



Quantifying COTS Components Selection using Multi Criteria Decision Analysis Method PROMETHEE

By Kulbir Kaur & Harshpreet Singh

Lovely Professional University, India

Abstract- Component Based Development relies on already existing components to develop the system. It offers various advantages as increase in productivity, reduced development effort and time. The biggest challenge is to select the appropriate component from number of alternatives based on the quality parameters. In this paper COTS component selection is reduced to a multi criteria decision problem by quantifying it with PROMETHEE method. PROMETHEE is an outranking method which better supports the evaluation and selection from various alternatives based on the functional and non-functional requirements. The aim of this paper is to show the application of PROMETHEE in evaluating, analysing and selecting the appropriate COTS component with respect to requirements. The paper also discusses the procedure and benefits of using PROMETHEE method over the other MCDA methods.

Keywords: COTS, CBD, MCDA, PROMETHEE, AHP, WSM.

GJCST-C Classification: K.6.3



Strictly as per the compliance and regulations of:



Quantifying COTS Components Selection using Multi Criteria Decision Analysis Method- PROMETHEE

Kulbir Kaur ^α & Harshpreet Singh ^σ

Abstract- Component Based Development relies on already existing components to develop the system. It offers various advantages as increase in productivity, reduced development effort and time. The biggest challenge is to select the appropriate component from number of alternatives based on the quality parameters. In this paper COTS component selection is reduced to a multi criteria decision problem by quantifying it with PROMETHEE method. PROMETHEE is an outranking method which better supports the evaluation and selection from various alternatives based on the functional and non-functional requirements. The aim of this paper is to show the application of PROMETHEE in evaluating, analysing and selecting the appropriate COTS component with respect to requirements. The paper also discusses the procedure and benefits of using PROMETHEE method over the other MCDA methods.

General Terms: selection, alternative, criteria, rank, degree, preference, profile.

Keywords: COTS, CBD, MCDA, PROMETHEE, AHP, WSM.

I. INTRODUCTION

Component Based Development (CBD) relies on reusable COTS components to build the software systems. Before integrating the components into the system, the components should be quantified according to the non-functional and functional requirements.

With the rapid growing and changing of technology, number of products or tools entering in the market also increases. So it becomes a big challenge to select the best component from a number of alternative components and to build a trust on the selected components.

Component selection and evaluation is a multi criteria problem in which a component from various alternatives is to be selected which best satisfies the maximum criteria than others. A chosen option should have greater rank on all criteria than others.

Various methods can be used as a solution of this problem like OSTO [2], CARE [8], AHP [3], WSM [3], Utility Theory [1], SMART [1], DesCOTS [9], UnHOS [7] etc. Multi Criteria Decision Analysis methods help

the decision maker to select the best option from number of multi criteria alternatives which best scores on multiple criteria. PROMETHEE is a multi criteria method proposed by JP Brans in 1982 [6]. It can be applied for the analysis and selection of components and solutions in various kinds of fields like Banking, Industrial Location, Manpower planning, Water resources, Investments, Medicine, Chemistry, Health care, Tourism, Ethics in OR, Dynamic management [6]. It can be applied to selection and evaluation of COTS components while making the decision to select components from repository to develop the software system. The aim of this paper is to apply PROMETHEE on the selection and evaluation of software packages and its benefits over others multi criteria methods.

II. LITERATURE REVIEW

COTS-Aware Requirements Engineering and Software Architecting (CARE/SA) proposed by Lawrence [8] for evaluating, matching and selecting of COTS components. CARE/SA method uses the architectural aspects, functional aspects and non-functional aspects of COTS components. It indicates that each component is represented by the unique attributes which consists of its architectural, functional and non-functional aspects.

Hamdy Ibrahim et al. in [7] proposed a method named 'UnHOS' (Uncertainty Handling in COTS Selection) method for the evaluation of COTS components and takes into account their uncertainty. It uses Analytic Hierarchy Process (AHP) for the evaluation of COTS components and Bayesian Belief Network (BBN) to indicate their uncertainty. It also presents a tool to support the usability of the UnHOS method.

