International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 6, December 2018, pp. 4735~4744
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i6.pp4735-4744

PORM: Predictive Optimization of Risk Management to Control Uncertainty Problems in Software Engineering

Salma Firdose[1], L. Manjunath Rao[2]
[1]Bharathiar University, India
[2]Department of MCA, Dr. Ambedkar Institute of Technology, India

Article Info

Article history:

Received Mar 6, 2018
Revised Jun 6, 2018
Accepted Jul 22, 2018

Keyword:

Risk factors
Risk management
Software engineering
Software projects
Software risk
Uncertainty

ABSTRACT

Irrespective of different research-based approaches toward risk management, developing a precise model towards risk management is found to be a computationally challenging task owing to critical and vague definition of the origination of the problems. This research work introduces a model called as PROM i.e. Predictive Optimization of Risk Management with the perspective of software engineering. The significant contribution of PORM is to offer a reliable computation of risk analysis by considering generalized practical scenario of software development practices in Information Technology (IT) industry. The proposed PORM system is also designed and equipped with better risk factor assessment with an aid of machine learning approach without having more involvement of iteration. The study outcome shows that PORM system offers computationally cost effective analysis of risk factor as assessed with respect to different quality standards of object oriented system involved in every software projects.

Corresponding Author:

Salma Firdose,
Bharathiar University,
Coimbatore, Tamilnadu, India.
Email: salmaf.phd@gmail.com

1. INTRODUCTION

With the increasing race towards offering quality delivery of software projects, the software development team in various IT industries is in continuous exploration process for such effective development methodologies. At present there are various standard software development methodologies [1], [2] as well as quality standard [3], [4] that are considered to be technological boon for every project development team. One of the techniques to ensure an effective software development practices is to ensure higher degree of risk control measures that calls for an effective risk management [5]. Basically, a risk management in software development industry is all about considering all sorts of possible factors that could degrade the product quality or invite some unfortunate challenges in near future during the development period that could possible cost some tangible resources [6], [7].

There are various literatures e.g. [8], [9] which states that various standard risk management models, frameworks, practices, etc are already existing. Inspite of this, there is always an unprecedented fear of risk due to following reason e.g. change of management, policy alternation, skill gap, employee attrition, requirement volatility, sudden adoption of unknown technology etc. Basically, there are various types of risk factors where some are quite known i.e. deterministic form while some of absolutely unknown. In the first type of risk, the team is completely aware of the risk and has all the chances to ensure an effective control of the risk. However, still various factors e.g. time involved in controlling the risk, resource involved, cost etc are some factors that are also to be considered while mitigating such forms of risk.

Unfortunately, the second form of risk factor is something that the software development team as well as stakeholder has absolutely no pre-defined information or clue about the form of the risk. It is a direct indication of involvement of uncertainty factor in the risk management. There are various studies where uncertainty problems [10], [11] have been discussed as the most challenging problem with respect to computational model. It is because such forms of problems cannot be mathematically formulated as variables corresponding to the problems cannot be discretely defined. Hence, an effective risk management actually suffers from such forms of problems where there is absolutely no benchmarked model or solution in order to assists such computationally challenging problems.

One way to develop such model will be to take a case study and define various constraints appropriate to case study and perform different forms of iterations to check how the model behaves in risk evaluation. Hence, adoption of different optimization algorithms [12], [13] are highly recommended in this case as they could offer a good balance between constraint satisfaction as well as minimize the occurances of risk. However, while doing such design approach, it calls for various rounds of iteration where some rounds of iterations could be sufficiently big just in order to obtain an elite outcome. Such approaches may be completely unpractical even if they offer good outcomes. Therefore, there is a need to design and develop an efficient computational model that is capable of controlling the risk factor to a good extent that is applicable in practical life [14]-[16].

This research paper introduces one such dedicated attempt where a simple and cost effective modeling is carried out using analytical methodology in order to compute the practical form of risk as well as to ensure the accuracy in it. The secondary research objective is to offer reliable outcome of risk computation with an aid of machine learning approach. The tertiary research objective of this paper is to ensure a good computational model that could take the real-time constrainsts as input and offer reliable risk evaluation to assists stakeholders for formulating countermeasures. Section 1.1 discusses about the existing literatures towards risk management followed by discussion of research problems in Section 1.2 and proposed solution in 1.3. Section 2 discusses about algorithm implementation followed by discussion of result analysis in Section 3. Finally, the conclusive remarks are provided in Section 4.

1.1. Background

This section discusses about the existing literatures studies towards risk management as an extension to our prior review work [17]. Discussion towards importance of risk as critical system was put forward by Laplante and DeFranco [18], Lutz and Huang [19]. A comparative illustration of diferent risk frameworks have been discussed by Pasha *et al.* [20]. It has been seen that risk management offers complementart benefits while exercised on existing quality standards.

