Article

Interactive software for classification and ranking procedures based on multi-criteria decision-making algorithms

M. Tlas, B. Abdul Ghani

Scientific Services Department, Atomic Energy Commission of Syria (AECS), P. O. Box 6091, Syria E-mail: pscientific31@aec.org.sy

Received 11 May 2020; Accepted 20 June 2020; Published 1 September 2020

Abstract

Interactive software that helps the decision maker to choose the optimal ranking for complex systems has been prepared according to multiple preference criteria. The code algorithm is based on famous mathematical methods well known in multi-criteria decision making, such as VIKOR, TOPSIS, and the weighted sum method (WSUM). This code is used at the AECS to assess and classify the performance of employees or for inter-departmental comparison according to their scientific and technical output performance. It can also be used in wider fields outside the AECS according to preference criteria carefully selected by the decision-maker. Beneficiaries might include Syrian private and state universities or academics. Criteria elements can, for example, be drafted along international standards of copyrighted internal reports, published papers, journal impact factor, journal citations, H-index and registered patents. Two real examples are explained in this paper to prove the validity and consistency of this computer code.

Keywords ranking and classification methods; multi-criteria optimization; multi-criteria decision-making; multi-objective optimization; multi-criteria decision aid.

Computational Ecology and Software ISSN 2220-721X URL: http://www.iaees.org/publications/journals/ces/online-version.asp RSS: http://www.iaees.org/publications/journals/ces/rss.xml E-mail: ces@iaees.org Editor-in-Chief: WenJun Zhang Publisher: International Academy of Ecology and Environmental Sciences

1 Introduction

Multi-criteria decision making (MCDM), multi-objective decision making (MODM), multi-criteria optimization (MCO), multi-objective optimization (MOO), multi-criteria decision aid (MCDA) or multi-criteria decision analysis(MCDA) are considered as the process of determining the best feasible solution according to established criteria which represent different effects. However, these criteria are usually conflicted and there may be contradictory with each other criterion which that implies to no feasible solution satisfying all criteria simultaneously at the time. Thus, the concept of Pareto optimality was introduced for a vector optimization problem. Pareto optimal solutions have the characteristic that, if one criterion is to be improved, at least one other criterion has to be made worse. In such cases, a system analyst can aid the decision making process by making a comprehensive analysis and by listing the important properties of the Pareto optimal non-inferior solutions.

However, in engineering, sustainable renewable energy development, Energy planning and management practice (Kumar et al., 2017; IAEA, 2019), it is often requested to select a final feasible solution to be recommended. An approach to determine a final feasible solution as a compromise was introduced by Yu (1973), and other distance-based techniques have also been developed by Chen and Hwang (1992), Ferrarini (2011, 2012) and Zhang et al. (2017). Three multi-criteria methods VIKOR, TOPSIS and the weighted sum (WSUM) are presented and implemented in this paper.

The VIKOR method was presented by Opricovic and Tzeng (2004) as a multi-criteria decision making system to solve discrete decision making problems with non-commensurable and conflicting criteria. This method deals with ranking and selecting from a set of alternatives, in order to determine compromise solutions for a problem with conflicting criteria. This method can aid decision makers to reach a final feasible decision. Thus, the compromise solution is a feasible one which is the closest to the ideal, and a compromise reflects an agreement established by mutual concessions. The area of application of VIKOR method is mechanical engineering, manufacturing engineering, energy policy, business management, and medicine and health.

However, another distance-based method is called TOPSIS (Hwang and Yoon, 1981). It aims at determining a feasible solution with the shortest distance from the ideal solution as well as the farthest distance from the negative-ideal solution. TOPSIS on the other hand does not consider the relative importance of these distances. The area of application of TOPSIS method is logistics, water resource management, energy management, and chemical engineering.

