Metric's Thresholds for Encoding Evolutionary Computing Representation in Software Engineering Problem

Abubakar D. Bakar, Abu Bakar Md. Sultan, Hazura Zulzalil, and Jamilah Din

***Abstract*—Recently, evolutionary computing is gaining more interest from software engineering community for using it to solve various types of related problems. The software metrics selection problem is among the problems implemented using this technique. For the problem to be implemented using this method, the initial process called representation should be established. Currently, the conversion of the problem into binary representation is done using human experts. However, experts' opinions in categorizing the defect and non-defect prone classes into bits are always questionable due to human weakness. This paper presents metrics thresholds encoding algorithm as an alternative to replace human experts in developing problem representation. Algorithm was tested to metric selection problem using Genetic Algorithm and the results obtained are promising.**

***Index Terms*—Genetic algorithm, software metrics, software metrics thresholds, evolutionary computation, and representation.**

I. Introduction

Genetic algorithm (GA) is the most used evolutionary computation for solving exhaustive problems in software engineering [1]. Among the extensive search problem of this type is software metrics selection done by [2]. He used GA method to propose the model for selecting the best software metrics in predicting software maintainability. It is known that for the problem to be solved by GA, the conversion of the problem into specific format is basic requirement [3]. This representation process determines the efficiency and accuracy of the final model. In this practice the appearance, behavior and physical qualities of individuals can be encoded into binary format [4].

In the GA problem by [4], software engineering experts were used to categorize defect and non-defect classes into binary digits. However, the study provides interesting results, but the use of expert opinions at the base stage of his GA model was entrusted. Like many other previous works which were based on human involvement, subjective strategies experienced a lot of criticisms. Among the critical argument is the nature of human decisions in the ranking process where decisions depend on what is in the mind of the people at the particular time [5]-[8].

This paper presents software metrics thresholds encoding algorithm as alternative to human experts at the presentation phase of the GA. This algorithm proposed as an empirical technique in opposing the current experts' opinions for the Evolutionary algorithms (EA) chromosome formulation. The algorithm was evaluated and validated by applying the technique to the practical model. The model was the initial implementation of GA software metrics selection model aimed to predict the software maintainability. The promising results of the final GA model are intuition toward other evolutionary computation methods. The final developed model was compared with Principal Component Analysis (PCA) and software metrics suite to evaluate its performance. In addition, the model used two different case studies to get more clarification.

The rest of the paper is organized as follows: Section II presents state-of-art of the evolutionary computation, specifically the representation phase. Section III presents the proposed algorithm. Section IV presents evaluation and validation of the algorithm. Results and discussions of the study are provided in Section V, and concluding remarks are given in Section VI.

II. Evolutionary Computation Representation

Any problem that is intended to be implemented using Evolutionary algorithms should pass the modification stage to make it appropriate for manipulation. This initial stage determines the quality of the final output of an algorithm. Many practitioners have acknowledged the potentiality of this stage in providing quality results of the final model. Reference [9] found a direct impact of problem representation toward the performances of intelligent systems. Similarly, [10] found the effect of representation as an important thing that can affect the search-based algorithms in software engineering. They cautioned that the mis-interpretation of the problem at this stage may lead to low performances of the final model being implemented. They also advised that the problem intended to be implemented using the GA should pass the appropriate modification stage to make it appropriate for manipulation. Thus, [9] found a direct impact of problem representation in the performance of intelligent systems, whereas [10] found the effect of representation stage in the search algorithms.

In this study, the problem used binary conversion which conforms to the GA context [4]. The list of individual entities in the problem domain is interpreted as series of bits called chromosomes. This method of encoding the genes into binary is simpler and most accurate, and it simplifies the crossover and mutation operations. Other representation processes like value encoding, and tree encoding are also used, but the

Manuscript received March 9, 2014; revised May 9, 2014.