The study of Albadarneh *et al.* has discussed a case study of agile methodology with respect to its significant benefits [21]. Sundararajan *et al.* have investigated towards the large scale of project development for assessing the risk associated with it [22]. Literatures have also witnessed implications of decision making towards risk management as seen in work of Aslam *et al.* [23]. Importance of similar practices of risk analysis towards agile methodology was also found supported in the work of Elbanna and Sarker [24]. Batbayar *et al.* [25] have presented a study where statistical tool has used for assessing risk along with applying fuzzy logic. The studies also shows that risk associated with the software design patterns can be also assessed for their risk factor using specific bound approach as reported by Bernardi *et al.* [26]. Adoption of modeling approach for using social and technical system has been proven to improve the software design as well as offer better risk management as reported by Bider and Otto [27]. Similar form of research work has been also reported by Chadli *et al.* [28].

Literatures have also introduced integrated-based approach where joint implementation of different forms of discrete process is found to assists in minimizing occurances of risk. This fact was discussed by Janjua *et al.* [29]. Work of Kendall *et al.* [30] have introduced a practical centric methodology for an effective governance of risk factor. Lueddemann *et al.* [31] have presented an experimental-based approach for assessing standard ISO risk associated with clinical devices. Pietrantuono *et al.* [32] have presented architecture for ensuring effective software reliability.

Adoption of probabilistic-based approach was reported to minimize the complexiy associated with evaluation of risk associated with medical devices. This fact was further found advocated in the work of Rao *et al.* [33], [34] that inclines towards identification of risk factor. Study towards taxonomies of risk management was discussed by Reyes *et al* [35]. Adoption of evolution technqiues are also reported to offer better management and control of risk during the software project development. This fact was discussed by Sarro *et al.* using multi-objective based approach in order to incorporate adaptivity in their design [36]. Usage of software intelligence was investigated by Susarev *et al.* in order to study about the safety factors essential for essential software management [37]. There exists various forms of research-based studies towards risk management but very less models are found to be investigated from the viewpoint of software engineering. Moreover, very less computational model with significant benchmarking has been reported. The next section outlines the research problems that are arrived from reviewing the existing approach towards software risk management.

1.2. The Problem

The significant research problems are as follows:

a. Existing research work are more theoretical / conceptualized model and lacks computational modeling aspect with respect to analytical solution approach.
b. Undertaking of real-time risk factor and formulating it in analytical modeling is less witnessed in existing approach that reduces the applicability of claimed success of model.
c. There is no major optimization technqiues implemented over software engineering with many of them are found not to focus on reducing the iteration.
d. More inclination towards modeling aspect and less towards exploring the applicability of the modeling in real-time scenario.

Therefore, the problem statement of the proposed study can be stated as "*Developing a cost effective computation model to offer a precise evaluation of the critical risk factor for the given set of operational data with reliable predictive performance is quite challenging task*". The next section discusses about the proposed methodology used to countermeasure the above stated research problem.

1.3. The Proposed Solution

The proposed work is basically an extension of the our prior design approach called as 3LRM [38] where the present work focuses on optimizing the performance of the computing risk factor involved in software engineering. The architecture of the proposed PORM is shown in Figure 1.