Anil Jadhav et al. in [3] tells that Multi Criteria Decision Making Methods helps the decision makers to solve the problem of selection and evaluation of software components in which problem is defined as a collection of multiple criteria that needs to be taken into account. It gives the overview of Multi Criteria Decision Making Methods like: Analytic Hierarchy Process (AHP), Weighted Scoring Method (WSM) and Hybrid Knowledge Based System (HKBS). It compares the three approaches and concludes that HKBS is better than AHP and WSM.

Author α: Student, School of CSE and IT, Lovely Professional University, Phagwara, Punjab. e-mail: gill.kulbir11@gmail.com

Author σ: Assistant Professor, School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab. e-mail: harsh08walia@gmail.com

Arvinder Kaur et al. in [2] provide a brief overview of the evolutionary techniques. It also derives a hierarchical decomposition method to draw goals from that impact factors. It introduces OSTO method for the selection of software components which compares the scores and cost associated to each alternative and their relative comparison. It introduces various factors in the selection of reusable software components. It also presents the evaluation criteria based on various classifications as functional requirements, product quality attributes, strategic concerns and architecture and domain compatibility. It gives the result of two case studies using OSTO method. The component which have good quality assurance score is selected for consideration.

III. MULTI CRITERIA DECISION ANALYSIS METHOD

Multi criteria problem involves the selection of the best option from a number of alternatives on the basis of multiple criteria satisfaction with higher degree. As component selection is a multi-criteria problem, there are number of alternatives for the solution of problem and we have to select a candidate component which best suits for the solution on the basis of satisfying maximum criteria than others with higher degree. So problem can be formulated as:

$$\max \{c_1a_n, c_2a_n, \dots, c_ka_n \mid a_n \in A\}.$$

Let $A = \{a_1, a_2, a_3, \dots, a_n\}$ be the set of 'n' alternatives for the solution of the problem.

$C = \{c_1, c_2, c_3, \dots, c_k\}$ be the set of 'k' criteria as a basis of evaluation and selection.

Let $w_1, w_2, w_3, \dots, w_k$ be the weight of each criterion respectively.

Each multi criteria decision analysis method proceeds with the decision table. Decision Table is shown in Table 1. Each column denotes the criteria, each row denotes the alternatives and 'c_{kan}' represents the score of alternative 'n' on criterion 'k'.

Table 1 : The decision table

Alternative	Criteria			
	w ₁	w ₂	w _k
	c ₁	c ₂	c _k
a ₁	c _{1a₁}	c _{2a₁}	c _{ka₁}
a ₂	c _{1a₂}	c _{2a₂}	c _{ka₂}
.
.
a _n	c _{1a_n}	c _{2a_n}	c _{ka_n}

a) PROMETHEE Method

There is need to have a method which is simpler and better helps in decision making while obtaining the solution of multi objective selection of trusted components from the number of available

alternatives. As COTS components selection is a multi-criteria problem. PROMTHEREE solves the problem in an optimal way with additional benefits than other MCDA methods.

PROMETHEE is Preference Ranking Organisation Method for Enrichment Evaluation. PROMETHEE is a multi criteria decision analysis method. It is an outranking method based on pair wise comparison of alternatives. It was developed by JP Brans in 1982[6]. Originally it was developed as PROMETHEE-1 (partial ranking) and PROMETHEE-2 (complete ranking). Later PROMETHEE-3 (ranking based on intervals) and PROMETHEE-4 (continuous case) were developed. PROMETHEE-5 (MCDA includes segmentation constraints) and PROMETHEE-6 (represents human brain) are also there. PROMETHEE is based on mathematical properties [6]. It can be applied on various fields for the selection and evaluation of winning solution in a multi criteria problem.

Steps for solving multi criteria problem with this method is as follows:

1. Determination of available alternatives to solve the problem.

Let $A = \{a_1, a_2, a_3, \dots, a_n\}$ be the set of 'n' alternatives for the solution of the problem.

2. Determination of evaluation criteria. Let $C = \{c_1, c_2, c_3, \dots, c_k\}$ be the set of 'k' criteria as a basis of evaluation and selection.

