Journal of New Results in Science (JNRS)
ISSN: 1304-7981
http://dergipark.gov.tr/en/pub/jnrs
Research Article
Open Access

Volume: 8
Issue: 2
Year: 2019
Pages: 42-52

Received: 06.12.2019 ***Accepted***: 23.12.2019 ***Published***: 27.12.2019

A New Approach to Group Decision-Making Method Based on TOPSIS Under Fuzzy Soft Environment

Serdar Enginoğlu[a,*], **Samet Memiş**[a], **Faruk Karaaslan**[b]

[a] *Department of Mathematics, Faculty of Arts and Sciences, Çanakkale Onsekiz Mart University, Çanakkale, Turkey*
[b] *Department of Mathematics, Faculty of Sciences, Çankırı Karatekin University, Çankırı, Turkey*
**Corresponding author, serdarenginoglu@gmail.com*

ABSTRACT: TOPSIS, developed in 1981 by Hwang and Yoon, is one of the known multi-criteria decision-making (MCDM) methods. In 2015, the group decision-making method based on TOPSIS under fuzzy soft environment was defined and applied to a decision-making problem by Eraslan and Karaaslan. Recently, this method has been configured by Enginoğlu and Memiş 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 have uncertainties. However, the configured method has two drawbacks which affect its running time and the ranking order negatively. We, in this study, improve this method by removing the disadvantages. We then compare the running time of these algorithms. The results show that the new method outperforms it, in particular, a large number of data come into question. For example, the proposed method offers up to 97.7672% of time advantage for ten objects and 9000 parameters. Afterwards, we apply the new method to a performance-based value assignment to seven state-of-art filters used in image denoising, 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

Molodtsov (1999) propounded the concept of soft sets, parameterized family of subsets of the set of alternatives, to deal with uncertainties, and up to now, many researchers have conducted various applied and theoretical studies on that (Maji et al., 2001, 2003; Çağman and Enginoğlu, 2010a; Çağman et al., 2011a; Çağman et al., 2011b; Karaaslan et al., 2012; Enginoğlu et al., 2015; Zorlutuna and Atmaca, 2016; Riaz and Hashmi, 2017, 2018; Şenel, 2016, 2017, 2018; Ullah et al., 2018; Sezgin et al., 2019). To able to use the ability of this concept in computer mathematics (or sciences), Çağman and Enginoğlu then presented the soft matrices (Çağman and Enginoğlu, 2010b), fuzzy soft matrices (Çağman and Enginoğlu, 2012), and fuzzy parameterized fuzzy soft matrices (*fpfs*-matrices) (Enginoğlu, 2012; Enginoğlu and Çağman, n.d.).

Since fuzzy/soft sets/matrices and their hybrid versions adept in decision-making, the studies containing soft decision-making (SDM) methods have been increased rapidly (Çağman et al., 2010; Razak and Mohamad, 2011, 2013; Deli and Çağman, 2015; Karaaslan, 2016; Riaz et al., 2018). For example, Eraslan and Karaaslan (2015) proposed the group decision-making method based on TOPSIS under fuzzy soft environment. The method therein, however, has two drawbacks which affect its running time and the ranking order negatively. Among these versions mentioned above, fuzzy parameterized fuzzy soft matrices (*fpfs*-matrices) are the most prominent in terms of performance. Therefore, Enginoğlu and Memiş

(2018a) have configured eighteen SDM methods via *fpfs*-matrices, faithfully to the original. Since the configurations have been made as faithfully to the originals, the drawbacks, such as mentioned above have also been transferred. On the other hand, the lack of the names of the configured methods causes difficulty of use. To overcome this drawback, Enginoğlu and Memiş have suggested a notation using the combination of the first letters of the authors' surname and the last two digits of the publication year of the paper. For example, the method constructed by Eraslan and Karaaslan in 2015 was denoted by EK15 therein. Moreover, the authors have pointed out that studies on the simplifications and different configurations of these configurated methods therein are worth doing. Therefore, several SDM algorithms provided in (Enginoğlu and Memiş, 2018a) have been simplified and applied (Enginoğlu and Memiş, 2018b,c; Enginoğlu et al., 2018a,b, 2019b) to a decision-making problem.

In this paper, we have focused on improving a new method free of the disadvantages mentioned above. In Section 2, we present the concept of *fpfs*-matrices (Enginoğlu, 2012; Enginoğlu and Çağman, n.d.) and give EK15 (Eraslan and Karaaslan, 2015; Enginoğlu and Memiş, 2018a). In Section 3, we propound a new method, namely EMK19. Here, EMK19 refers to the method constructed by Enginoğlu, Memiş, and Karaaslan in 2019. In Section 4, we compare the running time of these algorithms. In Section 5, we apply EMK19 to a decision-making problem in which the noise removal/image denoising filters can be ordered in terms of performance. We then compare the ranking orders by EMK19 with the ranking orders by EK15. Finally, we discuss the need for further research.

2. Preliminaries

In this section, firstly, definitions of *fpfs*-sets (Enginoğlu et al., 2010; Enginoğlu and Çağman, n.d.) and *fpfs*-matrices (Enginoğlu, 2012; Enginoğlu and Çağman, n.d.) have been presented. 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, a fuzzy set is denoted by $\left\{{}^{\mu(x)}x : x \in E\right\}$.

Definition 2. 1. (Enginoğlu et al., 2010; Enginoğlu and Çağman, n.d.) *Let U be a universal set, $\mu \in F(E)$, and α be a function from μ to $F(U)$. Then, the set $\{({}^{\mu(x)}x, \alpha({}^{\mu(x)}x)) : x \in E\}$ being the graphic of α is called a fuzzy parameterized fuzzy soft set (fpfs-set) parameterized via E over U (or briefly over 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,*

$$\alpha = \left\{\left({}^{0}x_1, \left\{{}^{0.9}u_2, {}^{0.6}u_5\right\}\right), \left({}^{0.5}x_2, \left\{{}^{0.8}u_1, {}^{0.4}u_3, {}^{0.7}u_4\right\}\right), \left({}^{0.7}x_3, \left\{{}^{0.2}u_1, {}^{1}u_3, {}^{0.8}u_5\right\}\right), \left({}^{1}x_4, \left\{{}^{0.5}u_2, {}^{0.8}u_4\right\}\right)\right\}$$

is an fpfs-set over U.