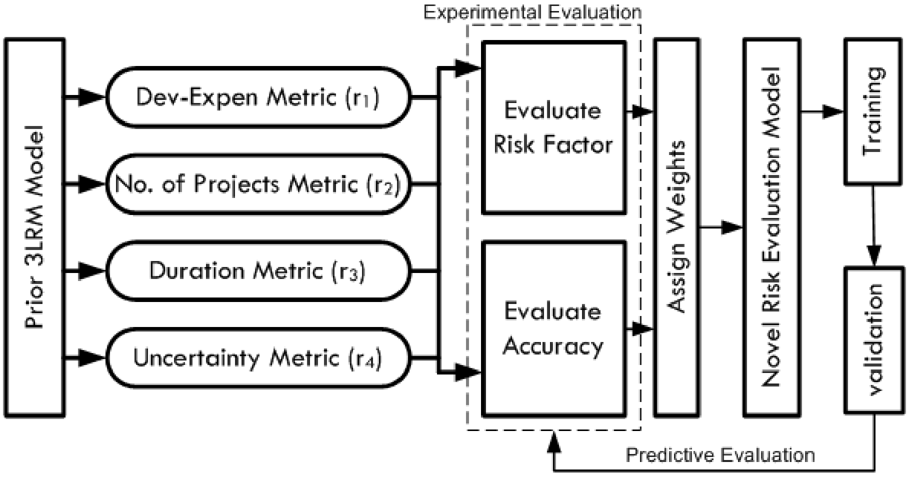


Figure 1. Proposed Architecture of PORM

The implementation of the proposed system is carriedout considering analytical research methodology where the emphasis is rendered on developing the inputs associated with the real-time development scenario in any IT organization with respect to its software projects. The proposed system takes four different forms of inputs e.g. i) development expenditures per software projects (r_1), ii) total number of allocated software projects per employee (r_2), iii) allocated development duration for each projects on an employees (r_3), and iv) uncertainty factor (r_4). The first three inputs (i.e. r_1, r_2, and r_3) can be captured from any real-time statistics of an organization while the fourth input .i.e. r_4 is randomly initialized.

The proposedsystem applies neural network and explicitely uses two different non-linear optimization functions while performing training over its hidden layer in order to obtain better form of outcome associated with risk. After the training process is accomplished, the proposed system offers numerical evaluation of the risk factor associated with the case study of an organization given. This outcome is of higher importance for any stakeholders of an organization so that they can formulate an effective decision for resisting an upcoming risk event. The undertaking of decision is further concretized by check the accuracy value of the proposed prediction operation followed by convergence test for an effective validation. The complete study outcome is assessed with respect to the performance parameters that are standardized in the area of software engineering and still used in the industry in order to assess the software quality. The next section illustrates about the algorithm implementation for the above discussed methodology.

2. ALGORITHM IMPLEMENTATION

The prime purpose of this algorithm is to design and develop an algorithm that bears the capability for precisely computing risk for a given state of information associated with the methodology adopted for software project development cycle. The algorithm construction is basically carried out in two phases where the first phase concentrates in computation of risk factor and second phase concentrates on computing the amount of reliability score associated with the computation. The proposed PORM system also applies machine learning in order to carry out predictive optimization operation. The algorithm takes in the input of *r* (risk assessment data) that after processing yields R_{deg} (Degree of Risk). The steps of the proposed algorithm are as follows:

Algorithm for Computing Risk Factor

Input: *r* (risk assessment data)

Output: R_{deg} (Degree of Risk)

```
Start
1. init $r$
2. $\alpha \rightarrow f_1(r_{in})$, $\beta \rightarrow f_1(r_{out})$
3. For i=1:$n_1$
4.     $r_{in} = w.\frac{r_{in}(i)-\alpha(i)}{\alpha_1(i)-\alpha_s(i)-c}$
5. End
6. For i=1:$n_2$
7.     $r_{out} = w.\frac{r_{out}(i)-\beta(i)}{\beta_1(i)-\beta_s(i)+c}$
8. End
9. For i=1:$n_3$
10.    $T_{out} = [\frac{T(i)-c}{w}.(\beta_2(i)-\beta_1(i))]+\beta_1(i)$
11.End
12. $r_{deg} \rightarrow f_1(r_{out}-T_{out})$
End
```

The description of the above algorithm is as follows: The input of this algorithm is basically represented as *r* (risk assessment data) which is considered to be consisting of i) development expenditures per software projects (r_1), ii) total number of allocated software projects per employee (r_2), iii) allocated development duration for each projects on an employees (r_3), and iv) uncertainty factor (r_4). A data in the form of .csv file is collected from a case study and is considered as an input (Line-1). The next part of the implementation is associated with processing the inputs correctly. For this purpose, a function f_1(x) is constructed that extract the minimum and maximum value of its input arguments r_{in} and r_{out} that represents the input and output arguments respectively (Line-2). A look is constructed for all the values of the input arguments n_1 (Line-3) and therefore the value of n1 is 4 i.e. r={ r_1, r_2, r_3, r_4}. It should be understood that r is a raw data and hence,it is essential to carry out normalization in order to ease down the computation process of optimization. The proposed algorithm computes the normalized input rin (Line-4) using a simplified empirical expression as follows:

$$r_{in} = w.\frac{r_{in}(i)-\alpha(i)}{\alpha_1(i)-\alpha_s(i)-c} \tag{1}$$

In the above expression, the computation of the normalized input associated with the risk factor is computed. The dependable variables used in the expression are w (weight), r_{in} initial risk input element corresponding to i^{th} element, where $i \leq n_1$, α (primary input corresponding to same i^{th} element of higher / lower order i.e. α_2 and $\alpha_1$respectively), and a statistical constant *c*. Almost similar mechanism is continued for computing the normalized output (rout) using the following expression