Abubakar D. Bakar, Abu Bakar Md. Sultan, Hazura Zulzalil, and Jamilah Din are with the System Department, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 Serdang, Selangor, Malaysia (e-mail: abakar@upm.edu.my).

DOI: 10.7763/LNSE.2015.V3.158

binary-based encoding has been acknowledged to have more impact in GA [4].

III. Proposed Algorithm

Algorithms are introduced in software engineering for solving complex problem faster and accurately. In this study, we introduced binary encoding algorithm for the Genetic algorithm representation. Algorithm was used in the conversion process of the software metrics values from the software classes into the binary format. Previously, the practice was done by the software engineering experts who classified error and non-error prone classes into bits. The proposed algorithm is illustrated in the script below.

Given the metric value λ and the metric threshold *t*, then the binary will prompt the user to enter metric value λ

```
Get metric threshold $t$
If $t$ is Lbound then
    If $\lambda > t$ then Assign 1
    Else Assign 0
Else    //if $t$ Ubound then
    If $\lambda > t$ then Assign 0
    Else Assign 1
```

The flow chart in Fig. 1 shows the conceptual modeling of the algorithm. The process starts with the user entering the metrics values collected from the class. The values are then compared with the threshold for the metric concerned, which will give the binary values based on the quality design of the property measured by the metric.

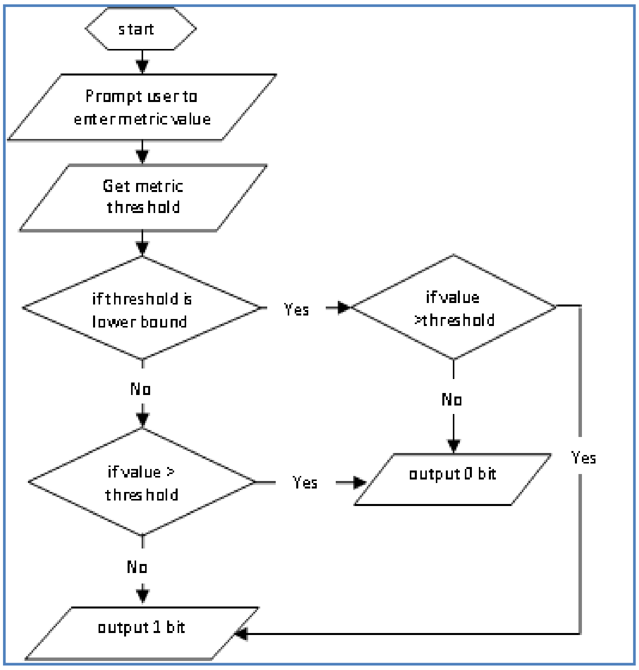

Fig. 1. Conceptual modeling of the algorithm.

IV. Evaluation and Validation

We employed the proposed algorithm in the encoding technique for the GA representation in software metrics selection model for predicting software maintainability. The metrics selected was evaluated using Linear Discriminant Analysis (LDA). LDA is a statistical technique used to calculate the classification performance of the three top most effective metrics selected by the GA. The following expression illustrates the relationship between the recall and precision in calculating the performance measurements of the selected metrics.

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

The LDA is used as a classifier in machine learning to find a linear combination of features. The methods used precision and recall to determine the performance measurement of the phenomena. In this case, the GA model was validated by comparing with other traditional classification models. The traditional techniques were software metrics suite and Principal Component Analysis (PCA). The model was then replicated to another software product as the representative for the unseen case.

The study was also utilizing statistical analysis to find the significance of the results. ANOVA (Analysis of Variance) was used to compare means of performance among different methods of metrics selection models. The treatment means with significance difference were then separated by Tukey's SR (HSD) [11] Test. Analyses were performed using Statistical analysis software (SAS 9.3).

