ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

Regression Testing Using AIGTCP Algorithm for Industry Based Applications

K. Hema Shankari, Dr. R. Thirumalai Selvi

Research Scholar, Bharath University, &

Assistant Professor, Dept. of Computer Science, Women's Christian College, Chennai, India

Research Supervisor,&

Assistant Professor, Dept. of Computer Science, Govt. Arts College, Nandanam, Chennai, India

ABSTRACT*:* Regression testing is an important phase in software maintenance activity to ensure the modification caused by debugging. Regression testing is a testing to test the modified software during the maintenance level. Regression testing is a costly but crucial problem in software development. Both the research community and the industry have paid much attention to this problem. This paper try to do the survey of and current practice in industry and also try to find out whether there are gaps between them. This research discusses the problems about current research on regression testing and quality control in application of regression testing in the engineering practice, and proposes a practical regression method, combing with change-impact-analysis, business rules model, cost risk assessment and test case management The proposed Advanced genetic algorithm for regression test case prioritization (AIGTCP) is compared with previous approach using APFD metric. The results represent that propose approach outperforms the earlier approach.

KEYWORDS*:* Regression testing; Cost-risk-assessment; Industry application; Business rules, AIGTCP, Test case prioritization, APFD metric.

I. Introduction

Regression testing is an iterative process, a new round of testing has a great similarity with the pre-test, so how to reuse historical accumulation of test resources, and improve testing automation to efficiently complete testing are worthy of further study. The Test case prioritization techniques [4] intend to arrange test cases for regression testing in such a manner, with the goal of amplifying some criteria. Rothermel et al. [1] and Elbaum et al. [3] proposed a variety of test case prioritization techniques to the boost fault detection rate.

The Advanced Genetic Algorithm is well suited for solving problems where solution space is huge and time taken to search exhaustively is very high. Another advantage of this algorithm is that it has the ability to solve problems with no previous knowledge.

II. Related work

Test case prioritization can address to boost a diversity of objective functions such as rate of fault detection, rate of detection of high-risk faults, likelihood of revealing regression errors, coverage of coverable code, and confidence in the reliability of the system under test [1]. Numerous techniques have been investigated to arrange test cases for regression testing, with an attempt to test modified software, nine different test case prioritization techniques have been explained by Rothermel et al. [1]. We have presented an approach for prioritizing regression test cases on the basis of three factors which are rate of fault detection (RFT), percentage of fault detected (PFD) and risk detection ability (RDA). RFT is defined as the average number of defects found per minute by a test case [7]. PFD is the percentage of fault detected by a test case. RDA is defined as the ability of test case to detect severe faults per unit time.

DOI: 10.15680/IJIRCCE.2015. 0309072

The previously mentioned test models are relying on software development process, so there is no practical implementation approach for regression testing. Different from the unit testing, integration testing and performance testing in development process, regression testing repeatedly emphasizes accumulation, which can be completed through the structure and the business rules modeling methods, so that the cycle of regression testing can proceed.

To build a supporting platform of regression testing for decision-making, at first, you need to scan and analyze the source code of the core business systems, and set up an application description model; meanwhile, a bank of expert knowledge of the industry should be established to collect and refine business information. And then, a model of business rules should be established to express business information. Finally, risk assessment model will be established, according to industry application and the characteristics of test implementation.

In regression testing, reusing of used cases can greatly improve test efficiency, and reduce time and duplication of effort. Therefore, there is a huge test case library at the supportive platform. It indexes all the used cases in catalogue to associate the specific cases with related businesses, and facilities the reference of cost-assessment model and the automatic generation of the test scripts.

III. Proposed Algorithm

A. *Design Considerations:*

If business systems change with the modification of demands, and with the changes of system maintenance and other reasons; if new versions of the software are produced by the development department, implementation steps regression testing of are as follows:

Step 1. Collect change requests

Step 2. Identify the scope of the next release and the scope of the next release and determine which change requests will be included in the next build.

Step 3. Document the requirements, functional requirements, functional specification and implementation plans for each grouping of change requests.

Step 4. Implement the change.

Step 5. Test or verify the change. Unit testing is done by the person who made the change, usually the programmer. Function testing tests a functional area of the system to see that everything works as expected. Regression testing is system-wide to insure that all areas of the
system still function as expected. This validates that the change caused no unexpected side effects and that the system still has the overall functionality it had before the change.

Step 6. Release.