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

Definition 2..2. (Enginoğlu, 2012; Enginoğlu and Çağman, n.d.) *Let $\alpha \in FPFS_E(U)$. Then, $[a_{ij}]$ is called the matrix representation of α (or briefly fpfs-matrix of α) and is 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 $i \in \{0,1,2,\cdots\}$ *and* $j \in \{1,2,\cdots\}$,

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

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

Example 2. 2. *The fpfs-matrix of* α *provided in Example 2.1 is as follows:*

$$[a_{ij}] = \begin{bmatrix} 0 & 0.5 & 0.7 & 1 \\ 0 & 0.8 & 0.2 & 0 \\ 0.9 & 0 & 0 & 0.5 \\ 0 & 0.4 & 1 & 0 \\ 0 & 0.7 & 0 & 0.8 \\ 0.6 & 0 & 0.8 & 0 \end{bmatrix}$$

From now on, the set of all *fpfs*-matrices parameterized via E over U is denoted by $FPFS_E[U]$.

Secondly, we present the algorithm of EK15. Here, EK15 refers to the method constructed by Eraslan and Karaaslan in 2015. Throughout this paper, $I_n = \{1,2,3,\dots,n\}$ and $I_n^* = \{0,1,2,3,\dots,n\}$.

EK15's Algorithm Steps

Step 1. Construct *fpfs*-matrices $[a_{ij}^1]$, $[a_{ij}^2]$, ..., $[a_{ij}^t]$

Step 2. Obtain $[b_{ij}]$ defined by $b_{ij} := a_{0j}^i$ such that $i \in I_t$ and $j \in I_n$

Step 3. Obtain $[c_{ij}]$ defined by

$$c_{ij} := \frac{b_{ij}}{\sqrt{\sum_{k=1}^{t} {b_{kj}}^2}}, \qquad i \in I_t \text{ and } j \in I_n$$

Step 4. Obtain $[d_{i1}]$ defined by

$$d_{i1} := \frac{1}{t}\sum_{j=1}^{t} c_{ji}, \qquad i \in I_n$$

Step 5. Obtain $[e_{i1}]$ defined by

$$e_{i1} := \frac{d_{i1}}{\sum_{k=1}^{n} d_{k1}}, \qquad i \in I_n$$

Step 6. Obtain $[f_{ij}]$ defined by

$$f_{ij} := \frac{1}{t}\sum_{k=1}^{t} a_{ij}^k, \quad i \in I_{m-1} \text{ and } j \in I_n$$

Step 7. Obtain $[g_{ij}]$ defined by $g_{ij} := e_{j1} f_{ij}$ such that $i \in I_{m-1}$ and $j \in I_n$

Step 8. Obtain $[g_{1j}^+]$ and $[g_{1j}^-]$ defined by

$$g_{1j}^+ := \max_i\{g_{ij}\} \quad \text{and} \quad g_{1j}^- := \min_i\{g_{ij}\}, \qquad i \in I_{m-1} \text{ and } j \in I_n$$

Step 9. Obtain $[s_{i1}^{+}]$ and $[s_{i1}^{-}]$ defined by

$$s_{i1}^{+} \coloneqq \sqrt{\textstyle\sum_{j=1}^{n} (g_{ij} - g_{1j}^{+})^2} \quad \text{and} \quad s_{i1}^{-} \coloneqq \sqrt{\textstyle\sum_{j=1}^{n} (g_{ij} - g_{1j}^{-})^2}, \; i \in I_{m-1} \text{ and } j \in I_n$$

Step 10. Obtain $[s_{i1}]$ defined by

$$s_{i1} \coloneqq \frac{s_{i1}^{-}}{s_{i1}^{+} + s_{i1}^{-}}, \quad i \in I_{m-1}$$

Step 11. Obtain the set $\{u_k \in U \mid s_{k1} = \max_i s_{i1}\}$

3. The SDM Method: EMK19

TOPSIS (Hwang and Yoon, 1981) is a very efficient fuzzy multi-criteria decision-making method. Eraslan and Karaaslan (2015) have introduced an SDM method EK15 to transfer TOPSIS's ability to fuzzy soft sets. The first step of EK15 contains a normalisation that makes the norms of columns of the weight matrix are 1. This normalisation leads to changing the weights given by the decision-makers. To this end, in this section, we propose a new SDM method referred to as EMK19 and which is an improved version of EK15.

Example 3. 1. Assume that $\begin{bmatrix} 0.8 & 0.4 \\ 0.7 & 0.5 \end{bmatrix}$ is a weight matrix. Then, the normalised matrix of it is $\begin{bmatrix} 0.53 & 0.44 \\ 0.46 & 0.55 \end{bmatrix}$. Changing the weights provided by the decision-makers brings about faults. For more details, see Section 5.

