

Prioritizing alternative using Evolutionary Computing procedure in AHP

¹NUR FARHA ZAKARIA, ²HALINA MOHAMED DAHLAN, ³AB RAZAK CHE HUSSIN

Department of Information Systems

Faculty of Computer Science & Information Systems

Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia.

¹nfharhaz@gmail.com, ²halina@utm.my, ³abrazak@utm.my

Abstract: - Making decision is the critical part in choosing the best solution. The decision maker can use Analytic Hierarchy Process (AHP) as a method in making decision. However, this method has been criticized mainly in priority derivation procedure. To solve this criticized; this paper proposes AHPEC which is using Evolutionary Computing (EC) to derive priorities in AHP. The AHPEC gives better result compare to the other prioritization methods based on accuracy of derived priorities.

Key-Words: - Decision making, Analytic Hierarchy Process (AHP), and Evolutionary Computing (EC)

1 Introduction

Decision Making is a process of choosing among alternative courses of action for the purpose of attaining a goal or goals [1]. In making decision, the decision maker can use Analytic Hierarchy Process (AHP). AHP has been use in many areas such as in social, personal, politic, education, government, management, sport, industry, and so on.

The main component of AHP is priority derivation procedure. This research found three main problems in priority derivation procedure in AHP. The first one is the problem with the inconsistency of the judgments. The consistency of the judgment is important in making decision. However, inconsistency of the judgment always happens and it will affect the accuracy in selecting alternative. Second, the effect of using non-EC prioritization methods in deriving priorities, and the last is the accuracy performance of the non-EC prioritization methods.

This paper proposed a priority derivation procedure based on Evolutionary Computing (EC) in AHP (AHPEC) in order to solve the mention problems.

2 Analytic Hierarchy Process (AHP) Problems

In 1977, Saaty [3] introduced AHP to solve the Multi Criteria Decision Making (MCDM) problem. MCDM is referring to making decision in the presence of multiple criteria.

Zahedi [2] summarizes the original AHP procedure by Saaty into four phases:

- 1) Break the decision problem into a hierarchy of interrelated problems.
- 2) Provide the matrix data for pairwise comparison of the decision elements.
- 3) Using Eigenvector Method (EV) as a prioritization method.
- 4) Aggregate the relative weights of the decision elements to obtain a rating for decision alternatives.

Since AHP has been introduced, it has been applied in numerous situations with impressive results. However, AHP has been also criticized in the literature review, mainly in priority derivation procedure. The priority derivation procedure is the process of deriving scores in order to set the relative order of importance of alternatives.

This paper has identified three mains problems in current priority derivation procedure which is explained in the next section.

2.1 Inconsistency of the judgment

The original prioritization method that has been proposed is Eigenvector Method (EV) by Saaty in 1977. Consistency in EV is measured by using consistency ratio (CR). It is designed in such a way if the value of CR greater than 0.10 which indicates an inconsistency in the judgments was happens. When inconsistency of the judgment happens, it will affect the accuracy in selecting alternative. Then, the decision maker must repeat the judgment again until the decision maker gets the value of CR equal to 0.10 or less.

Because of these problems, a new prioritization method which can derive priorities in AHP in both situations, consistent or inconsistent of the judgment is needed.

2.2 Non-evolutionary computing approach

In deriving priorities, there are two approaches in AHP: the non-optimization approach and the optimization approach. The non-optimization approach is referring to Eigenvector Method (EV), which is the original prioritization method in AHP. For the prioritization methods in optimization approach, they share the same characteristic where priorities are derived by solving single objective optimization problem using non-EC approach. Table 1 shows the list of non-EC prioritization methods that have been proposed by previous researchers to derive priorities in AHP.

Table 1: List of non-EC prioritization methods from previous researchers

Author	Prioritization Method
Srdjevic [3]	Combining different prioritization method
Mikhailov [4]	Fuzzy Preference Programming (FPP)
Bryson [5]	Goal Programming Method (LGP)
Crawford and Williams [6]	Logarithmic Least Square Method (LLS)
Chu et al. [7]	Weighted Least Square Method (WLS)
Saaty [8]	Eigenvector method (EV)

However, the non-EC approach has several weaknesses which are:

- When specific problem knowledge is not available, the optimal solution is difficult to generate.
- Several optimization runs are required to obtain the optimal solution.

