

Comparing method equivalence class partitioning and boundary value analysis with study case add medicine module

I G S Aryandana, A E Permanasari* and T B Adjii

Department of Electrical and Information Engineering, Faculty of Engineering,
Universitas Gajah Mada, Yogyakarta, Indonesia

*adhistya@ugm.ac.id

Abstract. Software Testing is helpful to alleviate the developers' task in improving the system. But in doing software testing, there are problems faced by developers, including information that become a benchmark in the feasibility of software testing. Many factors are used to obtain information or benchmarks. These factors can include line code, program structure, business processes and the time used in conducting trials. A test case is needed to ensure one of the two testing processes. The first is to make sure the data type of the entered data. The method of this process is called Equivalence Class Partitioning (ECP). The second is ensuring the lower bound value, the middle bound value, and the upper limit value of the number of strings as inputs of a system module. The method of this process is called Boundary Value Analysis (BVA). The purpose of this study is to compare the ECP and BVA methods. The expected result of this study is to find out the best method between ECP or BVA. Comparison of calculations using standard matrix testing. This study uses 5 parameters as benchmarks. Based on this study, the ECP method is superior to BVA with 5 parameters used as benchmarks.

1. Introduction

Software testing has an important role to know the quality of a software. Software quality is said to be feasible if it produces the expected output. Activities carried out in the testing aim to find out the software failures. Identifying software failures can improve the performance, accuracy, quality or information from the tested software. Testing can be considered good when it has the ability to find errors that have not been revealed [1]. Software testing is the need for developers to find out the software capabilities on whether it is running well or not [2]. The importance of software testing in testing the performance of the software is very influential to determine the reliability and level of user satisfaction in using the application. This is done to ensure the quality, accuracy, performance or information from software whether is feasible to use or not before reaching the user [3].

Black Box is one of the software testing technique that has five stages, namely Graph-Based Testing, Equivalent Partitioning, Boundary Value Analysis, Comparison Testing, and Orthogonal Arrays [4]. Two techniques are used in this study i.e. ECP and BVA. ECP is a testing technique that is useful for ascertaining the data type to be entered. Whereas BVA is a testing technique that is used to ascertain the lower bound value, the middle bound value, and the upper limit value of the number of strings as inputs of a system module. ECP and BVA are useful for displaying information that will later become a benchmark for the software [5,6].



The problem experienced is that the developer did not know the benchmark that causes the software to be said inappropriate or failed. It causes the developers to continue to make improvements without knowing the benchmarks. The number of methods of software testing that are available does not ensure that each trial process becomes easier. This is because the process of identifying each method is different and the information obtained is different and the time needed to execute the software becomes longer [7]. The methods that generally use in software testing are Equivalence Partitioning and Boundary Value Analysis. But the problem is the ECP and BVA methods have their own weaknesses and strengths. Therefore the purpose of this study is to compare the Equivalence Partitioning and Boundary Value Analysis methods to determine the comparisons and weaknesses and strengths owned by the ECP or BVA method.

2. Related work

Software testing is an activity that aims to evaluate the software. Software testing is used to determine whether the program functionality is compatible with the desired results. Software testing is divided into 3 activities, namely testing, verification, validation. Testing is a process that checks the program to find errors before being given to the user. Verification is a process to find out whether the software meets its specifications. Validation is the process of building the right device, whether it fits the user needs. Testing software can be done with several existing approaches. The software testing approach can use ECP and BVA to process whether the software produces good performance or not [8].

Ling Fang et al conducted a case test selection study with Equivalence Class Partitioning (ECP). This research resulted in a strategy and test suite selection to obtain efficiency and retesting. Efficiency and effectiveness that can lead to a reduction in conditions from the ECP method. The weakness of the Ling fang et al research is that it does not produce a more specific test case using parameters as a benchmark. So there is no reference or value parameter used to prove the level of efficiency and effectiveness. Letras et al conducted a study using Field Programmable Gate Array (FPGA) as a measurement object and using the ECP method as a determinant of its condition [9]. The results obtained from the research of Letras et al are proposing a parallel architecture of hardware to get an easier search strategy.

