2019 IEEE 7th Conference on Systems, Process and Control (ICSPC 2019), 13–14 December 2019, Melaka, Malaysia

A Survey on Test Case Prioritization and Optimization Techniques in Software Regression Testing

Mohammed Hazim Alkawaz
Faculty of Information Sciences & Engineering
Management and Science University
Shah Alam, Selangor, Malaysia
mohammed_hazim@msu.edu.my

Abrahmi Silvarajoo
School of Graduates Studies
Management and Science University
Shah Alam, Selangor, Malaysia
abrahmisilvarajoo@gmail.com

Abstract—**Software Testing (ST) is vital for software development to check the degree to which it meets client's prerequisites, watch the errors or bugs in the software code and find a way to correct the mistakes to make software advantageous. For approval of changes in software, Regression Testing (RT) must be connected. RT plays it role during the software maintenance phase. It guarantees mistake free software after change during maintenance. Test cases and test suites are set up for testing, furthermore, it ought to be done in least time where Test Case Prioritization and Optimization (TCPAO) strategies are required.The fundamental point of TCPAO is to test software in least time and with greatest proficiency. This paper discusses, an empirical study on the TCPAO techniques in software RT. This study focouses on the beginners and intermediate researches. The most pertinent studies were chosen from the appropriate repositories by utilizing a lot of search keywords, inclusion/exclusion criteria. The study also discusses on the most recent papers on TCPAO technique.**

Keywords: Software Development Life Cycle (SDLC), Regression Testing, Software Testing, Test Case Prioritization (TCP), Test Case Optimization (TCO), Test Case Selection(TCS), Test Case Minimization(TCM)

I. INTRODUCTION

ST is a huge time of SDLC, furthermore, it can be the most costly stage [1]. In any case, the utilization of TCPAO seems to upgrade test suitability in ST action.

RT makes sure that any newly added code does not impact the previous functionality of the software. As the RT ongoing during the maintenance phase the test suite will be always updated in order to check the old and new functionalities [2]. The primary test suite is constantly accessible for re-use in RT, yet whole net worthy test suite can't be executed because of time and spending objectives .Thus we have to choose and organize the test suite accordingly [3]. RT methodologies grouped into three categories; minimization, selection, and prioritization [4].

TSM approaches expect to recognize tedious examinations and to wipe out test cases from a test suite execution with a particular objective. Minimization is now and then called 'test suite reduction', which means the end of test cases is changeless [5].

TCS approach likewise plans to reduce the quantity of the test cases, in any case, the fundamental thought of choice methodology is that it is proposed to be change mindful [6].

Lastly, TCP intends to arrange a lot of test cases to accomplish an early improvement dependent on favored properties. It enables a way to deal with execute profoundly critical test cases initially as per some measure, and produce the ideal result, for example, uncovering deficiencies prior and giving input to the testers [5].

II. LITERATURE REVIEW

A. Software develoment

SDLC comprises of subtleties steps and exercises which portray how to design, maintain, replace, alter, adjust, improve, test or even dispatch as software. The exercises can be separated into a very detail level and yet they can be assemble into five center classes, which are Requirement Analysis, Design, Implementation, Testing and Evolution [7].The following is the realistic portrayal which shows an ordinary SDLC.

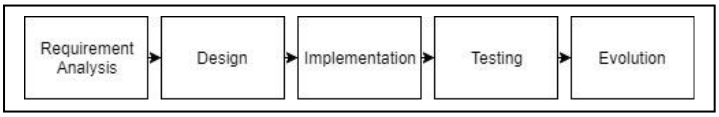

Figure 1.Software Development Life Cycle (SDLC)

B. Software testing

ST is a process of checking whether the software works as expected. ST is to guarantee that product should always be defect free and effectively kept up [8].It's an exceptionally wide territory, which includes numerous other specialized and non-specialized regions, for example, detail, design and implementation, maintenance, procedure and the board issues in software engineering. At the end of the day ST is verification

978-1-7281-1826-0/19/$31.00 ©2019 IEEE

and validation process [9]. RT in ST involves during maintenance phase of ST.

C. *Regression testing*

RT is one sort of ST that is performed with changes of current software. It is fundamental to give certainty that such changes, which are recently presented, don't block the conduct of the unaltered existing pieces of the software. RT is exorbitant, and speaks to an essential issue in the software development [10].

Regular techniques for RT are re running recently finished test cases and after that checking whether the program conduct has been changed or the recently fixed issues have re-developed [11]. This requires a great deal of expense and time as the size and complexity of the software increase. Retest All, Hybrid Approach, Prioritization of Test Cases and Regression Test Case Selection are RT techniques.