3. Problem statement stated as $\max\{c_1(a_n), c_2(a_n), c_3(a_n), \dots, c_j(a_n), \dots, c_k(a_n) \mid a_n \in A\}$ Where 'c_k(a_n)' represents the value of alternative 'a_n' on the criterion 'k'.

4. Create an evaluation table or (n*k) matrix with 'n' rows (number of alternatives) and 'k' columns (number of evaluation criteria) and place the score value of each alternative based on each criterion i.e. 'c_{ka_n}'.

5. Assign weight to each criterion i.e. w_j where (j=1,2,3,...k) and w₁+w₂+w₃+...w_k=1.

6. Find the difference between each pair of alternatives based on each criterion i.e. d_j(a,b)= c_j(a)-c_j(b) where (j=1, 2...k) and (a,b∈A).

7. Find the preference of the one alternative over the other as a function of difference between each pair of alternatives based on each criterion i.e. P_j (a,b)= F_j [d_j(a,b)] where (a,b∈ A) and (j=1, 2...k) and 0≤P_j(a,b)≤1. In case of minimizing the criteria preference P_j(a,b)=F_j[-d_j(a,b)]. Preference function values can be taken on the basis of particular criterion function and the parameter value which you have selected.

8. Calculate the degree to which preferred option is better than other alternative on all criteria i.e. $\pi(a,b) = P_1(a,b)w_1 + P_2(a,b)w_2 + \dots + P_k(a,b)w_k$. And 0≤π (a,b)≤ 1.

9. If the degree of preference nearly equals to 'zero'; it means there is weak preference of alternative 'a'

over alternative 'b'. And if the degree of preference nearly equals to 'one'; it means there is strong preference of alternative 'a' over alternative 'b'.

- Calculate the positive and negative outranking flow of each option and then compute the net outranking flow of the option and if it comes out to be greater than 'zero' then the option outranking the other options and if lower than 'zero' then it means that the option is outranked by the other options on all criteria.

Positive outranking flow (option outranks others):
 $\Phi+(a) = 1/(n-1)[\pi(a1,a2)+\pi(a1,a3)... \dots \pi(a1,an)]$
 Negative outranks flow (option is outranked by others):
 $\Phi-(a) = 1/(n-1)[\pi(a2,a1)+\pi(a3,a1)... \dots \pi(an,a1)]$

Net outrank flow of an option:
 $\Phi(a) = \Phi+(a) - \Phi-(a)$.
 We can say that 'a' is preferred over 'b' if $\Phi(a) > \Phi(b)$ and $0 \leq \Phi(a) \leq 1$. Moreover $\{\Phi(a1) + \Phi(a2) + \dots + \Phi(an) = 0\}$

- Obtain the outrank flow of each option on each criterion as:

$\Phi_j(a) = 1/(n-1)[(P1(a,b) - P1(b,a)) + (P2(a,b) - P2(b,a)) + P3(a,b) - P3(b,a)) \dots \dots + Pk(a,b) - Pk(b,a)]$.

- Obtain the profile of an alternative on all the criteria as:

$\Phi(a) = \Phi1(a)w1 + \Phi2(a)w2 + \dots \Phik(a)wk$.

Profile of an alternative indicates the quality of an alternative on each criterion. Profile is shown in figure 1.

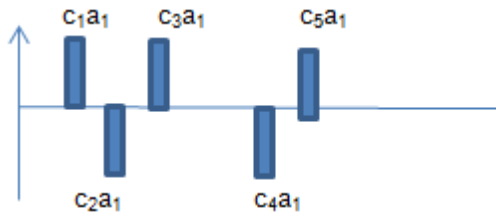


Figure 1 : Profile of an alternative

- Select the alternative which has high ' $\Phi(a)$ '. Values of ' $\Phi(a)$ ' gives the complete rank of the alternatives.

b) Promethee on Cots Components Selection

While developing system from COTS components it becomes very difficult to select best one if number of alternatives are available and to evaluate those alternatives. Application of PROMETHEE methodology on the COTS components selection better supports us in decision making.