Bhat and Quadri reviewed the ECP and BVA methods as a comparison. The results obtained from the review of Bhat and Quadri are an analysis of the functions of the ECP and BVA methods and their framework. The drawbacks do not explain the factors or parameters that can be used in the ECP and BVA methods so that they can be a picture of the results of the framework obtained [10].

Zhang et al conducted a study using Boundary Value Analysis as its test generation. The results obtained from Zhang's research are getting the lower bound value, middle boundary value, and upper limit value as the chosen execution conditions. The weakness of Zhang's research is the absence of benchmarks to determine the performance of the BVA method [11]. Vij and Feng conducted a study using Boundary Value Analysis with a Devide and Rule Approach. The results obtained from the research of Vij and Feng are finding new rules that cannot be produced by BVA [12].

This research mentioned above has not yet done the test case calculation. The results of the test case obtained are then classified using valid, invalid and defect values. Where the classification results are used to perform standard testing metrics calculations so that they can find out the system performance of the ECP and BVA trial methods by using parameters from the standard testing metrics.

3. Research and methodology

3.1. Equivalence class partitioning

Equivalence Class Partitioning (ECP) is a software testing method that oriented to create several conditions based on the obtained test case. The Test Case is used to produce a trial of the system module to be tested. The trial are conducted to obtain the results. The results obtained from this condition is a valid or invalid value. The input condition in ECP may be numeric value, range value, or Boolean condition [5].

3.1.1. *Advantages of ECP.* The advantages of the ECP method are as follows.

- It can determine the test case more specifically.
- It can recognize system defects based on the test case being tested.

3.1.2. *Disadvantages of ECP.* The disadvantages of the ECP method are as follows.

- Identification of test cases still uses manual method and requires expertise of a tester.
- The length of time used in testing the test case [5].

3.2. *Boundary value analysis*

Boundary Value Analysis (BVA) is a software testing technique useful for entering input data by using a number of Strings entered. These input data produce boundary values which is useful as a parameter. There are 3 boundary values determined by the BVA method, namely the lower boundary value, the middle boundary value, and the upper boundary value, each of the specified boundary value has its own value. These values can be determined depending on the parameter used [8,13].

3.2.1. *Advantages of BVA.* The advantages of the BVA method are as follows.

- It can facilitate in identifying the test case.
- It can control expenses based on the number of the generated test cases.
- BVA is a strongly suitable method for User Input Trouble.

3.2.2. *Disadvantages of BVA.* The disadvantages of the BVA method are as follows.

- It cannot test all test case input values.
- BVA technique is not suitable with Boolean Variable, because Boolean is a type of data which have values of 1 and 0.
- Dependence with test case variables that already have conditions for input values [6].

3.3. *Dataset*

Pharmacy Information System is a management system useful for managing pharmacy information process through provided modules. Modules provided in the pharmacy information system are add medicine module, sales module, purchase module, and inventory module. Each module has specification for the data to be entered.

The data to be entered can be in the form of string and integer. Each data input that uses string can be characters from 1 character to 30 characters. The integer can be positive integer numbers. Each input data uses positive integers a pharmacy information system module. The string and integers be dataset for the modules.

The modules used in this research are the Add Medicine (Add Medicine) module that contain fields such as name of medicine (*Nama Obat*), storage (*Penyimpanan*), number of stock (*Banyak Stok*), unit (*Unit*), category (*Kategori*), expiration date (*Tanggal Kadaluarsa*), description (*Deskripsi*), buying price (*Harga Beli*), selling price (*Harga Jual*), and name of supplier (*Nama Pemasok*).

In this study, the author conducted experiments using the ECP and BVA methods. The interface display of the Add Medicine (Add Medicine) menu is as follows.

Figure 1 is an interface display of Add Medicine (Add Medicine) module in the pharmacy information system.