$$r_{out} = w.\frac{r_{out}(i)-\beta(i)}{\beta_1(i)-\beta_s(i)+c} \tag{2}$$

The proposed study considers that there are two outcomes of study i.e. i) Quantized Risk and ii) Predictive Accuracy. In the above expression, the dependable variables are r_{out} initial risk output and β (primary output corresponding to same i^{th} element of higher / lower order i.e. β_2 and $\beta_1$respectively). It also uses same weight and statistical constant (Line-7). The above two empirical expression significant assists to extract the normalized input and output of the risk factor in most computationally efficient manner as it involves faster time of extraction with reducing the dependencies of reading massive elements of the raw database of the risk management within an organization. This process is a stepping stone of optimization where without involving any extra resources, the proposed system is attempting to obtain better numerical values with higher degree of accuracy and reliability. The last phase of the algorithm implementation again constructs a loop with iteration restricted to n_3 that corresponds to number of outcomes i.e. 2 for the proposed system. The expression introduced for computing the test outcome of the proposed system is as follows:

$$T_{out} = [\frac{T(i) - c}{w} .(\beta_2(i) - \beta_1(i))] + \beta_1(i) \tag{3}$$

However, it is interesting to understand the formulation of the test-outcome T_{out} (Line-10) that finally leads to applying similar function $f_1(x)$ offer the difference of r_{out} normalized output of risk with T_{out} test outcome in order to obtain r_{deg} degree of risk (Line-12). The computation of the test outcome is carried out in an explicit manner with involvement of the neural network-based predictive operation. The steps of the algorithm are as follows:

```
Algorithm for Optimization and Computing Test Outcome
Input: s (size of network), rin/rout (normalized input and output of risk factor)
Output: $T_{out}$ (validated test outcome
Start
1. init s
2. $\gamma \rightarrow f_1(r_{in})$
3. $u \rightarrow \tau(\gamma, s, \omega)$
4. $op \rightarrow f_2(u, r_{in}, r_{out})$
5. $T_{out} \rightarrow f_3(u, r_{in})$
End
```

As the algorithm implements neural network, it is essential that it should posses an effective dimension of its network (consisting of input, hidden, and output layer). The variable *s* represents size of the network followed by implementation of similar function $f_1(x)$ considering the input argument of normalized input i.e. r_{in} (Line-2). The proposed system than applies a feed forward backpropagation function τ considering the input arguments of sort out normalized input value obtained from Line-2, size of the network, and non-linear function ω. The application of the feed forward function τ assists in forming a network that is trained in one-direction and is completely independent on formation of any kinds of computational loops. The proposed optimization is also carried out by considering two non-linear optimization process i.e. tansig and purelin. The tansig is basically a form of transfer function that is responsible for computing the output of a layer for a given value of input layer. The numerical outcome of this function is equivalent to hyperbolic function of tangent. The purelin function is another form of transfer function used Figure 2.

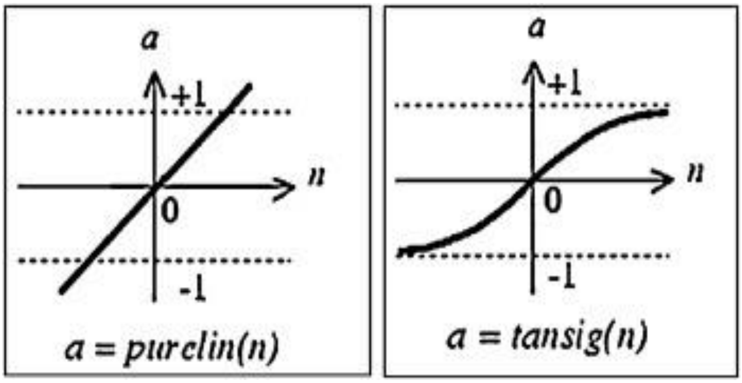


Figure 2. Graphical Representation of *purelin* and *tansig* Functions

The algorithm implements a defined training process f_2(x) considering the input arguments of network u and normalized input/output variables i.e. r_{in} / r_{out} (Line-4). Finally, an aggregate function f2(x) for simulating the neural network model is implemented and applied over the training network i.e u and normalized input value of the risk factor rin (Line-5). This numerical assessment of this model is carried out by calculating the T_{out} obtained by considering the network generated considering the rin (where r_{in}=r_1, r_2, r_3, and r_4) well-defined by numerical attributes of its elements. Therefore, the proposed optimization model offers a significant information about the risk factors considering the ner-real world numerical values associated with 4 variables of risk. However, the last risk variable i.e. r4 is considered as 1 that is highest value of probability to perform an investigation of its impact on risk anlaysis. The next section outlines the results obtained by implementing the proposed predictive optimization technique.

3. RESULT ANALYSIS

The implementation of the proposed PORM system is carried out using MATLAB where the algorithm implemented in prior section has been executed under various test-cases associated with initialized value of risk factor r. The proposed implement programmatically controls the different value of r_1, r_2, r_3, and r_4 followed by usage of neural network toolbox for assessing the outcome of the predictive optimization.

