International Journal of Future Generation Communication and Networking
Vol. 14, No. 1, (2021), pp. 3394–3404

An Automated Approach for Web Service Test Case Optimization Using Regression Testing

[1]Sirisha K L S , [2]Dr.M.Chandra Mohan

[1]Research Scholar, JNTUH, Telangana, India.
[2]Professor, Department of CSE, JNTUH, Telangana, India.

ABSTRACT

Web services technology enabled enterprise applications to have heterogeneous platform support to make the application development easier. In fact, web services provide inter-operability to enable seamless integration of heterogeneous applications of chain of related businesses. In this regard, there are different services that are developed in house and that are used with API calls provided by third parties. Therefore, web services testing has become very important research area as it is so complex in nature. There are many existing methods for generating test suits for web services and even generation of test cases automatically when changes are made to web services.

A framework is proposed to generate test cases with optimization for regression testing. It has an approach that could accurately determine changes made to web service and the test cases needed to be created or updated and the generation of optimized test cases required by regression testing. Towards this end three different algorithms are proposed and implemented. The experimental results revealed that the proposed framework is useful and paves way for further research in the area of test case generation with more sophistication in future.

Keywords *– Web services technology, dynamic test case generation, regression testing, change detection, test case optimization*

1. INTRODUCTION

Web services technology enabled enterprise applications to have heterogeneous platform support to make the application development easier. In fact, web services provide inter-operability to enable seamless integration of heterogeneous applications of chain of related businesses. In this regard, there are different services that are developed in house and that are used with API calls provided by third parties. Therefore, web services testing has become very important research area as it is so complex in nature. There are many existing methods for generating test suits for web services and even generation of test cases automatically when changes are made to web services.

Masood *et al*. [3] focused on regression testing that is made automatically in web services based applications. They developed a methodology to generate test cases required by regression testing automatically. Konsaard and Ramingwong [4] proposed a methodology that takes care of total coverage based regression test case generation and prioritization using Genetic Algorithm (GA).Zarrad [9] made an extensive review of regression testing procedures in web services based applications while Rohatgi *et al*. [10] also did the same with empirical study on the automatic regression testing.

ISSN: 2233-7857IJFGCN

Tarhini *et al*. [11] focused on web services based applications and the concept of regression testing and its importance in leveraging such applications. Sahoo and Ray [12] focused on the optimization of regression testing and its test cases when test cases are to be modified based on the changes made to underlying web services.Engström *et al*. [14] focused on review of many regression testing techniques used for testing web services.Zhong *et al*. [16] developed a framework known as TestSage that performs regression test selection for testing web services in large scale.

Yoo and Harman [17] focused on regression testing and also prioritization of test cases.Masood *et al*. [19] developed an automated approach to regression testing. From the literature, it is understood that there is more comprehensive approach towards automatic detection of modifications in existing web services and optimize test cases using regression testing. Towards this end, in this paper, we proposed required methodology and algorithms. Our contributions in this paper are as follows.

1. A framework is proposed to support automatic update of test cases when web services are subjected to modifications.
2. Three algorithms such as Web Service Change Detection (WSCD), Forward Slicing (FS) and Regression Test (RT) are defined to realize the framework. These algorithms are executed in the same sequence to obtain optimized test cases for modified web services.
3. A prototype application to show the usage of the proposed framework and the underlying algorithms.

The remainder of the paper is structured as follows. Section 2 reviews literature on existing methods associated with test cases generation. Section 3 presents the proposed framework and the algorithms defined. Section 4 presents a case study and results. Section 5 concludes the paper and provides directions for future work.

2. RELATED WORK

This section provides review of relevant literature associated with automatic web services test case generation. Lizarralde *et al*. [1] developed deep autoencoder for detection of web service changes that helps in generation of test cases automatically. Zhang *et al*. [2] on the other hand proposed a framework for metaheuristics based optimization leading to better performance. Masood *et al*. [3] focused on regression testing that is made automatically in web services based applications. They developed a methodology to generate test cases required by regression testing automatically. Konsaard and Ramingwong [4] proposed a methodology that takes care of total coverage based regression test case generation and prioritization using Genetic Algorithm (GA). They found that the evolutionary approach could provide better results when compared to the state of the art. Fuentes*et al*. [5] proposed a system for detection of outliers and optimizations in distributed environments.

