

Parameter Estimation Techniques of Software Reliability Growth Models: A Critical Research with Experimentation



Sreedhar Y, Krishna Mohan G

Abstract— Ensuring software reliability is a challenging task in software development. Software reliability is the probability of software to provide its intended functionality over a specified time. A couple of testing procedures during the phases of software development provides the data to approximate software reliability. This approximation guides the development team to plan necessary corrective actions. A variety of Software Reliability Growth Models (SRGMs) are in use to predict software reliability. A common task for every SRGM is to calculate reliability growth models attributes as a part of reliability estimation. Optimal calculation of such attributes is influenced by the relationships among the parameters of an SRGM. Therefore parametric SRGMs rely on parameter estimation techniques. The present paper has undertaken the study of existing parameter estimation techniques with the main goal of understanding the pros and cons of each technique in order to design a better technique of parameter estimation for SRGM's in use. A critical review of existing techniques of parameter techniques is given in this paper detailing the categories, approaches, problems relating to the techniques. One of the most successful swarm intelligence techniques named Gray Wolf Optimization (GWO) along with its variants are applied to estimate the parameters of SRGMs. The results of this application along with the comparative analysis are given. The variants of GWO played a significant role in parameter estimation of the SRGMs considered for the experiments. An attempt is made to propose new ways of parameter estimation to achieve optimization.

Keywords— Software Reliability, Software Reliability Growth Models; Parameter Estimation, Grey Wolf Optimizer, Modified Grey Wolf Optimizer, Augmented Grey Wolf Optimizer, Weighted Distance Grey Wolf Optimizer.

I. INTRODUCTION

Software reliability is the probability of software to function as supposed to and according to the specified time. Industries, Academia, Business, transportation, health sector are more dependent on software for day to day functions. Therefore the reliability of software is a key issue [4]. The

reliability value helps the software development team and the management in scheduling the software milestones. Software reliability measures associate with system engineering, product management, reengineering, and evaluation of software development. Reliability is proportional to testing. Test and debugging costs depend on the reliability value. Reliability of the Software is the key indicator to judge the performance of the software [14]. A SRGM identifies and describes the reliability variations using a set of reliability functions with parameters such as time, properties of the product (size, complexity, structure and so on) and process (SE tools and techniques and the experience of the personnel involved). The phases of operations and testing are the right points to track the variation. The models describe the time to failure and/or failure rate in a time interval. Software failures generally found in two categories, finite and infinite failures. The form of the distribution of time and failures of finite category belongs to Poisson, Binomial. Based on the application area and context variety of software reliability models come into existence. The criteria to get differentiation among various models include predictive capability, usefulness, assumption's quality, applicability, and simplicity. The ability of a model to fit with actual data is the primary criterion for selecting a better model.

The structure of this paper has seven divisions. The current section provides an introduction to software reliability. The following two sections elaborate on the concept of SRGM's and the study of the existing parameter estimation techniques. The fourth section briefed the problems identified in the current practices of parameter estimation for SRGMs. The fifth section is devoted to cover the experiments done as an extension of the present study. A detailed analysis of the results is given here. The proposals to improve the parameter estimation techniques for SRGMs are made in the sixth section followed by the conclusion in the next section.

II. EVOLUTION OF SOFTWARE RELIABILITY GROWTH MODELS

Two types of SRGM models are present one is prediction modeling and other is estimation modeling. The last half-century witnessed with more than 250 models. Software reliability estimation models also known as SRGM'S rely on failure data from testing. Software developers rely on models of software reliability growth process and its trend prediction. A group of SRGMs has been constructed by various researchers [6].

Manuscript published on November 30, 2019.

* Correspondence Author

Sreedhar Y*, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India.

Krishna Mohan G, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

Open-source software is not bounded to a particular organization. The reliability measurement of such software is dependent on stochastic failure behavior. A parallel group of SRGMs has been developed to cope with such behavior [19].

Exponential Weibull testing effort function was incorporated into a special type of SRGMs named s-shaped models by *Ahmad et al.* [12]. To cover interdependence among successive runs of software Goseva-Popstojanova and Trivedi tried Markov processes of renewal [8]. To deal with multiple failures that occur simultaneously some authors [10] introduced a reliability model based on Poisson distribution. To deal with the burning problems of optimal release time under different criteria several authors suggested the minimization of cost-constrained by reliability [2].

SRGM model to estimate the fault detection rate can be observed in [20]. This model also reflects changes in operating environments guided by a calibration factor. Three of the much-known models are used in our experiment presented below.

Exponential Model (EXPM):

This model [3] is proposed by Goel and Okumoto formulated as

$$m(t) = a(1 - e^{-bt})$$

$$\lambda(t) = a b e^{-bt}$$

Where $m(t)$ represent MVF and $\lambda(t)$ represent failure intensity function. Here a, b are associated with the number of failures expected, and the rate of decreasing defects respectively.