Test Cases can be prioritized as far as random, optimal, statement coverage total or additional, branch coverage total or additional, failure rates, or total fault exposing potential of the test cases [12]. Numerous systems have been utilized for prioritization as indicated by at least one of the picked criteria [13]. Various activities carried out while performing the RT.

D. *Test case prioritization*

TCP is planning of test cases in some request that they distinguish the blames extremely quick and effectively in that specific request instead of some irregular request [15]. As it is outstanding that there is no such indicated spending plan and time for RT, so the point is to discover most extreme faults in least time; for that course of action of test cases or test suites is done in some restricted request[16]. This even characterizes a few criteria like flaw identification and code inclusion as per which test cases are organized [17].

- **Search-Based:** Search based methodology can fluctuate as far as usage: For instance, Greedy algorithm was found profoundly identified with coverage-based goal.
- **Coverage-Based:** This methodology is a key testing approach and reviews the code legitimately. Code, statement, branch, or function inclusion exists in two principle classes: total and additional coverage.
- **Fault-Based:** Fault based methodologies produce a succession of test cases to identify focused on deficiencies, which may ease researchers to have an objective on which blames that ought to be identified
- **Requirement-based:** Requirement based uses prerequisites data during necessity elicitation, for example, client need, blunder inclined worth, instability worth, and execution troubles for prioritization criteria.
- **History-based:** History-based uses history information as a contribution to execute experiment prioritization. In light of the perception in the essential examinations, history-based usage components have to do with its information that is effectively accessible for any pursuit based exploratory arrangement.
- **Risk-based:** This methodology is for the most part utilized in a venture that commonly worries on risk qualities identified with the product to be created.
- **Bayesian-Network based:** Usage of Bayesian Networks anticipates the plausibility of mistake to be found in each and every test cases by the methods for various available information.
- **Cost-Effective-based:** Cost-aware test case prioritization procedure integrates historical data to reduce testing cost.
- **Topic-Model-based:** Method that uses phonetic information, for example, identifier markers, comments, and string literals to help separate their usefulness.

III. RESEARCH METHOD

In the primary stage, four fundamental research questions were created to answer the principle point of this paper. Next, determination of important repositories was performed. The yield of the search stage was then moved into the study selection phase.

A. *Research questions*

Q1: What are the taxonomies of TCP techniques?
Q2: Why TCP is important in RT?
Q3: What are the strengths and limitations of RT Approaches?
Q4: What will be the empirical and SLR studies of TCP?
Q5: What are the latest related studies of TCPAO techniques?

B. *Selection of repositories& Search Strategies*

The authors began this determination technique by entering **'TCP', 'RT', 'TCPAO techniques** 'as search strings with the exact phrase on Google Scholar and Research Gate database. The picked repositories are **IEEE Xplore**, **ACM Digital Library**, and **Spring.**

C. *Study selection*

Table 1, shows the inclusion and exclusion criteria of the paper.

TABLE 1. INCLUSION AND EXCLUSION CRITTERIA OF THE PAPER

Inclusion Criteria	*Exclusion Criteria*
English Paper	Non-English paper
Paper concentrating on TCP procedures	Paper doesn't have any connection with TCP approaches
Paper that can answer to at least one research question	Similar studies, latest paper included.

IV. RESULT AND DISCUSSION

In this section each research questions were answered.

1. **What are the taxonomies of TCP techniques?**

Figure 2, shows the detailed taxonomies of the TCP which helps the researchers easily understand the basic of T

2. **Why TCPAO is important in RT?**

Test Case prioritization is planning of test cases in some request that they distinguish the blames extremely quick and effectively in that specific request instead of some irregular request. As it is outstanding that there is no such indicated spending plan and time for RT, so the point is to discover most extreme faults in least time; for that course of action of test cases or test suites is done in some restricted request.

3. **What are the strengths and limitations of RT Approaches?**

Table 2, shows the strengths and limitation of the RT approaches.

TABLE 2. STRENGTHS AND LIMITATIONS OF REGRESSION TESTING APPROACHES

RT Approach	*Strength*	*Limitation*
Test Case Minimization	Effective in reducing test cases	Test cases are not change mindful.
Test Case Selection	Successful in choosing alterations mindful test cases.	New test cases might be missed out in the temporary selection that is modification-aware.
Test Case Prioritization	Valuable when new test cases will consistently be considered in the test case stage.	Time-consuming, larger test suite.

