Evaluation of educational open-source software using multicriteria decision analysis methods

Georgia Paschalidou[1], Nikolaos Vesyropoulos[1], Vassilis Kostoglou[2], Emmanouil Stiakakis[1] and Christos K. Georgiadis[1]

[1] University of Macedonia, Department of Applied Informatics, Greece

[2] Alexander TEI of Thessaloniki, Department of Information Technology, Greece

81st Meeting of the European Working Group «Multiple Criteria Decision Aiding», Annecy, France 26-28 March 2015

Presentation Overview

- What is Open Source Software
- What is Software Quality
- Open Source Maturity Model (OSMM)
- Weighted Sum Model (WSM) in OSMM
- Application of MCDA methods based on OSMM
  - AHP
  - PROMETHEE
  - ELECTRE III
  - REGIME
- Rankings Comparison
- Conclusion – Future Work

What is Open Source Software (OSS)

- The term OSS refers to software whose source code is available for examination, use and modification
  - Usually designed by a group of volunteer programmers
  - Publicly accessed
  - Improvement and evolution of the software often conducted by software communities
  - Wide variety of existing open source software products

- **Evaluation of open source software products:**
  Matter of paramount importance for organizations in order to decide which one best fits the organization's requirements

What is Software Quality

- Quality of Open Source Software (OSS) products
  - Affected by many variables
  - Varying strongly in different products
- Limitless list of indicators providing estimation of an OSS project's quality
  - e.g. number of users, longevity of the project, documentation available on-line, etc.
- Need for aggregation of indicators' values
- Several approaches dealing with the OSS quality assessment
  - Open Business Readiness Rating (OpenBRR)
  - Qualification and Selection of Open Source software (QSOS)
  - **Open Source Maturity Model (OSMM)** from Navica

Open Source Maturity Model (OSMM)

- A software quality model which evaluates the overall maturity of a product
  - Six dimensions (criteria) for maturity evaluation
    - ✓ Software product
    - ✓ Technical support
    - ✓ Documentation
    - ✓ Training
    - ✓ Product integrations
    - ✓ Professional services
  - For each criterion
    - Some weight indicating its relative importance
    - A maturity score, in a scale of 1 to 10, after examining specific factors

Open Source Maturity Model (OSMM)

- Factors examined for each dimension (criterion)
  - Software product
    - product functionality
    - longevity
    - quality of the product
    - quality of the technical team
  - Technical support
    - community support
    - paid support
    - advanced self-support
  - Documentation
    - web postings
    - developer-created documentation
    - commercially-published documentation

Open Source Maturity Model (OSMM)

- Factors examined for each dimension (criterion)
  - Training
    - existence of web-based mini-tutorials
    - developer-created or commercial tutorials
    - classroom delivered by development team
    - classroom delivered by commercial entity
  - Integration
    - existent integration with third party software
    - necessary integrations via self-development
    - necessary integrations by commercial vendors
  - Professional services
    - product team services
    - services available from external firms

Weighted Sum Model (WSM) in OSMM

- **Overall maturity score of a software product = Weighted sum of the criteria maturity scores**
  - according to their predefined weighting factors

- Weighted Sum Model (WSM), an MCDA method, is actually applied
  - The total importance of alternative A_i, denoted as $A_i^{\text{WSM-score}}$, is defined as:

$$A_i^{WSM-score} = \sum_{j=1}^{n} w_j \, a_{ij}, \text{ for } i = 1,2,\ldots,n$$

where: m = number of alternatives
n = number of criteria
w_j = relative weight of importance of the criterion C_j
a_{ij} = performance value of alternative A_i in terms of criterion C_j

Weighted Sum Model (WSM) in OSMM

Illustrative Example

5 Educational Open-Source Software (EOSS) Projects

Criterion	Actual Score					Weighting Factor
	EOSS1	EOSS2	EOSS3	EOSS4	EOSS5	
Product Software	7	8	6	8	7	4
Support	6	6	7	6	6	2
Documentation	7	7	7	5	6	1
Training	6	6	6	5	5	1
Product Integrations	7	7	6	6	7	1
Professional Services	5	6	5	6	6	1