Delayed S-Shaped Model (DSSM):

This model is proposed by *Yamada et al*[21] describe the software reliability process as a delayed S-shape.

$$m(t) = a(1 - (1 + bt)e^{-bt})$$

$$\lambda(t) = a b^2 t e^{-bt}$$

Rayleigh Model (RAYM):

This model is proposed by Yamada et al.[21] which is a simple NHPP model.

$$m(t) = a(1 - e^{-(bt)^2})$$

$$\lambda(t) = 2 a b^2 t e^{-(bt)^2}$$

The selection of a particular model depends on the kind of process development. Each type of model has its own merits and limitations. Some stages of a software development process behave differently in terms of a number of failures or failures at time t . Therefore failure data at different stages fit towards a different model. The majority of this likeliness may lead to the selection of a particular modal or combination of a set of models.

III. EXISTING PARAMETER ESTIMATION TECHNIQUES

Generally, all software reliability growth models resemble a mathematical model where the mathematical functions take time as a key parameter. The accuracy of an SRGM model is influenced by the parameter values that fit the data [12]. Therefore parameter estimation is a challenging task. A

number of parameters and the type of functions that constitute the model decide the reliability estimation accuracy.

3.1 Traditional Techniques

A traditional technique of parameter estimation is nothing but the equation solving techniques using mathematics. Maximum likelihood estimation (MLE), least-square estimation (LSE)[17] etc belong to this category. These estimation techniques work well for two-parameter data. Further, these mathematical techniques are most suitable for the function sets with linear relationships. Generally, the data of software testing may or may not fit into linear relationships. In such a context of nonlinearity, the techniques do not work well. While solving equations the continuity and derivability of relationship functions pose limitations on solving those functions.

3.2 Current Techniques

To overcome the limitations posed by the traditional techniques alternative techniques are found in the literature. Some researchers proposed meta-heuristic techniques for parameter estimation using genetic algorithms [5][16]. Kennedy and Eberhart [7] proposed a meta-heuristic algorithm based on particle swarm optimization (PSO) with the basic concepts of self-learning and group learning.

PSO is a particle based optimization procedure that resembles the collective behavior of swarms. Each particle represents a possible solution to the current optimization problem. The initialized solution and its velocity are arbitrary at the beginning. Based on some evaluation criteria the particle values will be modified towards optimization. When updating reached the convergence level the process will be stopped [15][1].

GWO is a population-centric stochastic procedure designed by *Mirjalili et al.* [18]. It has better convergence capacity comparing to other swarm intelligence techniques and in addition, it is easier to implement with fewer parameters to be adjusted. The process starts with the initial construction of the population. The fittest, second and third best solutions of present iteration guide the positional changes in the following iterations. The positional changes result in modified fitness values which will guide the following iterations. The process continues until a stopping criterion is met.

IV. PROBLEMS IDENTIFIED FROM THE EXISTING PARAMETER ESTIMATION TECHNIQUES

The traditional ways of parameter estimation technique for SRGMs plays a vital for software reliability estimation. But these techniques may or may not provide optimal solutions as they are limited by issues such as the size of the sample, biasing and parameter initialization. Traditional techniques are most suitable when the number of parameters is two and have linear relationships. For non-linear data, these techniques fit with high error terms which are not desirable.



To overcome such limitations and constraints developers of SRGMs started moving towards swarm intelligence and related strategies for better optimization in parameter estimation. Though the parameter estimation techniques embedded with swarm intelligence and other intelligent algorithms able to provide better optimization all the techniques are at their infant stage of the application and are data specific. Most of the existing swarm-based algorithms have a common limitation that they are vulnerable to local optimization. Some of the existing methods have weaknesses like immature convergence, poor convergence when the dimensionality is high. Getting a good convergence rate is also a problem. There is a need to improve more reliable parameter techniques that suit for trending and mimicking nature of software testing data.

V. APPLICATION OF VARIANTS OF GREY WOLF OPTIMIZATION FOR PARAMETER ESTIMATION IN SRGM'S

One of the successful optimization techniques called Grey Wolf Optimization is applied to estimate the parameters of SRGMs. The basic version of the GWO and its variants are applied one by one in order to know the strength and weaknesses of different variants and their suitability in parameter estimation.

5.1.1 Conventional Grey Wolf Optimization

GWO is a very popular and efficient optimization algorithm developed by *Mirjalili et al.* [18]. GWO is a population-based stochastic search method inspired by grey wolves, which simulate the social hierarchy and the hunting mechanism of the grey wolves. Grey wolves live in groups and they follow social hierarchy in hunting the prey. The typical size of the group is from 5-12. Alpha (α) wolf lead the group followed by beta (β) and delta (δ) wolves which will help the leader to make a decision in hunting. The lowest ranking wolves in the group are omega (ω) wolves.

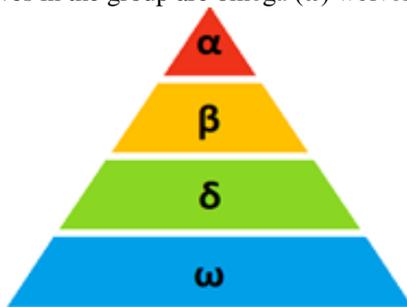


Fig 1. Hierarchy of the grey wolves

The main steps of hunting:

The hunting process constituted by the following paradigm:

- 1) Searching (Exploration) : Finding the Prey
- 2) Encircling Prey (Exploration) : During hunting first, they encircle the prey
- 3) Attacking Prey (Exploitation) : Alpha guides Beta and Delta wolf.