The algorithm of EMK19 is as follows:

EMK19's Algorithm Steps

Step 1. Construct *fpfs*-matrices $[a_{ij}^{1}]$, $[a_{ij}^{2}]$, ..., $[a_{ij}^{t}]$

Step 2. Obtain $[b_{ij}]$ defined by

$$b_{ij} \coloneqq \frac{1}{t}\sum_{k=1}^{t} a_{ij}^{k}, \qquad i \in I_{m-1}^{*} \text{ and } j \in I_n$$

Step 3. Obtain $[c_{ij}]$ defined by $c_{ij} \coloneqq b_{01}b_{ij}$ such that $i \in I_{m-1}$ and $j \in I_n$

Step 4. Obtain the Positive Ideal Solution matrix $[c_{1j}^{+}]$ and Negative Ideal Solution matrix $[c_{1j}^{-}]$ defined by

$$c_{1j}^{+} \coloneqq \max_i\{c_{ij}\} \quad \text{and} \quad c_{1j}^{-} \coloneqq \min_i\{c_{ij}\}, \qquad i \in I_{m-1} \text{ and } j \in I_n$$

Step 5. Obtain $[s_{i1}^{+}]$ and $[s_{i1}^{-}]$ defined by

$$s_{i1}^{+} \coloneqq \sqrt{\textstyle\sum_{j=1}^{n} (c_{ij} - c_{1j}^{+})^2} \quad \text{and} \quad s_{i1}^{-} \coloneqq \sqrt{\textstyle\sum_{j=1}^{n} (c_{ij} - c_{1j}^{-})^2}, \; i \in I_{m-1} \text{ and } j \in I_n$$

Step 6. Obtain $[s_{i1}]$ defined by

$$s_{i1} \coloneqq \frac{s_{i1}^{-}}{s_{i1}^{+} + s_{i1}^{-}}, \qquad i \in I_{m-1}$$

Step 7. Obtain the set $\{u_k \in U \mid s_{k1} = \max_i s_{i1}\}$

4. Simulation Results

In this section, we compare the running time of EK15 and EMK19 by using MATLAB R2019b and a laptop with I(R) Core(TM) CPU i5-4200H @ 2.7 GHz and 8 GB RAM in this study. We present the running time of EK15 and EMK19 in Table 1 and Fig. 1 for ten objects and the parameters ranging from 10 to 100 and in Table 2 and Fig. 2 for ten objects and the parameters ranging from 1000 to 10000. The results show that EMK19 offers up 93.1230% and 97.7672% running time advantages over EK15, respectively.

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

	10	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
EK15	0.0259	0.0079	0.0019	0.0021	0.0106	0.0091	0.0065	0.0070	0.0077	0.0100
EMK19	0.0084	0.0041	0.0004	0.0006	0.0052	0.0018	0.0005	0.0005	0.0005	0.0016
Difference	0.0175	0.0038	0.0015	0.0014	0.0054	0.0074	0.0060	0.0065	0.0072	0.0084
Advantage (%)	67.381	47.7115	77.6380	69.5378	50.5395	80.5389	92.1688	92.9776	93.1230	84.4345

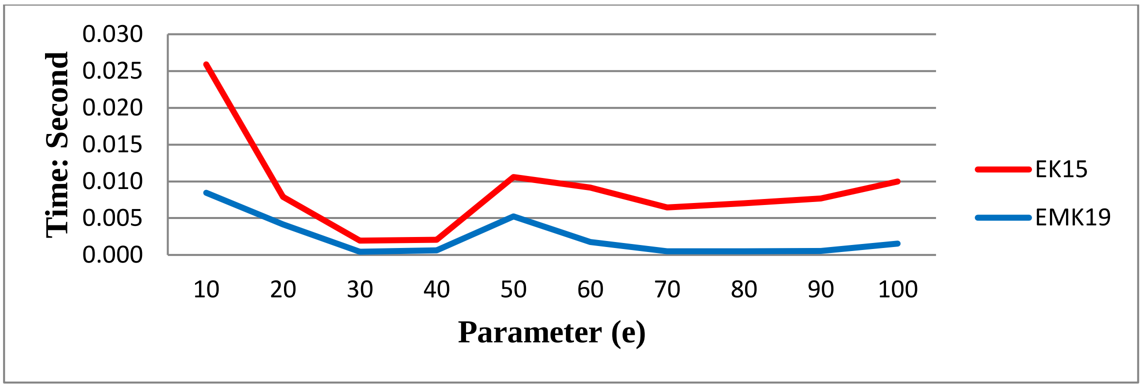

Fig. 1. The figure for Table 1

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

	1000	**2000**	**3000**	**4000**	**5000**	**6000**	**7000**	**8000**	**9000**	**10000**
EK15	0.1268	0.1318	0.1428	0.1790	0.3185	0.3367	0.3654	0.4763	0.5295	0.5478
EMK19	0.0172	0.0127	0.0047	0.0078	0.0103	0.0178	0.0091	0.0175	0.0118	0.0130
Difference	0.1096	0.1191	0.1380	0.1712	0.3082	0.3190	0.3563	0.4588	0.5177	0.5348
Advantage (%)	86.412	90.3990	96.6908	95.6312	96.7686	94.7260	97.5066	96.3297	97.7672	97.6232

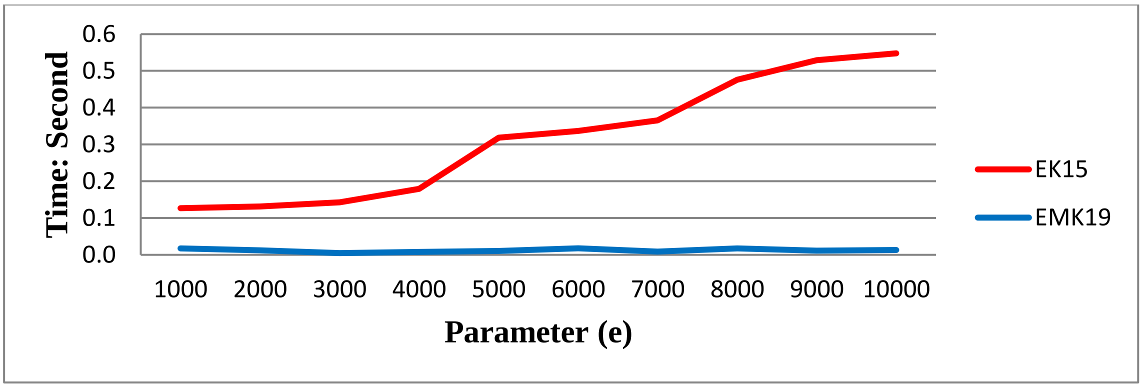

Fig. 2. The figure for Table 2

We then provide their running time in Table 3 and Fig. 3 for ten parameters and the objects ranging from 10 to 100 and in Table 4 and Fig. 4 for ten parameters and the objects ranging from 1000 to 10000. The results show that EMK19 offers up 97.3944% and 89.2337% running time advantages over EK15, respectively.