4. **What will be the empirical and SLR studies of TCP?**

Table 3, shows the empirical and SLR studies of TCP.

5. **What are the latest related studies of TCPAO techniques?**

Table 4, shows the related studies of TCPAO techniques.

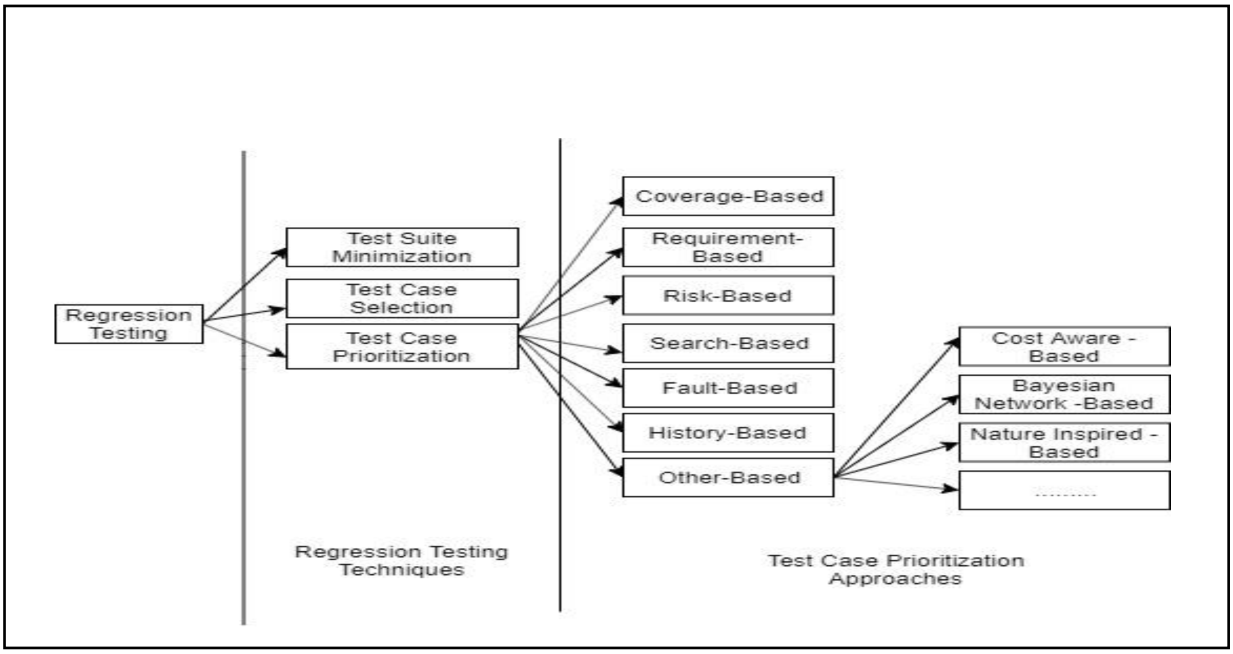

Figure 2.Taxonomies of Test Case Prioritization and Optimization Techniques

TABLE 3: EMPIRICAL AND SLR STUDIES OF TCP

Study Type	Reference ID	Study Focus	Years of Publication	Total Studies Reviewed	Year Covered
Empirical Study	.[15]	Mutation-based test case prioritization techniques	2019	NA	NA
	[16]	Various Regression Testing Approaches	2019	NA	NA
	[17]	Defect Prediction for Test Case Prioritization	2019	NA	NA
	[18]	Impact of Parameter Settings and Operators Role for Genetic Algorithm based Test Case Prioritization	2019	NA	NA
	[1]	Test Case Prioritization and Optimization Techniques	2018	NA	NA
	[19]	Prioritization Regression Testing Test Case	2017	NA	NA
	[20]	Test-case prioritization: achievements and challenges.	2016	NA	NA
Systematic Literature Review (SLR) Mapping Study	[21]	Test Case Prioritization Approaches in Regression Testing	2019	27	2000-2019
	[22]	Regression Test-Case Prioritization	2019	191	1997-2016
	[23]	Selecting an appropriate Requirements Based Test Case Prioritization Technique	2019	91	2000 -2018.
	[24]	Different Approaches for software test case prioritization	2018	90	2001-2018
	[25]	Regression test case prioritization approaches	2018	36	2004-2018
	[26]	Software Regression Testing Techniques	2016	25	2000-2014