Suppose a set of software components i.e. Alternatives set as $A = \{A1, A2, A3, A4, A5\}$ and evaluation criteria set as $C = \{Performance, Reliability, Maintainability, Cost, Integrability\}$ and the weight of each criterion respectively is as: 0.3, 0.2, 0.1, 0.2, 0.2.

Let criteria can be written as C1, C2, C3, C4 and C5. Scale and units for criteria are as follows:
 C1: VG, G, A, B, VB

C2: no. of failures per 1000 hours of service
 C3: VG, G, A, B, VB
 C4: Rs. (Rupees)
 C5: VG, G, A, B, VB

Where VG, G, A, B, VB stands for very good, good, average, bad, very bad. Score for each grade is as in table 2.

Table 2 : Grade scores

Grade	VG	G	A	B	VB
Score	5	4	3	2	1

Evaluation table is shown in table 3.

Table 3 : Evaluation table

	C1	C2	C3	C4	C5
	0.3	0.2	0.1	0.2	0.2
A1	3	2	4	1000	5
A2	2	1	3	1200	4
A3	5	3	5	900	3
A4	4	7	3	1000	2
A5	4	2	2	1100	1

Preference function may be used as Usual, U-Shaped, V-Shaped, Level, Gaussian or V-Shape with indifference criterion function. Let in the example U-Shaped criterion function is taken for C2 and C4. Level criterion function is taken for C1, C3 and C5. Level and U-Shaped criteria are shown in figure 2 and figure 3 respectively.

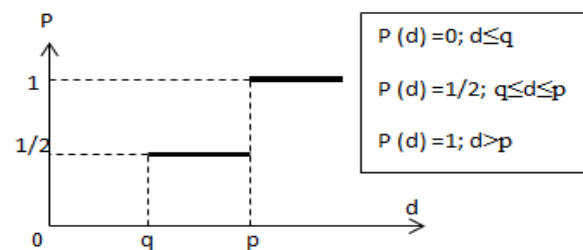


Figure 2 : Level Criterion

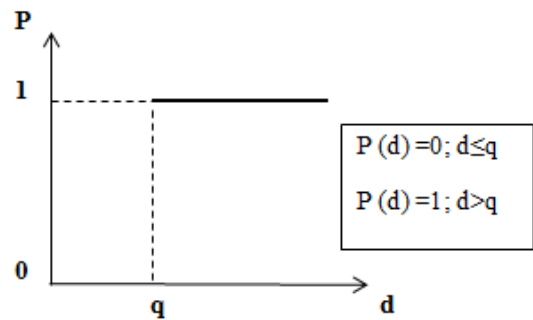


Figure 3 : U-Shaped Criterion

Let parameter values for each criterion is as follows:
 For C1, C3 and C5; q=2 and p=4
 For C2; q=4

For C4; q=100

Relative difference between alternatives on each criterion is shown in tables 4, 5, 6, 7 and 8.

Table 4 : Difference between alternatives with respect to performance

$d_1(a,b)$	A1	A2	A3	A4	A5
A1	0	1	-2	-1	-1
A2	-1	0	-3	-2	-2
A3	2	3	0	1	1
A4	1	2	-1	0	0
A5	1	2	-1	0	0

Table 5 : Difference between alternatives with respect to reliability

$d_2(a,b)$	A1	A2	A3	A4	A5
A1	0	-1	1	5	0
A2	1	0	2	6	1
A3	-1	-2	0	4	-1
A4	-5	-6	-4	0	-5
A5	0	-1	1	5	0

Table 6 : Difference between alternatives with respect to maintainability

$d_3(a,b)$	A1	A2	A3	A4	A5
A1	0	1	-1	1	2
A2	-1	0	-2	0	1
A3	1	2	0	2	3
A4	-1	0	-2	0	1
A5	-2	-1	-3	-1	0

Table 7 : Difference between alternatives with respect to cost

$d_4(a,b)$	A1	A2	A3	A4	A5
A1	0	200	-100	0	100
A2	-200	0	-300	-200	-100
A3	100	300	0	100	200
A4	0	200	-100	0	100
A5	-100	100	-200	-100	0

Table 8 : Difference between alternatives with respect to integrability

$d_5(a,b)$	A1	A2	A3	A4	A5
A1	0	1	2	3	4
A2	-1	0	1	2	3
A3	-2	-1	0	1	2
A4	-3	-2	-1	0	1
A5	-4	-3	-2	-1	0

Preference function value of each alternative over other on all criteria is shown in table 9, 10, 11, 12 and 13.