Weighted Sum Model (WSM) in OSMM

Element	Actual Score					Weighting Factor
	EOSS1	EOSS2	EOSS3	EOSS4	EOSS5	
Product Software	7	8	6	8	7	4
Support	6	6	7	6	6	2
Documentation	7	7	7	5	6	1
Training	6	6	6	5	5	1
Product Integrations	7	7	6	6	7	1
Professional Services	5	6	5	6	6	1
Total Maturity Score for EOSS1 (out of 100)	65	70	62	66	64	

Education Open Source Software	Total Maturity Score	OSMM Ranking
EOSS1	65	3
EOSS2	70	1
EOSS3	62	5
EOSS4	66	2
EOSS5	64	4

Application of MCDA methods based on OSMM

- Motivation
  - Employment of Weighted Sum Model (WSM) in OSMM

- Purpose
  - Comparison of the results obtained by other MCDA methods based on the criteria used in OSMM

- MCDA Methods used
  - AHP (Analytic Hierarchy Process)
  - PROMETHEE (Preference Ranking Organization METHod for Enrichment of Evaluations)
  - ELECTRE (ELiminationEt ChoixTraduisant la REalité)
  - REGIME (Regime Analysis)

Analytic Hierarchy Process (AHP)

- Methodology consisting of four discrete steps:
  - Creating the hierarchy of the problem's district components
  - Applying the criteria priorities through pairwise comparisons
  - Calculating the rate of the alternatives for each criteria
  - Calculating the relative score for each alternative.

- In more detail:
  - Assuming the final pairwise matrix is named S_{mn} we calculate the sum of each row

$$Sum_m = \sum_{n=1}^{k} S_{mn}$$

where *k* is the number of columns

Analytic Hierarchy Process (AHP)

- An eigenvector is being computed with the formula

$$EV_m = \frac{SUM_m}{\sum_1^{\mathrm{k}} SUM_{\mathrm{m}}}$$

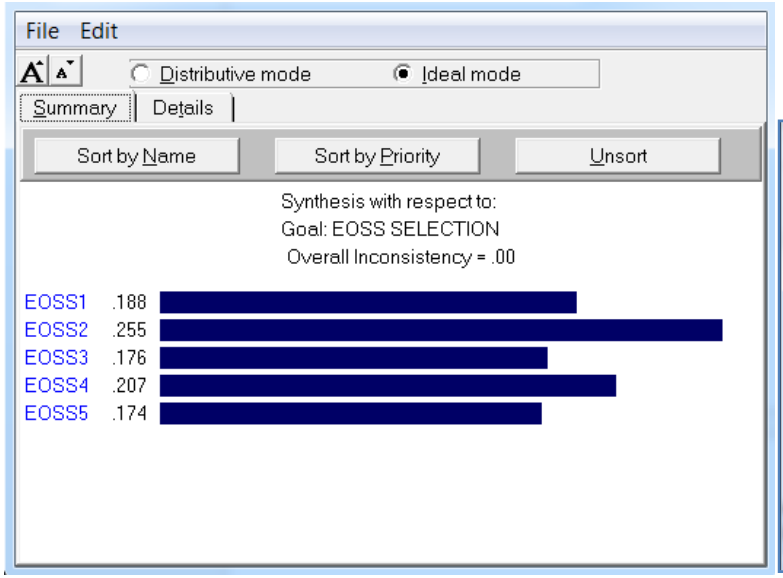
- Same procedure followed for the eigenvector of each criterion table
- The matrix consisting of the rank of the alternatives, is computed based on the formula