The WSUM method is considered as the simplest procedure in multi-criteria decision making for ranking and classification. The area of application of WSUM method is structural optimization and energy planning. The three methods of MCDM are implemented and programmed in this computer code VIKOR, TOPSIS and the weighted sum method (WSUM). The VIKOR method is based on closeness to a specified ideal solution and it is guided by weighting set and by the decision maker. The TOPSIS method is based on choosing the alternative which has the shortest distance from the ideal solution as well as the farthest distance from the negative-ideal solution. The well-known weighted sum method (WSUM) operates directly on weights specified by the decision maker. Using the above mentioned three methods may help and aid the decision maker to make the best decision and to have the clear view of the alternatives.

2 VIKOR Method

The VIKOR method was presented and developed by Opricovic and Tzeng (2002) to solve the following multi-criteria optimization problem:

Optimize
$$\{ (C_j(A_i), i = 1, ..., m), j = 1, ..., n \},$$

where *m* is the number of feasible alternatives; $A_i = \{x_1, x_2, ...\}$ is the *i*th alternative obtained or generated with certain values of system variables *x*; $f_{ij} = C_j(A_i) i = 1, ..., m; j = 1, ..., n$ is the value of the *j*th criterion function for the alternatives A_i ; *n* is the number of criteria; *optimize* denotes the operator of a multi-criteria decision making procedure for selecting the best (compromise) alternative in multi-criteria sense. Values $f_{ij} = C_j(A_i) i = 1, ..., m; j = 1, ..., n$ forms a matrix with *m* rows represent the alternatives $A_i (i = 1, ..., m)$ and *n* columns represent the criterion functions $C_j (j = 1, ..., n)$, this matrix is called the decision matrix.

Assuming that each alternative is evaluated according to all criterions, the compromise based ranking could be performed by comparing the measure of closeness to the ideal solution F^* (is the best values of criterions). The compromise ranking algorithm VIKOR has the coming steps:

Step 1: Determine the best value (ideal) f_i^* and the worst value (nadir) f_i^- of all criterion functions

$$C_{j}(j = 1,...,n)$$

 $f_j^* = \max_i f_{ij}, \quad f_j^- = \min_i f_{ij}$, if the *j*th criterion function represents a benefit (maximization) $f_j^* = \min_i f_{ij}, \quad f_j^- = \max_i f_{ij}$, if the *j*th criterion function represents a cost (minimization)

Step 2: Compute the following quantities for all alternatives A_i (i = 1, ..., m) by the relations:

$$S_{i} = \sum_{j=1}^{n} \frac{w_{j} \left(f_{j}^{*} - f_{ij}\right)}{\left(f_{j}^{*} - f_{j}^{-}\right)}, R_{i} = \max_{j} \frac{w_{j} \left(f_{j}^{*} - f_{ij}\right)}{\left(f_{j}^{*} - f_{j}^{-}\right)}, \quad (j = 1, ..., n)$$

where w_j (j = 1,...,n) is a positive number, it is the weight of the *j*th criterion function C_j (j = 1,...,n), expressing the decision maker preference as the relative importance of the criterion, $\sum_{j=1}^{n} w_j = 1$ (normalized).

Step 3: Compute the following quantities for all alternatives A_i (i = 1, ..., m) by the relations:

$$S^{*} = \min_{i} S_{i}, S^{-} = \max_{i} S_{i}, R^{*} = \min_{i} R_{i}, R^{-} = \max_{i} R_{i}, (i = 1, ..., m)$$
$$Q_{i} = v \frac{(S_{i} - S^{*})}{(S^{-} - S^{*})} + (1 - v) \frac{(R_{i} - R^{*})}{(R^{-} - R^{*})}, (i = 1, ..., m)$$

where $\nu \in [0,1]$ is introduced as a strategy weight of the majority of criteria or the maximum group utility, it can be chosen arbitrary but it is preferred to be 0.5.

Step 4: Rank the alternatives A_i (i = 1, ..., m), sorting the values S_i, R_i , and Q_i (i = 1, ..., m) in decreasing order. The results are three ranking lists.

Step 5: Suggest as a compromise solution the alternative which is the best ranked by the measure Q minimum or (1-Q) maximum.