The complete assessment of the study outcome is carried out considering the numerical values as shown in Table 1. The PORM system takes the 4 different types of input of risk r_{in} i.e. i) development expenditures per software projects (r_1), ii) total number of allocated software projects per employee (r_2), iii) allocated development duration for each projects on an employees (r_3), and iv) uncertainty factor (r_4). After applying function f_1(x), the numerical outcomes are shown as minimum and maximum values in 3rd and 4th column of the above table. The proposed system also uses a scaling factor for further normalizing the numeric for better convergence outcome.

Figure 3 shows the graphical outcome of the convergence test, where it can be observed that curve for trainng (obtained from numerical outcome of PORM) as well as best fit just overlaps each other showing a good aggrement with the proposed system. It shows that porposed mechanism of computing risk factor is highly reliable mathematically for a given scenario of constraints (i.e. r_{in}). The best part of the study outcome is that it is completely free from higher ranges of iterative operation as it just need3-4 iterations in order to obtain reliable outcome of risk factor. An interesting factor to observe from Figure3(a) and Figure3(b) is that the proposed system minimizes the final epoch of 4011 to 2178 out of total of 20,000 epoch, which directly means that irrespective of any total number of initialized epoch, the proposed system performs optimization at the cost of extremely reduced number of epoch thereby exhibiting highly reduced dependencies of computational resources.

Table 1 Summary of Numerical Values of Parameters of PORM

Parameters	Parameter	Min-Value	Max-Value	Scale Factor
r_{in}	r_1	0.3	0.5	0.7
	r_2	2	3	4
	r_3	1	3	3
	r_4	1	4	4
r_{out}	r_{out1}	0.6	0.8	1.3
	r_{out2}	4.9	13.8	18.7

The study outcome of the proposed PROM system has been compared with the existing system. The procedure adopted are:-the hypothesis of the proposed system concerning about risk factor is truelly associated with the software engineering field where there are multiple performance parameers required to assess any software design. The proposed system considers 5 frequently adopted software paramaters e.g. PF-1 (Weighted Method per Class), PF-2(Response for Classes), PF-3 (Depth of Inheritance Tree), PF-4 (Coupling Between Objects), and PF-5 (Number of Childern). These are the standard metric that controls the quality of software design and degradation in any of these quality factors would call for risk factor. Hence, the study selects the work carried out by Zhou and Lenug [39] and performs comparison with each other. The outcomes are discussed with respect to the performance metric:

a. *Analysis of PF-1 (Weighted Method per Class)*: As children will inherit maximum number of methods for a given class and greater value of it will act as an impediement towards single design usage of software design that may be obsolete in future. Hence, lower value of RF-1 shown by PORM shows that proposed system offers good design reuse flexibility.

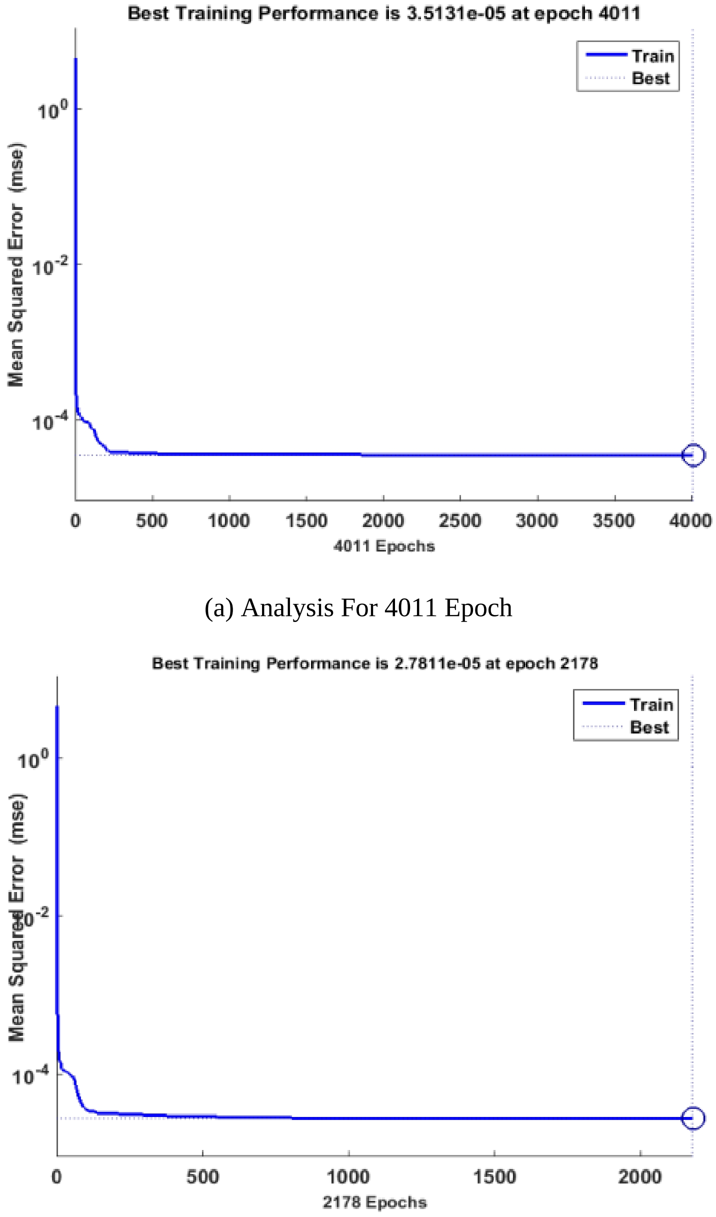
b. *Analysis of PF-2 (Response for Classes)*: Higher value of this performance metric will only invite higher computational effort of debugging and hence result in complexity. It can be seen that PORM offers highly reduced complexity in this case.