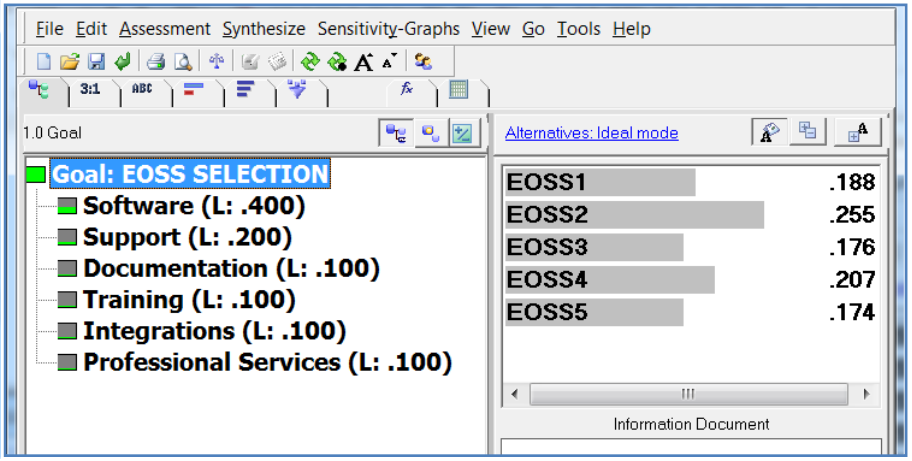
$$AR[i] = \sum_{n=1}^{\mathrm{k}} CEV_n[\mathrm{i}] * EV[n]$$

where AR refers to the final rankings matrix

CEV_n refers to the eigenvector formulated for each criterion

Analytic Hierarchy Process (AHP)





Education Open Source Software	Total Maturity Score	AHP Ranking
EOSS1	0.188	3
EOSS2	0.255	1
EOSS3	0.176	5
EOSS4	0.207	2
EOSS5	0.174	4

PROMETHEE

- Initialized by calculating the evaluation table
  - Constitutes of the alternatives and their evaluation
  - The relative importance of each criteria
- For each pair of alternatives (a, b):
  - One is preferable if it has equal or higher values for each and every criterion
  - Cannot be compared if each alternative outranks the other in different criteria
- Method requests an indifference threshold and smallest deviation.
  - For solving incomparability issues
- The preferences P_j between a pair of alternatives for a specific criterion are computed
- The sum of all preferences P_j are multiplied by corresponding weights
  - produce the overall degree of preference

PROMETHEE

- Each alternative is compared with the rest
  - the positive and negative outranking flows are calculated as follows:

$$\varphi^{+}(a) = \frac{1}{n-1}\sum_{x \in A} \pi(a, x)$$

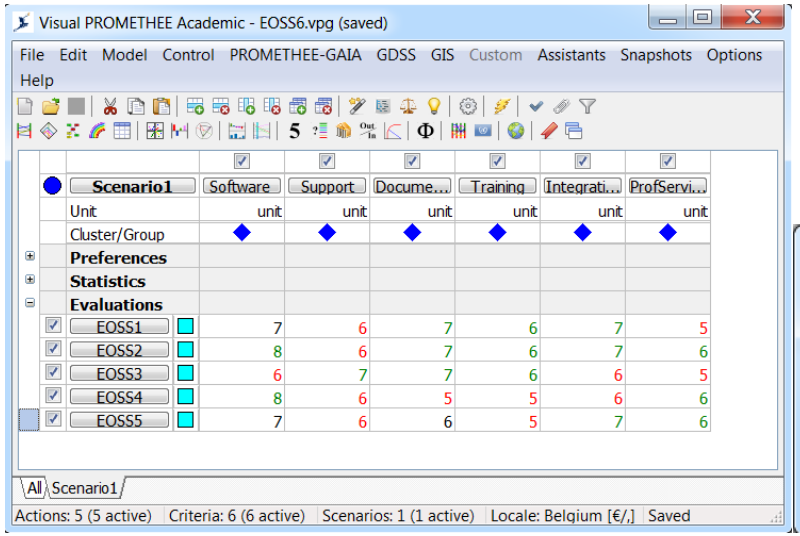
$$\varphi^{-}(a) = \frac{1}{n-1}\sum_{x \in A} \pi(x, a)$$