2.3 Accuracy performance of the prioritization method

The accuracy of the solution can be measured by using Total Deviation (TD). TD is used as an objective function to measure accuracy of the solution to solve SOP. The smaller or close to zero value of TD is the most accurate in term of solution. TD is presented as:

$$TD = \sum_i \left(\sum_j \left(\frac{w_i}{w_j} - a_{ij} \right)^2 \right)^{1/2}$$

Table 2 shows the comparison of performance between non-EC prioritization methods based on TD as objective function for two case studies. For the case study 1 which is problem of reservoir storage allocation, the best prioritization method is LLS. However, for case study 2 which is selecting the high school, the best prioritization method is by using FPP.

Table 2: Comparison of accuracy performance of prioritization methods

Case Study	Ranking of the accuracy performance of the solution (TD value)				
	1	2	3	4	5
1. Reservoir storage allocation [3]	LLS	EV	WLS	LGP	FPP
2. Selecting the high school [3]	FPP	WLS	LLS	EV	LGP

Based on the Table 2, even though the performance of LLS is the best method in case study 1, somehow it is not the best method in case study 2. In addition, according to the case study 1, FPP gives the worst result which is contrast to the case study 2 where the method gives the best result among to the other methods. From the results, we can conclude that there is no method is favorable for both cases in term of accuracy performance.

Therefore, this paper proposes AHPEC in solving the problems that has been faced in AHP.

3 The AHPEC

This paper proposed AHPEC which is using EC to derive priorities in AHP. EC uses evolutionary approach which is based on computational model of natural selection and genetics [9].

In solving single objective optimization problem for deriving priorities by using AHPEC, it involves three operators which are selection, crossover, and mutation. In this paper, Roulette wheel selection (RWS) has been selected as a method in doing selection. For crossover, this paper using single-point crossover and using uniform mutation for mutation type. In order to ensure that only the best population always survives, elitism has also been applied as an additional selection strategy.

To implement EC in AHP, the first step is parent population of population size n is created by generate random number. Then, the fitness value is calculated by using TD as objective function. After that, the operation selection, crossover, mutation and elitism are applied. From that, the new offspring are ready to go to the next generation. The process continued until it reaches the generation size.

Fig. 1 shows the procedure of AHPEC in deriving priorities in AHP, and Fig. 2 represents the pseudo code of the implementation of AHPEC.

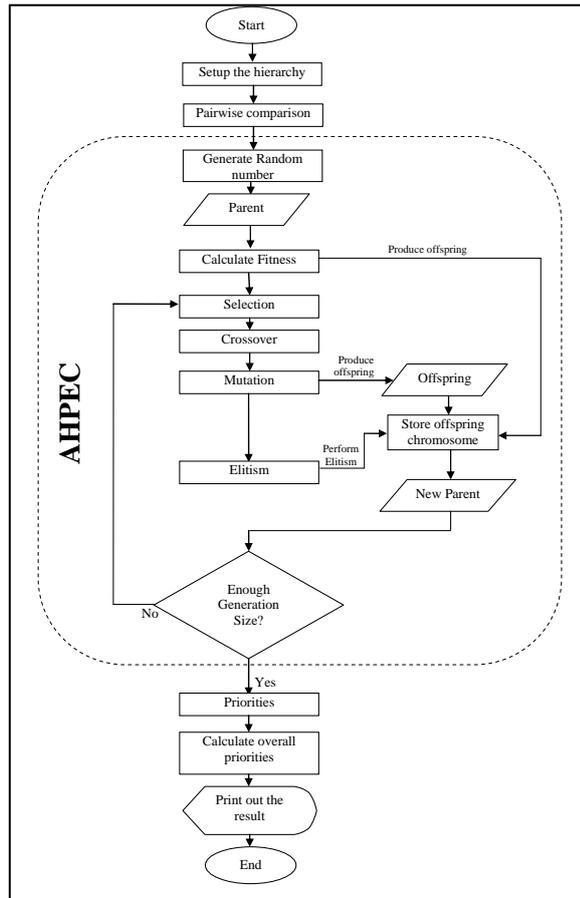


Fig. 1 The procedure of AHPEC

3.1 Parameter Setting

By using EC, parameter setting is the important part in getting the better result. The performance of EC is greatly dependent on its turning of parameter. This paper proposes a new parameter setting. It is tested with two case studies from Srdjevic paper [3]. Table 3 shows the parameter settings for the previous researchers which are Dejong setting [10], Grefenstette's setting [11], and MicroGA setting [12] and the proposed parameter setting for this paper.