1) Encircling prey:

Grey wolves encircle the prey prior to the hunt. Mathematically expressed below equations

$$\vec{K} = |\vec{C} \cdot \vec{M}_P - \vec{M}(t)| \tag{1.1}$$

$$\vec{M}(t+1) = \vec{M}_P(t) - \vec{A} \cdot \vec{K} \tag{1.2}$$

Where $\vec{M}_P(t)$, $\vec{M}(t)$ are current location of the prey and the location of the wolf, t indicates current iteration.

\vec{A} , \vec{C} are the coefficient vectors and calculated as follows:

$$\vec{A} = 2 \cdot \vec{a} \cdot \vec{n}_1 - \vec{a}$$

$$\vec{C} = 2 \cdot \vec{n}_2$$

\vec{a} is decreased linearly from 2 to 0 and \vec{n}_1 , \vec{n}_2 are random vectors in the range of [0,1].

2) Hunting:

GWO algorithm saves the top three solutions from the search space obtained so far and based on that rest of the wolves update their positions. Hunting process is represented mathematically by the following equations [18]

$$\vec{K}_\alpha = |\vec{C}_1 \cdot \vec{M}_\alpha - \vec{M}| \tag{2.1}$$

$$\vec{M}_1 = \vec{M}_\alpha - \vec{A}_1 \cdot \vec{K}_\alpha \tag{2.2}$$

$$\vec{K}_\beta = |\vec{C}_2 \cdot \vec{M}_\beta - \vec{M}| \tag{2.3}$$

$$\vec{M}_2 = \vec{M}_\beta - \vec{A}_2 \cdot \vec{K}_\beta \tag{2.4}$$

$$\vec{K}_\delta = |\vec{C}_3 \cdot \vec{M}_\delta - \vec{M}| \tag{2.5}$$

$$\vec{M}_3 = \vec{M}_\delta - \vec{A}_3 \cdot \vec{K}_\delta \tag{2.6}$$

$$\vec{M}(t+1) = (\vec{M}_1 + \vec{M}_2 + \vec{M}_3) / 3 \tag{2.7}$$

```

Initialize the grey wolf population  $M_i (i = 1, 2, \dots, n)$  Initialize the
parameters  $a$ ,  $A$ , and  $C$ 
Calculation of fitness of each search agent
 $M_\alpha$  = best search agent
 $M_\beta$  = second best search agent
 $M_\delta$  = third best search agent
while ( $t < \text{Max number of iterations}$ )
    for each search agent
        Update the position of the current search agent
         $\vec{M}(t+1) = (\vec{M}_1 + \vec{M}_2 + \vec{M}_3) / 3$  (2.7)
    end for
    Update  $a$ ,  $A$ , and  $C$ 
    Calculate the fitness of all search agents
    Update  $M_\alpha$ ,  $M_\beta$ ,  $M_\delta$ 
     $t = t + 1$ 
end while
return  $M_\alpha$ .
    
```

Fig 2. GWO Algorithm - Pseudo code

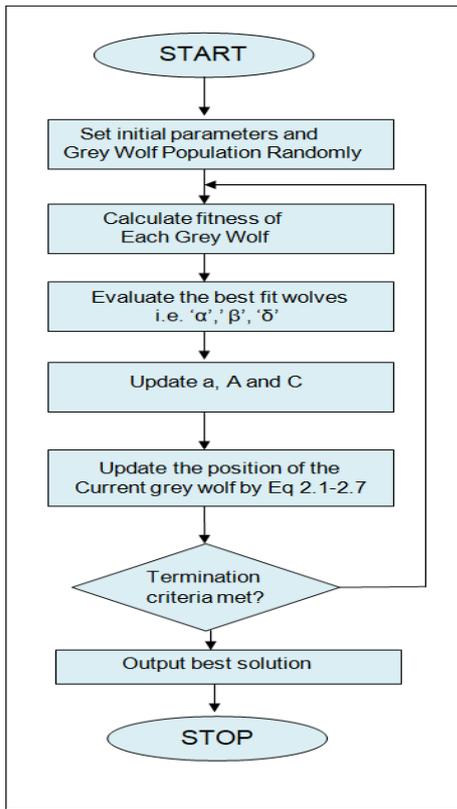


Fig 3.Flow-Chart of GWO Algorithm

Divergence and Convergence in GWO

The next position of the wolf is defined mainly based on the parameter C. C value range of [0,2] irrespective of the iteration number. If C>1 the contribution of prey will be more in the calculation of the next position.

The parameter A is another control parameter for which control the divergence and convergence. The range of A is from the interval [-2,2] and it depends on the value at which decreases from 2 to 0. If A>1 or A< -1 the wolves go for divergence path and the rest of the time they will go in convergence way. To obtain accurate solutions the balance between divergence and convergence is very much essential.

GWO applied to various optimization problems successfully.GWO is very popular in the optimization domain and because of the below reasons.