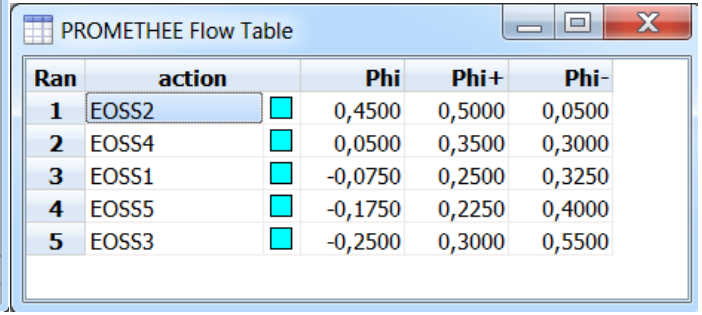
- A complete ranking is calculated based on the difference of ranking flows
  - Higher $\varphi(\mathrm{a})$ value corresponds to a better alternative

$$\begin{cases} \mathrm{a}P^{I}\mathrm{b} & if\ f \quad \varphi(\mathrm{a}) > \varphi(\mathrm{b}) \\ \mathrm{a}I^{I}\mathrm{b} & if\ f \quad \varphi(\mathrm{a}) = \varphi(\mathrm{b}) \end{cases}$$

- All alternatives are comparable and there are no inconsistencies

PROMETHEE





Education Open Source Software	Total Maturity Score	PROMETHEE Ranking
EOSS1	-0.075	3
EOSS2	0.450	1
EOSS3	-0.250	5
EOSS4	0.050	2
EOSS5	-0.175	4

ELECTRE III

- Requires the definition of thresholds, to handle levels of imprecision.
  - the indifference threshold q_j
  - the preference threshold p_j
  - the veto threshold v_j
- Low level of criteria compensation
- The concordance index for two alternatives *(a,b)* is computed based on the formula:

$$c_j(a,b)=\begin{cases} 0 & g_j(b)-g_j(a) \leq q_j \\ 1 & g_j(b)-g_j(a) \geq p_j \\ \frac{p_j+g_j(a)-g_j(b)}{p_j-q_j} & q_j \leq g_j(b)-g_j(a) \leq p_j \end{cases}$$

where $g_j(a)$ corresponds to the evaluation of the alternative a on criterion *j*.

ELECTRE III

- The discordance index is computed based on the formula:

$$d_j(a,b)=\begin{cases} 0 & g_j(b)-g_j(a) \leq p_j \\ 1 & g_j(b)-g_j(a) \geq v_j \\ \frac{g_j(a)-g_j(b)-p_j}{v_j-p_j} & p_j \leq g_j(b)-g_j(a) \leq v_j \end{cases}$$

- The overall concordance index is then computed by:

$$C(a,b)=\frac{1}{\sum_{j=1}^{r} k_j} \sum_{j=1}^{r} k_j \; c_j(a,b)$$

where k_j is equal to the weight given by the decision maker

ELECTRE III

- The credibility index is calculated, corresponding to the credibility of comparisons between *a* over *b*
  - Based on the overall concordance index

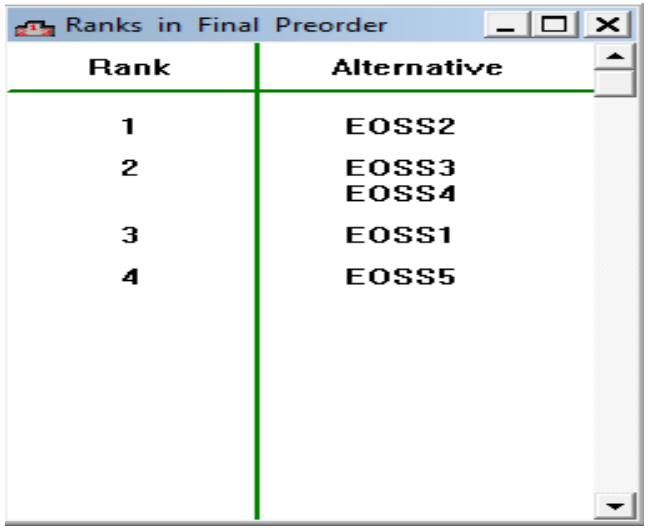
$$S(a,b)=\begin{cases} \mathrm{C}(a,b)d_j \leq \mathrm{C}(a,b) \ \ \forall j \\ \mathrm{C}(a,b)\prod_{j\in J(ab)}\frac{1-d_j(a,b)}{1-C(ab)} \end{cases} \quad \begin{matrix} \\ d_j(a,b) > C(a,b) \\ J(a,b)\colon d_j(a,b) > C(a,b) \end{matrix}$$