c. *Analysis of PF-3 (Depth of Inheritance Tree)*: Higher value of this performance metric will call for higher inheritance making the software deisgn more unpredictable and more complex. Hence, lower value of this performance metric on PROM shows better outcome.

d. *Analysis of PF-4 (Coupling Between Objects)*: Increased value of it will only mean increase maintainance which is absolutely not cost effective. Therefore, proposed PORM offers cost effective optimization outcome.

e. *Analysis of PF-5 (Number of Childern)*: Increased value of these metric calls for challenge in obtaining inefficient abstraction by the parent class that results in increasing design complexity. Hence, PROM doesn't offer any such complexity.

Apart from this, the proposed system consumes 0.23354 seconds while existing system took 1.87762 seconds of computational time. Hence, in every respect, the proposed system can be said to offer better predictive performance of optimization in software engineering.



(a) Analysis For 4011 Epoch

(b) Analysis For 2178 Epoch

Figure 3. Outcome of Convergence Analysis

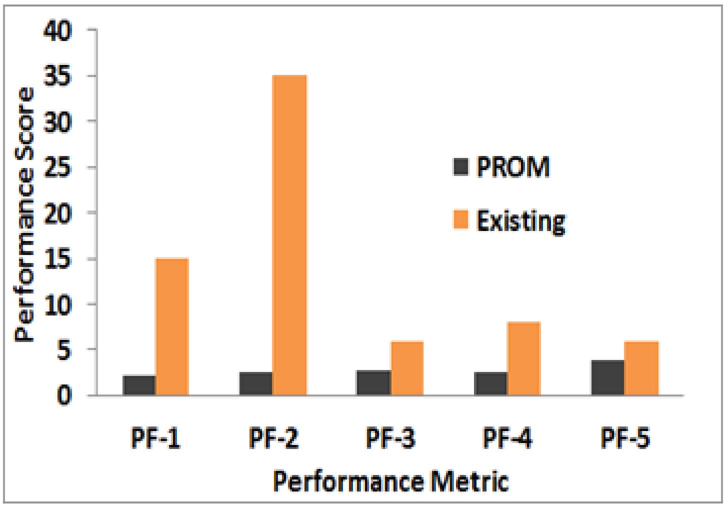


Figure 4. Comparative Analysis

4. CONCLUSION

The main idea of this paper is to introduce a concept that an analytical modeling could be designed for overcoming the research problem associated with risk manegment. Well, risk management is a very vast concept and the problems associated with its also voluminous. This is also one of the reason that why the existing research-based approaches doesn't encapsulate all the problems. After reviewing existing approaches, it was found that problems originated from the uncertainty factor of risk are found not to be addressed and hence this proposed PORM is designed exclusively to address this problem. The study outcome shows that proposed PORM offers good convergence behaviour proving the technical correctness of the proposed concept and it also proved to be computationally cost effective as it has extremely very less iteration involvement as well as it offers faster processing time. Apart from this, the outcome of comparative analysis also proved that it could successfully upgrade the software quality apart from the risk-related problems.