3 TOPSIS Method

The TOPSIS method is presented by Chen and Hwang (1992), with reference to Hwang and Yoon (1981). The basic principal is that the chosen alternative should have the shortest distance from the ideal solution as well as the farthest distance from the negative-ideal solution (Zhang et al., 2017).

The TOPSIS procedure is composed from the coming steps:

Step 1: Calculate the normalized decision matrix. The normalized values r_{ij} (i = 1, ..., m; j = 1, ..., n) are calculated as:

$$r_{ij} = \frac{f_{ij}}{\sqrt{\sum_{i=1}^{m} f_{ij}^{2}}} (i = 1, ..., m \& j = 1, ..., n)$$

Step 2: Calculate the weighted normalized decision matrix. The weighted normalized values v_{ij} (i = 1, ..., m; j = 1, ..., n) are calculated as: $v_{ij} = w_j r_{ij}$ (i = 1, ..., m; j = 1, ..., n)

where $w_j (j = 1,...,n)$ is a positive number it is the weight of the *j*th criterion function $C_j (j = 1,...,n)$, $\sum_{i=1}^{n} w_j = 1$. Step 3: Obtain the ideal and the negative-ideal solutions as:

$$A^{*} = \left\{ v_{1}^{*}, ..., v_{n}^{*} \right\} = \left\{ \left(\max_{i} v_{ij} / j \in I^{'} \right), \left(\min_{i} v_{ij} / j \in I^{''} \right) \right\}$$
$$A^{-} = \left\{ v_{1}^{-}, ..., v_{n}^{-} \right\} = \left\{ \left(\min_{i} v_{ij} / j \in I^{''} \right), \left(\max_{i} v_{ij} / j \in I^{''} \right) \right\}$$

where I' is associated with benefit criteria (maximization), and I'' is associated with cost criteria (minimization).

Step 4: Obtain the separation measures, using the *n*-dimensional Euclidean distance. The separation of each alternative A_i (i = 1, ..., m) from the ideal solution is given as:

$$D_{i}^{*} = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_{j}^{*} \right)^{2}} (i = 1, ..., m)$$

Similarly, the separation from the negative-ideal solution is given as:

$$D_i^{-} = \sqrt{\sum_{j=1}^n \left(v_{ij} - v_j^{-} \right)^2} \ (i = 1, ..., m)$$

Step 5: Compute the relative closeness to the ideal solution. The relative closeness of the alternative A_i (i = 1, ..., m) with respect to A^* is defined as:

$$C_{i}^{*} = \frac{D_{i}^{-}}{\left(D_{i}^{*} + D_{i}^{-}\right)} (i = 1, ..., m)$$

Step 6: The alternative which is the best ranked by the measure C^* maximum or $(1-C^*)$ minimum.

4 Weighted Sum Method (WSUM)

The WSUM method is considered as the simplest procedure in multi-criteria decision making for ranking and classification (Zhang et al., 2017). It is simply consisted of the next steps:

Step 1: calculate the following quantities for all alternatives A_i (i = 1, ..., m) by the relations:

$$\theta_i = \sum_{j=1}^n w_j f_{ij} \quad (i = 1, ..., m)$$

where w_i (j = 1,...,n) is a positive number it is the weight of the *i*th criterion function C_i (j = 1,...,n),

$$\sum_{j=1}^{n} w_j = 1$$

Step 2: The alternative which is the best ranked by the measure θ maximum.

For more detailed explanation and for the complete version of these methods, readers are invited to see and consult the references (Opricovic and Tzeng, 2004, 2007; Ferrarini, 2012; Zhang et al., 2017).

www.iaees.org

5 Interactive Computer Code Algorithm

This interactive computer code is implemented by C-Sharp programming language and Dot-Net version 4 and more (see supplementary material). The code algorithm is composed from the next steps:

Step 1: Construct the decision matrix with m rows represent the alternatives A_i (i = 1, ..., m) and n columns

represents the criterions C_i (j = 1, ..., n) as depicted in Fig. 1.