What need to emphasize here is that the regression testing is required to be system-wide, also mentioned that in the industry, multi level regression testing is required to insure the software quality. Since in practice, large systems may be developed in different stages and by different teams. During the process, testing is involved in almost every stage, such as unit testing, functional testing, acceptance testing and field testing. So in practice, regression testing is required to be embedded in every mature software development process, since for every change of the software, no matter how tiny it is, regression testing must be applied.

B. *Description of the Proposed Algorithm:*

We consider three factors for proposed prioritization technique. These factors are discussed as follows.

(i) Rate of Fault Detection

The rate of fault detection (RFD) is defined as the average number of defects found per minute by a test case For the test case f.

$$RFD_{f} = (N_{i}/ \text{ time } f) * 6 \qquad (1)$$

(ii) Percentage of Fault Detected

The percentage of fault detected (PFD) for test case Tk can be computed by using number of faults found by test case Tk and total number of faults, expressed as follows.

$$PFD_{f} = (N_{f}/ N) * 6 \qquad (2)$$

(iii) Risk Detection Ability

It can be defined as the ability of test case to detect severe faults per unit time. Testing efficacy could be progressed by emphasizing on test cases which detect greater percentage of severe faults (RDA). Risk value was allocated to every fault depending on the faults impact on software. To every fault a Risk value has been allocated based on a 10 point scale expressed as follows.

Very High Risk: RV of 10
High Risk: RV of 8
Medium Risk : RV of 6
Less Risk: RV of 4
Least Risk : RV of 2.

For test case Tf, RDAf have been computed using severity value Sf, Nf is the number of defects found by Tf, and timef is the time needed by Tf to find those defects. The equation for RDA can be expressed as follows.

$$RDA = (S_{f} * N_{f})/\text{time } f \qquad (3)$$

Test Case Ranking

Test case Ranking is the summation of the three factors which are RFD, PFD and RDA. For test case T_f, Test case ranking (TCR_f) can be calculated by the equation given below:

$$TCR_{f} = RFD_{f} + PFD_{f} + RDA_{f} \qquad (4)$$

IV. PSEUDO CODE

The proposed prioritization technique expressed as follows.
Input: Test suite ST, and test case ranking (TCR) for every test case are inputs of the algorithm.
Output: Prioritized order of test cases.

```
Step1. Start
Step 2. Set ST empty
Step 3. For each test case Tf ε ST do
Step 4. Calculate test case ranking using equation (4)
Step 5. end for
Step 6. Sort ST according to descending order of TCR value
Step 7. Let ST be T
Step 8. end
```

V. Simulation Results

For the purpose of motivation this example assumes a priori knowledge of the faults detected by T in the program P.

TABLE 1: Sample data of Test cases

Test cases	No of faults covered	Execution time	Risk severity
T1	2	12	8
T2	3	14	10
T3	1	11	4
T4	4	10	20
T5	2	10	12
T6	2	13	6

Table 2. RFD, PFD, RDA for test cases T1..T6

Test cases	RFD	PFD	RDA
T1	1.66	2	1.333
T2	2.142	3	2.142
T3	0.9	1	0.3636
T4	4.0	4	8
T5	2.0	2	2.4
T6	1.538	2	0.923

The values of rate of fault detection (RFD), percentage of fault detected (PFD) and risk detection ability (RDA) for test cases T1..T10 is calculated by using equation (1), equation (2) and equation (4) respectively. Table 2represents the values for all three factors which are RFD, PFD, RDA for test case T1..T6 respectively.

Table 3. Test case ranking for T1..T6 respectively

Test cases	Test case ranking TCR=RFD+PFD+RDA
T1	4.993
T2	7.284
T3	2.263
T4	16
T5	6.4
T6	4.461

Table 4: Test cases ordering for proposed order

Test cases	Prioritized order
T1	T4
T2	T2
T3	T5
T4	T1
T5	T6
T6	T3

For test cases, T1..T6, TCR value computed from equation (4) as given below. Table 3 shows test case ranking for each test case. Table 4 shows the prioritized order.