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
EK15	0.0265	0.0082	0.0019	0.0019	0.0108	0.0083	0.0077	0.0201	0.0076	0.0077
EMK19	0.0084	0.0042	0.0005	0.0006	0.0047	0.0019	0.0007	0.0005	0.0008	0.0006
Difference	0.0181	0.0039	0.0015	0.0013	0.0060	0.0064	0.0071	0.0196	0.0068	0.0071
Advantage (%)	68.213	48.0682	76.0231	66.6493	56.0540	77.4725	91.4512	97.3944	89.4101	92.3568

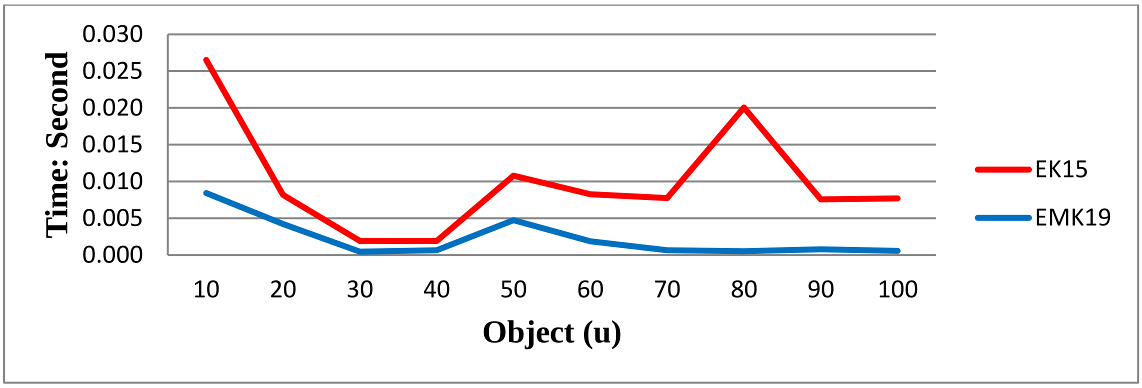

Fig. 3. The figure for Table 3

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
EK15	0.1246	0.1010	0.1890	0.1970	0.2970	0.4219	0.4762	0.5702	0.6538	0.7521
EMK19	0.0209	0.0223	0.0203	0.0349	0.0981	0.1037	0.1110	0.1446	0.1779	0.2207
Difference	0.1036	0.0787	0.1686	0.1621	0.1989	0.3182	0.3651	0.4256	0.4760	0.5314
Advantage (%)	83.1842	77.9321	89.2337	82.2880	66.9633	75.4141	76.6789	74.6398	72.7968	70.6547

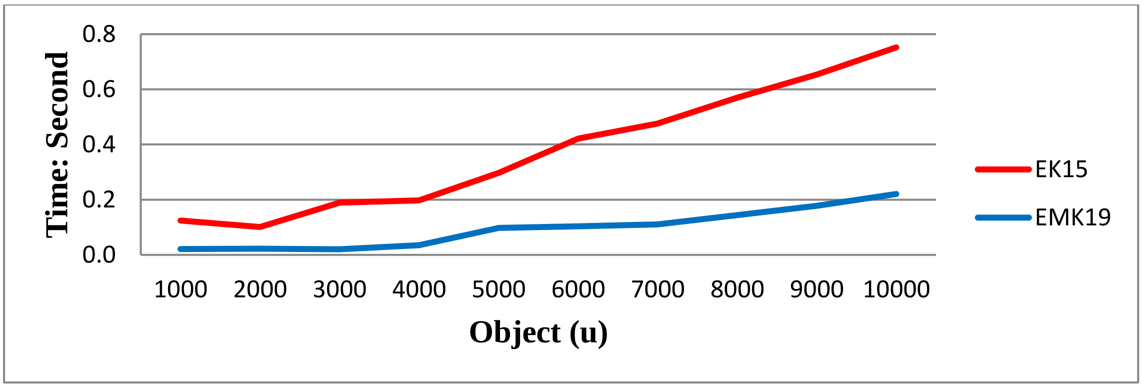

Fig. 4. The figure for Table 4

Finally, we provide the running time 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 EMK19 performs better than EK15 in any number of parameter.

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
EK15	0.0260	0.0094	0.0062	0.0144	0.0497	0.0290	0.0337	0.0282	0.0352	0.0512
EMK19	0.0090	0.0046	0.0011	0.0012	0.0038	0.0024	0.0008	0.0009	0.0019	0.0014
Difference	0.0170	0.0048	0.0051	0.0132	0.0459	0.0266	0.0330	0.0273	0.0333	0.0499
Advantage (%)	65.4306	50.6152	81.8680	91.5282	92.3300	91.7701	97.7494	96.7116	94.5352	97.3419