Table 3: The parameter setting

Parameter setting	Dejong	Grefenstette	MicroGA	AHPEC
Population size	50	30	5	10
Number of generation	1,000	Not Specified	100	100
Crossover rate	60%	90%	50%	75%
Mutation rate	0.1%	1%	2% and 4%	10%

Fig. 3 and Fig. 4 show the comparison between the proposed setting and others parameter settings.

Based on Fig. 3 and Fig. 4, the AHPEC parameter settings give much better result compare to the others parameter settings.

n = size of population
 m = size of matrix
 k = generation size
Input: the inconsistent matrix $A = (a_{ij})$ or the consistent matrix $A' = (a'_{ij})$

Output: the priorities estimation in the AHP

1. **for** ($i = 1$ to $n, j = 1$ to n) **do**
2. generate random number
3. construct column normalized matrix $B = (b_{ij})$;
4. generate random number which the priority $W = [w_1, w_2, \dots, w_m]$;
5. calculate Total Deviation $C = (c_{ij}), c_{ij}$

$$= \sum_i \left(\sum_j \left(\frac{w_i}{w_j} - b_{ij} \right)^2 \right)^{1/2}$$

6. **end for**
7. **do** selection, crossover, and mutation
8. double $\maxTotalDeviation = 0$; double $\minTotalDeviation = 0$;
9. **for** ($i = 1$ to $n, j = 1$ to n) **do**
10. double $TotalDeviation_{ij} \leftarrow c_{ij}$;
11. **if** ($TotalDeviation_{ij} > \maxTotalDeviation$) **then**
12. $\maxTotalDeviation \leftarrow TotalDeviation_{ij}$;
13. **end if**
14. **else** ($TotalDeviation_{ij} < \minTotalDeviation$) **then**
15. $\minTotalDeviation_{ij} \leftarrow TotalDeviation_{ij}$;
16. **end for**
17. **do** elitism
18. **if** ($! = k$), goto 1 to take the next turn.

Fig. 2 The pseudo code of AHPEC

3.2 The results

In applying the procedure of AHPEC and AHPEC parameter setting, an experiment has been conducted based on data that are taken from two case study in Srdjevic's study [3]. The results obtained from his study and the result using AHPEC are list in table 4 for the first case study which is problem of reservoir storage allocation. The P1, P2, P3, ..., P6 represent the pairwise comparison matrix used in the case study. Table 5 shows the second case study which is problem of choosing high schools.

