



Regression Test Suite Reduction

Pankaj Girase¹, Asst.Prof. KrishnaKant Kishore²,

PG Student, Dept. Of Computer, Jagadguru Dattatray College of Technology, Indore, M.P., India¹

HOD, Dept. Of Computer, Jagadguru Dattatray College of Technology, Indore, M.P, India²

ABSTRACT— Regression testing is a software maintenance activity. Regression testing is the process of testing changes to computer programs to make sure that the older programming still works with the new adjustments. Test suites tend to growing size as software evolve, often making it too costly to execute entire test suite sin the given time and constraints. Hence it becomes an essential to minimize the test suite and choose a subset of test cases from test suite which will be execute din least time and has the capability to cover all the errors, Thus it is advisable to choose certain test cases and prioritized them with the objective to under size the whole test suite dimension. A number of different approaches have been studied to maximize the value of the accrued test suite: minimization, selection and prioritization. Test suite minimization seeks to eliminate redundant test cases in order to reduce the number of tests to run. Test case selection seeks to identify the test cases that a irrelevant to some set of recent changes. Test case prioritization seeks to order test cases in such a way that early fault detection is maximized.

KEYWORDS – Regression Testing, Test Case Selection, Set Theory.

I. INTRODUCTION

Software examining is really a growing place. We are able to determine that since “Software testing” could be the technique of making this program or perhaps a software program using the intent associated with finding mistakes. For this reason there exists about to make the test out conditions with the performance so that it can easily obtain the mistakes. Because software sizing grows up you will find thousands of test out conditions to get performed. Because time is really a crucial, we should instead lessen how big a new test out room specifically while in regression is examining, that is conducted whenever a change has become devote almost any application plan. In these cases testers are actually utilize a few decline techniques to minimize how big a new test-suite is. Test room minimization regarding marketing could be the operations associated with picking out those test out conditions that satisfies a few presented requirement as well as doing away with dozens of are generally come to be a tautology.

Regression examining is completed whenever improvements are made to current software; the purpose of regression examining would be to present self-confidence that this fresh launched improvements will not prevent the behaviours with the current, unchanged area of the software. This can be a complicated method which is even more difficult because of a lot of the latest trends within software development paradigms. As an example, the element based Software development method has a tendency to end in use of a lot of black-box factors, frequently used from your third-party. Any kind of change inside the third-party factors may perhaps interfere with other software program, still it is tricky to accomplish regression examining considering that the internals with the third-party factors usually are not



recognized to their people. The actual smaller life-cycle associated with software development, including the just one proposed with the agile encoding control, furthermore imposes limits and limitations on what regression examining can be carried out within minimal means.

Naturally, one of the simplest method to this challenge would be to basically implement every one of the current test out conditions inside the test out room; this really is called a new retest-all approach. Nevertheless, since software evolves, the test out room has a tendency to increase, which suggests it might be prohibitively costly to implement your entire test out room.

That constraint aids thing to consider associated with techniques that find to cut back your time essential for regression examining in a variety of means. Several unique solutions are actually studied to help you the regression examining process. The actual several main limbs include things like test out room minimization, test out situation collection and test out situation prioritization. Test room minimization is really a process that looks for to recognize after which get rid of the obsolete as well as a tautology test out conditions from your test out room. Test situation collection deals with the condition associated with selecting a subset associated with test out conditions which is to be employed to test out the improved areas of it. Lastly, test out situation prioritization worries the id. With the 'ideal' getting associated with test out conditions that enhances attractive houses, for instance first failing diagnosis.

Software package development is definitely an iterative practice: you'll find brand new specifications, a number of should be reworked, eliminated, put into practice brand new characteristics, and so on. As well as the best recent insect conjecture approaches within the novels estimate in which alterations about authentic application introduce pests within 78% of that time period.

