Chapter 7

Conclusion

Software reliability is one of the key measures for determining the quality of software. Most of the effort made for software reliability prediction; estimation and assessment are based on the information and data obtainment in the late implementation and testing phase. In order to utilize reliability prediction for enhancing the reliability of a software product it is essential that efforts be made in the design phase.

When design phase is concentrated the approach i.e., traditional or object oriented needs to be taken into consideration. This work has presented a new approach for predicting the reliability of the object-oriented software in the early design phase. As object oriented Design is concentrated, Object oriented design metrics are used as a source for finding defect density and thereafter predicting the reliability of the object oriented software.

Chapter 6 shows the experiments and results related to RFQMOOD model. Next section discusses and interprets those results.

7.1 Discussions and Interpretation

The RFQMOOD model can be used for predicting the reliability of the object oriented software in the early design phase. The detailed class diagram obtained from the object oriented design phase and the organizational information like the CMM level, team factor can be used as a source for computing the defect density and thereafter the reliability of the object oriented software.
The RFQMOOD model is run across five case studies as described in chapter 6 and appendix B. The correlation across the class metrics for these five case studies shows that there is weak correlation between WMC and CBO, which contradicts with the original authors prediction. Table 7.1 shows that if the computation of the CBO for DRS case study is made using the relationship WMC = RFC + CBO, there is increase in reliability prediction of the overall system in around 87% of cases. But that might seem misleading seems the actual coupling between objects may some thing different thereby reducing the reliability estimate.

<table>
<thead>
<tr>
<th>CMM 1</th>
<th>Reliability for CBO identified from Collaboration between Objects</th>
<th>Reliability for CBO identified from Relation between RFC, WMC and CBO</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>0.860226</td>
<td>0.837841</td>
<td>0.022385</td>
</tr>
<tr>
<td>Average</td>
<td>0.703384</td>
<td>0.664555</td>
<td>0.03883</td>
</tr>
<tr>
<td>Low</td>
<td>0.469718</td>
<td>0.423296</td>
<td>0.046422</td>
</tr>
</tbody>
</table>

Table 7.2 shows that if an error is induced in the computation of DIT, like for example in case of ATM if the DIT is not properly identified this single metric can make difference in the reliability prediction of the system. All though the difference seems too low but it still affects the overall predictability of reliability of the entire system due to single metric. Thus careful attention should be made while identifying the DIT of the system.
The A I graph in section 6.3.1.3 for all the case studies shows that packages have good coupling ratio the packages which are in zone of pain or zone of uselessness are difficult to maintain and requires high testability. However it should be noted that it cannot be expected for all the packages to be on the main sequence line since it will define that package almost abstract and thereby no communication with the outside packages. This seems almost impractical.

After the class metrics and package metrics are obtained the defect density is found out. The results of validation of defect density model is shown in section 6.3.3, which shows the values of R-Square and adjusted R-Square values are close to 1 and that of p values are almost 0 thereby indicating the acceptance of the model.

![Figure 7.1 Reliability Increase As Failure Intensity Decrease For Exponential Model Parameters](image-url)
Chapter 7 Conclusion

After estimating the defect density the next process follows with the estimation of failure intensity and reliability. The results obtained from reliability estimation in section 6.3.4 predict that as the failure intensity goes on decreasing, reliability tends to increase. This results are obtained both for exponential model parameters and logarithmic model parameters as shown in figure 7.1 and 7.2. This shows that as the testing time precedes the number of defects in the software system are removed and the failure intensity goes on decreasing thereby increasing the reliability.

Usually while estimating $\beta_0^E$ and $\beta_1^E$ the K that is the fault exposure ratio is considered constant throughout. However, instead to being restricted to some arbitrary constant value equation 5.12 is used for the estimation of K. However it is observed that considering K as constant or computing it using equation 5.12 has almost no effect on the individual package reliability prediction. So either can be used.

Figure 7.3 shows that as the CMM level of the organization increases, the reliability prediction also increases. Thus it can be said that the approach used in the software development process has lot to do with the reliability of the software.
Figure 7.3 Increase in Reliability with Increasing CMM Level

Figure 7.4 shows that for all the four case studies as the team level increase, a increase in reliability prediction is observed. Thus it can be predicted that as the experience of the team increases in the software development process they go towards developing more reliable software.