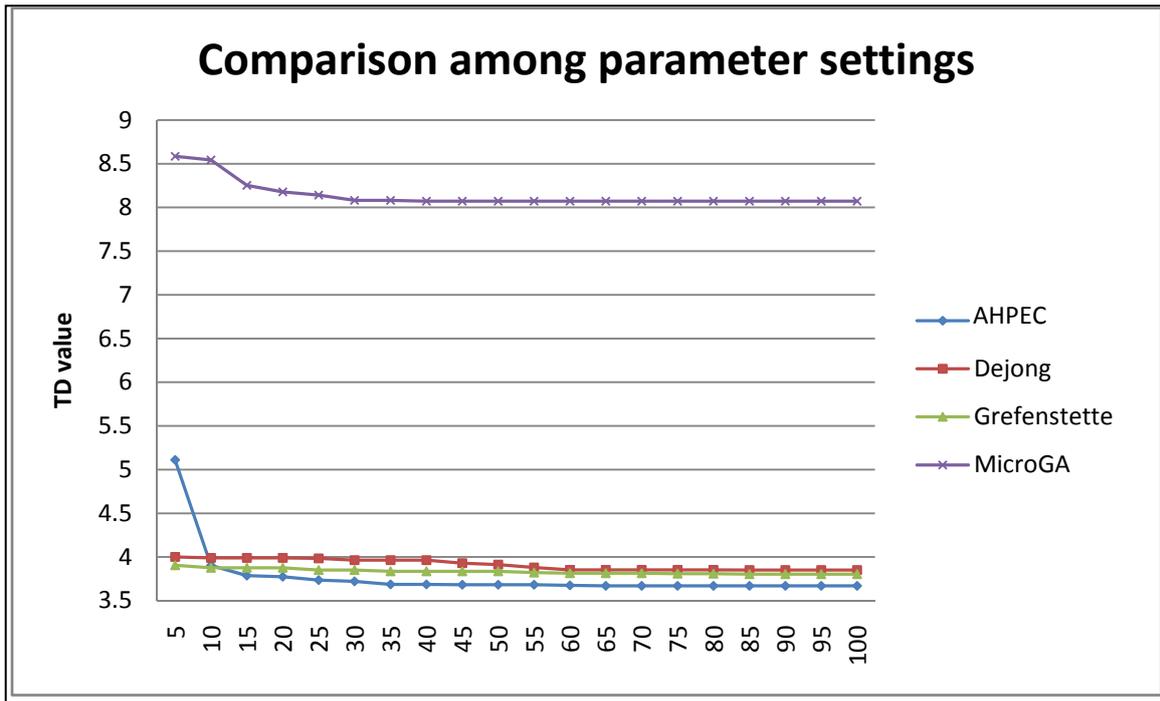


Fig. 3 The result based on four parameter setting (case study 1)

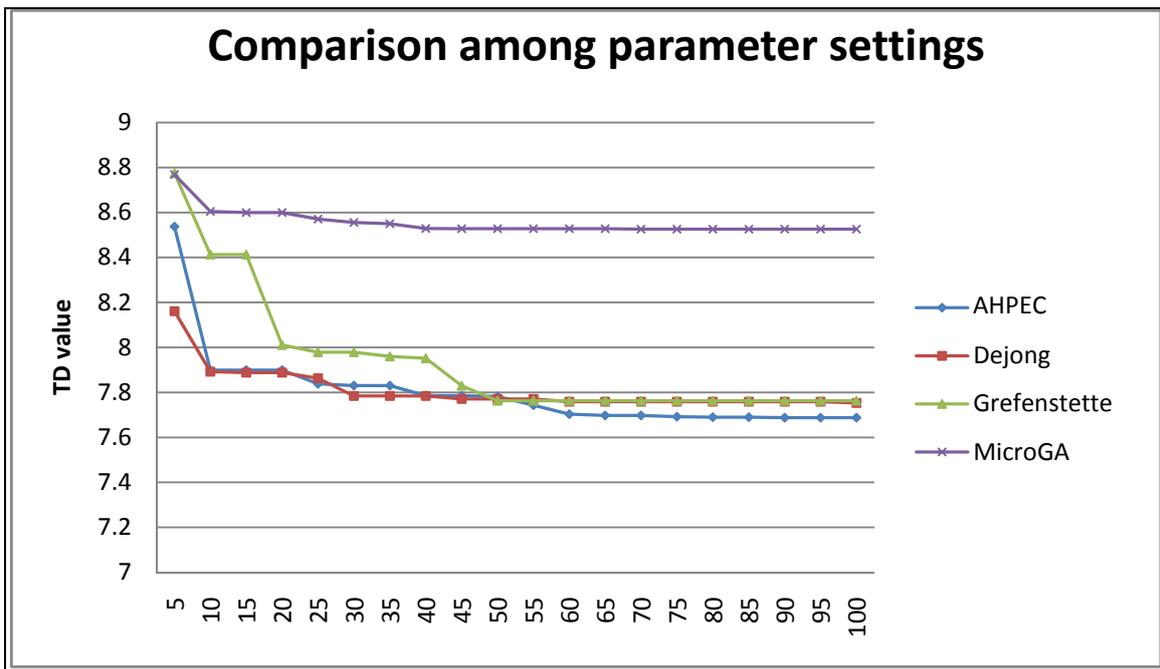


Fig. 4 The result based on four parameter setting (case study 2)

Table 4: The result between non-EC prioritization methods and AHPEC method in term of value of TD in case study 1 from Srdjevic's study [3].

No.	Prioritization method	Value of TD					
		P1	P2	P3	P4	P5	P6
1.	Additive normalization (AN)	4.583	6.209	5.305	6.797	6.107	4.786
2.	Eigenvector (EV)	4.961	6.255	5.359	7.382	6.451	5.055
3.	Weighted least-squares (WLS)	5.508	6.937	5.204	7.114	7.054	5.331
4.	Logarithmic least-square (LLS)	4.813	6.119	5.289	7.327	6.627	4.642
5.	Logarithmic Goal programming (LGP)	5.444	8.027	6.904	7.318	6.634	8.740
6.	Fuzzy preference programming (FPP)	4.550	8.227	5.607	7.005	7.643	8.162
7.	AHP using Evolutionary Computing (AHPEC)	3.667	5.671	4.608	5.849	5.091	4.497
The smaller value of TD		AHPEC	AHPEC	AHPEC	AHPEC	AHPEC	AHPEC

Table 5: The result between non-EC prioritization methods and AHPEC method in term of value of TD in case study 2 from Srdjevic's study [3].