TABLE 4: RELATED STUDIES OF TCOAP TECHNIQUES

Reference ID	Algorithm	Technique	Coverage and Implementation	Metric used
[27]	TCP algorithm	Tie breaker approach	• ROCKET Prioritization for Continuous Regression Testing	Average Percentage Faults Detected (APFD)
[28]	Law of gravitation algorithm	Location-based TCP technique	• Smart mall test gravitation Consolidates three components (test case information, fault information, and location information), to prioritize test cases.	_
[29]	Ant Colony Optimization Algorithm	No ordering, random prioritization and reverse prioritization techniques	Test Case Prioritization-ANT (TCP-ANT)	APFD
[30]	Fingerprint algorithms	Fingerprinting software	Telemetry Fingerprinting	_
[31]	Dynamic Programming Algorithm	Dynamic Programming approach	Concentrate on low memory utilization in pseudo-polynomial time multifaceted applicable in both selection and selection-prioritization problems over sets of test cases or test suites.	_
[32]	Genetic Algorithm	Test Case Prioritization and Optimization strategy	• Accept Saving Details for an Income Tax Calculator. Total statement coverage, necessity need factor esteem, risk exposure, and execution time	-
[33]	Rule mining algorithms, Search algorithms	Mining and Multiobjective search (named as REMAP)	Detect faults earlier	APFD per cost
[34]	modified Page-Rank algorithm	Model-based approach	The methodology permits modeling of necessities and test related data and their relationships.	_

V. CONCLUSION

In this paper, an empirical survey has been performed on the strategies utilized in TCPAO. All the accessible related papers of the research have been taken from the literature. They are assessed, and each paper has been dissected based on strategies or algorithm utilized, metrics utilized for productivity, coverage that has been taken and execution premise of that paper. Subsequent to perusing this present paper, researchers can get a quick review of the papers. They can likewise know the profundity of the papers and their research work.

ACKNOWLEDGEMENT

Authors are thankful to the School of Graduate Studies and the Faculty of Information Sciences and Engineering, Management and Science University, Malaysia for their help.

REFERENCES

[1] Saraswat, P., Singhal, A., & Bansal, A. (2019). A Review of Test Case Prioritization and Optimization Techniques *Software Engineering* (pp. 507-516): Springer.

[2] Solanki, K., Singh, Y., & Dalal, S. (2015). *Test case prioritization: an approach based on modified ant colony optimization (m-ACO).* Paper presented at the 2015 International Conference on Computer, Communication and Control (IC4).

[3] Ansari, A., Khan, A., Khan, A., & Mukadam, K. (2016). Optimized regression test using test case prioritization. *Procedia Computer Science, 79*, 152-160.

[4] Khatibsyarbini, M., Isa, M. A., Jawawi, D. N., & Tumeng, R. (2018). Test case prioritization approaches in regression testing: A systematic literature review. *Information and Software Technology, 93*, 74-93.

[5] Fraser, G., & Rojas, J. M. (2019). Software Testing *Handbook of Software Engineering* (pp. 123-192): Springer.

[6] Steffens, A., Lichter, H., & Moscher, M. (2018). *Towards Data-driven Continuous Compliance Testing.* Paper presented at the Software Engineering (Workshops).

[7] Rosero, R. H., Gómez, O. S., & Rodríguez, G. (2016). 15 years of software regression testing techniques—A survey. *International Journal of Software Engineering and Knowledge Engineering, 26*(05), 675-689.

[8] Do, H. (2016). Recent advances in regression testing techniques *Advances in Computers* (Vol. 103, pp. 53-77): Elsevier.

[9] Khanna, E. (2016). Regression Testing based on Genetic Algorithm. *International Journal of Computer Applications, 154*(8).

[10] Mittal, S., & Sangwan, O. P. (2018). Prioritizing test cases for regression techniques using metaheuristic techniques. *Journal of Information and Optimization Sciences, 39*(1), 39-51.

[11] Bajaj, A., & Sangwan, O. P. (2018). *A survey on regression testing using nature-inspired approaches.* Paper presented at the 2018 4th International Conference on Computing Communication and Automation (ICCCA).

[12] Wang, S., Nam, J., & Tan, L. (2017). *QTEP: quality-aware test case prioritization.* Paper presented at the Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering.

[13] Huang, R., Zhou, Y., Zong, W., Towey, D., & Chen, J. (2017). *An empirical comparison of similarity measures for abstract test case prioritization.* Paper presented at the 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC).

[14] Sultan, Z., Bhatti, S. N., Abbas, R., & Shah, S. A. A. (2017). Analytical review on test cases prioritization techniques: An empirical study.