7.4 Increase in Reliability with the Increase in Team Level
All the above graphs in figure 7.5 shows that as we move from CMM level 1 to CMM level 5 the relative difference in reliability prediction between the high, average and low goes on decreasing. From this observation it can be predicted that the organizations, which are of CMM level 5, have little effect on their software development process, whichever be the team level. Thus it shows that as we go towards CMM level 5 the organizations obtaining CMM level 5 thrives to obtain same productivity regardless of its team level. A point to be recalled here is that the classification of the team levels here is on the experience basis that is the number of years a person or people in the team have serviced in the software development area.

Figure 7.6 shows that if reliability is estimated using exponential or logarithmic model parameters the difference is reliability prediction is very low.

Estimation of $\beta_1^E$ has taken into account system metrics, K and $T_L$ while $\beta_1^L$ has been estimated using $D_0$, $D_{\text{min}}$, K and $T_L$. Also $\beta_0^E$ is estimated using equation 5.8 and $\beta_0^L$ is estimated using equation 5.9. It is found that $\beta_1^E$ varies from $\beta_1^L$ by 15% to 20% where as $\beta_0^E$ varies from $\beta_0^L$ by 4% (refer table 6.58). The overall effect on testing time varies by 6% that is, testing time predicted by logarithmic model parameters is 6% more than that predicted by exponential parameters. However the overall prediction varies by just 0.01%, which can be considered almost negligible. Thus it proves that the use of system
metrics made for the estimation of $k$ is justifiable. Also use of Logarithmic model or exponential model for the estimation of $\beta_0$ and $\beta_1$ parameters make almost no difference to the reliability prediction.

![Figure 7.6 Differences In Reliability Estimation Using Exponential And Logarithmic Model Parameters.](image)

### 7.2 Conclusion

This work examines the various aspects of software reliability in the light of software engineering and object orientation. It was found that there is no specific set of aspects that can be regarded as full-fledged set for software reliability prediction. Therefore concentration to design aspects is given to predict the reliability of the object-oriented software in the early design phase.
This work produces a detail exploration to software reliability from the point of view of models, trend analysis, defect density and exponential and logarithmic model parameters. It is found that the existing software reliability models are used late in the implementation phase or operational phase, which seems too late to be utilized for enhancing the reliability of the software. Since changes made to the software design after the product is almost complete requires around 20% to 80% of additional cost. This motivates for having a model for reliability prediction in the early design phase of object-oriented software.

A study of existing object oriented quality models, frameworks and metrics, shows that there is no such approach for predicting the reliability of the object oriented software in the early design phase. Therefore taking the base of the existing models, metrics and frameworks a new approach to predict the reliability of the software in the early design phase is brought forth through RFQMOOD model.

The RFQMOOD model comprises of a framework called FPROOD, defect density model and finally reliability estimation using exponential and logarithmic model parameters. The FPROOD framework is distributed across two vectors category and granularity. At the category level design metric, size metric and complexity metrics are placed. At the granularity level there are class, package and system. Here direct consideration to method has not been given because to extract method level metrics there is no detail information regarding it in the early design phase.

After extracting the metrics from the FPROOD framework, the Fods factor of the defect density model is estimated which is further utilized for estimating defect density. Once the defect density is estimated the $\beta_0$ and $\beta_1$ are estimated using exponential and logarithmic model. These values are further used for estimating failure intensity and thereafter the reliability of the software product in the early design phase.

The practical exploration to the RFQMOOD model using five case shows that the R-square and adjusted R-square values close to 1, and 0 value of p shows the model validity.

Reliability estimated using exponential and logarithmic model shows that there is only 0.01% variation using the two approaches. The testing time required by logarithmic parameters is 6% more than that by exponential parameters.

The reliability estimation shows that as the CMM level goes on increasing the reliability go on increasing also as the team level increase the reliability increase. Here it is assumed single team handles, one single software development process.

Thus the reliability estimation made in the early design phase in this research work is utilized for enhancing the reliability of the software product as it gives a detail exploration to all design aspects.
7.3 Future Work

There are several metrics, which affects the final reliability of the object oriented software products. A careful collection and development of more metrics and their relationship can be further explored and utilized for enhancing the reliability of the object oriented software. Also the RFQMOOD model can be applied to real world case studies and further enhancement to the model can be suggested using the statistical information. A new approach for further enhancing the object oriented software reliability in testing and implementation phase can be rooted out on the basis of RFQMOOD model for the assessment of software reliability predicted in the object oriented design phase, which can form a new dimension for assessing the reliability of the software product.