Tambah Obat Baru

Nama Obat	<input type="text"/>
Penyimpanan	<input type="text"/>
Banyak Stok	<input type="text"/>
Unit	<input type="text"/>
Kategori	<input type="text"/>
Tanggal Kedaluwarsa	<input type="text"/>
Deskripsi	<input type="text"/>
Harga Beli (Rp)	<input type="text"/>
Harga Jual (Rp)	<input type="text"/>
Nama Pemasok	<input type="text"/>

Figure 1. Add medicine module.

The Add Medicine module is useful for managing medicines, such as insert medicine, update medicine, and delete medicine. In the Add Medicine (Add Medicine) module, that contain fields such as name of medicine name (Medicine Name), storage (Storage), number of stock (Stock), unit (Unit), category (Category), expiration date (Expired Date), description (Description), buying price (Purchase Price), selling price (Selling Price), supplier name (Supplier Name).

3.4. Classification equivalence class partitioning

Table 1 is an Equivalence Class Partitioning classification with 13 predetermined conditions.

Table 1. Classification equivalence class partitioning.

No	Input Condition Specification
1	The name of the drug entered can be alphabetical and numerical
2	The supplier name entered is alphabetical
3	The name of the drug entered uses a symbol
4	Supplier name entered using symbol
5	The number of input characters entered is at least 1 character
6	The number of input characters entered is less than 1 character
7	Input characters for integer drug stock
8	Character input for stock of decimal drugs
9	The input character for the drug stock is number
10	The input character for the drug stock is the alphabet
11	The input character for the drug category is the alphabet
12	Input characters for the drug category are number
13	The input characters for the drug category are symbols

Thirteen (13) conditions in Table 1 will then be tested and produce Valid dan Invalid output as in Table 3. Classification is obtained based on existing system conditions. These conditions identified to look for every possibility that occurs in the pharmacy information system. Each obtained condition will produce output which later will be used to determine whether the condition is running well or not in the system. ECP Classification is more specific in determining conditions. This condition can determine data with numerical, decimal or symbolic characteristic.

3.5. Classification boundary value analysis

Table 2 is a BVA classification with eight (8) conditions obtained from the Add Medicine module (Add Medicine).

Table 2. Classification boundary value analysis.

No	Input Condition Specification	Output
1	Test case input drug name is less than 1 character	Invalid
2	Test case input drug name more than 1 character	Valid
3	Test case for input drug stock is less than 1	Invalid
4	Test case for more than 1 drug stock input	Valid
5	Test case input supplier name less than 1 character	Invalid
6	Test case input of supplier name more than 1 character	Valid
7	Test case input for drug category is less than 1 character	Invalid
8	Test case input for drug categories is more than 1 character	Valid

Eight (8) conditions obtained are then tested by producing valid and invalid output. Classification is obtained based on existing system conditions. The conditions carried out by BVA are actually almost the same as ECP. What distinguishes the condition is the nature of classification of BVA method. BVA Classification is more general and depends on the parameter or the number of characters to be entered into the system. Specified number of characters will produce output whether the system can run properly or not.

4. Result and analysis

4.1. Testing analysis equivalence class partitioning

Table 3 is the result of trials that have been carried out based on predetermined conditions.

Table 3. Testing analysis ECP.

No	Test Data	Output	Classes Covered
1	Medicined	Valid	1,5
2	Suppliername	Valid	2
3	Medicine@##\$	Valid	3
4	Supplier!@#	Valid	4
5	Medicine	Valid	5
6	-	Invalid	6
7	12	Valid	7
8	1.567	Invalid	8
9	100	Valid	9
10	Medicine100	Valid	10
11	Medicinecategory04	Valid	11,12
12	Category!@#\$\$%	Valid	13

Results obtained from Table 3 are trials that have been done by a tester. From 12 test case produced output with a total value for Valid of 10 and Invalid of 2.

4.2. Testing analysis boundary value analysis

Figure 2 is the parameter or boundary value of the output result obtained from BVA classification in Table 2.