A new application insect may be the period used in informatics to go into detail a new drawback, miscalculation, or mistake within personal computer method in which produces a great inappropriate or unexpected consequence. The effects regarding pests could be really severe. Breakdowns within application take place each day, along with precisely what is even worse can be men and women are burning off tasks and in some cases his or her liberty pertaining to application failures that is preventable. Examples regarding application failures within 2011:

□ Computer method pests result in Asian kitchenware checking amenities recovery time “Computer method complications at certainly one of Japan’s most significant financial institutions led to a new across the country ATM system greater than 5. 800 machines planning traditional every day and night, net checking solutions staying shut down pertaining to 3 times. ”

□ Cash device insect positive aspects clients by providing them more money “An Hawaiian bank began offering significant amounts regarding funds from 50 dollars machines all over one particular city. Officers in the organization said we were looking at running within stand-by mode, consequently could not identify the particular consideration balances regarding clients. ”

□ Bugs within social media iPhone app pertaining to pill just hours soon after postponed release “Just hours soon after its release, this particular social media sites’ long-awaited pill iphone app had been acquiring document with

regards to trivial pests from clicking through which webpages via cell symbols in order to complications posting responses.”

Additional instance occurred within 2002: “An examine commissioned because of the YOU Office regarding Commerce’ Nationwide Company regarding Standards along with Engineering figured application pests, or blunders, are consequently frequent therefore adverse they cost the united states economic system approximately \$59 billion dollars every year, or with regards to 0.6% on the gross home product”². Software package pests can be a consequence of undesirable top quality nation-wide politics or the nature regarding individual blunders within the encoding activity. And so, it is crucial testing the software program.

II. LITERATURE SURVEY

Regression Testing Computer software which has been changed (to resolve some recognize mistake or to delete/add brand new requirement) must be authenticated (retested) using focus from the next targets:

- i. Assure that the revolutionary prerequisites are executed effectively;
- ii. Be sure that brand new prerequisites does not affect preceding uses (which continue functioning since expected);
- iii. Test out people regions of the application which may have not really been recently checked out previous to. The task associated with retesting the application to ensure that revised software still operate effectively just how test situations utilized to test authentic software (objective 2) is termed regression screening.

Lately, the application screening analysis community have got granted ample importance to help the topic of regression screening. A number of approaches are offered to maximise your test collection of every SUT: minimization, variety as well as prioritization. Test out collection minimization will be the way to represent what test situations are generally repetitive or maybe out of date and remove them from the test collection. Test out collection variety go with a subset associated with test event that is to be utilized to test your pieces which were changed in computer software. Finally, test collection prioritization identify the top purchase associated with test event which take full advantage of some house, including mistake recognition.

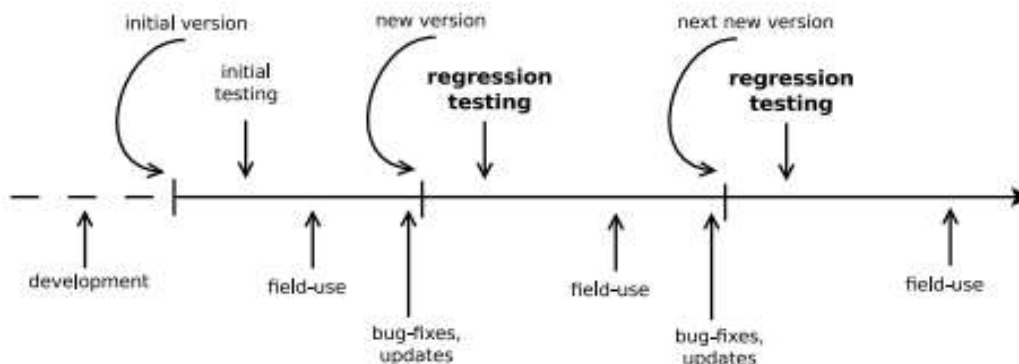


Fig 1. Timeline during System



Volume 1, Issue 6, November 2015

Over Figure talks about a typical instance of your schedule throughout the life of your computer software program. Even as we are able to see, your setup associated with regression screening is a significant small percentage with the system's lifetime. Regrettably, regression screening can't be done as a result of typical improvements as well as revisions of your program. Every time a software has become revised, it is vital to make sure the improvements operate effectively but also to ensure that your unmodified quests with the software are yet to been recently affected because of the adjustments. It is extremely needed mainly because possibly smaller adjustments, in a single a part of a program, often have a poor impact in various other self-sufficient regions of the program.