V. Results and Discussions

The paper intended to presents the proposed software metrics encoding algorithm for the GA representation. This algorithm was used in the representation phase of the proposed GA software metrics selection model for predicting software maintainability. The motive behind the introduction of this encoding algorithm is to reduce the human involvement in models development in combating the criticisms of subjective opinions in the development of models. This algorithm facilitates the quick availability of GA chromosomes without human involvement.

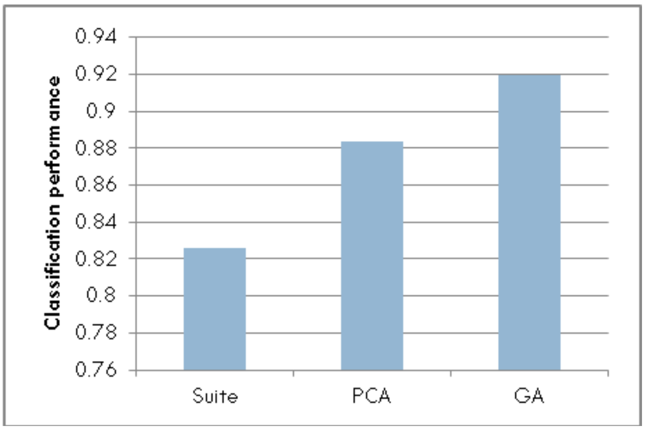

Fig. 2. Geotool results: GA with other classical approach.

In the first case where Geotool software was used, the algorithm seemed to give interesting results. The GA was found to have noteworthy performance over the Principal Component Analysis (PCA) and the software metrics suite. The classification performances were 92%, 88%, and 82%

for GA, PCA, and Suite, respectively. Fig. 2 shows the classification performances for three different classification models.

The same scenario was replicated using another case, and similar results were observed. In the second dataset called Geoserver, the metrics selection performance for the GA counts was 90%, whereas for the PCA was 89%, and for the CK suite was 79%. Fig. 3 shows the performances for the metrics selected to predict Geoserver software systems maintainability. Again, the results followed the similar trend.

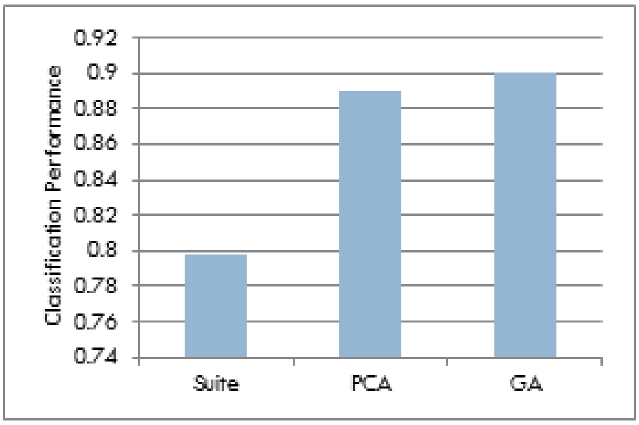

Fig. 3. Geoserver results: GA compared with other classification methods.

In terms of spotlighting a broad-spectrum of the results presented, the two assistant techniques showed different results in terms of overall selection performance. Fig. 4 shows the trends of the techniques for the two datasets used.

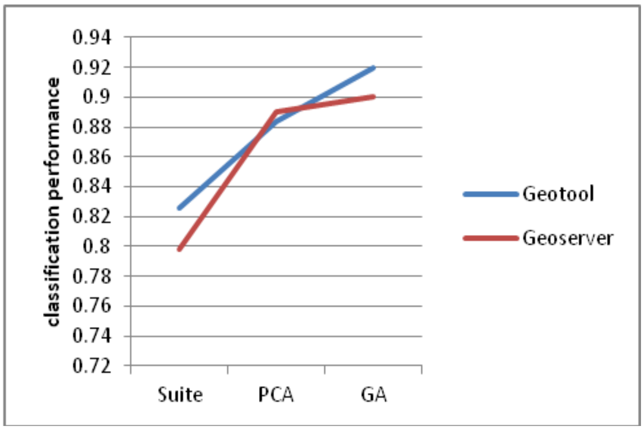

Fig. 4. The trends of the results obtained from two datasets.