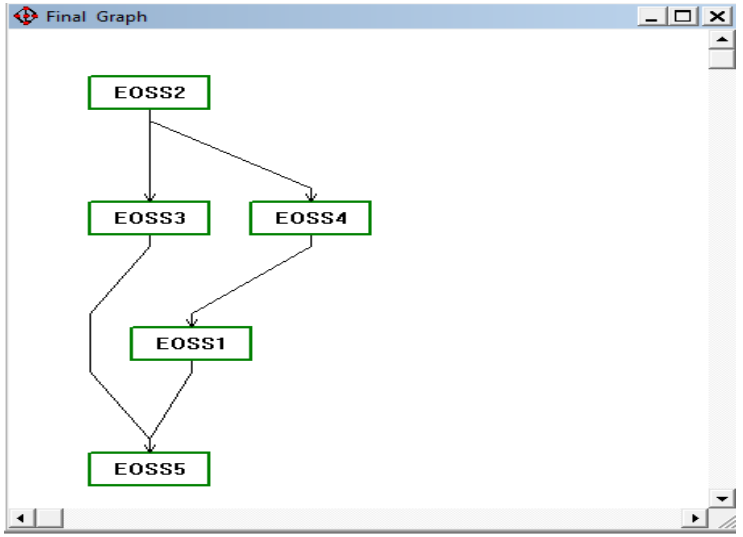
- The λ_{max} value (maximum value), is computed as follows:

$$\lambda_{max} = \max S(a,b)$$

- The λ_{max} values allows the computation of the ascending and descending distillation of the preferences
  - Their conjunction corresponds to the final ranking

ELECTRE III





Education Open Source Software	ELECTRE Ranking
EOSS1	4
EOSS2	1
EOSS3	2
EOSS4	2
EOSS5	5

REGIME

- Regime Analysis can handle various types of data:
  - Binary
  - Ordinal
  - Categorical
  - Cardinal (ratio and interval scale)
  - Mixed data

- Based on concordance analysis for ranking alternatives
  - procedure similar to PROMETHEE and ELECTRE methods
  - basic idea of concordance analysis → rank a set of actions by means of their pairwise comparisons in relation to the chosen criteria

REGIME

- Concordance index C_{ik} = the sum of the weights that are related to the criteria for which *i* is better than *k*
- Concordance index C_{ki} considering the criteria for which *k* is better than *i*
- Net concordance index

$$\mu_{ij} = C_{ik} - C_{ki}$$

- The method applies the performance indicator p_{ij} for the dominance of criteria *i* with respect to *j*