Table 9 : Preference value on performance

$P_1(a,b)$	A1	A2	A3	A4	A5
A1	0	0	0	0	0
A2	0	0	0	0	0
A3	.5	.5	0	0	0
A4	0	.5	0	0	0
A5	0	.5	0	0	0

Table 10 : Preference value on reliability

$P_2(a,b)$	A1	A2	A3	A4	A5
A1	0	0	0	1	0
A2	0	0	0	1	0
A3	0	0	0	0	.5
A4	0	0	0	0	0
A5	0	0	0	1	0

Table 11 : Preference value on maintainability

$P_3(a,b)$	A1	A2	A3	A4	A5
A1	0	0	0	0	.5
A2	0	0	0	0	0
A3	0	.5	0	.5	.5
A4	0	0	0	0	0
A5	0	0	0	0	0

Table 12 : Preference value on cost

$P_4(a,b)$	A1	A2	A3	A4	A5
A1	0	1	0	0	0
A2	0	0	0	0	0
A3	0	1	0	0	1
A4	0	1	0	0	0
A5	0	0	0	0	0

Table 13 : Preference value on integrability

$P_5(a,b)$	A1	A2	A3	A4	A5
A1	0	0	.5	.5	.5
A2	0	0	0	.5	.5
A3	0	0	0	0	.5
A4	0	0	0	0	0
A5	0	0	0	0	0

Degree of preference of one alternative over other is shown in table 14.

Table 14 : Degree of preference $\Pi(a,b)$

$\Pi(a,b)$	A1	A2	A3	A4	A5
A1	0	.20	.10	.30	.15
A2	0	0	0	.30	.10
A3	.15	.40	0	.05	.35
A4	0	.35	0	0	0
A5	0	.15	0	.20	0

Positive, negative and net outrank flow of each alternative is shown in table 15.

Table 15 : Positive, negative and net outrank flow

	$\Phi^+(a)$	$\Phi^-(a)$	$\Phi(a)$
A1	.1875	.0375	.1500
A2	.1000	.2750	-0.1750
A3	.2375	.0250	.2125
A4	.0875	.2125	-0.1250
A5	.0875	.1500	-0.0625

PROMETHEE-1 Partial ranking of each alternative is shown in figure 4 and PROMETHEE-2 Complete ranking of each alternative is shown in figure 5.

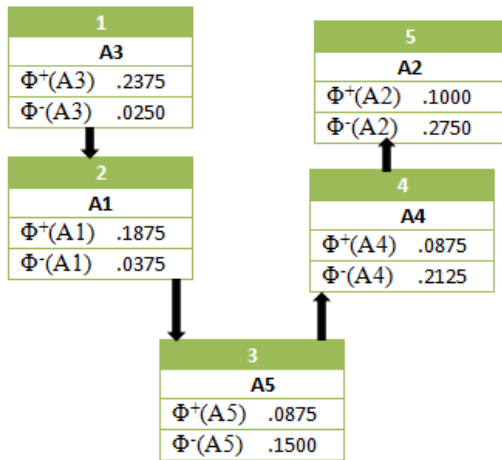


Figure 4 : Partial ranking

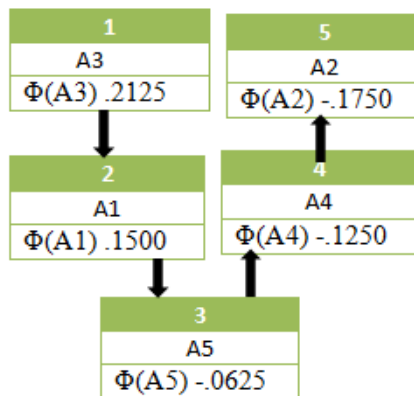


Figure 5 : Complete ranking

Profile of each alternative on all criteria is shown in table 16.