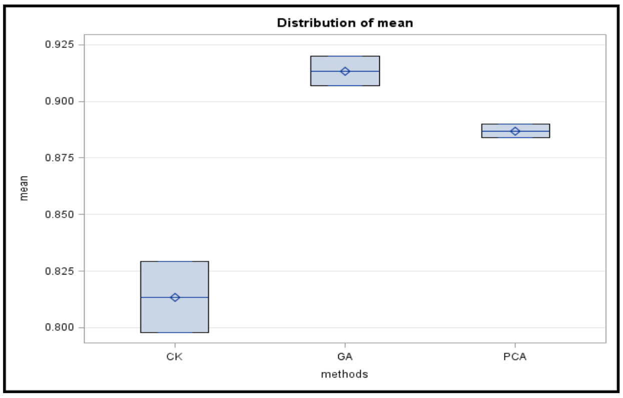

Fig. 5. Classification performance distribution mean for the GA model in comparison with the other classical approach .

The performances of the three methods were also validated using statistical analysis. The study used Statistical analysis software (SAS) for two specific aims. The first aim was to investigate the distribution for the selection performance among two treatments. Fig. 5 represents the distribution mean for the three classification methods.

The second aim was to compare the result of classification performance in the two treatments; Geotool and Geoserver. The outcome is presented in Fig. 6. The figure shows the selection performance mean for the three software metrics selection models in both treatments.

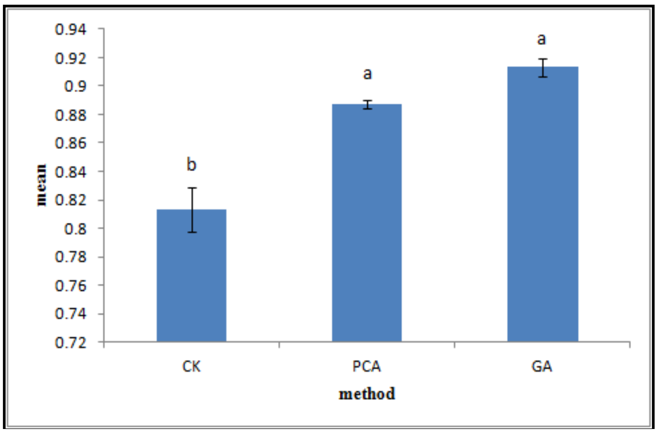

Fig. 6. Classification performance mean for the GA model in comparison with the other classical approaches.

Significant difference was observed among the different software (*F*= 26.92, *P*=0.01). According to results, the mean of GA and PCA were significantly higher than CK, however, the difference was observed between GA and PCA was insignificant. The mean number of GA, PCA and CK were (0.91338 ± 0.00647), (0.88694 ± 0.00290) and (0.81351± 0.01575) respectively.

The results are promising. The algorithm has not only introduced the encoding strategy and technique that eradicate human dependent in model development, but also the technique that is more efficient and promising. The results proved that the algorithm is applicable in different classification methods and in using unseen datasets. In addition, practitioners can use this algorithm in other evolutionary computations as representation is mandatory in any EC problem.

VI. Conclusions and Recommendations

This article increases our understanding on how evolutionary algorithms, particularly, GA could assist in solving various software engineering problems. The threshold method was presented to replace human expertise to encode problem representation in metrics selection problem. The threshold method provides more accurate values compared with human judgment thus yielding better prediction results for metrics selection model. Our future plan is to further investigate the practicality of metrics thresholds for different types of software engineering problem.

References

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems*, Ann Arbor, MI: The University of Michigan Press, 1975, ch. 1.