Zheng *et al*. [6] studied the importance of automatic test case generation in the real world applications. Kim and Wu [7] focused on reinforcement learning based optimization. Sellami *et al*. [8] focused on the automatic discovery and composition of web services to make enterprise level applications. Zarrad [9] made an extensive review of regression testing procedures in web services based applications while Rohatgi *et al*. [10] also did the same with empirical study on the automatic regression testing.

Tarhini *et al*. [11] focused on web services based applications and the concept of regression testing and its importance in leveraging such applications. Sahoo and Ray [12] focused on the optimization of regression testing and its test cases when test cases are to be modified based on the changes made to underlying web services. Mukherjee and Patnaik [13] investigated on the test cases prioritization. In

ISSN: 2233-7857IJFGCN

the field of software engineering, they found the importance of prioritizing test cases in order to have better results. Engström*et al*. [14] focused on review of many regression testing techniques used for testing web services. Lakshmi *et al*. [15] investigated on the testing of web applications in the contemporary era with automation testing.

Zhong *et al*. [16] developed a framework known as TestSage that performs regression test selection for testing web services in large scale. Yoo and Harman [17] focused on regression testing and also prioritization of test cases. Almulla*et al*. [18] used fuzzy logic based system to visualize real world web services associated with distributed applications. Masood*et al*. [19] developed an automated approach to regression testing while Bozkurt*et al*. [20] focused on testing web services automatically. From the literature, it is understood that there is more comprehensive approach towards automatic detection of modifications in existing web services and optimize test cases using regression testing. Towards this end, in this paper, we proposed required methodology and algorithms.

3. PROPOSED SYSTEM

We proposed a framework for automatic execution of test cases for new web service and generation of optimized test cases for modified web services for regression testing. The framework has different mechanisms to deal with the intended operations. It is used by test engineers when they need to test new web service or any modified web service. The automation of the process is taken care of the proposed framework. It ensures that the test engineer gets desired services for either new web service or an existing web service that has been modified.

3.1 THE FRAMEWORK

As presented in Figure 1, the proposed framework has operations such as generation of operations tree, detection of change to the signatures of functions associated with the web service, dynamic forward slicing to detect the kind of change with ease and apply optimized regression test algorithm to generate optimized test cases for changed web services. For web services that are new (not subjected to modifications), the existing test cases are executed.