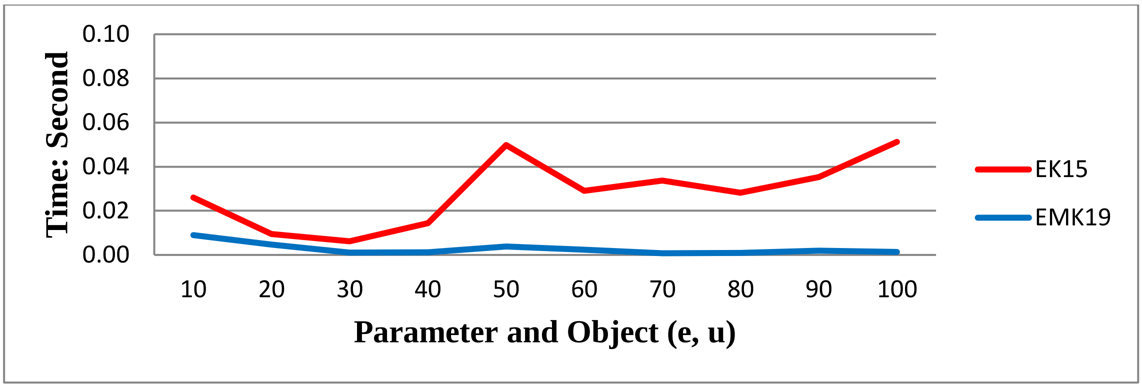

Fig. 5. The figure for Table 5

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
EK15	0.1096	0.2234	0.4419	0.7348	0.9994	1.4881	2.0419	2.8349	3.7299	5.2057
EMK19	0.0180	0.0101	0.0127	0.0231	0.0686	0.0879	0.1797	0.2726	0.4585	0.5438
Difference	0.0915	0.2133	0.4292	0.7117	0.9308	1.4002	1.8622	2.5623	3.2714	4.6620
Advantage (%)	83.5328	95.4762	97.1204	96.8567	93.1375	94.0954	91.1989	90.3841	87.7071	89.5547

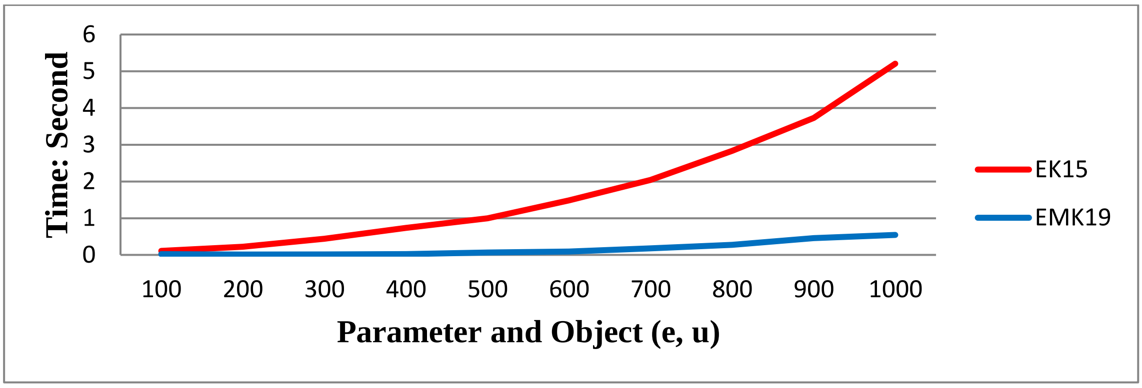

Fig. 6. The figure for Table 6

5. An Application of EMK19 to Performance-Based Value Assignment

In this section, we apply EMK19 to a performance-based value assignment problem for seven state-of-art images denoising filters concerning salt-and-pepper noise (SPN) removal performance by using the results provided in (Enginoğlu et al., 2019a). We present the results in Table 7-9 as follows:

Table 7. The mean-PSNR results for the 20 traditional images with different SPN ratios

Filters/Noise Density	10%	20%	30%	40%	50%	60%	70%	80%	90%
DBA	37.52	34.29	31.96	29.83	27.86	25.89	23.90	21.55	18.55
MDBUTMF	36.80	32.18	29.02	28.48	28.81	28.34	26.95	23.42	15.29
BPDF	36.98	33.54	31.03	28.88	26.82	24.60	21.98	17.74	10.51
NAFSMF	36.08	33.27	31.49	30.15	29.02	27.96	26.82	25.47	22.34
AWMF	36.34	35.00	33.83	32.69	31.47	30.14	28.68	26.99	24.70
DAMF	39.58	36.33	34.14	32.45	30.99	29.64	28.28	26.69	24.35
ARmF	**40.04**	**37.12**	**35.14**	**33.53**	**31.99**	**30.45**	**28.86**	**27.08**	**24.74**

Table 8. The mean-SSIM results for the 20 traditional images with different SPN ratios

Filters/Noise Density	10%	20%	30%	40%	50%	60%	70%	80%	90%
DBA	0.9796	0.9584	0.9315	0.8968	0.8520	0.7949	0.7213	0.6265	0.4966
MDBUTMF	0.9774	0.9197	0.8117	0.7973	0.8399	0.8410	0.8025	0.7023	0.3566
BPDF	0.9783	0.9536	0.9229	0.8838	0.8323	0.7634	0.6680	0.5096	0.2585
NAFSMF	0.9748	0.9504	0.9248	0.8973	0.8666	0.8320	0.7910	0.7357	0.6190
AWMF	0.9728	0.9622	0.9484	0.9315	0.9098	0.8816	0.8437	0.7904	0.7028
DAMF	0.9854	0.9699	0.9516	0.9303	0.9051	0.8748	0.8368	0.7846	0.6964
ARmF	**0.9868**	**0.9735**	**0.9581**	**0.9400**	**0.9173**	**0.8880**	**0.8491**	**0.7947**	**0.7056**

Table 9. The mean-VIF results for the 20 traditional images with different SPN ratios

Filters/Noise Density	10%	20%	30%	40%	50%	60%	70%	80%	90%
DBA	0.8548	0.7319	0.6179	0.5119	0.4095	0.3128	0.2229	0.1365	0.0635
MDBUTMF	0.8272	0.6713	0.5044	0.4420	0.4310	0.3978	0.3302	0.2212	0.0730
BPDF	0.8188	0.6858	0.5659	0.4564	0.3529	0.2541	0.1614	0.0783	0.0334
NAFSMF	0.7902	0.6751	0.5828	0.5030	0.4307	0.3604	0.2897	0.2129	0.1226
AWMF	0.7896	0.7366	0.6789	0.6181	0.5533	0.4833	0.4066	0.3129	0.1928
DAMF	0.8787	0.7816	0.6943	0.6162	0.5437	0.4731	0.3998	0.3096	0.1913
ARmF	**0.8832**	**0.7975**	**0.7210**	**0.6474**	**0.5741**	**0.4974**	**0.4158**	**0.3182**	**0.1955**