-5, -4, -3, -2, -1, 0	1, 2, 3, 4, 5, 6
Invalid	Valid

Figure 2. Testing analysis BVA.

Figure 2 has a valid boundary value from 1 to 5 which means the trials are successful. While the invalid boundary value is from 0 to -5 which means the trials performed are experiencing failure. Output results can be seen in Table 2

4.3. Result

From the existing analysis, a trial can be performed on the number of test cases that have been tested. The analysis is done by utilizing the Add Medicine module in the pharmacy information system. This study uses standard testing metrics with formulas [7].

$$\text{Test Case Pass} = \left(\frac{\text{Test Case Passed}}{\text{Total Test Case}} \right) \times 100\% \quad (1)$$

Based on the results in Table 3, Valid output values are 10 out of 12 test cases, while Invalid output values are 2 out of 12 test cases. The formula used is as follows. ECP Test case pass $(10/12) \times 100\% = 83,33\%$, Based on the results in Table 2, Valid output values are 4 out of 8 test cases, while Invalid output values are 4 out of 8 test cases. The formula used is as follows. BVA Test case pass $(4/8) \times 100\% = 50\%$.

$$\text{Defect Leakage} = \left(\frac{\text{Find Defect Leakage}}{\text{Total Test Case}} \right) \times 100\% \quad (2)$$

Based on the results in Table 3, the value of Defect Leakage ECP obtained during the trial is 1 output. The value of 1 is obtained because the output produced is defective. The formula used is as follows. ECP Defect Leakage $(1/12) \times 100 = 8,3\%$. Based on the results in Table 2, the value of BVA Defect Leakage obtained during the trial is 1 output. The value of 1 is obtained because the output produced is defective. The formula used is as follows. BVA Defect Leakage $(1/8) \times 100\% = 13\%$.

Based on the results in Table 3, the total ECP test cases are 12 test cases. Based on the results in Table 2, the total BVA test cases are 8 test cases.

$$\text{Test Case Failed} = \left(\frac{\text{Find Test Case Failed}}{\text{Total Test Case}} \right) \times 100\% \quad (3)$$

Based on the results in Table 3, ECP Test Case Failed is $(2/10) \times 100 = 16,67\%$. Based on the results in Table 2, BVA Test Case Failed is $(4/8) \times 100 = 50\%$.

$$\text{Cost of Finding Defect} = \left(\frac{\text{Effort Spent on Testing}}{\text{Defect Found on Testing}} \right) \times 100\% \quad (4)$$

Based on the results in Table 3, ECP Cost finding defect is 12/1 with a result of 1. Based on the results in Table 2, BVA Cost finding defect is 8/1 with a result of 8.

In the operation services section synchronized with the inpatient section by combining the two parts so that the animal owner does not need to be asked again whether to be hospitalized again or not. For the down payment has also been determined based on the services taken and inpatients using a software

that functions to make a decision table, namely Prologa version 5.6. The conditions, actions and rules section are filled in as shown in Figure 3:

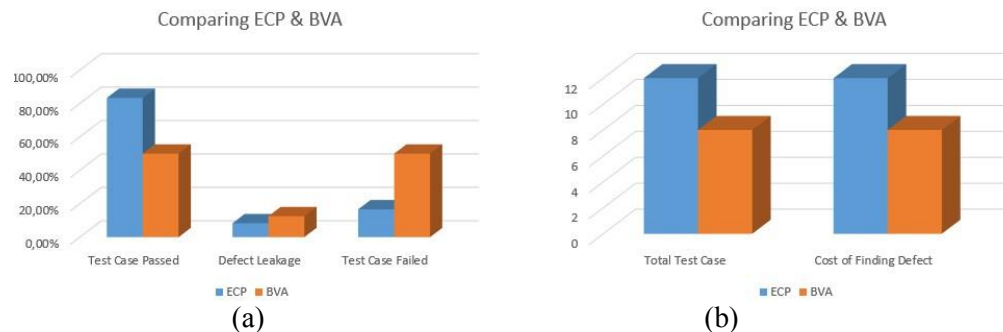


Figure 3. Comparing ECP & BVA performance (a), Comparing ECP & BVA test case (b).