No.	Prioritization method	Value of TD				
		P1	P2	P3	P4	P5
1.	Additive normalization (AN)	8.795	1.063	0.000	0.000	5.011
2.	Eigenvector (EV)	9.479	1.091	0.000	0.000	6.032
3.	Weighted least-squares (WLS)	11.757	0.960	0.000	0.000	3.856
4.	Logarithmic least-square (LLS)	9.235	1.091	0.000	0.000	6.032
5.	Logarithmic Goal programming (LGP)	9.285	1.126	0.000	0.000	3.645
6.	Fuzzy preference programming (FPP)	28.667	1.118	0.000	0.000	3.764
7.	AHP using Evolutionary Computing (AHPEC)	7.748	0.932	0.000	0.000	3.622
The smaller value of TD		AHPEC	AHPEC	AHPEC	AHPEC	AHPEC

Based on Table 4 and 5, AHPEC produce the smaller or close to zero the value of TD for every single P1, P2, ..., and P6. These result also show that the AHPEC can be use to derive priorities for both situation of inconsistency and consistency of the judgment.

Nowadays, EC has been used in many areas and it gives an excellence result to solve many problems. It also happens in this paper when EC give an excellent result in deriving priorities in AHP.

The advantages of using EC in AHPEC are:

1. EC does not have much mathematical requirements about the optimization problems. EC can handle any kind of objective functions and any kind of constraints (linear or nonlinear), defined on discrete, continuous or mixed.
2. By using EC, with single optimal run, the optimal solution can be found.
3. EC provides adaptability and give a good performance in solving the problem as shown in Table 4 and Table 5.

4 Conclusion

This paper proposed AHPEC as an approach to derive priorities in AHP. This paper SOP optimization problem, by maximizing accuracy of the solution which is by using TD as criterion. AHPEC gives much better result compared to the other non-EC prioritization methods. From the experiment that has been done, we can conclude that AHPEC is more adaptable, accurate, and flexible. This feature enables it to be applied to all case studies.

For the further research, the design and development of the prototype will be done to demonstrate the practicability of AHPEC.

References:

1. Turban, E. and J.E. Aronson, *Decision Support Systems and Intelligent Systems*, ed. 5th. 1990, New Jersey: Prentice Hall.
2. Zahedi, F., *The Analytic Hierarchy Process- A Survey of the Method and its Application*. Interfaces, 1986. **16**: p. 96-108.
3. Srdjevic, B., *Combining different prioritization methods in Analytic Hierarchy Process synthesis*. Computer & Operations Research, 2005. **32**: p. 1897-1919.
4. Mikhailov, L. and M.G. Singh. *Comparison analysis of deriving priorities in the analytic hierarchy process*. in *proceeding of the IEEE International Conference on Systems, Man and Cybernetics*. 1999.
5. Byson, N., *A goal programming method for generating priorities vectors*. Journal of Operational Research Society, 1995. **46**: p. 641-648.
6. Crawford, G. and C. Williams, *A note on the analysis of subjective judgment matrices*. Journal of Mathematical Psychology, 1985. **29**: p. 387-405.
7. Chu, A., R. Kalaba, and K. Springam, *A comparison of two methods for determining the weights of belonging to fuzzy sets*. Journal of Optimization Theory and Applications, 1979. **27**: p. 531-541.
8. Saaty, T.L., *A Scaling Method for Priorities in Hierarchical Structures*. Journal of Mathematical Psychology, 1977. **15**: p. 234-281.
9. Negnevitsky, M., *Artificial Intelligence*, ed. s. edition. 2005: Addison-Wesley.
10. Dejong, K.A. and W.M. Spears. *An analysis of the Interacting Roles of Population Size and Crossover in Genetic Algorithms*. in *Proc. First Workshop Parallel problem Solving from Nature*. 1990. Berlin: Springer-Verlag.
11. Grefenstette, J.J., *Optimization of control parameters for Genetic Algorithms*. IEEE Trans. System, Man, and Cybernetics, 1986. **16**: p. 122-128.
12. Carroll, D.L. *Genetic Algorithm (GA) Tips*. [cited 2009 8 Sept 2009]; Available from: <http://cuaerospace.com/carroll/gatips.html>.