ISSN: 2233-7857IJFGCN

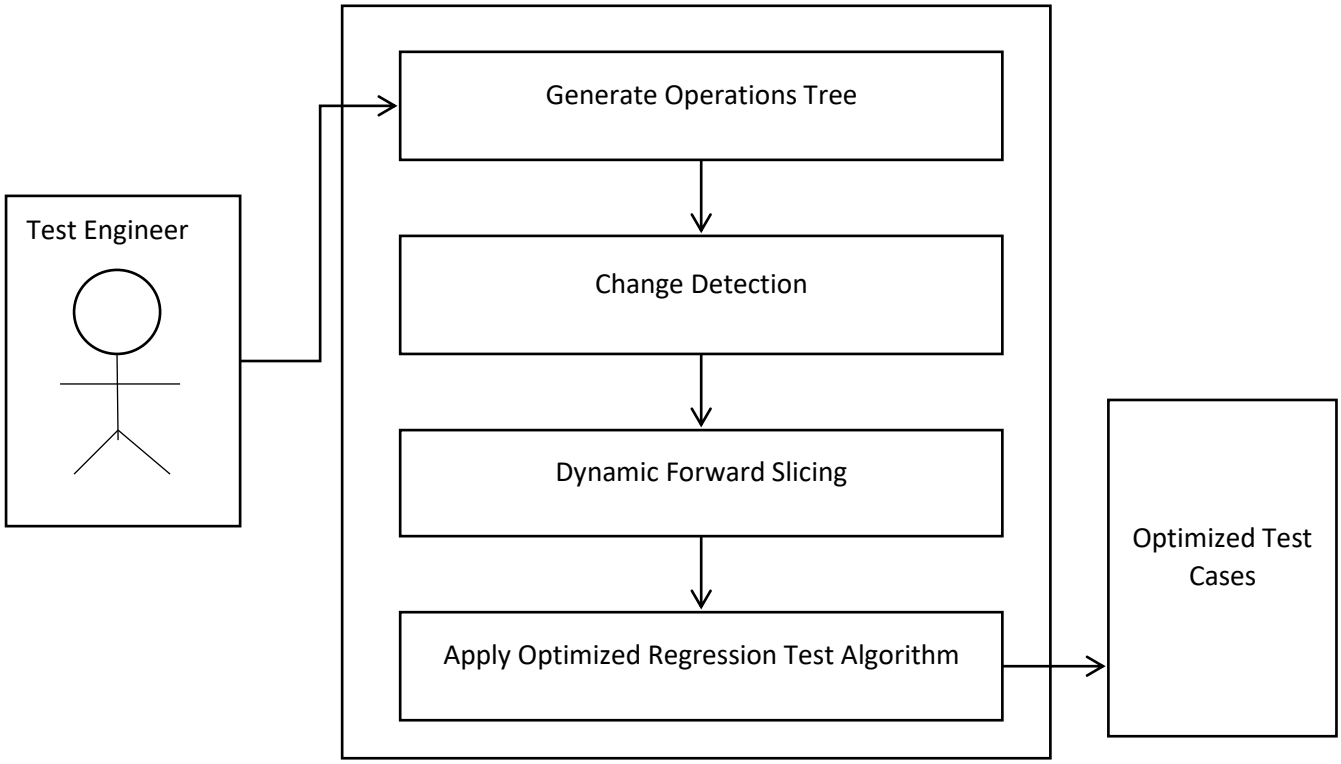
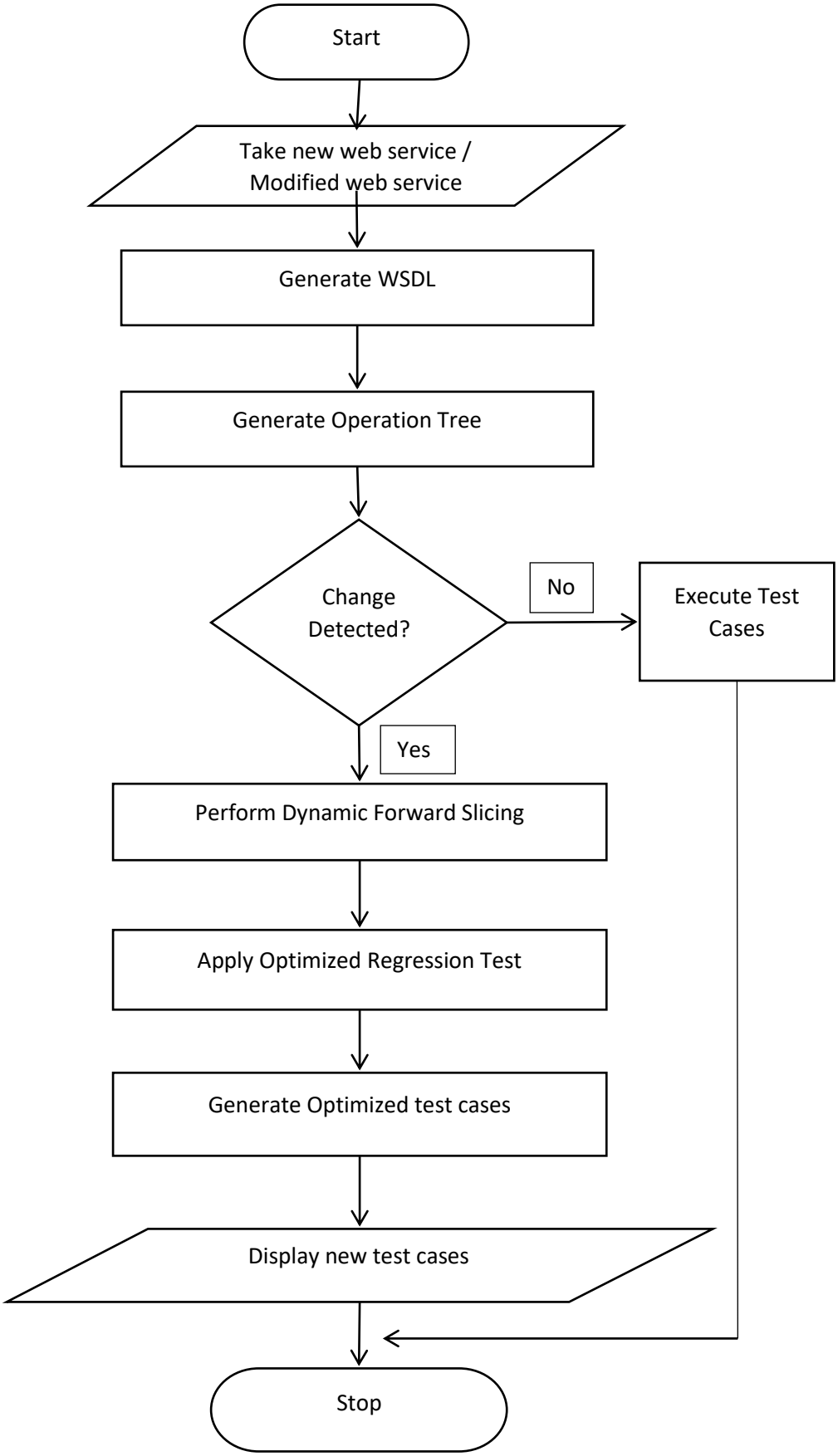