Figure 3 (a) shows that the ECP graph is superior compared to BVA. With parameter values Test Case Passed ECP 83.33%, Test Case Passed BVA 50%, Defect Leakage ECP 8.3%, Defect Leakage BVA 13%, Test Case Failed ECP 16.67%, Test Case Failed BVA 50%.

Figure 3(b) shows that the ECP graph is superior to the BVA graph. With a total ECP test case obtained as many as 12, and a total BA test case as much as 8. While for the cost of finding the Defect ECP gets as many as 12 and a BVA cost of finding as much as 8.

5. Conclusion

This study presents a comparison of ECP and BVA methods in testing the drug added module. The ECP method can do more specific tests than BVA, this is supported by a higher number of ECP test cases.

The conclusion from this study is that ECP is superior to BVA. This is because the ECP value is better on the 5 parameters used compared to BVA consisting of Test Case passed, Defect Leakage, Test Case Failed, Total Test Case and Cost of Finding Defect. This research can be developed by combining the ECP and BVA methods so that they can cover the weaknesses of each method.

Acknowledgments

The authors would like to acknowledge funding from DIKTI through the PDUPT program in 2019, Universitas Gadjah Mada.

References

- [1] Satrio C D, Saputra M C and Rachmadi A 2017 Perbandingan Test Case Generation dengan Pendekatan Genetic Algorithm Mutation Analysis dan Sampling *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer* **2**(1) 246-255
- [2] Angraini A and Pertiwi I D 2017 Analisa Pengelolaan Risiko Penerapan Teknologi Informasi Menggunakan ISO 31000 *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi* **3**(2) 70-76
- [3] Suhartono J 2016 *Software Testing* (Jakarta: Binus University) [Online] Retrieved from: <https://sis.binus.ac.id/2016/12/16/software-testing/> [Accessed: 01-Sep-2018]
- [4] Vijay 2018 *Black Box Testing: An In-depth Tutorial with Examples and Techniques* (softwaretestinghelp) [Online] Retrieved from: <https://www.softwaretestinghelp.com/black-box-testing/> [Accessed: 01-Sep-2018]
- [5] Rana K 2017 *Equivalence Class Partitioning* (artoftesting) [Online] Retrieved from: <https://artoftesting.com/manualTesting/equivalence-class-partitioning.html> [Accessed: 15-Dec-2018]
- [6] Rana K 2017 *Boundary value analysis* (artoftesting) [Online] Retrieved from: <https://artoftesting.com/manualTesting/boundary-value-analysis.html> [Accessed: 15-Dec-2018]

- 2018]
- [7] Akshatha V and Illango V 2018 A Comparative Analysis On Equivalence class partitioning And Boundary value analysis *Int. J. Recent Trends Eng. Res.* **4**(3) 542–554
 - [8] Fang L and Li G 2016 Test Selection with Equivalence Clab Partitioning *Proc. - 2015 2nd Int. Symp. Dependable Comput. Internet Things, DCIT 2015* 40–49
 - [9] Letras M, Hernández-León R and Cumplido R 2016 Hardware architectures for frequent itemset mining based on equivalence classes partitioning *Proc. - 2016 IEEE 30th Int. Parallel Distrib. Process. Symp. IPDPS 2016* 289–294
 - [10] Bhat A and Quadri S M K 2015 Equivalence class partitioning and boundary value analysis-A review *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)* 1557-1562
 - [11] Zhang Z, Wu T and Zhang J 2015 Boundary value analysis in automatic white-box test generation *2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)* 239-249
 - [12] Vij K and Feng W 2008 Boundary value analysis using divide-and-rule approach *Fifth International Conference on Information Technology: New Generations (itng 2008)* 70-75
 - [13] Mora Perdiguero J 2016 *Development of an efficient data coverage strategy for testmanager* (Doctoral dissertation, Amsterdam: ETSI_Informatica)