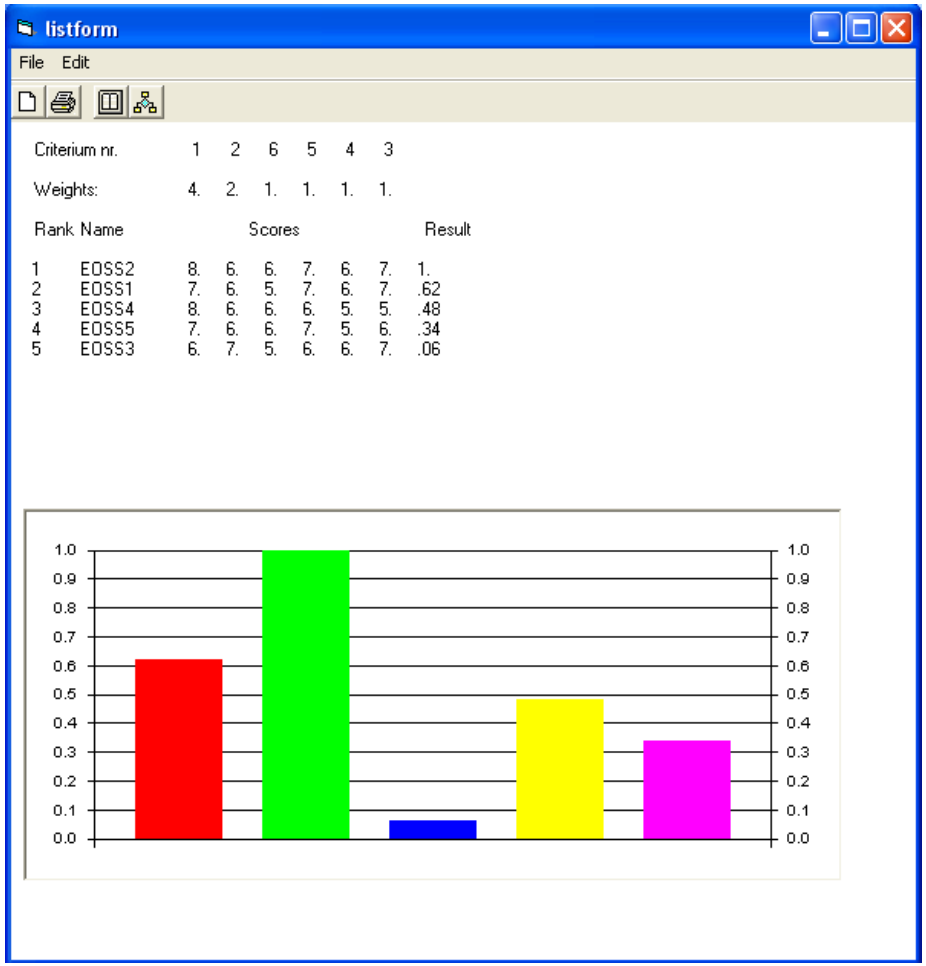
$$p_{ij} = prob(\mu_{ij} > 0)$$

- Finally the performance score is calculated based on

$$p_i = \frac{1}{I-1}\sum_{j \neq i} p_{ij}$$

where *I* is the number of chosen alternatives

REGIME



Education Open Source Software	Total Maturity Score	REGIME Ranking
EOSS1	0.62	2
EOSS2	1	1
EOSS3	0.06	5
EOSS4	0.48	3
EOSS5	0.34	4