REFERENCES

[1]. Capers Jones, Software Methodologies: A Quantitative Guide, CRC Press, 2017

[2]. Saïd Assar, Imed Boughzala, Isabelle Boydens, Practical Studies in E-Government: Best Practices from Around the World, Springer Science & Business Media, 2010

[3]. Alain April, Claude Y. Laporte, Software Quality Assurance, John Wiley & Sons, 2018

[4]. Capers Jones, A Guide to Selecting Software Measures and Metrics, CRC Press, 2017

[5]. Alex Sidorenko, Elena Demidenko, Guide to Effective Risk Management 3.0, CreateSpace Independent Publishing Platform, 2017

[6]. Aparna Gupta, Risk Management and Simulation, CRC Press, 2016

[7]. Mr Kit Sadgrove, The Complete Guide to Business Risk Management, Gower Publishing, 2015

[8]. A. Nordin, L. M. Abdullah, F. D. Mohamad Fadzil and N. A. S. Roselan, "Requirements elicitation and analysis: Towards the automation of software project risk management," *2014 8th. Malaysian Software Engineering Conference (MySEC)*, Langkawi, 2014, pp. 78-83.

[9]. J. T. Sambantha Moorthy, S. bin Ibrahim and M. N. Mahrin, "Identifying usability risk: A survey study," *2014 8th. Malaysian Software Engineering Conference (MySEC)*, Langkawi, 2014, pp. 148-153.

[10]. Y. Li, J. Chen and L. Feng, "Dealing with Uncertainty: A Survey of Theories and Practices," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2463-2482, Nov. 2013.

[11]. A. F. Zakaria and S. C. J. Lim, "A preliminary survey on modeling customer requirements from product reviews under preference uncertainty," *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, Bandar Sunway, 2014, pp. 1096-1100.

[12]. S. Z. Qasim and M. A. Ismail, "Research problems in Search-Based Software Engineering for many-objective optimization," *2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT)*, Karachi, 2017, pp. 1-6.

[13]. H. Iscan and M. Gunduz, "A Survey on Fruit Fly Optimization Algorithm," *2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, Bangkok, 2015, pp. 520-527.

[14]. Jeong, Eun Joo, Ji Hwan Bae, and Seung Ryul Jeong. "Guidelines aimed at reducing the risks of user acceptance delay in the context of an IT service project management plan." International Journal of Electrical and Computer Engineering 5.4 (2015): 832.

[15]. Firdose, Salma, and L. Manjunath Rao. "3LRM-3 Layer Risk Mitigation Modelling of ICT Software Development Projects." International Journal of Electrical and Computer Engineering 6.1 (2016): 349.

[16]. BHATIA, SUGANDH, and JYOTEESH MALHOTRA. "CSPCR: Cloud Security, Privacy and Compliance Readiness-A Trustworthy Framework." International Journal of Electrical and Computer Engineering (IJECE) 8.6 (2018).

[17]. Salma Firdose, L. Manjunath Rao, " Investigational Analysis of Existing Risk Management Models and Prior Research Contribution: A Review", International Journal of Advanced Research in Computer Science and Software Engineering, vol.4., issue.12, 2014

[18]. P. A. Laplante and J. F. DeFranco, "Software Engineering of Safety-Critical Systems: Themes From Practitioners," in *IEEE Transactions on Reliability*, vol. 66, no. 3, pp. 825-836, Sept. 2017.

[19]. R. Lutz and J. Cleland-Huang, "The Risk of Overly Strict Requirements," in *IEEE Software*, vol. 34, no. 2, pp. 26-29, Mar.-Apr. 2017.

[20]. M. Pasha, G. Qaiser and U. Pasha, "A Critical Analysis of Software Risk Management Techniques in Large Scale Systems," in *IEEE Access*, vol. PP, no. 99, pp. 1-1.

[21]. A. Albadarneh, I. Albadarneh and A. Qusef, "Risk management in Agile software development: A comparative study," *2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, Amman, 2015, pp. 1-6.

[22]. S. Sundararajan, M. Bhasi and P. K. Vijayaraghavan, "Case study on risk management practice in large offshore-outsourced Agile software projects," in *IET Software*, vol. 8, no. 6, pp. 245-257, 12 2014.

[23]. A. Aslam *et al*., "Decision Support System for Risk Assessment and Management Strategies in Distributed Software Development," in *IEEE Access*, vol. 5, pp. 20349-20373, 2017.

[24]. A. Elbanna and S. Sarker, "The Risks of Agile Software Development: Learning from Adopters," in *IEEE Software*, vol. 33, no. 5, pp. 72-79, Sept.-Oct. 2016.

[25]. K. Batbayar, M. Takács and M. Kozlovszky, "Medical device software risk assessment using FMEA and fuzzy linguistic approach: Case study," *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, Timisoara, 2016, pp. 197-202.

[26]. S. Bernardi, J. Campos and J. Merseguer, "Timing-Failure Risk Assessment of UML Design Using Time Petri Net Bound Techniques," in *IEEE Transactions on Industrial Informatics*, vol. 7, no. 1, pp. 90-104, Feb. 2011.