[15] Shin, D., Yoo, S., Papadakis, M., & Bae, D. H. (2019). Empirical evaluation of mutation-based test case prioritization techniques. *Software Testing, Verification and Reliability, 29*(1-2), e1695.

[16] Solanki, K., & Dalal, S. (2019). Retrospection on Various Regression Testing Approaches.

[17] Paterson, D., Campos, J., Abreu, R., Kapfhammer, G. M., Fraser, G., & McMinn, P. (2019). *An Empirical Study on the Use of Defect Prediction for Test Case Prioritization.* Paper presented at the 2019 12th IEEE Conference on Software Testing, Validation and Verification (ICST).

[18] Bajaj, A., & Sangwan, O. P. (2019). Study the Impact of Parameter Settings and Operators Role for Genetic Algorithm Based Test Case Prioritization. *Available at SSRN 3356318.*

[19] Suleiman, D., Alian, M., & Hudaib, A. (2017). *A survey on prioritization regression testing test case.* Paper presented at the 2017 8th International Conference on Information Technology (ICIT).

[20] Hao, D., Zhang, L., & Mei, H. (2016). Test-case prioritization: achievements and challenges. *Frontiers of Computer Science, 10*(5), 769-777.

[21] Paygude, Priyanka. (2019). Comparative analysis of Test Case Prioritization Approaches in Regression Testing. International Journal of Advanced Trends in Computer Science and Engineering. 1260-1267. 10.30534/ijatcse/2019/36842019.

[22] Yiling, Lou & Chen, Junjie & Zhang, Lingming & Hao, Dan. (2018). A Survey on Regression Test-Case Prioritization. 10.1016/bs.adcom.2018.10.001.

[23] Dongoor, S. P. (2019). Selecting an appropriate RequirementsBased Test Case PrioritizationTechnqique.

[24] Mukherjee, Rajendrani & Patnaik, K Sridhar. (2018). A Survey on Different Approaches for Software Test Case Prioritization. Journal of King Saud University - Computer and Information Sciences. 10.1016/j.jksuci.2018.09.005.

[25] Dahiya, Omdev & Solanki, Dr. (2018). A systematic literature study of regression test case prioritization approaches. Vol 7, No 4 (2018). 2184-2191. 10.14419/ijet.v7i4.15805.

[26] Rosero Miranda, Raul & Gómez, Omar S. & Rodriguez, Glen. (2016). 15 Years of Software Regression Testing Techniques -- A Survey. International Journal of Software Engineering and Knowledge Engineering. 26. 675-689. 10.1142/S0218194016300013.

[27] Marijan, Dusica & Gotlieb, Arnaud & Sen, Sagar. (2013). Test Case Prioritization for Continuous Regression Testing: An Industrial Case Study. IEEE International Conference on Software Maintenance, ICSM. 540-543. 10.1109/ICSM.2013.91.

[28] Wang, X., Zeng, H., Gao, H., Miao, H., & Lin, W. (2019). Location-Based Test Case Prioritization for Software Embedded in Mobile Devices Using the Law of Gravitation. *Mobile Information Systems, 2019.*

[29] Vescan, A., Pintea, C.-M., & Pop, P. C. (2019). *Solving the Test Case Prioritization Problem with Secure Features Using Ant Colony System.* Paper presented at the International Joint Conference: 12th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2019) and 10th International Conference on EUropean Transnational Education (ICEUTE 2019).

[30] Anderson, J., Azizi, M., Salem, S., & Do, H. (2019). On the use of usage patterns from telemetry data for test case prioritization. *Information and Software Technology, 113*, 110-130.

[31] Banias, O. (2019). Test case selection-prioritization approach based on memoization dynamic programming algorithm. *Information and Software Technology*.

[32] Mishra, D. B., Mishra, R., Acharya, A. A., & Das, K. N. (2019). Test Case Optimization and Prioritization Based on Multi-objective Genetic Algorithm *Harmony Search and Nature Inspired Optimization Algorithms* (pp. 371-381): Springer.

[33] Pradhan, D., Wang, S., Ali, S., Yue, T., & Liaaen, M. (2019). Employing rule mining and multi-objective search for dynamic test case prioritization. *Journal of Systems and Software, 153*, 86-104.

[34] Abbas, M., Inayat, I., Saadatmand, M., & Jan, N. (2019). *Requirements Dependencies-Based Test Case Prioritization for Extra-Functional Properties.* Paper presented at the 2019 IEEE International Conference on Software Testing, Verification and Validation Workshops (IC