Table 16 : Profile of alternative

	$\Phi_1(a)$	$\Phi_2(a)$	$\Phi_3(a)$	$\Phi_4(a)$	$\Phi_5(a)$
A1	-.125	.250	.125	.250	.375
A2	-.375	.250	-.125	-.750	.250
A3	.250	0	.375	.500	0
A4	.125	-.750	-.125	.250	-.250
A5	.125	.250	-.250	-.250	-.375

Profile of alternative A1 on all criteria is shown in figure 6.

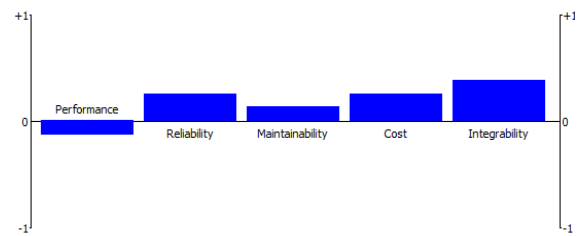


Figure 6 : Profile of A1

Profile of alternative A2 on all criteria is shown in figure 7.

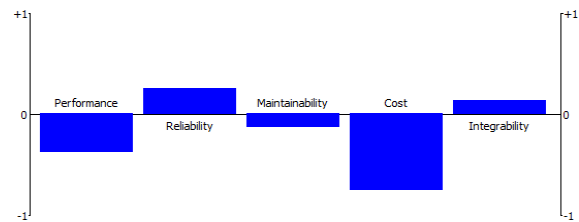


Figure 7 : Profile of A2

Profile of alternative A3 on all criteria is shown in figure 8.

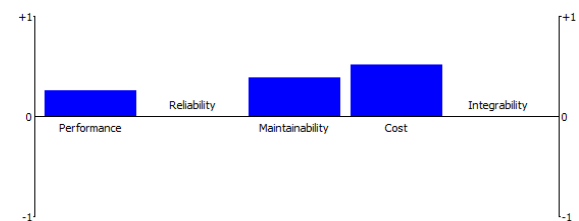


Figure 8 : Profile of A3

Profile of alternative A4 on all criteria is shown in figure 9.

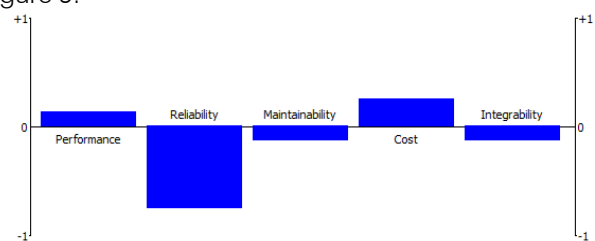


Figure 9 : Profile of A4

Profile of alternative A5 on all criteria is shown in figure 10.

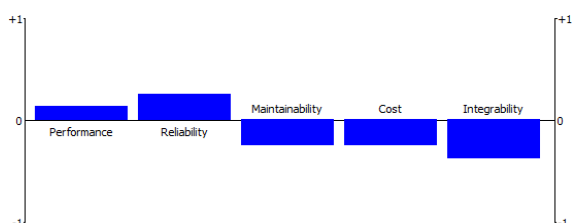


Figure 10 : Profile of A5

Ranking of all alternatives on all criteria is shown in figure 11.