[27]. I. Bider and H. Otto, "Modeling a Global Software Development Project as a Complex Socio-Technical System to Facilitate Risk Management and Improve the Project Structure," *2015 IEEE 10th International Conference on Global Software Engineering*, Ciudad Real, 2015, pp. 1-12.

[28]. S. Y. Chadli, A. Idri, J. L. Fernández-Alemán, J. N. Ros and A. Toval, "Identifying risks of software project management in Global Software Development: An integrative framework," *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, Agadir, 2016, pp. 1-7.

[29]. U. I. Janjua, J. Jaafar and I. B. A. Aziz, "Integration of supportive processes with elementary processes for making current practices of software project risk management more effective," *2015 International Symposium on Mathematical Sciences and Computing Research (iSMSC)*, Ipon, 2015, pp. 292-297.

[30]. R. P. Kendall *et al*., "A Risk-Based, Practice-Centered Approach to Project Management for HPCMP CREATE," in *Computing in Science & Engineering*, vol. 18, no. 1, pp. 40-51, Jan.-Feb. 2016.

[31]. T. Lueddemann, S. Sahin, J. Pfeiffer and T. C. Lueth, "Experimental evaluation of a novel ISO 14971 risk management software for medical devices," *2016 IEEE/SICE International Symposium on System Integration (SII)*, Sapporo, 2016, pp. 162-167.

[32]. R. Pietrantuono, S. Russo and K. S. Trivedi, "Software Reliability and Testing Time Allocation: An Architecture-Based Approach," in *IEEE Transactions on Software Engineering*, vol. 36, no. 3, pp. 323-337, May-June 2010.

[33]. A. Rao, N. Carreón, R. Lysecky and J. Rozenblit, "Probabilistic Threat Detection for Risk Management in Cyber-physical Medical Systems," in *IEEE Software*, vol. 35, no. 1, pp. 38-43, January/February 2018.

[34]. A. Rao, N. Carreon Rascon, R. Lysecky and J. W. Rozenblit, "Probabilistic Security Threat Detection for Risk Management in Cyber-Physical Medical Systems," in *IEEE Software*, vol. PP, no. 99, pp. 1-1

[35]. C. E. C. Reyes, R. Kiesel and R. Schmitt, "First-class risk management from second-use data sources: How intelligent data processing could make risk management more efficient and affordable for SMEs," *2017 IEEE International Systems Engineering Symposium (ISSE)*, Vienna, 2017, pp. 1-7.

[36]. F. Sarro, F. Ferrucci, M. Harman, A. Manna and J. Ren, "Adaptive Multi-Objective Evolutionary Algorithms for Overtime Planning in Software Projects," in *IEEE Transactions on Software Engineering*, vol. 43, no. 10, pp. 898-917, Oct. 1 2017.

[37]. S. V. Susarev, N. G. Gubanov, D. A. Melnikova, Y. V. Sarbitova and A. A. Odintsova, "The intelligent software application method: Real time mode lengering object monitoring based on safety management," *2017 IEEE II International Conference on Control in Technical Systems (CTS)*, St. Petersburg, 2017, pp. 188-191.

[38]. Salma Firdose*, L. Manjunath Rao, "3LRM-3 Layer Risk Mitigation Modelling of ICT Software Development Projects", *International Journal of Electrical and Computer Engineering (IJECE)*, SCOPUS INDEXED, Vol. 6, No. 1, February 2016, pp. 349~356

[39]. Y. Zhou and H. Leung, "Empirical analysis of object - oriented design metrics for predicting high and low severity faults". *IEEE Transactions on Software Engineering*, Vol. 32, No. 10, pp. 771-789, 2006

BIOGRAPHIES OF AUTHORS

Salma Firdose did her Bachelor of Science from Bangalore University from 2000 to 2003.In 2003 received the Bachelor degree. She Studied Masters of Computer Application from Bangalore University from 2003 to 2006 and was awarded masters in the same year.In 2007 to 2009 did Master of Philosophy from Bharathiar Universiy, Coimbatore. Now she is a Ph.D. student 5th year of CSE at Bharathiar University, Coimbatore, India. She worked as lecturer for 8 years at Bangalore, India and 2 years in abroad. Now currently working at Bangalore, India

Dr.L.Manjunatha Rao is working as Professor and Head , Department of MCA, Dr.AIT, Bangalore. He has got 25 years of teaching experience. He did his Bachelor of Science from Bangalore University in the year 1990. He Studied Masters of Computer Application from Madhurai Kamaraj University and was awarded in the year 1999. In 2002 did Master of Philosophy from Mononmanium Sundaranar University. He has awarded Ph.D from Vinayaka Mission University, Tamil Nadu. He has published research papers in both national and international Journals and has authored 2 textbooks.