Comparison with the previous work

In this section, the proposed prioritized order is compared with previous work Table 7 represents proposed order of test cases and the prioritized order proposed

Prioritization Technique	APFD %
Non Prioritized	49%
Random approach	56%
AGRTCP	78%

Table 7: APFD % for no prioritization, Random

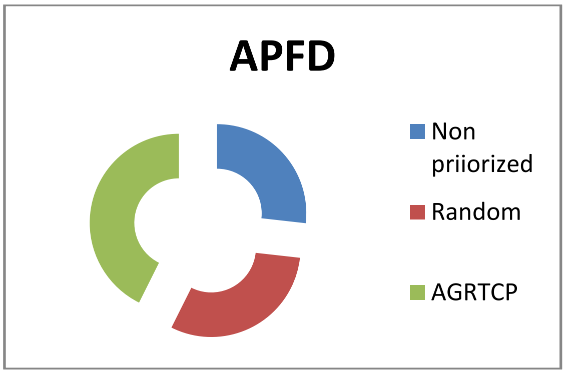

fig 1 : APFD Percentage for no order,andom and AIGTCP

VI. Conclusion and Future Work

Software tests, especially software represented by regression testing, it accompanies the whole life cycle of industrial application system. This paper presents a regression testing method for industry-oriented applications to solve issues, such as the low degree of automation of large-scale business systems and difficulty of defining test coverage. For every test case all the three factors are calculated and test case ranking is computed by adding these factors for each test case. To solve the problem of test case prioritization we prioritize test cases, according to decreasing order of test case ranking value, and we obtain the prioritized order of test cases. The proposed approach is compared with different prioritization techniques such as no ordering, using APFD metric. The APFD is calculated by taking the weighted average of the number of faults detected during the execution of the test suite

References

[1] G. Rothermel, R. Untch, C. Chu and M. Harrold, "Test case prioritization: An empirical study," *In Software Maintenance, 1999. (ICSM' 99) proceedings. IEEE International conference, on pages 179-188 IEEE, 1999.*

[2] A.Pravin and Dr. S. Srinivasan,"An Efficient Algorithm for Reducing the Test Cases which is Used for Performing Regression Testing," 2nd International Conference on Computational Techniques and Artificial Intelligence (ICCTAI'2013) March 17-18, 2013

[3] S. Elbaum, A. Malishevsky, and G. Rothermel, "Prioritizing test cases for regression testing," *Proc. The 2000 ACM SIGSOFT International Symposium on Software Testing and Analysis, Portland, Oregon, U.S.A., August 2000, 102–112.*

[4] W. Wong, J. Horgan, S. London and H. Agrawal, "A study of effective regression testing in practice," *In Proc. of the Eighth Intl. Symp. on Softw. Rel. Engr., pages 230–238, Nov. 1997.*

[5] R.Beena, Dr.S.Sarala, "CODE COVERAGE BASED TEST CASE SELECTION AND PRIORITIZATION," International Journal of Software Engineering & Applications (IJSEA), Vol.4, No.6, November 2013.

[6] Alex Groce, Todd Kulesza, Chaoqiang Zhang, ShaliniShamasunder, Margaret Burnett, Weng-Keen Wong,SimoneStumpf, Shubhomoy Das, Amber Shinsel, Forrest Bice, and Kevin McIntosh," You Are the Only Possible Oracle: Effective Test Selection for End Users of Interactive Machine Learning Systems," IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 40, NO. 3, MARCH 2014

[7] R. Kavitha, N. Sureshkumar, "Test Case Prioritization for Regression Testing based on Severity of Fault*," College of Engineering and Technology Madurai, Tamilnadu, India (IJCSE) International Jthenal on Computer Science and Engineering 2010.*

[8] Samaila Musa, Abu BakarMd Sultan, Abdul Azim Bin AbdGhani, SalmiBaharom,"A Regression Test Case Selection and Prioritization for Object-Oriented Programs using Dependency Graph and Genetic Algorithm" *Research Inventy: International Journal of Engineering And Science Vol.4, Issue 7 (July 2014), PP 54-64 Issn (e): 2278-4721, Issn (p):2319-6483.*

[9]S. Elbaum, A. Malishevsky, and G. Rothermel.(2000), Prioritizing test cases for regression testing".

[10] G. Rothermel, R. Untch, C. Chu, and M. J. Harrold. "Test case prioritization: an empirical study". Testing "European Journal of Scientific Research ,ISSN 1450-216X Vol.55 No.2 (2011), pp.261-274

[11] S. Elbaum, A. G. Malishevsky and G. Rothermel,(2001), "Incorporating varying test costs and fault severities into test case Prioritization" ,23rd International Conference of Software Engineering, pages329-338

[12]http://www.infosys.com/industries/banking/automated_order_management.asp?page=iemsl

BIOGRAPHY

K. Hema Shankari Research scholar of Bharath University and Assistant Professor Department of Computer science in Women's Christian College with more than 12 years of teaching experience and **Dr. R. Thriumali Selvi** is a Research Supervisor and Assistant Professor in the Department of Computer Science Nandanam Govt. Arts College with more than 20 years of Teaching experience.