Assume that the success in high noise densities is more important than in the others. In that case, the values given in Table 7-9 can be represented with three *fpfs*-matrices as follows:

$$[a_{ij}] := \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\ 0.9371 & 0.8564 & 0.7982 & 0.7450 & 0.6958 & 0.6466 & 0.5969 & 0.5382 & 0.4633 \\ 0.9191 & 0.8037 & 0.7248 & 0.7113 & 0.7195 & 0.7078 & 0.6731 & 0.5849 & 0.3819 \\ 0.9236 & 0.8377 & 0.7750 & 0.7213 & 0.6698 & 0.6144 & 0.5490 & 0.4431 & 0.2625 \\ 0.9011 & 0.8309 & 0.7865 & 0.7530 & 0.7248 & 0.6983 & 0.6698 & 0.6361 & 0.5579 \\ 0.9076 & 0.8741 & 0.8449 & 0.8164 & 0.7860 & 0.7527 & 0.7163 & 0.6741 & 0.6169 \\ 0.9885 & 0.9073 & 0.8526 & 0.8104 & 0.7740 & 0.7403 & 0.7063 & 0.6666 & 0.6081 \\ 1.0000 & 0.9271 & 0.8776 & 0.8374 & 0.7990 & 0.7605 & 0.7208 & 0.6763 & 0.6179 \end{bmatrix}$$

$$[b_{ij}] := \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\ 0.9796 & 0.9584 & 0.9315 & 0.8968 & 0.8520 & 0.7949 & 0.7213 & 0.6265 & 0.4966 \\ 0.9774 & 0.9197 & 0.8117 & 0.7973 & 0.8399 & 0.8410 & 0.8025 & 0.7023 & 0.3566 \\ 0.9783 & 0.9536 & 0.9229 & 0.8838 & 0.8323 & 0.7634 & 0.6680 & 0.5096 & 0.2585 \\ 0.9748 & 0.9504 & 0.9248 & 0.8973 & 0.8666 & 0.8320 & 0.7910 & 0.7357 & 0.6190 \\ 0.9728 & 0.9622 & 0.9484 & 0.9315 & 0.9098 & 0.8816 & 0.8437 & 0.7904 & 0.7028 \\ 0.9854 & 0.9699 & 0.9516 & 0.9303 & 0.9051 & 0.8748 & 0.8368 & 0.7846 & 0.6964 \\ 0.9868 & 0.9735 & 0.9581 & 0.9400 & 0.9173 & 0.8880 & 0.8491 & 0.7947 & 0.7056 \end{bmatrix}$$

and

$$[c_{ij}] := \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\ 0.8548 & 0.7319 & 0.6179 & 0.5119 & 0.4095 & 0.3128 & 0.2229 & 0.1365 & 0.0635 \\ 0.8272 & 0.6713 & 0.5044 & 0.4420 & 0.4310 & 0.3978 & 0.3302 & 0.2212 & 0.0730 \\ 0.8188 & 0.6858 & 0.5659 & 0.4564 & 0.3529 & 0.2541 & 0.1614 & 0.0783 & 0.0334 \\ 0.7902 & 0.6751 & 0.5828 & 0.5030 & 0.4307 & 0.3604 & 0.2897 & 0.2129 & 0.1226 \\ 0.7896 & 0.7366 & 0.6789 & 0.6181 & 0.5533 & 0.4833 & 0.4066 & 0.3129 & 0.1928 \\ 0.8787 & 0.7816 & 0.6943 & 0.6162 & 0.5437 & 0.4731 & 0.3998 & 0.3096 & 0.1913 \\ 0.8832 & 0.7975 & 0.7210 & 0.6474 & 0.5741 & 0.4974 & 0.4158 & 0.3182 & 0.1955 \end{bmatrix}$$

Here, the entries of $[a_{ij}]$ except for its first row have been obtained by normalising via the maximum value provided in Table 7.

If we apply EMK19 to the *fpfs*-matrices $[a_{ij}]$, $[b_{ij}]$, and $[c_{ij}]$, then the score matrix and the decision set are as follows:

$$[s_{i1}] = [0.4089 \quad 0.4313 \quad 0.0633 \quad 0.6885 \quad 0.9547 \quad 0.9415 \quad 1]^T$$

and

$$\{{}^{0.4089}\text{DBA}, {}^{0.4313}\text{MDBUTMF}, {}^{0.0633}\text{BPDF}, {}^{0.6885}\text{NAFSMF}, {}^{0.9547}\text{AWMF}, {}^{0.9415}\text{DAMF}, {}^{1}\text{ARmF}\}$$

The scores show that ARmF outperforms the others and the ranking order BPDF≺DBA≺MDBUTMF ≺NAFSMF≺DAMF≺AWMF≺ARmF is valid.

Assume that the success in low noise densities is more important than in the others. In that case, the values given in Table 7-9 can be represented with three *fpfs*-matrices as follows:

$$[d_{ij}] \coloneqq \begin{bmatrix} 0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\ 0.9371 & 0.8564 & 0.7982 & 0.7450 & 0.6958 & 0.6466 & 0.5969 & 0.5382 & 0.4633 \\ 0.9191 & 0.8037 & 0.7248 & 0.7113 & 0.7195 & 0.7078 & 0.6731 & 0.5849 & 0.3819 \\ 0.9236 & 0.8377 & 0.7750 & 0.7213 & 0.6698 & 0.6144 & 0.5490 & 0.4431 & 0.2625 \\ 0.9011 & 0.8309 & 0.7865 & 0.7530 & 0.7248 & 0.6983 & 0.6698 & 0.6361 & 0.5579 \\ 0.9076 & 0.8741 & 0.8449 & 0.8164 & 0.7860 & 0.7527 & 0.7163 & 0.6741 & 0.6169 \\ 0.9885 & 0.9073 & 0.8526 & 0.8104 & 0.7740 & 0.7403 & 0.7063 & 0.6666 & 0.6081 \\ 1.0000 & 0.9271 & 0.8776 & 0.8374 & 0.7990 & 0.7605 & 0.7208 & 0.6763 & 0.6179 \end{bmatrix}$$