III. PROBLEM DEFINITION

Regression tests is a repair period application tests task that is completed to ensure there are no more insects inside application. The idea should be to do the many analyse cases which might be engendered during the advancement period in the application. But due to the moment in addition to charge difficulties, it's not at all possible to be able to do the full analyse package. Regression analyse variety is a means of lowering the analyse package on the unique analyse package to be able to nominal pair of analyse cases that could handle the many faults with nominal moment.

Agile Technique seems to be the particular desired strategy as soon as a lending broker must help to make typical produces to generation as well as uses a more rapidly launch procedure together with new adjustments, improvement, or perhaps functionalities within every single launch to generation. Regression tests is usually an crucial step up the whole tests procedure, the one that re-tests the particular already-functional areas of a new application undertaking immediately after every single launch. Right after any kind of improvement, new efficiency, or perhaps irritate deal with, it is crucial for your tester to make certain that different fully-working procedures from the software program will still be doing work not surprisingly just before it brain away to generation. Regression tests ought to be obtained since seriously as if they may be tests the particular part with regard to the first time. Regression analyse package should specify critical as well as very obvious efficiency together with to the point ways to do the particular analyse situation. Regression analyse package should include tests areas in which improvement did not help to make any kind of signal adjustments inside a module- often there is the opportunity of undetectable insects. Log regression insects effectively; this will likely assist improvement recreate the particular irritate as well as take care of quickly. Monitor just about all regression insects observed in the course of every single run. Updating/changing regression testing immediately after each and every launch ought to be the tester's obligation and is a vital section of the procedure. Regression tests is a basic section of the excellent procedure as well as helps to ensure that signal adjustments don't have bad has an effect on around the current functionalities. Efficient regression tests in the end saves a company moment as well as funds.

IV. PROPOSED SOLUTION

The proposed algorithm aims to reduce the cost of regression testing by test case suite minimization. We use the concept of Set theory for test case minimization from a larger test suite. The proposed technique can be proved to produce better results. The Algorithm finds the minimal set of test cases from the given test suite that cover all the faults in minimum execution time.

Set theory is the area of mathematical logic that deals with sets, which are collections of items or objects. Any type of object can be collected into a set. Set theory provides us some binary functions such as union (U), intersection (\cap), Set
Copyright to IJASMT www.ijarsmt.com 4



difference (-), Cartesian product(X) that can be applied to set. The union function of set theory is used as a tool to generate a minimal set of test case suite. The suggested technique introduces the concept of regression matrix. Each cell of regression matrix, c_{ij} where $1 \leq i \leq n$, $1 \leq j \leq n$, indicates the union of test case/s t_i and t_j .

Algorithm Steps:

Initialization:

1. Set threshold value, $\alpha=5$.
2. Set $n = 8$, where n is number of test cases.
3. Set $i = 2$.
4. Set $S = \phi$ (Each test case T_i is assigned a value which is equal to the number of faults covered by that test case.)
5. For ($p=1$ to n)
6. $T_p = f_p$ (Where f_p is the number of faults covered by T_i .)
7. Do {
8. If ($\alpha < 10$)
(The value of each c_{ij} of regression matrix is calculated by the union operation of row t_i and column t_j .)
9. for ($p=1$ to n)
10. for ($q=1$ to n)
11. $cpq = f\{t_p\} \cup f\{t_q\}$ (Numerically this value is the total number of faults covered by set $\{t_p\}$ and set $\{t_q\}$.)
12. if ($cpq > \alpha$)
13. $S = \{S \cup \{t_p t_q\}, ET\}$ Where $ET =$ Execution Time taken by test case $\{t_p\}$ and test case $\{t_q\}$. (S is the reduced regression matrix generated after each iteration i .)
14. $i++$
15. $\alpha = \alpha + 2$
16. while ($i \leq n$)

The reduction in the original test suite could be computed

According to the following formula:

$$C_{red} [\%] = ((CR - C_{min}) / CR) * 100$$

Where:

CR Original regression test suite

C_{min} Reduced regression test suite

V. RESULTS ANALYSIS

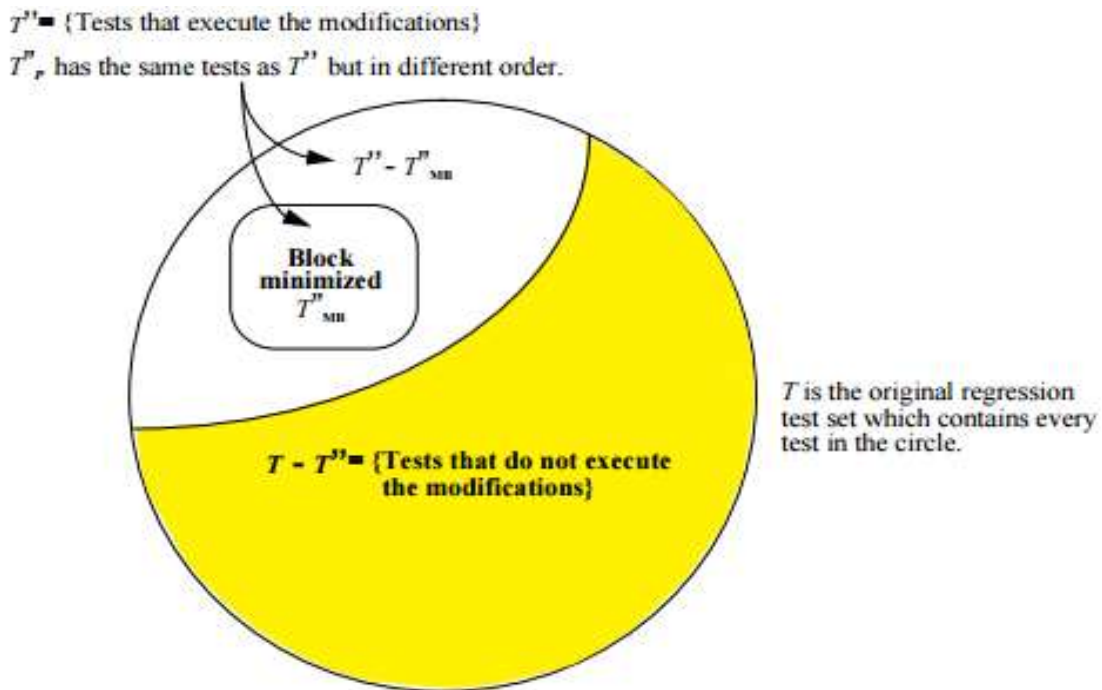
5.1 RESULT EVALUATION FACTORS

The particular proposed algorithm aims to reduce the price of regression examining by simply analyze case room minimization. Most of us utilize the technique of Arranged hypothesis for analyze case minimization from your bigger

Volume 1, Issue 6, November 2015

analyse room. The particular proposed process could be shown to create far better final results. The particular Criteria detects this little number of analyse circumstances in the presented analyse room in which protect the many errors throughout minimum amount performance time.

We have formulated some sort of internet based request to show these algorithm. The particular process formulated using this approach isolates along with lessens this analyse files. The particular approach endows far better results in the first iteration with the full method. It offers beneficial response and therefore the item causes improved options throughout maximum time. Scope of upcoming research may integrate automation with the encouraged process along with using it with complex software. The particular qualification for the proposed approach can be a analyse room. The below figure which shows the status of modified test cases as well as same test cases. The role of regression testing is started from these stage only. The original regression test set which contains every test in the circle.





VI.CONCLUSION

Regression testing is a kind of testing that helps developers make sure that there are no defects after the application has been changed. We have proposed test case selection approach from a large test suite using the concept of set theory. This approach has been tested for several examples. In above figure the dashboard page shows the Scope of future research may incorporate automation of the suggested technique and applying it on large and complex projects. We also intend to compare it with other test case reduction techniques. We can also see the result in the form of percentage.

Scope for Future Extension from application perspective

- Adding defect management activities in this web application.
- Or Allows integration of this web application with other Defect management tool. (E.g. JIRA)
- Integrate the result of automation execution.