- a) It possesses a very simple structure.
- b) GWO is a good candidate for hybridization because of its simple structure.
- c) GWO performs well in unknown environments
- d) GWO requires fewer parameters.

The drawbacks of the GWO are

- 1) Local optima trapping.
- 2) The adverse effect of Convergence rate on the later period of evaluation
- 3) The size of parameters is a bottleneck for performance.

Many researchers developed variants of GWO to improve performance. In this paper, three variants of GWO were considered and applied to estimate the parameters of SRGM.

5.1.2 Modified grey wolf optimizer (mGWO)

Mittal et al.[13] proposed exponential decay function to update the control parameter a to balance exploration and exploitation.

$$a = 2(1 - t^2 / T^2)$$

This is a function of the iteration number (t), and the maximum number of iterations (T). The local optima stagnation problem is improved by this exponential decay function by devoting 70% of iterations to exploration and 30 % to exploitation.

5.1.3 Augmented Grey Wolf Optimization (AGWO)

Md H Qais et al.[9] proposed two changes to the original GWO to augment the exploration and exploitation to overcome local optima stagnant problem and named the algorithm as Augmented Grey Wolf Optimization (AGWO). Exploration is augmented by updating the control parameter non-linearly by using the cosine function. Exploitation is increased by taking the top 2 solutions alpha, beta only in calculating the position of the next wolves.

Control parameter a change from 2 to 1 with the relation:

$$a = 2 - (\cos(rand) \times t / T^2)$$

The hunting process is controlled by only Alpha (α) and beta (β) wolves.

$$\vec{M}(t+1) = (\vec{M}_1 + \vec{M}_2) / 2$$

5.1.4 Weighted distance Grey Wolf Optimizer (WGWO)

Md Raphiyoddin S.Malik [11]introduced a weight-based location updating method to overcome premature convergence in solving complex problems with higher dimensions and named the algorithm as Weighted distance Grey Wolf Optimizer (WGWO). Weights are calculated based on the co-efficient \vec{A} and \vec{C} . Updating of the location individuals wolves is calculated based weighted average of top 3 locations in the pack. The position updating equation is changed as below.

$$W_1 = A_1 * C_1, W_2 = A_2 * C_2, W_3 = A_3 * C_3 \tag{5.4}$$

$$\vec{M}(t+1) = (W_1 * \vec{M}_1 + W_2 * \vec{M}_2 + W_3 * \vec{M}_3) / (W_1 + W_2 + W_3) \tag{5.5}$$

5.2 Assumptions of the experiments

Matlab R2015a is used with a computer with core i5, 8 GB RAM, Windows 10 for coding. The sample size of search agents taken as 30, the maximum number of iterations taken as 10 and the number of runs have taken as 10 to compare RMSE values of each algorithm.

In our experiment, we estimate the parameters of three SRGM models using original GWO and it variants mGWO, AGWO, WGWO algorithms by taking three real-time failure datasets. Exponential (EXPM), Delayed S-shape (DSSM) and Rayleigh Model (RAYM) are the SRGMs reused in this analysis. Convergence curves are generated for all four variants of GWO. In each iteration and for every individual wolf the fitness values are calculated for all the entries of failure dataset. The RMSE value is calculated based on the fitness values calculated in the previous step.



The individual who is having the least RMSE value of all the iterations is taken as the best solution.

$$RMSE = \sqrt{\frac{1}{N} \sum_{j=1}^N (actual - estimated)^2}$$

Here N is the number of estimations.

5.3 Dataset Information:

Three real-time error datasets are used in these experiments. The first dataset consists of 46 Measurements, second and third datasets consist of 109 and 111 measurements respectively which were present in[1].

5.4 Results, observations, comparisons, and discussion

The search space for the parameters is taken as a [0, 1000] and b [0,1]. For each model, the parameters are estimated by using all four methods for three failure datasets and the results are captured in the tables respectively.

Parameter Estimation for Exponential Model (EXPM):

Dataset 1 consists of 46 measurements and Figure 5.4.1 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the Exponential model. The results of 10 independent runs of three datasets are presented in table 1. Here it can be observed that AGWO outperformed followed by mGWO, GWO, and WGWO in terms of average RMSE.

Dataset 2 consists of 109 measurements and Figure 5.4.2 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the Exponential model. GWO outperformed AGWO marginally followed by mGWO and WGWO in terms of average RMSE.

Dataset 3 consists of 111 measurements and Figure 5.4.3 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the Exponential model. GWO outperformed mGWO marginally followed by AGWO and WGWO in terms of average RMSE.