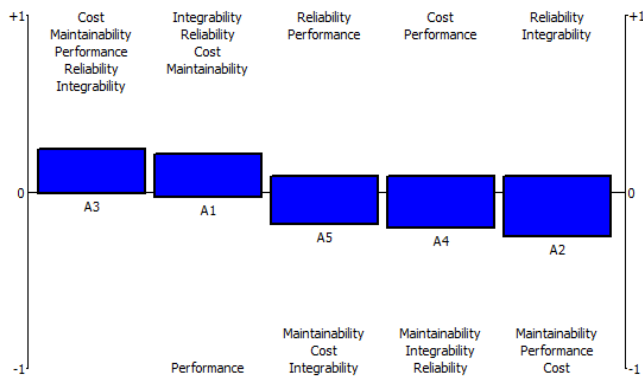


Figure 11 : Ranking of all alternatives on all criteria

c) *Benefits of using Promethee*

PROMETHEE methodology helps in decision making better than other MCDA methods in number of ways:

1. The supporting software packages of PROMETHEE like D-Sight, PROMCALC, Decision Lab, Visual PROMETHEE etc. are very user friendly.
2. PROMETHEE-GDSS supports group decision making in this the final solution is obtained by weighted sum of net outrank flow of each alternative.
3. PROMETHEE provides partial, complete, interval based ranking of alternatives.
4. Unlike other MCDA methods, PROMETHEE's preference degree tells us the degree by which an option is preferred over other.
5. PROMETHEE needs very less input for further operations as compare to other MCDA methods.
6. Unlike other MCDA methods, new alternatives can be added without doing much change in others.
7. PROMETHEE does not include normalization for normalizing the units of measurement of each criterion so there are fewer chances of errors as compare to many other MCDA methods.
8. PROMETHEE's extensions can be used as sorting purposes.

IV. CONCLUSION

Component selection is a wide comparison of components using a common set of criteria. Selecting the appropriate and relevant component significantly reduces the chances of risks associated with the COTS components with no source code available with them and improves the corporate competitiveness. Using PROMETHEE-GAIA methodology for the complete ranking of alternatives help decision makers to choose and analyse the highest rank component on all criteria and help to build confidence on the selected component.

REFERENCES RÉFÉRENCES REFERENCIAS

1. Hutchinson, J., & Kotonya, G. (2006, August). A Review of Negotiation Techniques in Component Based Software Engineering. In *Software Engineering and Advanced Applications*, 2006. SEAA'06. 32nd EUROMICRO Conference on (pp. 152-159). IEEE.
2. Kaur, A., & Mann, K. S. (2010). Component Selection for Component-Based Software Engineering. *International Journal of Computer Applications*, 2(1), 109-114.
3. Jadhav, A., & Sonar, R. (2009, December). Analytic Hierarchy Process (AHP), Weighted Scoring Method (WSM), and Hybrid Knowledge Based System (HKBS) for Software Selection: A Comparative Study. In *Emerging Trends in Engineering and Technology (ICETET)*, 2009 2nd International Conference on (pp. 991-997). IEEE.
4. Becker, C., Kraxner, M., Plangg, M., & Rauber, A. (2013, January). Improving decision support for software component selection through systematic cross-referencing and analysis of multiple decision criteria. In *System Sciences (HICSS)*, 2013 46th Hawaii International Conference on (pp. 1193-1202). IEEE.
5. Halim, A., Sudrajat, A., Sunandar, A., Arthana, I. K. R., Megawan, S., & Mursanto, P. (2011, December). Analytical Hierarchy Process and PROMETHEE application in measuring object oriented software quality. In *Advanced Computer Science and Information System (ICACSIS)*, 2011 International Conference on (pp. 165-170). IEEE.
6. Brans, J. P., & Mareschal, B. (2005). PROMETHEE methods. In *Multiple criteria decision analysis: state of the art surveys* (pp. 163-186). Springer New York.
7. Ibrahim, H., Elamy, A. H. H., Far, B. H., & Eberlein, A. (2011). UnHOS: A Method for Uncertainty Handling in Commercial Off-The-Shelf (COTS) Selection. *International Journal of Energy, Information & Communications*, 2(3).
8. Chung, L., Cooper, K., & Courtney, S. (2004). COTS-Aware Requirements Engineering and Software Architecting. In *Software Engineering Research and Practice* (pp. 57-63).
9. Grau, G., Carvallo, J. P., Franch, X., & Quer, C. (2004, August). DesCOTS: a software system for selecting COTS components. In *Euromicro Conference*, 2004. Proceedings. 30th (pp. 118-126). IEEE.