$$[e_{ij}] \coloneqq \begin{bmatrix} 0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\ 0.9796 & 0.9584 & 0.9315 & 0.8968 & 0.8520 & 0.7949 & 0.7213 & 0.6265 & 0.4966 \\ 0.9774 & 0.9197 & 0.8117 & 0.7973 & 0.8399 & 0.8410 & 0.8025 & 0.7023 & 0.3566 \\ 0.9783 & 0.9536 & 0.9229 & 0.8838 & 0.8323 & 0.7634 & 0.6680 & 0.5096 & 0.2585 \\ 0.9748 & 0.9504 & 0.9248 & 0.8973 & 0.8666 & 0.8320 & 0.7910 & 0.7357 & 0.6190 \\ 0.9728 & 0.9622 & 0.9484 & 0.9315 & 0.9098 & 0.8816 & 0.8437 & 0.7904 & 0.7028 \\ 0.9854 & 0.9699 & 0.9516 & 0.9303 & 0.9051 & 0.8748 & 0.8368 & 0.7846 & 0.6964 \\ 0.9868 & 0.9735 & 0.9581 & 0.9400 & 0.9173 & 0.8880 & 0.8491 & 0.7947 & 0.7056 \end{bmatrix}$$

and

$$[f_{ij}] \coloneqq \begin{bmatrix} 0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 & 0.3 & 0.2 & 0.1 \\ 0.8548 & 0.7319 & 0.6179 & 0.5119 & 0.4095 & 0.3128 & 0.2229 & 0.1365 & 0.0635 \\ 0.8272 & 0.6713 & 0.5044 & 0.4420 & 0.4310 & 0.3978 & 0.3302 & 0.2212 & 0.0730 \\ 0.8188 & 0.6858 & 0.5659 & 0.4564 & 0.3529 & 0.2541 & 0.1614 & 0.0783 & 0.0334 \\ 0.7902 & 0.6751 & 0.5828 & 0.5030 & 0.4307 & 0.3604 & 0.2897 & 0.2129 & 0.1226 \\ 0.7896 & 0.7366 & 0.6789 & 0.6181 & 0.5533 & 0.4833 & 0.4066 & 0.3129 & 0.1928 \\ 0.8787 & 0.7816 & 0.6943 & 0.6162 & 0.5437 & 0.4731 & 0.3998 & 0.3096 & 0.1913 \\ 0.8832 & 0.7975 & 0.7210 & 0.6474 & 0.5741 & 0.4974 & 0.4158 & 0.3182 & 0.1955 \end{bmatrix}$$

Here, the entries of $[d_{ij}]$ except for its first row have been obtained by normalising via the maximum value provided in Table 7.

If we apply EMK19 to the *fpfs*-matrices $[d_{ij}]$, $[e_{ij}]$, and $[f_{ij}]$, then the score matrix and the decision set are as follows:

$$[s_{i1}] = [0.4440 \quad 0.2796 \quad 0.2580 \quad 0.4321 \quad 0.7190 \quad 0.8822 \quad 1]^T$$

and

$$\{{}^{0.4440}\text{DBA}, {}^{0.2796}\text{MDBUTMF}, {}^{0.2580}\text{BPDF}, {}^{0.4321}\text{NAFSMF}, {}^{0.7190}\text{AWMF}, {}^{0.8822}\text{DAMF}, {}^{1}\text{ARmF}\}$$

The scores show that ARmF outperforms the others and the ranking order BPDF≺MDBUTMF≺NAFSMF≺DBA≺AWMF≺DAMF≺ARmF is valid.

The ranking orders obtained by EK15, EMK19, and expert's view are in Table 10. It must be noted that EK15 has produced the same ranking order in both cases. Besides, EK15 is not in compliance with the expert's view, yet EMK19 is. All results show that EMK19 outperforms EK15.

Table 10. The ranking orders of the filters for EK15, EMK19, and expert's view

Matrices	Algorithms	Ranking Orders
$[a_{ij}]$,$[b_{ij}]$,$[c_{ij}]$	EK15	BPDF≺MDBUTMF≺DBA≺AWMF≺NAFSMF≺DAMF≺ARmF
	EMK19	BPDF≺DBA≺MDBUTMF ≺NAFSMF≺DAMF≺AWMF≺ARmF
	Expert's View	BPDF≺DBA≺MDBUTMF ≺NAFSMF≺DAMF≺AWMF≺ARmF
$[d_{ij}]$, $[e_{ij}]$,$[f_{ij}]$	EK15	BPDF≺MDBUTMF≺DBA≺AWMF≺NAFSMF≺DAMF≺ARmF
	EMK19	BPDF≺MDBUTMF≺NAFSMF≺DBA≺AWMF≺DAMF≺ARmF
	Expert's View	BPDF≺MDBUTMF≺NAFSMF≺DBA≺AWMF≺DAMF≺ARmF

6. Conclusion

The group decision-making method based on TOPSIS under fuzzy soft environment was defined in 2015 (Eraslan and Karaaslan, 2015). Afterwards, this method configured (Enginoğlu and Memiş, 2018a) via *fpfs*-matrices (Enginoğlu, 2012, Enginoğlu and Çağman, n.d.). However, the normalisation step in the EK15 leads to faults in the ranking order. To overcome this problem, in this study, we proposed a new SDM methods EMK19. We then compared the running time of EMK19 and EK15. In addition to the results in Section 4, the results in Table 11 show that EMK19 outperforms EK15 in any number of data. Finally, we applied EMK19 to the performance-based value assignment problem for the filters used in (Enginoğlu et al. 2019a). EMK19 can be successfully applied to such problem many areas, such as in machine learning and image enhancement.