Figure 1: Model of the proposed framework

Different algorithms are defined to fulfil the operations of the proposed framework. One important algorithm is the change detection algorithm that is crucial for dynamically updating test cases by detecting kind of change using regression testing. In order to choose only modified test cases, there is an algorithm defined known as dynamic forward slicing. Optimized regression test algorithm provides a set of optimized test cases for regression testing.

ISSN: 2233-7857IJFGCN



ISSN: 2233-7857IJFGCN

Figure 2: Abstract view of proposed methodology

The methodology illustrated in Figure 2 helps for two different use cases. One use case is when a new web service is to be handled while the second use case is when an existing web service that has been subjected to changes is to be handled. In either case, there is utility with respect to dynamically testing or generating optimized test cases. After generating WSDL and operations tree, different algorithms are possible to detect changes, find test cases that are to be changed and generate optimized test cases. The algorithms are defined in the following sub sections.

3.2 ALGORITHM FOR CHANGE DETECTION

This algorithm is meant for detecting different kinds of changes made to given web service. This will help in taking further steps in automatic test case generation or update of existing test cases.

Algorithm 1: Web service change detection algorithm

Algorithm: Web Service Change Detection (WSCD)
Inputs: Operations tree t1, modified operations tree t2
Output: List of changes

1. Extract nodes from t1 and t2
2. If root nodes of t1 and t2 do not match Then
3. Deletion is detected
4. End If
5. If Operation Name not matched Then
6. Deletion is detected
7. End If
8. IF input node of t1 and t2 matches Then
9. Next child of input node is to be compared
10. Else
11. Updated it with new value and continue the process
12. End If
13. IF message node of t1 and t2 matches Then
14. Next child of message node is to be compared
15. Else
16. Updated it with new value and continue the process
17. End If
18. IF part node of t1 and t2 matches Then
19. Next child of part node is to be compared
20. Else
21. Updated it with new value and continue the process
22. End If

As presented in Algorithm 1, it takes operations tree of web service and the modified operations tree and compare them both node by node in order to arrive at change detection accurately.

3.3 ALGORITHM FOR FORWARD SLICING

This algorithm is meant for finding the test cases to be modified provided a changed operations tree of a web service.

ISSN: 2233-7857IJFGCN

Algorithm 2: Forward slicing algorithm

Algorithm: Forward Slicing (FS)
Input: Operations tree for modified web service
Output: slicing list

1. IF changed operation is insert Then
2. Child nodes are traversed and dependencies are verified
3. Update slice list
4. End If
5. IF changed operation is modify Then
6. Other nodes are traversed and dependencies are verified
7. Update slice list
8. End If
9. IF changed operation is delete Then
10. Other nodes are traversed and dependencies are verified
11. Update slice list
12. End If

As presented in Algorithm 2, it takes operations tree of web service that has been modified and traverse all nodes to generate slicing list based on detected changes in order to facilitate test case generation.

3.4 ALGORITHM FOR REGRESSION TESTING

This algorithm is meant for generating test cases for regression testing.

