	89.1005	99.3912	0.1999
Table 6	100-1000	100-1000	99.5480	99.9955	751.5236

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

Atmaca, S., 2017. Relationship between Fuzzy Soft Topological Spaces and (X, τ_e) Parameter Spaces. Cumhuriyet Science Journal. 38: 77-85.

Atmaca, S., Zorlutuna, İ., 2014. On Topological Structures of Fuzzy Parametrized Soft Sets. The Scientific World Journal. Article ID: 164176. 8 pages.

Bera, S., Roy, S.K., Karaaslan, F., Çağman, N., 2017. Soft Congruence Relation over Lattice. Hacettepe Journal of Mathematics and Statistics. 46(6): 1035-1042.

Çağman, N., Çıtak, F., Enginoğlu, S., 2010. Fuzzy Parameterized Fuzzy Soft Set Theory and Its Applications. Turkish Journal of Fuzzy Systems. 1(1): 21-35.

Çağman, N., Çıtak, F., Enginoğlu, S., 2011. FP-Soft Set Theory and Its Applications. Annals of Fuzzy Mathematics and Informatics. 2(2): 219-226.

Çağman, N., Deli, İ., 2012. Means of FP-Soft Sets and Their Applications. Hacettepe Journal of Mathematics and Statistics. 41(5): 615-625.

Çağman, N., Enginoğlu, S., 2010a. Soft Set Theory and Uni-Int Decision Making. European Journal of Operational Research. 207: 848-855.

Çağman, N., Enginoğlu, S., 2010b. Soft Matrix Theory and Its Decision Making. Computers and Mathematics with Applications. 59: 3308-3314.

Çağman, N., Enginoğlu, S., 2012. Fuzzy Soft Matrix Theory and Its Application in Decision Making. Iranian Journal of Fuzzy Systems. 9(1): 109-119.

Çağman, N., Enginoğlu, S., Çıtak, F., 2011. Fuzzy Soft Set Theory and Its Applications. Iranian Journal of Fuzzy Systems. 8(3): 137-147.

Çıtak, F., Çağman, N., 2015. Soft Int-Rings and Its Algebraic Applications. Journal of Intelligent and Fuzzy Systems. 28: 1225-1233.

Çıtak, F., Çağman, N., 2017. Soft k-Int-Ideals of Semirings and Its Algebraic Structures. Annals of Fuzzy Mathematics and Informatics. 13(4): 531-538.

Das, P.K., Borgohain, R., 2012. An Application of Fuzzy Soft Set in Multicriteria Decision Making Problem. International Journal of Computer Applications. 38(12): 33-37.

Enginoğlu, S., 2012. Soft Matrices. PhD dissertation. Gaziosmanpaşa University, *Graduate School of Natural and Applied Sciences, Department of Mathematics, p.85, Tokat –Turkey (in Turkish)*.

Enginoğlu, S., Çağman, N., Karataş, S., Aydın, T., 2015. On Soft Topology. El-Cezerî Journal of Science and Engineering. 2(3): 23-38.

Enginoğlu, S., Memiş, S., 2018. A Configuration of Some Soft Decision-Making Algorithms via *fpfs*-Matrices. Cumhuriyet Science Journal. 39(4): In Press.

Enginoğlu, S., Memiş, S., Öngel, T., 2018. A Fast and Simple Soft Decision-Making Algorithm: EMO18o. Eds: Akgül, M., Yılmaz, İ., İpek, A. International Conference on Mathematical Studies and Applications, October 4-6, Karaman, Turkey.

Eraslan, S., 2015. A Decision Making Method via TOPSIS on Soft Sets. Journal of New Results in Science. 8: 57-71.

Eraslan, S., Karaaslan, F., 2015. A Group Decision Making Method Based on TOPSIS Under Fuzzy Soft Environment. Journal of New Theory. 3: 30-40.

Erkan, U., Gökrem, L., Enginoğlu, S., 2018. Different Applied Median Filter in Salt and Pepper Noise. Computers and Electrical Engineering. 70: 789-798.