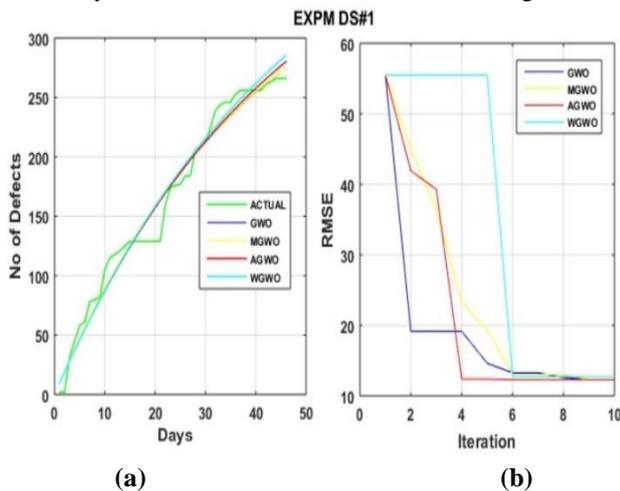


Fig 5.4.1 (a) Actual and Estimated failure curves of four methods
(b) Convergence curves of four methods for Dataset -1

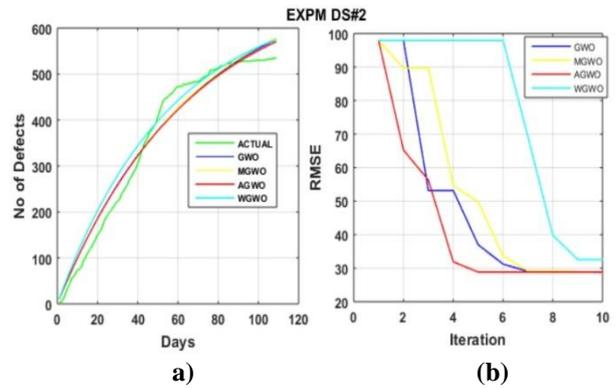


Fig 5.4.2 (a) Actual and Estimated failure curves of four methods
(b) Convergence curves of four methods for Dataset -2

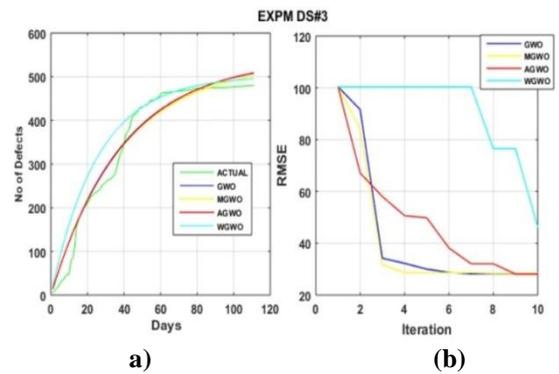


Fig 5.4.3(a) Actual and Estimated failure curves of four methods
(b) Convergence curves of four methods for Dataset -3

Table 1: RMSE Values for Exponential Model

Method	Dataset #1		Dataset #2		Dataset #3	
	AVG	STD	AVG	STD	AVG	STD
GWO	13.33	1.7415	29.232	1.2401	28.149	0.02849
mGWO	12.742	0.7325	30.339	4.464	28.285	0.16859
WGWO	22.672	14.0125	37.77	18.3345	36.613	7.70976
AGWO	12.707	0.5769	29.479	0.9067	28.993	0.98309

Parameter Estimation for Delayed S- Shape Model (DSSM):

Dataset 1 consists of 46 measurements and Figure 5.4.4 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the DSSM model. The results of 10 independent runs are reported in table 2 for 3 datasets. It is found in the table that GWO outperformed mGWO marginally followed by AGWO and WGWO in terms of average RMSE.

Dataset 2 consists of 109 measurements and Figure 5.4.5 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the DSSM model. It is found that GWO outperformed mGWO, AGWO, and WGWO in terms of average RMSE.

Dataset 3 consists of 111 measurements Figure 5.4.6 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for the DSSM model. It is found in the table that GWO outperformed mGWO marginally followed by AGWO and WGWO in terms of average RMSE.

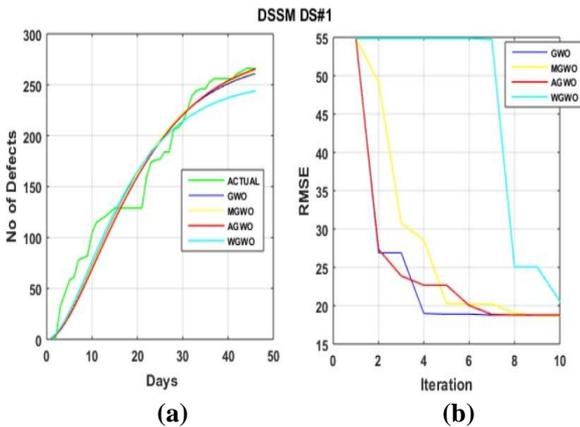


Fig 5.4.4 (a) Actual and Estimated failure curves of four methods
 (b) Convergence curves of four methods for Dataset -1

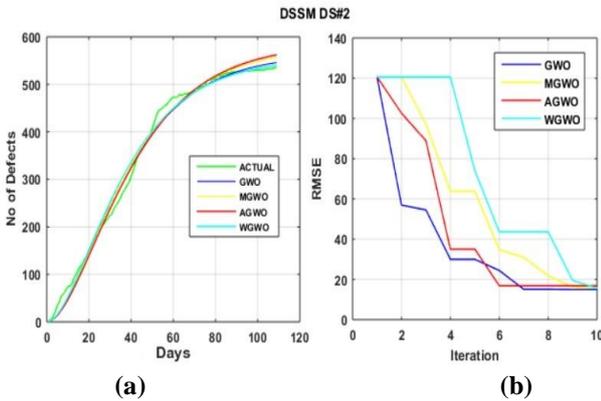


Fig 5.4.5 (a) Actual and Estimated failure curves of four methods
 (b) Convergence curves of four methods for Dataset -2