REFERENCES

- [1] S. Yoo, M. Harman, "Regression Testing Minimization, Selection and Prioritization : A Survey", King's College London, Centre for Research on Evolution, Search & Testing, Strand, London, WC2R 2LS, UK,2010.
- [2] Harish R Madhu B K Loksha V "A Sophisticated Study on Best Practices of Agile Software Testing" International Journal of Electronics Communication and Computer Engineering Vol 3, Issue (1) NCRTCST, ISSN 2249 – 071X ,2012.
- [3] W. E. Wong, J. R. Horgan, S. London, and A. P. Mathur, "Effect of test set minimization on fault detection effectiveness". Software Practice and Experience, 28(4):347,369, April 1998.
- [4] W. E.Wong, J. R. Horgan, S. London and H.Agrawal, "A study of effective regression testing in practice," In Proceedings of the 8th IEEE International Symposium on Software Reliability Engineering (ISSRE' 97), pages 264-274, November 1997.
- [5] Rothermel, Gregg, et al. "Prioritizing test cases for regression testing."Software Engineering, IEEE Transactions on 27.10 (2001): 929-948.
- [6] Singh, Yogesh, Arvinder Kaur, and Bharti Suri. "Test case prioritization using ant colony optimization." ACM SIGSOFT Software Engineering Notes 35.4 (2010): 1-7.
- [7] Suri, Bharti, and Shweta Singhal. "Implementing Ant Colony Optimization for Test Case Selection and Prioritization." International Journal on Computer Science & Engineering 3.5 (2011).
- [8] Suri, Bharti, Isha Mangal, and Varun Srivastava. "Regression Test Suite Reduction using an Hybrid Technique Based on BCO And Genetic Algorithm."Special Issue of International Journal of Computer Science & Informatics (IJCSI), ISSN (2011): 2231-5292.
- [9] Gerard Meszaros "Agile Regression Testing Using Record & Playback", ACM 1-58113-751-6/03/0010, 2003.
- [10] G.Rothermal, R.H.Untch,C.Chu & M.J.Harrold.(2001),"Prioritizing test cases for Regression Testing",IEEE Transactions on software Engineering.
- [11] D. Hao, L. Zhang, X. Wu, H. Mei, and G. Rothermel, "On-Demand Test Suite Reduction," Proc. 34th Int. Conf. Soft. Eng., June 2012, pp. 769-779, doi:10. 1109/ICSE. 2012.
- [12] S. Yoo and M. Harman, "Regression testing minimisation, selection and prioritisation: A survey," Journal of Software Testing, Verification and Reliability, vol. 22, no. 2, pp. 67-120, 2012.
- [13] J. von Ronne, Test Suite Minimization an Empirical Study. Master's Thesis, Oregon State University, 1999.
- [14] Suri, Bharti, and Shweta Singhal. "Implementing Ant Colony Optimization for Test Case Selection and Prioritization." International Journal on Computer Science & Engineering 3.5 (2011).
- [15] Suri, Bharti, Isha Mangal, and Varun Srivastava. "Regression Test Suite Reduction using an Hybrid Technique Based on BCO And Genetic Algorithm."Special Issue of International Journal of Computer Science & Informatics (IJCSI), ISSN (2011): 2231-5292.



-
- [16] Pargas, Roy P., Mary Jean Harrold, and Robert R. Peck. "Test-data generation using genetic algorithms." *Software testing verification and reliability* 9.4 (1999): 263-282.
- [17] G.Duggal, B.Suri,"Understanding Regression Testing Techniques", COIT, 2008, India.
- [18] H. Leung and L. White, "Insights into regression testing," In *Proceedings of the Conference on Software Maintenance*, pages 60-69, Oct. 1989.
- [19] K.K.Agarwal & Yogesh Singh, —*Software Engineering Programs Documentation, Operating Procedures*, New Age International Publishers, Revised Second Edition – 2005.
- [20] Glenn Vanderburg, "A Simple Model of Agile Software Practices - or - Extreme Programming Annealed," in *Object-Oriented Programming, Systems, Languages, and Applications*, New York, 2005, pp. 539-545.