together during the early test phases because the models are used in the late test phases and because developers and testers are separated in the late test phases.

To deal with these limitations, a new model was proposed that considers the test time and the debug time together, making it possible to manage software reliability from the early test phases to the late test phases through the proposed model. Therefore, the failure removal cost will be reduced.

An experiment was conducted to compare the existing model and the proposed model with actually collected industrial data, and a Paired T Test was used to confirm the difference between the proposed model and the existing model and between the actual collected values and the estimated values by the proposed model. This experiment confirmed whether there was a statistically significant difference.

Through the proposed model, it was possible to provide more accurate estimation results on the test side. Moreover, the failure removal cost during the late test phases can be reduced on the development side. On the management side, the three steps of the reliability process, the software reliability prediction models, early estimation models, and late estimation models, can supported while reducing the schedule management cost as well as the total cost. On the project side, the high possibility of software development is enhanced.

Currently, the proposed model is based on Exponential models. Therefore, data that is fitted to S-shaped models cannot be used with the proposed model. This is a limitation of the proposed model. To deal with this limitation, the proposed model will be expanded to use data that is fitted to S-shaped models. The expanded model will provide MRE and MSE values to select more accurate models according to the estimated results. Additional experiments will also be conducted to confirm accuracy of the model based on data collected during integration testing to provide more accurate evidence for use with the model.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0018020).

References

1. Reliability Analysis Center, “Introduction to Software Reliability: a State of the Art Review”, Rome Laboratory, 1996
2. Michael R. Lyu, “Software Reliability Engineering: A Roadmap”, FOSE, 2007.
3. “IEEE Recommend Practice on Software Reliability”, IEEE Reliability Society, June, 2008.
4. M. Xie, “Software Reliability Modeling”, WorldScientific, 1991.
5. Ch. Ali Asad Muhammad Irfan Ullah, Muhammad Jaffar-Ur Rehman, “An Approach for Software Reliability Model Selection”, Computer Software and Applications Conference, 2004.
6. Yinong Chen and Jean Arlat, “An Input Domain-Based Reliability Growth Model and Its Applications in Comparing Software Testing Strategies”, LAAS REPORT, April, 1995.
7. Michael R. Lyu, “Handbook of Software Reliability Engineering”, IEEE Computer Society Press, 1997..
8. Norman F. Schneidewind, “Reliability Modeling for Safety-Critical Software”, IEEE Transactions on Reliability, March, 1997.
9. Norman F. Schneidewind, “Modeling the Fault Correction Process”, 12th International Symposium on software Reliability, November, 2001.
10. Jung-Hua Lo, Chin-Yu Huang, “An Integration of Fault Detection and Correction Processes in Software Reliability Analysis”, The Journal of Systems and Software, 2006.
11. Robert V. Hogg, Joseph W. McKean, Allen T. Craig, “Introduction to Mathematical Statistics”, Pearson, 2005.
12. Allen Nikora, “CASRE-A Computer-Aided Software Reliability Estimation Tool”, http://www.openchannelfoundation.org/projects/CASRE_3.0.
13. William Farr, Oliver Smith, “SMERFS-Statistical Modeling and Estimation of Reliability Functions for Systems”, 1996.
14. Myungmuk Kang, Taewan Gu, Jongmoon Baik, “A User Friendly software Reliability Analysis Tool based on Development Process to Iteratively Manage Software Reliability”, International Symposium on Software Reliability Engineering, 2009.
15. S. Yamada, M. Ohba, O. Osaki, “S-Shaped Reliability Growth Modeling for Software Error Detection, IEEE Transactions on Reliability”, Vol, R-32, no. 5475-5478, 1983.