[2] R. Vivanco, "Improving Predictive Models of Software Quality Using an Evolutionary Computational Approach," presented at IEEE International Conference on Software Maintenance, October 2-4, pp. 503-504, October 2-5, 2007.

[3] R. Popa, *Genetic Algorithms in Applications*, Intech, Croatia, 2012.

[4] F. Rothlauf, *Representations for Genetic and Evolutionary Algorithms*, 2nd ed., Berlin Heidelberg: Springer-Verlag, 2006, ch. 3.

[5] K. Khosravi and Y. Guéhéneuc, "On issues with software quality models," in *Proc. the 11th Working Conference on Reverse Engineering*, vol. 11, pp. 172-181, 2004, Canada: GEODES.

[6] R. Knauf, S. Tsuruta, and A. Gonzalez, "Overcoming Human Weaknesses in Validation of Knowledge-Based Systems," *Leipziger Informatik-Tage*, vol. 72, pp. 254-263, 2005.

[7] R. Knauf, S. Tsuruta and A. Gonzalez, "Toward Reducing Human Involvement in Validation of Knowledge-Based Systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 37, pp. 120-131, 2007.

[8] R. Baggen, J. Correia, K. Schill, and J. Visser, "Standardized Code Quality Benchmarking for Improving Software Maintainability," *Software Quality Journal*, vol. 20, no. 2, pp. 287–307, 2012.

[9] R. Zimmer, R. Holte, and A. MacDonald, "The Impact of Representation on the Efficacy of Artificial Intelligence: The Case of Genetic Algorithms," *AI & Society*, vol. 11, no. 1, pp. 76-87, 1997.

[10] Y. Zhang, K. Wang, D. Shaw, J. Miles, I. Parmee, and A. Kwan, "Representation and Its Impact on Topological Search in Evolutionary Computation," *Joint International Conference on Computing and Decision making in Civil and Building Engineering*, June 14-16, 2006, Montreal, Canada, pp. 2359-2368.

[11] *SAS/STAT user's guide*, SAS Institute Inc., Cary, NC. USA, 2010.

Abubakar D. Bakar was born in Zanzibar, Tanzania in 1977. He obtained his BSc in computer science in 2004. He received his MSc in computer science at University of Dar es salaam Tanzania, 2010. Currently, he is a PhD student in the Faculty of Computer Science and Information System at University Putra Malaysia (UPM).

His research interest is search-based software engineering. Mr. Abubakar is a member of software engineering group, Faculty of Computer Science at University Putra Malaysia.

Abu Bakar Md. Sultan was born in Melaka, Malaysia in 1965. He holds a PhD in artificial intelligence from University Putra Malaysia (UPM) in 2007.

His main research areas are artificial intelligence and software engineering, particularly search-based software engineering (SBSE). He has published several articles in conferences and various journals related to SBSE.

Associate professor Dr Abu Bakar Md. Sultan currently is the head of Information System Department, Faculty of Computer Science and Information Technology, UPM.

Hazura Z. was born in Kuching, Sarawak on the 28th of February 1972. She received her bachelor degree in computer science from University Agriculture of Malaysia in 1995 and master degree in software engineering from University Putra Malaysia in 1998. She obtained her PhD in Software Engineering from the same university in 2011.

Currently, Dr. Hazura is a senior lecturer at the Faculty of Computer Science and Information Technology, University Putra Malaysia.

Her research interests are software metrics, software quality and software product evaluation.

Jamilah Din was born in 1965 in Malaysia. She received her BSc in computing science from University of Evansville, Indiana 1987, MSc in computer science from University Putra Malaysia (UPM) 2002 and PhD in software engineering from National University of Malaysia (UKM) in 2009. She has more than six years of experience in industry before joining the academic field.

Her research interests and works are in the areas of object-oriented analysis and design and software engineering.

Dr. Jamilah is now a Senior lecturer at Faculty of Computer Science and Information Technology, UPM.