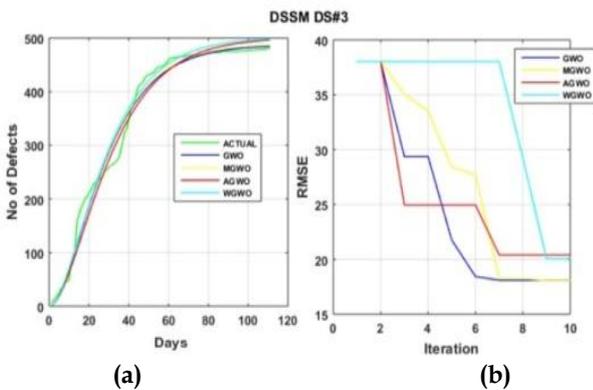


Fig 5.4.6(a) Actual and Estimated failure curves of four methods
 (b) Convergence curves of four methods for Dataset -3

Table 2 RMSE values for DSSM model

Method	Dataset #1		Dataset #2		Dataset #3	
	AVG	STD	AVG	STD	AVG	STD
GWO	18.71429	0.016997	15.086	0.199095	18.13501	0.089443
mGWO	18.74718	0.044786	15.584	0.815135	18.29722	0.257413
WGWO	20.28043	1.940776	20.532	18.3345	21.71793	3.204285
AGWO	19.24769	0.35374	17.007	2.946958	19.86942	1.711044

Parameter Estimation for Rayleigh Model (RAYM) :

Dataset 1 consists of 46 measurements and Figure 5.4.7 show the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for RAYM model. The results of 10 independent runs are reported in table 3 for 3 datasets. It is found in the table that GWO outperformed AGWO, mGWO and WGWO in terms of average RMSE.

Dataset 2 consists of 109 measurements and Figure 5.4.8 show the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for RAYM model. It is observed that GWO outperformed AGWO, mGWO and WGWO in terms of average RMSE

Dataset 3 consists of 111 measurements Figure 5.4.9 shows the actual and estimated failure curve, convergence curve of GWO, mGWO, AGWO, WGMO methods for RAYM model. It is found in the table that GWO outperformed AGWO, mGWO, and WGWO in terms of average RMSE.

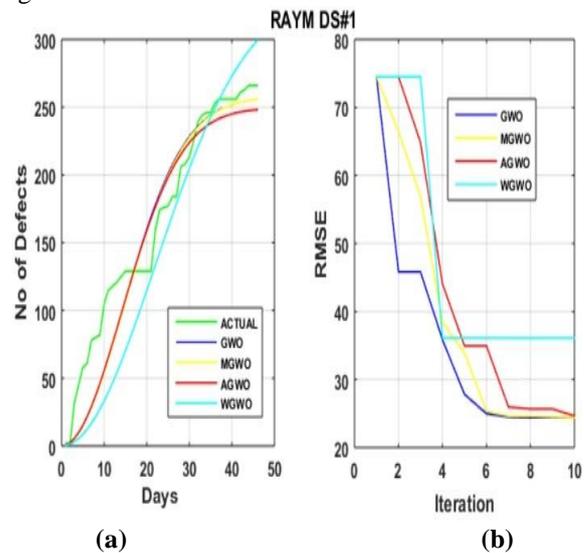


Fig 5.4.7 (a) Actual and Estimated failure curves of four methods
 (b) Convergence curves of four methods for Dataset -1

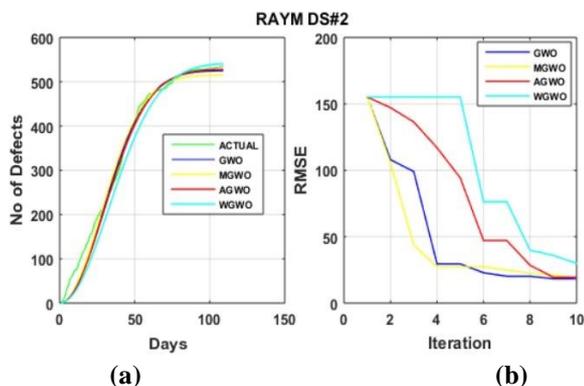


Fig 5.4.8 (a) Actual and Estimated failure curves of four methods
(b) Convergence curves of four methods for Dataset -2

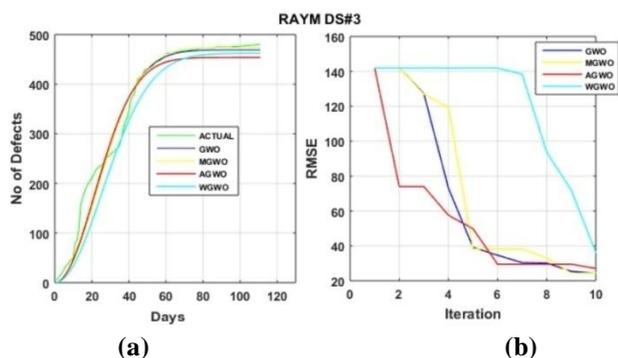


Fig 5.4.9(a) Actual and Estimated failure curves of four methods
(b) Convergence curves of four methods for Dataset -3
Table 3 RMSE values for RAYM model