Esakkirajan, S., Veerakumar, T., Subramanyam, A. N., PremChand, C.H. 2011. Removal of High Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter. IEEE Signal Processing Letters. 18(5): 287-290.

Karaaslan, F., 2016. Soft Classes and Soft Rough Classes with Applications in Decision Making. Mathematical Problems in Engineering. Article ID: 1584528, 11 pages.

Khan, N., Khan, F.H., Thakur, G.S., 2013. Weighted Fuzzy Soft Matrix Theory and Its Decision Making. International Journal of Advances in Computer Science and Technology. 2(10): 214-218.

Maji, P.K., Biswas, R., Roy, A.R., 2001. Fuzzy Soft Sets. The Journal of Fuzzy Mathematics. 9(3): 589-602.

Maji, P.K., Roy, A.R., Biswas, R., 2002. An Application of Soft Sets in a Decision Making Problem. Computers and Mathematics with Applications. 44: 1077-1083.

Maji, P.K., Biswas, R., Roy, A.R., 2003. Soft Set Theory. Computers and Mathematics with Applications. 45: 555-562.

Molodtsov, D., 1999. Soft Set Theory-First Results. Computers and Mathematics with Applications. 37: 19-31.

Muştuoğlu, E., Sezgin, A., Türk, Z. K., 2016. Some Characterizations on Soft Uni-Groups and Normal Soft Uni-Groups. International Journal of Computer Applications. 155(10), 8 pages.

Pattnaik, A., Agarwal, S., Chand, S., 2012. A New and Efficient Method for Removal of High Density Salt and Pepper Noise Through Cascade Decision Based Filtering Algorithm. Procedia Technology. 6: 108-117.

Razak, S.A., Mohamad, D., 2011. A Soft Set Based Group Decision Making Method with Criteria Weight. World Academy of Sciences, Engineering and Technology. 5(10): 1641-1646.

Razak, S.A., Mohamad, D., 2013. A Decision Making Method Using Fuzzy Soft Sets. Malaysian Journal of Fundamental and Applied Sciences. 9(2): 99-104.

Sezgin, A., 2016. A New Approach to Semigroup Theory I: Soft Union Semigroups, Ideals and Bi-Ideals. Algebra Letters. 2016:3, 46 pages.

Sezgin, A., Çağman, N., Çıtak, F., 2019. α-Inclusions Applied to Group Theory via Soft Set and Logic. Communications Faculty of Sciences University of Ankara Series A1 Mathematics and Statistics 68(1): 334-352.

Toh, K.K.V., Isa, N.A.M., 2010. Noise Adaptive Fuzzy Switching Median Filter for Salt-and-Pepper Noise Reduction. IEEE Signal Processing Letters. 17(3): 281-284.

Tunçay, M., Sezgin, A., 2016. Soft Union Ring and Its Applications to Ring Theory. International Journal of Computer Applications. 151(9): 7-13.

Ullah, A., Karaaslan, F., Ahmad, I., 2018. Soft Uni-Abel-Grassmann's Groups. European Journal of Pure and Applied Mathematics. 11(2): 517-536.

Vijayabalaji, S., Ramesh, A., 2013. A New Decision Making Theory in Soft Matrices. International Journal of Pure and Apllied Mathematics. 86(6): 927-939.

Wang, Z., Zhang, D., 1999. Progressive Switching Median Filter for the Removal of Impulse Noise from Highly Corrupted Images. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing. 46(1): 78-80.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. IEEE Transactions on Image Processing. 13(4): 600-612.

Zadeh, L. A., 1965. Fuzzy Sets. Information and Control. 8: 338-353.
Zhu, K., Zhan, J., 2016. Fuzzy Parameterized Fuzzy Soft Sets and Decision Making. International Journal of Machine Learning and Cybernetics. 7: 1207-1212.
Zorlutuna, İ., Atmaca, S., 2016. Fuzzy Parametrized Fuzzy Soft Topology. New Trends in Mathematical Sciences. 4(1): 142-152.