	C_1	C_2		C_n
A _l	f_{11}	f_{12}		f_{1n}
A_2	f_{21}	f_{22}		f_{2n}
A_m	f_{m1}	f_{m2}		f_{mn}
	W =	$[w_1, w_2,$	$, w_n]$	

Fig. 1 Decision matrix.

Dimesions (Criteria * Alternatives*V):
8*5*0.5
Weights:
0.073
0.172
0.073
0.280
0.026
0.201
0.035
0.141
Criteria:
23 23 33 0.482 0.374 2.9 0.022 0.281
54 115 122 0.063 0.104 15.18 0.024 0.002
11 40 34 0.255 0.044 2.97 0.04 0.171
75 98 119 0.1 0.374 2.65 0.02 0
32 36 19 0.1 0.104 2.87 0.04 0.263

Fig. 2 Input text file.

There are three ways to build the decision matrix explained as follows:

(1) First way, it can construct this matrix by hand, where the code requests the user to enter the number of criterion (*n*), the number of alternatives (*m*), and the factor value of $v \in [0,1]$ in suitable bins, after that the code requests again to enter the criterions C_j (j = 1,...,n) with their normalized weights w_j (j = 1,...,n) and alternatives A_i (i = 1,...,m). After that the code shows empty bins to fill by hand the values of criterions

 C_j (j = 1,...,n) at alternatives A_i (i = 1,...,m).

(2) Second way, it can import the decision matrix from text-file as described in Fig. 2, where we put the number of criterions $(n)^*$ number of alternatives $(m)^*$ value of $v \in [0,1]$ separated by stars in second line. Weights of criterions will be written starting from the third line (one weight by one line). Separate between weights and decision matrix by the word "criteria".

(3) Third way, it can import the decision matrix from Excel-sheet as described in Fig. 3.

	Α	В	С	D	Е	F	G	Н		J	K	L
1	Criteria	Alternatives	۷	1	C1	C2	C3	C4	C5	C6	C7	C8
2	8	10	0.5	Weights	0.25	0.25	0.25	0.05	0.05	0.05	0.05	0.05
3				A1	21.807	88.795	51	1.205	56.477	1.457	5.2	8.299
4				A2	14.7	37.355	35	0.675	25	1.346	3.923	7.88
5				A 3	21.905	27.925	30	1.007	18.689	1.875	9.5	7.928
6				A4	7.908	18.784	24	0.452	12.356	0.706	1.941	7.529
7				A 5	13.979	13.695	21	0.843	12.035	0.913	4.217	6.361
8				A 6	7.189	17.919	23	0.522	14.965	0.92	2.4	7.316
9				A 7	9.012	17.564	17	0.738	8.388	0.739	2.565	6.905
10				A 8	12.102	3.363	25	0.669	2.139	0.833	3.367	8.003
11				A9	6.11	20.993	10	0.93	18.651	0.385	1.538	6.098
12				A10	3.17	21.671	10	0.146	14.503	0.769	1.692	6.675
13												

Fig. 3 Input Excel file.

Enter the number of criterions n in bin A2, the number of alternatives m in bin B2 and the value of $\nu \in [0,1]$ in bin C2.

Enter the names of criterions C_i (j = 1, ..., n) in line 1, starting from bin E1.

Enter the names of alternatives A_i (i = 1, ..., m) in colon D, starting from bin D3.

Enter the normalized weights w_i (j = 1, ..., n) of criterions in line 3, starting from bin E2.

Enter the values of decision matrix in empty bins as displayed in Fig. 3. Import the constructed decision matrix to get the table as clarified in Fig. 4.

	C1	C2	C3	C4	C5	C6	C7	C8
Wi	0.073	0.172	0.073	0.28	0.026	0.201	0.035	0.14
A1	23	23	33	0.482	0.374	2.9	0.022	0.28
A2	54	115	122	0.063	0.104	15.18	0.024	0.00
A3	11	40	34	0.255	0.044	2.97	0.04	0.17
A4	75	98	119	0.1	0.374	2.65	0.02	0
A5	32	36	19	0.1	0.104	2.87	0.04	0.26

Fig. 4 Decision Matrix.