Algorithm 3: Regression test algorithm

Algorithm: Regression Test (RT)
Input: slicing list, latest test cases
Output: Optimized test cases

1. Compare latest test cases and slicing list
2. If slicing list reflects insert operation Then
3. Update test case for insert operation
4. End If
5. Execute Step 1
6. If slicing list reflects modification operation Then
7. Search for value of changed attribute
8. If new value is found Then
9. Add new test case
10. Else
11. Update test case to optimize
12. End If
13. End If
14. Execute Step 1
15. If slicing list reflects unchanged operation Then
16. Discard test cases
17. End If

ISSN: 2233-7857IJFGCN

As presented in Algorithm 3, the regression test algorithm generates optimized test cases that can be used for regression testing.

4. IMPLEMENTATION CASE STUDY

Calculator web service is considered for evaluating the proposed framework and observe the results of different algorithms.

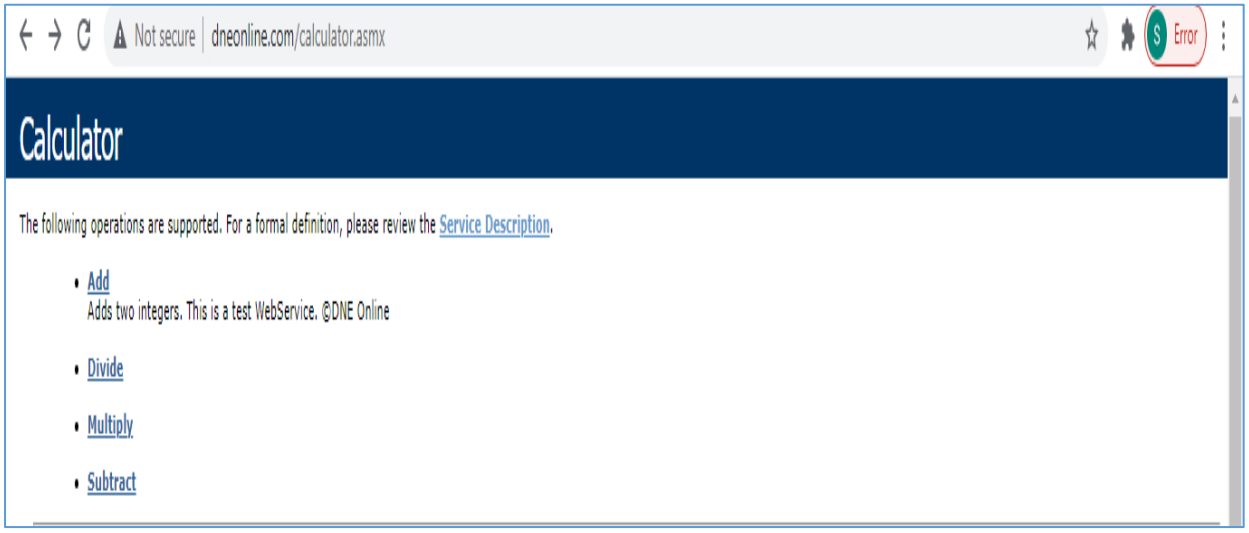


Figure 3: Shows existing Calculator web service

Figure 4: Shows WSDL of Calculator web service

ISSN: 2233-7857IJFGCN

As presented in Figure 3 and Figure 4, the Calculator web service has its own operations and the WSDL reflects operations and can be used programmatically to generate client application and then test all operations.