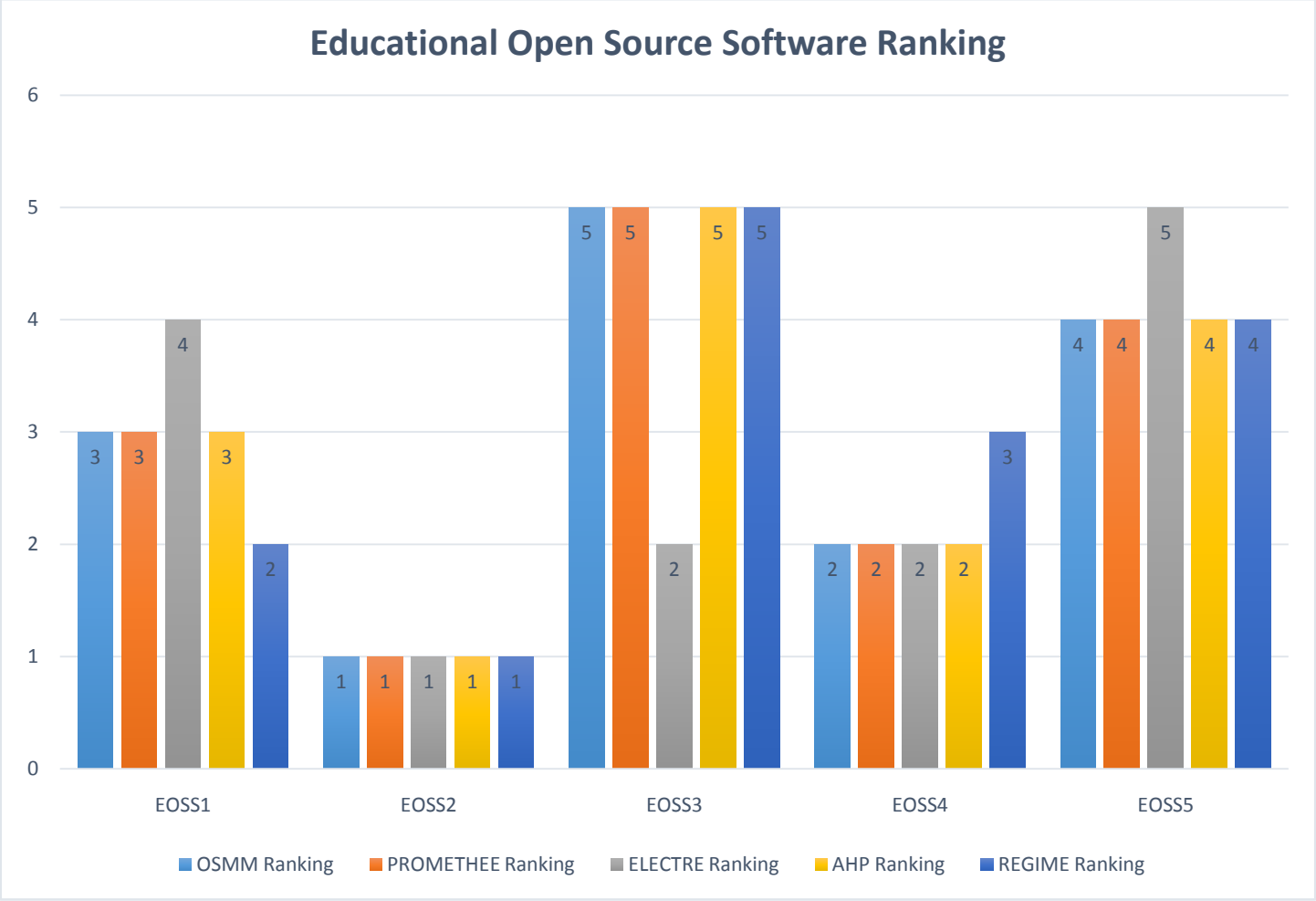
Rankings Comparison

- The different techniques generated minor differences
- WSM, PROMETHEE and AHP rankings are in the vast majority of cases in total agreement
- ELECTRE III and REGIME present minor differentiations
- However, if methods were deployed for the selection of the optimal alternative, the variation of technique would result to the same option

Rankings Comparison

EducationOpenSource Software	OSMM Ranking	PROMETHEE Ranking	ELECTRE Ranking	AHP Ranking	REGIME Ranking
EOSS1	3	3	4	3	2
EOSS2	1	1	1	1	1
EOSS3	5	5	2	5	5
EOSS4	2	2	2	2	3
EOSS5	4	4	5	4	4

Rankings Comparison



Pearson Correlation Coefficients

Correlation between sets of data is a measure of how well they are related.

Pearson Correlation Coefficient	OSMM Ranking	PROMETHEE Ranking	ELECTRE Ranking	AHP Ranking	REGIME Ranking
OSMM Ranking	1.000				
PROMETHEE Ranking	1.000	1.000			
ELECTRE Ranking	0.481	0.481	1.000		
AHP Ranking	1.000	1.000	0.481	1.000	
REGIME Ranking	0.900	0.900	0.289	0.900	1.000

High correlation: 0.5 to 1.0 or -0.5 to 1.0

Medium correlation: 0.3 to 0.5 or -0.3 to 0.5

Low correlation: 0.1 to 0.3 or -0.1 to -0.3

Conclusions

The purpose of this analysis was to test the level of convergence between the final rankings of alternatives produced by the five MCDA methods

- Results show that changing MCDA technique generates minor differences
- WSM, PROMETHEE and AHP rankings are in the vast majority of cases in total agreement
- ELECTRE III and REGIME present minor differentiations in the final order of preference
- If the particular methods were being deployed **exclusively** for the selection of the optimal alternative, then the variation of technique would result to the same option

Future Work

- Implementation of MCDA methods, based on the criteria and weights proposed by OSMM, on widely used open source software projects
  - Currently focus on educational virtual learning environments
- Comparison of the results generated by MCDA methods adopting varying weights
- Extension of the analysis relying on the criteria and weights introduced by other open source quality assessment models

Acknowledgement

This research is implemented through the Operational Program "Education and Lifelong Learning" and is co-financed by the European Union (European Social Fund) and Greek national funds. Project title: “Digital map of higher technological education professions"

Thank you for your attention