We classify the criterions as benefit function in green color and cost function in red color, we change between them by pressing on the name of criterion to change the color.

The largest matrix can be constructed one hundred of criterions and one thousand of alternatives.

Step 2: Construct the criterions C_j (j = 1, ..., n)

The code requests the user to enter the name and a detail description of each criterion C_{j} (j = 1, ..., n). We

can enter the criterion and the full description by hand one by one, or we can import them from text-file each criterion written on one line only. The user can save these criterions in text-file to re-use them other times as seen in Fig. 5.

Decision Matrix	Criteria	Alternatives	Preference Factors	Ranking	VIKOR Chart	TOPSIS Chart
	Criteria Ac Cr ur	a Table dd Criteria be iteria from te normalised y	elow with weights r ext file. You can en weights.	manually, ter norm	, or load alised or	
	A	dd descriptior	n for each criterion :		Weight	
	C1	:			0.05	
			Add Criterion	A	dd Weight	
		R	emove Criterion	Ren	nove Weight	
	C	1:Quantity of 2:Quality of r	research esearch		0.25 0.25	
	C	3:Distribution	of citations		0.25	
	C4	4:No.Docume	ents/No.Authors		0.05	
	C	5:H-Index/No	. ofpublications ve	ars	0.05	
	C	7:No.Docume	ents/No.publication	years	0.05	
	C	8:References	5		0.05	
		oad Criteria	Save As TXT		Clear Criteria	

Fig. 5 Input of criterions.

Step 3: Construct the alternatives A_i (i = 1, ..., m)

The code requests the user to enter the name and a detail description of each alternative A_i (i = 1, ..., m). We

can enter the alternatives and the full description by hand one by one, or we can import them from text-file each alternative written on one line only. The user can save these alternatives in text-file to re-use them other times as seen in Fig. 6.

If you fi	II the Decision Matrix manually, press Show Decision Matrix below after filling the tables. Alternatives Table
	Add description for each Alternative :
	Add Alternative Remove Alternative A1:MuyldermansSerge V. A2:Ward Kenneth D.
	A3:Maziak Wasim A4:Thomas RichardJames A5:Izzat Mohammad Bashar A6:AI-Masri Mohammad Saied A7:Baum Michael A8:Allaf Abdul Wahab A9:Mohammad Yousser K.
	A10:Singh Murari K.

Fig. 6 Input of alternatives.

Note 1: when filling the decision matrix manually by hand, we should press on "show Decision Matrix" to show the decision matrix.

Step 4: Processing

We press on "Process" the code starts to compute the specific preference coefficients for each classification algorithms VIKOR, TOPSIS, and WSUM as appeared in Fig. 7.



Fig. 7 Preference coefficients.

Step 5: Ranking and Classification

We press on "Rank" the code starts to compute the ranking of alternatives from the best to the worst for each classification algorithms VIKOR, TOPSIS, and WSUM as represented in Fig. 8.

Decision Matrix	Criteria Alternative	Preference Factors	Ranking	VIKOR Chart	TOPSIS Chart
		The Best Alternativ Descending Orde	ve in er:		
	VIKOR	TOPSIS		WSUM	
	A1 0 A2 0.51 A3 0.52 A5 0.79 A4 0.8 A6 0.81 A7 0.82 A8 0.87 A9 0.97 A10 1	A1 0.938 A2 0.485 A3 0.485 A5 0.28 A8 0.242 A4 0.234 A6 0.219 A7 0.204 A9 0.178 A10 0.158	A1 A2 A3 A4 A5 A6 A7 A8 A9 A10	44.03 23.7 21.91 13.82 13.39 13.33 11.86 10.87 10.66 9.9	
Save	, e Ranking Results As T	XT File	Save R	lankig Results As	s docx File

Fig. 8 Ranking results.

Step 6: Exportation of Ranking and Classification

We can save the ranking results in word file or in text file, including, a simple explanation of ranking algorithms, criterions table with their weights and classification table due to ranking algorithms.