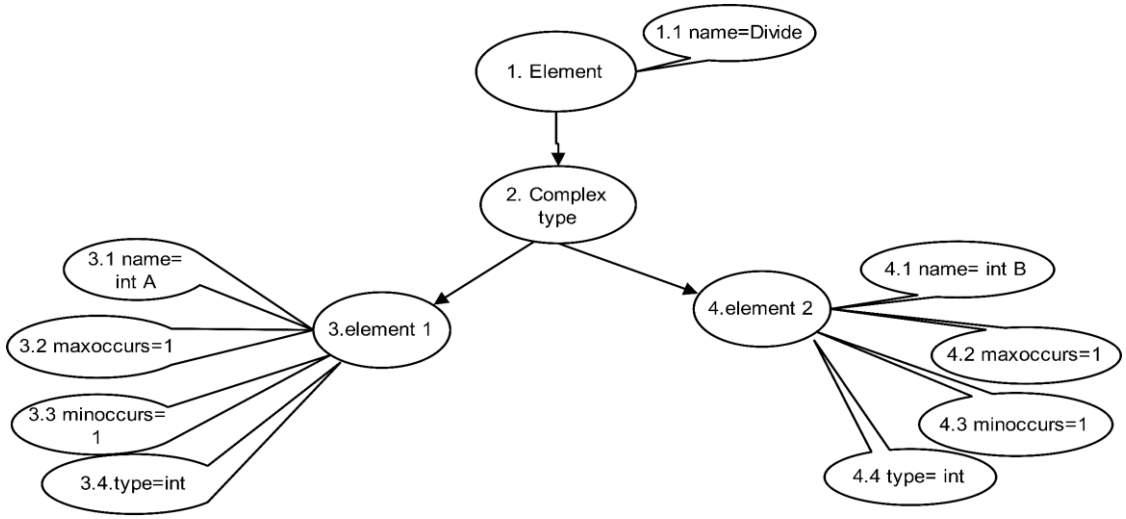


Figure 5: Operations tree of Calculator web service

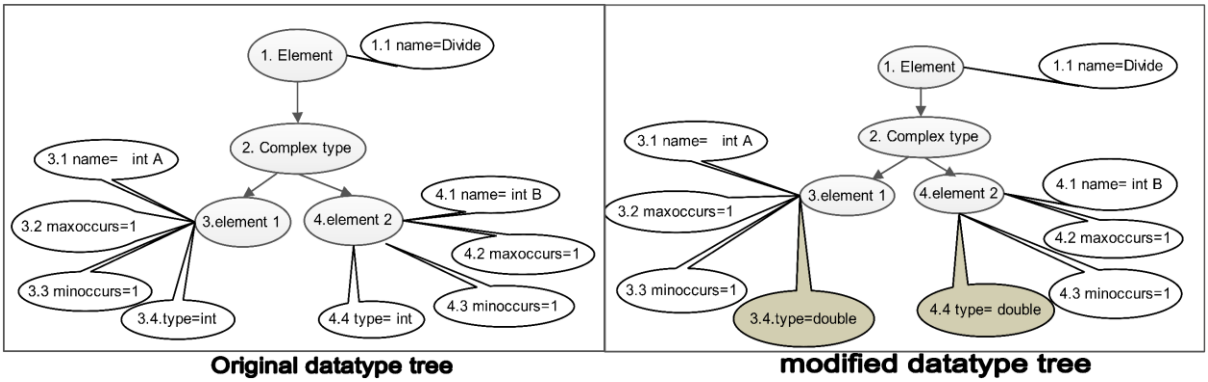


Figure 6: Shows difference between original datatype tree and modified datatype tree

As can be seen in Figure 5 and Figure 6, it is evident that the operations tree is able to distinguish changes. When it is subjected to the three algorithms defined in the previous section, it results in generation of optimized test cases.

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed a framework to generate test cases with optimization for regression testing. It has an approach that could accurately determine changes made to web service and the test cases needed to be created or updated and the generation of optimized test cases required by regression testing. Three different algorithms are defined to fulfil the operations of the proposed framework. One important algorithm is the change detection algorithm that is crucial for dynamically updating test cases by detecting kind of change using regression testing. In order to choose only modified test cases, there is an algorithm defined known as dynamic forward slicing. Optimized regression test algorithm provides a set of optimized test cases for regression testing. Particularly there are two use cases. One use case is when a new web service is to be handled while the second use case is when an existing web service that has been subjected to changes is to be handled. In either case, there is utility with respect to dynamically testing or generating optimized test cases. After generating WSDL and

ISSN: 2233-7857IJFGCN

operations tree, different algorithms are possible to detect changes, find test cases that are to be changed and generate optimized test cases.The experimental results revealed that the proposed framework is useful and paves way for further research in the area of test case generation with more sophistication in future.