Method	Dataset #1		Dataset #2		Dataset #3	
	AVG	STD	AVG	STD	AVG	STD
GWO	24.4336	0.0501	18.4854	0.1733	24.3503	0.1123
mGWO	24.4896	0.1015	18.7802	0.6624	24.5559	0.2833
WGWO	27.0462	4.0463	32.8741	26.8567	37.9985	16.5405
AGWO	24.7045	0.2043	20.2614	1.4555	26.4445	1.8498

ANOVA:

The statistical tool ANOVA is conducted to test the significance indifference among the models/SRGM

The result of the statistical analysis is given below:

Table 4: statistics

SUMMARY	Count	Sum	Average	Variance
GWO	3	65.13760	21.7125	14.0166
mGWO	3	65.73964	21.9132	13.9408
WGWO	3	88.45850	29.4862	53.3018
AGWO	3	68.50500	22.8350	11.1242
EXPM	4	108.08780	27.0219	18.1946
DSSM	4	75.30963	18.8274	2.6466
RAYM	4	104.44340	26.1108	27.9990

Anova: Two-Factor Without Replication

From the summary statistics, it can be observed that the GWO techniques exhibiting the smallest mean RMSE, but the averages are more stable with all SRGMs. Other variants of GWO are performing differently with different SRGMs.

It is also observed that the DSSM model is able to fit in the data with least average RMSE and it is stable with all variants of GWO.

Table 5: Anova

Source of Variation	SS	Df	MS	F	P-value	F crit
Rows	123.1252	3	41.04173	10.52568	0.008354	4.757063
Columns	161.3717	2	80.68585	20.69293	0.00203	5.143253
Error	23.39519	6	3.899199			
Total	307.8921	11				

Interpretation:

The significant probability value for rows is $p=0.008354$ which is less than 0.05. Therefore the difference among the variants of GWO is significant at 95%, and GWO is providing best RSME closely followed by its variants mGWO and AGWO. The significant probability value for columns is $p=0.00203$ which is less than 0.05. Therefore the difference among the SRGMs is statistically significant at 95%, where DSSM is most promising.

From the statistical tests, it can be inferred that the variants of GWO applied for parameter estimation provided good estimations. Among the variants, GWO, mGWO, and AGWO performed almost similar where GWO performed the best. The estimation of parameters for DSSM model using GWO technique offered the results with minimum RMSE.

VI. PROPOSALS TO IMPROVE THE CURRENT TECHNIQUES OF PARAMETER ESTIMATION OF SRGM FUNCTIONS

It is observed from the current study that the accuracy in parameter estimation of SRGMs depends on the search initialization, search optimization, and the combination of techniques to make the search progressive. After reviewing the existing variants of swarm intelligent techniques along with their merits and limitations some proposals are made to obtain more optimization and accuracy. The proposal specifically concentrated to improve the current variants of Gray Wolf optimization so that the improved variant will provide more accuracy in parameter estimation of SRGMs.

The proposed improvements include:

1. This proposal is going to change the poor individuals from the search space and reposition them around the best individuals and coupled with changes in repositioning the individual can provide a more accurate estimation with less number of iterations.
2. This proposal is going to make use of data clustering. Here the test data points undergo a clustering process to group the test data points into homogeneous clusters. Clustered data elevates the significant patterns of data. Cluster centers generally represent the average behavior of the cluster. Using these cluster centers as the basis for search positions in GWO processes it is possible to get better estimations.
3. This proposal is introducing a new weighting technique to update the search positions based on the fitness of the best wolfs which can provide better estimations.

VII. CONCLUSION

Parameter Estimation of SRGM's is very crucial and sensitive as these estimations are going to guide the developers in deciding various milestones of software development. The traditional models of parameter estimation have limitations in terms of computational efforts, the dimensionality of SRGMs and high error terms. In this study, an attempt is made to study the current trends and techniques of parameter estimation for SRGMs. The applicability and suitability of GWO technique in parameter estimation of SRGMs is critically reviewed with a suitable experimental setup. The results obtained are analyzed to understand the strengths and weaknesses of the variants of GWO. The detailed analysis, observations are presented. Among the SRGMs considered DSSM is most likely. Among the variants of GWO, AGWO is providing the best estimation. Based on the observations of the study and the result analysis some improvements are proposed to the existing GWO variants to get some more accuracy in parameter estimation. This paper is certainly useful as a reference to further research in parameter estimation of SRGMs. There is a scope to design better techniques of parameter estimation with suitable modifications in the technical process guided by GWO.