Step 7: Drawing charts of results

We can draw charts; the columns represent the alternatives versus and against their ranking due to the three ranking algorithms as depicted in Fig. 9, 10 and 11.



Fig. 9 VIKOR chart – alternatives A_i (i = 1, ..., m) versus the preference coefficient $(1 - Q_i)(i = 1, ..., m)$ maximum.



Fig. 10 TOPSIS chart-alternatives A_i (i = 1, ..., m) versus the preference coefficient C_i^* (i = 1, ..., m) maximum.



Fig. 11 WSUM chart-alternatives A_i (i = 1, ..., m) versus the preference coefficient θ_i (i = 1, ..., m) maximum.

We can save these charts as images with index "JEPG" on computer.

Note 2: If the number of alternatives is huge, in this case, the code permits to apply the "Zoom in" on a fixed zone of the chart to maximize or minimize it.

6 Testing Validity and Accuracy of Computer Code

Two problems of multi-criteria decision aid are treated in this work to prove the validity and accuracy of this code

Problem 1: Three-storey reinforced concrete building designed to be representative of pre-seismic code constructions in southern Europe (Caterino el al., 2009).

The alternatives and theirs descriptions are explained in Table 1.

Symbols of Alternatives	Description of alternatives
A_1	Confinement by glass fiber reinforced polymers
A_2	Steel bracing
A_3	Concrete jacketing of columns
A_4	Base isolation
A_5	Adding passive viscous dampers

Table 1 Alternatives.

The multi-criteria and weights are listed in Table 2.

Symbols of criterions	Description of criterions	Weight of criterions
C_1	Installation cost	0.073
C 2	Maintenance cost	0.172
<i>C</i> 3	Duration of works/disruption of use	0.073
C_4	Functional compatibility	0.280
C 5	Skilled labor requirements/needed technology level	0.026
C 6	Significance of the needed intervention at foundations	0.201
<i>C</i> ₇	Significant damage risk	0.035
C_8	Damage limitation risk	0.141

	Table	2	Eval	luation	criteria	1.
--	-------	---	------	---------	----------	----

The values of all criterions at all alternatives are shown in Table 3

	$C_1(\epsilon)$	$C_{2}\left(\in ight)$	$C_3(\mathrm{days})$	C_4	C_5	C_{6}	C_7	C_8
A_1	23	23	33	0.482	0.374	2.90	0.022	0.281
A_2	54	115	122	0.063	0.104	15.18	0.024	0.002
A_3	11	40	34	0.255	0.044	2.97	0.040	0.171
A_4	75	98	119	0.100	0.374	2.65	0.020	0.000
A_5	32	36	19	0.100	0.104	2.87	0.040	0.263

Table 3 Decision matrix.

The results driven by the method is listed in Table 4.

www.iaees.org

A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467
A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A1 0 0.745 A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467
A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A2 1 0.254 A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467
A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467	A3 0.142 0.619 A4 0.722 0.461 A5 0.649 0.467
A4 0.722 0.461 A5 0.649 0.467	A4 0.722 0.461 A5 0.649 0.467	A4 0.722 0.461 A5 0.649 0.467
A5 0.649 0.467	A5 0.649 0.467	A5 0.649 0.467

Table 4 Results driven by the code to methods VIKOR and TOPSIS.

Problem 2: Selection of facility location of a label production company in Turkey (Dag and Onder). The alternatives of this problem are the names of suggested cites situated in Turkey as in Table 5.

Table 5 Alternatives.							
Symbols of Alternatives	Names of Cites						
A_1	Avcilar						
A_2	Cerkezkoy						
A_3	Hadimkoy						
A_4	Ikitelli						
A_5	Tuzla						

Table 5 Alt

The multi-criteria and weights are given in Table 6.