REFERENCES

1. [1] Lizarralde, I., Mateos, C., Zunino, A., Majchrzak, T. A., &Grønli, T.-M. (2020). *Discovering web services in social web service repositories using deep variationalautoencoders. Information Processing & Management, 57(4),* p1-19.
2. [2] Zhang, F., Sun, Q., Mehrabadi, M., Khoshnevisan, B., Zhang, Y., Fan, X., … Liu, H. (2021). *Joint analytical hierarchy and metaheuristic optimization as a framework to mitigate fertilizer-based pollution. Journal of Environmental Management, 278,* p1-12.
3. [3] Tehreem Masood, Aamer Nadeem, S. Ali. (2013) "An automated approach to regression testing of web services based on WSDL operation changes", In Proceedings of 9 th IEEE International Conference on Emerging Technologies (ICET), pp. 1-5.
4. [4] Konsaard, P., & Ramingwong, L. (2015). *Total coverage based regression test case prioritization using genetic algorithm. 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON).* P1-6.
5. [5] Fuentes, R., Gardner, P., Mineo, C., Rogers, T. J., Pierce, S. G., Worden, K., … Cross, E. J. (2020). *Autonomous ultrasonic inspection using Bayesian optimisation and robust outlier analysis. Mechanical Systems and Signal Processing, 145,* p1-20.
6. [6] Zheng, Z., Chen, Y., Guo, bing, Wang, Y., Liu, W., Sun, J., & Wang, X. (2020). *Magnesium-organic framework-based stimuli-responsive systems that optimize the bone microenvironment for enhanced bone regeneration. Chemical Engineering Journal*,p1-12.
7. [7] YoungGeun Kim and Carole-Jean Wu. (2020) AutoScale: Energy Efficiency Optimization for Stochastic Edge Inference Using Reinforcement Learning, *53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, p1-15.
8. [8] Sellami, M., Mezni, H., &Hacid, M. S. (2020). *On the use of big data frameworks for big service composition. Journal of Network and Computer Applications,* p1-22.
9. [9] Anis Zarrad. (2015) A Systematic Review on Regression Testing for Web-Based Applications, *Journal of Software*, 10, (8), p1-20.
10. [10] Divya Rohatgi, GyanendraDwivedi and Tulika Pandey. (2019) Automated Regression Testing for Web Services, *International Journal of Engineering and Advanced Technology (IJEAT)*, 9, (2), p1-4.
11. [11] Tarhini, A., Fouchal, H., & Mansour, N. (2006). *Regression Testing Web Services-based Applications. IEEE International Conference on Computer Systems and Applications,*p1-8.
12. [12] Sahoo, S., & Ray, A. (2017). *A framework for optimization of regression testing of web services using slicing. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI).*p1-6.
13. [13] Mukherjee, R., & Sridhar Patnaik, K. (2018). *A Survey on Different Approaches for Software Test Case Prioritization. Journal of King Saud University - Computer and Information Sciences.* P1-21.
14. [14] E. Engström, P. Runeson, and M. Skoglund, "A systematic review on regression test selection techniques," Inf. Softw. Technol., vol. 52, no. 1, pp. 14–30, Jan. 2010.

ISSN: 2233-7857IJFGCN

15. [15] D.Rajya Lakshmi and S. SugunaMallika. (2017) A Review on Web Application Testing and its Current Research Directions, *International Journal of Electrical and Computer Engineering (IJECE)*, 7, (4), p2132-2141.
16. [16] Zhong, H., Zhang, L., &Khurshid, S. (2019). *TestSage: Regression Test Selection for Large-Scale Web Service Testing. 2019 12th IEEE Conference on Software Testing, Validation and Verification (ICST).*p1-11.
17. [17] S. Yoo, M. Harman. (2007) Regression Testing Minimisation, Selection and Prioritisation : A Survey, *Software Testing, Verification And Reliability*, p1-60.
18. [18] Almulla, M., Yahyaoui, H., &Almatori, K. (2012). *Visualization of Real-World Web Services Based on Fuzzy Logic. 2012 IEEE Eighth World Congress on Services.* P1-6.
19. [19] Tehreem Masood, Aamer Nadeem, S. Ali, "An automated approach to regression testing of web services based on WSDL operation changes" In Proceedings of 9 th IEEE International Conference on Emerging Technologies (ICET), pp. 1-5.
20. [20] Mustafa Bozkurt, Mark Harman and Youssef Hassoun, "Testing web services: A survey", Technical report: Centre for Research on Evolution, Search & Testing Kings College London Strand, London WC2R 2LS, UK, 2011. P1-49.

ISSN: 2233-7857IJFGCN