REFERENCES

- Alaa Sheta. "Reliability growth modeling for software fault detection using particle swarm optimization." *Evolutionary Computation, 2006.CEC 2006. IEEE Congress on.* IEEE,11 Sep 2006, DOI: 10.1109/CEC.2006.1688697
- C.Y.Huang and C.T Lin, "Software reliability analysis by considering fault dependency and debugging time lag," *IEEE Transactions on Reliability*, vol. 55, no. 3, pp. 436–450,3 Sep 2006, DOI: 10.1109/TR.2006.879607
- Goel, A.L., Okumoto, K., 1979. Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE Trans. Reliab.* 28,206–211, DOI: 10.1109/TR.1979.5220566
- Hoang Pham, *System Software Reliability, Springer Series in Reliability Engineering*, 2006.
- Hsu, Chao-Jung., and Chin-Yu Huang. "A study on the applicability of modified genetic algorithms for the parameter estimation of software reliability modeling." *COMPSAC 2010 IEEE 34th Annual.*19 July 2010, IEEE, DOI: 10.1109/COMPSAC.2010.59.
- J.D.Musa, K.Okumoto, "A logarithmic Poisson execution time model for software reliability measurement". *In: Proceedings of the 7th International Conference on Software Engineering*, pp. 230–238,26 Mar 1984, DOI:10.1109/HICSS.1999.772984
- J.Kennedy, RC Eberhart." Particle swarm optimization", *Proceedings of the IEEE - INNS*, pp 1942–1948,1 Dec 1995, DOI: 10.1109/ICNN.1995.488968.
- K.Go'seva-Popstojanova and K.Trivedi, "Failure correlation in software reliability models," *ISSRE 99*, pp. 232–241, November 1999, DOI: 10.1109/ISSRE.1999.809328.
- Mohammed H.Qais., Hany M.Hasanien., Saad Alghuwainem., "Augmented Grey Wolf Optimizer for Grid-connected PMSG-based Wind Energy Conversion Systems", *Applied Soft Computing Journal*,1 May 2018 <https://DOI.org/10.1016/j.asoc.2018.05.006>
- M Sahinoglu, "Compound-Poisson software reliability model," *IEEE Transactions on Software Engineering*, vol. 18, no. 7, pp. 624–630,7 Jul 1992, DOI:10.1109/32.148480.
- MRS Malik, ER Mohideen, L Ali."Weighted Distance grey wolf optimizer for global optimization Problems". *In: 2015 Neural Computing and Applications ICCIC.* IEEE, pp 1–6,21 Mar 2016, DOI: 10.1109/ICCIC.2015.7435714
- N.Ahmad, M. G. M.Khan, and L.S. Rafi, "A study - of testing effort dependent inflexion S-shaped software reliability growth models with imperfect debugging," *IJORM*, vol. 27, no. 1, pp. 89–110, Jan 2010, DOI: 10.1108/02656711011009335.
- N.Mittal, U. Singh, BS. Sohi," Modified grey wolf optimizer for global engineering optimization." *Appl Comput Intell Soft Comput*, 3 Apr 2016:8, <http://dx.DOI.org/10.1155/2016/7950348>
- P.A.Hamilton, and J.D Musa., "Measuring the reliability of computation centre software," *Proc. 3rd Int. Conf. Soft. Eng.,Atlanta,Georgia,* 10-12,10 May1978, pp.29-36,[https://DOI.org/10.1016/0164-1212\(79\)90023-2](https://DOI.org/10.1016/0164-1212(79)90023-2)
- R Malhotra, & A.Negi." "Reliability modeling using Particle Swarm Optimization", *Int J Syst Assur Eng Manag*,1 Feb 2013, DOI:link.springer.com/article/10.1007/s13198-012-0139-0.
- R Satyaprasad, Ch SuryaKiran and G Krishna mohan., "SPC based software reliability using Modified Genetic Algorithm: Rayleigh model", *JARDCS*, Vol. 10, 04-Special Issue,10 Apr 2018, <http://jardcs.org/papers/v10/s201870.pdf>
- R Satya Prasad,V Suryanarayana,&G Krishna Mohan. (2016). Gompertz based SPRT: MLE. *Journal of Theoretical and Applied Information Technology*, 86(2), 266-271.
- S.Mirjalili,S.M Mirjalili,A.Lewis, "Grey wolf optimizer", *Adv. Eng. Softw.* 69(3) 46-61,21 Jan 2014, <http://dx.DOI.org/10.1016/j.advengsoft.2013.12.007>
- Y Tamura, and S Yamada., "A component-oriented reliability assessment method for open source software," *IJRQSE*, vol. 15, no. 1, pp. 33–53, Feb 2008, DOI: 10.1142/S0218539308002915
- Y Tamura, and S.Yamada., "Comparison of software reliability assessment methods for open source software," *in Proceedings of the 11th ICPDS*, vol. 2, pp. 488–492, IEEE, Fukuoka, Japan, July 2005,21 Nov 2005, DOI: 10.1109/ICPADS.2005.111.
- Yamada, S., Ohba, M., Osaki S., 1983. "S-shaped reliability growth modeling for soft-ware error detection". *IEEE Trans. Reliab.* 32, 475–484.DOI: 10.1109/TR.1983.5221735