Symbols of criterions	Symbols of criterions Description of criterions			
C_1	Raw material supply	0.112		
<i>C</i> ₂	Proximity to customer	0.028		
<i>C</i> ₃	Proximity to airport	0.019		
C_4	Proximity to harbor	0.024		
<i>C</i> ₅	Transportation cost	0.090		
C_{6}	Availability of skill labor	0.266		
<i>C</i> ₇	Labor cost	0.155		
C_8	Proximity to industrial zone	0.063		
<i>C</i> ₉	Government facilities	0.051		
C ₁₀	Construction cost(investment cost)	0.193		

 Table 6 Evaluation criteria.

The values of all criterions at all alternatives are represented in Table 7.

	C_1	C_{2}	C_3	C_4	C_5	C_{6}	<i>C</i> ₇	C_8	<i>C</i> ₉	C_{10}
A_1	5	3	19.7	8.7	1	3	1500	1	1	7
A_2	8	5	101	96.4	5	9	800	6	10	1
A_3	7	2	43.7	23.6	2	7	1200	2	4	6
A_4	6	1	13.6	24.8	4	10	1200	3	2	5
A_5	10	7	7.6	12.8	2	9	800	1	7	9

Table 7 Decision matrix.

The results driven by the method is listed in Table 8.

Q.VIKOR A1 1 A2 0 A3 0.288 A4 0.155 A5 0.295

Table 8 Results driven by the code to method VIKOR.

7 Conclusions

Easy interactive computer software has been proposed. The code algorithm is mainly based on famous mathematical methods well-known in multi-criteria decision making such as: VIKOR, TOPSIS, and the weighted sum method (WSUM). Two real examples are shown to prove the validity and accuracy of this code. The code is implemented and programmed by the c-sharp under dot-net; it is easy to use and easy to be developed. Examples input data file, testing examples, and the setup file of this software are appended in a rar file with this paper.

Acknowledgments

Authors wish to thank Prof. I. Othman, the DG of the AECS for his valuable support and encouragement throughout this work. The anonymous reviewers are cordially thanked for their critics, remarks and suggestions that considerably improved the final version of this paper.

References

- Caterino C, Iervolino I, Manfredi G, Cosenza E. 2009. Applicability and effectiveness of different decision making methods for seismic upgrading building structures. ANIDIS BOLOGNA. Italy
- Chen SJ, Hwang CL. 1992. Fuzzy Multiple Attribute Decision Making: Methods and Applications. Sprenger Verlag, Berlin, Germany
- Dağ S, Onder E. 2013. Decision-Making for facility location using VIKOR method. Journal of international scientific publication: Economy and Business, 7(1): 308-330
- Ferrarini A. 2011. A new fuzzy algorithm for ecological ranking. Computational Ecology and Software, 1(3): 186-188

- Ferrarini A. 2012. WORTHY: a new model for ecological ranking and evaluation. Computational Ecology and Software, 2(4): 213-219
- Hwang CL, Yoon K. 1981. Multiple attribute decision making. Lecture Note in Economics and Mathematical Systems. 186, Springer Verlag, Berlin, Germany
- IAEA (International Atomic Energy Agency). 2019. Application of Multi-criteria Decision Analysis Methods to Comparative Evaluation of Nuclear Energy System Options: Final Report of the INPRO Collaborative Project KIND. NO. NG-T- 3.20
- Kumar A, Sah B, Singh A R, Deng Y, He X, Kumar P, Bansal RC. 2017. A review of multi-criteria decision making (MCDM) towards sustainable renewable energy development. Renewable and Sustainable Energy Reviews, 69: 596-609
- Opricovic S, Tzeng GH. 2002. Multi-criteria planning of post-earthquake sustainable reconstruction. The Journal of Computer-Aided Civil and Infrastructure Engineering, 17(3): 211-220
- Opricovic S, Tzeng GH. 2004. Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. European Journal of Operational Research, 156: 445-455
- Opricovic S, Tzeng GH. 2007. Extended VIKOR method in comparison with outranking methods. European Journal of Operational Research, 178: 514-529
- Yu PL. 1973. A class of solutions for group decision problems. Management Science, 38: 15499-15516
- Zhang WJ, Qi YH, Li X. 2017. MATASS: the software for multi-attribute assessment problems. Computational Ecology and Software, 7(2): 38-48