Table 11. The mean/max advantage and max difference values of EMK19 over EK15

Location	Objects	Parameters	Mean Advantage %	Max Advantage %	Max Difference
Table 1	10	10-100	75.6051	93.1230	0.0175
Table 2	10	1000-10000	94.9855	97.7672	0.5348
Table 3	10-100	10	76.3093	97.3944	0.0196
Table 4	1000-10000	10	76.9786	89.2337	0.5314
Table 5	10-100	10-100	85.9880	97.7494	0.0499
Table 6	100-1000	100-1000	91.9064	97.1204	4.6620

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.

References

Çağman, N., Enginoğlu, S., 2010a. Soft Set Theory and Uni-Int Decision Making. European Journal of Operational Research, 207(2): 848-855.

Çağman, N., Enginoğlu, S., 2010b. Soft Matrix Theory and Its Decision Making. Computer and Mathematics with Applications, 59(10): 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., 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., 2011a. FP-Soft Set Theory and Its Applications. Annals of Fuzzy Mathematics and Informatics, 2(2): 219-226.

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

Deli, İ., Çağman, N., 2015. Relations on FP-Soft Sets Applied to Decision Making Problems. Journal of New Theory, 3: 98-107.

Enginoğlu, S., 2012. Soft Matrices. PhD Dissertation. *Gaziosmanpaşa University. Graduate School of Natural and Applied Sciences, Department of Mathematics, Tokat, Turkey (In Turkish).*

Enginoğlu, S., Çağman, N., n.d. Fuzzy Parameterized Fuzzy Soft Matrices and Their Application in Decision-Making. TWMS Journal of Applied and Engineering Mathematics, In Press.

Enginoğlu, S., Erkan, U., Memiş, S., 2019a. Pixel Similarity-Based Adaptive Riesz Mean Filter for Salt-and-Pepper Noise Removal. Multimedia Tools and Applications, 78(24): 35401-35418.

Enginoğlu, S., Memiş, S., 2018a. A Configuration of Some Soft Decision-Making Algorithms via *fpfs*-matrices. Cumhuriyet Science Journal, 39(4): 871-881.

Enginoğlu, S., Memiş, S., 2018b. Comment on "Fuzzy soft sets" [The Journal of Fuzzy Mathematics, 9(3), 2001, 589–602]. International Journal of Latest Engineering Research and Applications, 3(9): 1-9.

Enginoğlu, S., Memiş, S., 2018c. A Review on An Application of Fuzzy Soft Set in Multicriteria Decision Making Problem [P.K. Das, R. Borgohain, International Journal of Computer Applications 38 (12) (2012) 33-37]. Eds: Akgül, M., Yılmaz, İ., İpek, A. International Conference on Mathematical Studies and Applications, pp. 173-178, October 4-6, Karaman, Turkey.

Enginoğlu, S., Memiş, S., Arslan, B., 2018a. Comment (2) on Soft Set Theory and uni-int Decision Making [European Journal of Operational Research, (2010) 207, 848-855]. Journal of New Theory, 25: 85-102.

Enginoğlu, S., Memiş, S., Çağman, N., 2019b. A Generalisation of Fuzzy Soft Max-Min Decision-Making Method and Its Application to A Performance-Based Value Assignment in Image Denoising. El-Cezerî Journal of Science and Engineering, 6(3): 466-481.

Enginoğlu, S., Memiş, S., Öngel, T., 2018b. Comment on Soft Set Theory and Uni-Int Decision Making [European Journal of Operational Research, (2010) 207, 848-855]. Journal of New Results in Science, 7(3): 28-43.

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

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

Hwang, C.L., Yoon, K., 1981. Multiple Attribute Decision Making: Methods and applications, Springer-Verlag, New York.

Karaaslan, F., 2016. Soft Classes and Soft Rough Classes with Applications in Decision Making. Mathematical Problems in Engineering, Article ID1584528, 11 pages.

Karaaslan, F., Çağman, N., Enginoğlu, S., 2012. Soft Lattices. Journal of New Results in Sciences, 1(1): 5-17.

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

Maji, P.K., Biswas, R., Roy, A.R., 2003. Soft Set Theory. Computer and Mathematics with Applications, 45(4-5): 555-562.

Molodtsov, D., 1999. Soft Set Theory-First Results. Computers and Mathematics with Applications, 37(4-5): 19-31.

Razak, S.A., Mohamad, D., 2011. A Soft Set Based Group Decision Making Method with Criteria Weight. International Scholarly and Scientific Research & Innovation, 5(10): 1641-1646.

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

Riaz, M., Hashmi, R., 2017. Fuzzy Parameterized Fuzzy Soft Topology with Applications. Annals of Fuzzy Mathematics and Informatics, 13(5): 593-613.

Riaz, M., Hashmi, R., 2018. Fuzzy Parameterized Fuzzy Soft Compact Spaces with Decision-Making, Punjab University Journal of Mathematics, 50(2): 131-145.

Riaz, M., Hashmi, R., Farooq, A., 2018. Fuzzy Parameterized Fuzzy Soft Metric Spaces. Journal of Mathematical Analysis, 9(2): 25-36.

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.

Şenel, G., 2016. A New Approach to Hausdorff Space Theory via the Soft Sets. Mathematical Problems in Engineering, Article ID 2196743, 6 pages.

Şenel, G., 2017. The Parameterization Reduction of Soft Point and its Applications with Soft Matrix. International Journal of Computer Applications, 164(1): 1-6.

Şenel, G., 2018. Analyzing The Locus of Soft Spheres: Illustrative Cases and Drawings. European Journal of Pure and Applied Mathematics, 11(4): 946-957.

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

Zorlutuna, İ., Atmaca, S., 2016. Fuzzy Parametrized Fuzzy Soft Topology. New Trends in Mathematical Sciences, 4(1): 142-152.