Code coverage measurement methodology for MMI software of safety-class I&C system

Lee Eun Hyung, Jung Beom Young, Choi Seok Joo
*Corresponding author: free2123@suresofttech.com

1. Introduction

MMI (Man-Machine Interface) software of the safety instrumentation and control system used in nuclear power plants carry out an important functions, such as displaying and transmitting the commend to another system, and change setpoints the safety-related information.

Yet, this has been recognized reliability of the MMI software plays an important role in enhancing nuclear power plants are operating, regulatory standards have been strengthened with it.

Strengthening of regulatory standards has affected even perform software testing soon, and accordingly, the current regulatory require the measurement of code coverage with legal standard. [1]

In this paper, it poses a problem of the conventional method used for measuring the above-mentioned code coverage, presents a new coverage measuring method for solving the exposed problems.

2. Methods and Results

This section describes the test coverage information for the existing measuring method and new measuring method applied to MMI software of safety grade instrumentation and control system, and it presents the results of applying the two test coverage measures.

2.1 Existing test coverage measuring method

Existing test coverage measurement methods in such a way that the tester measure the test coverage using white-box basis testing tool in target system environment, procedure for coverage measuring are such as follow.

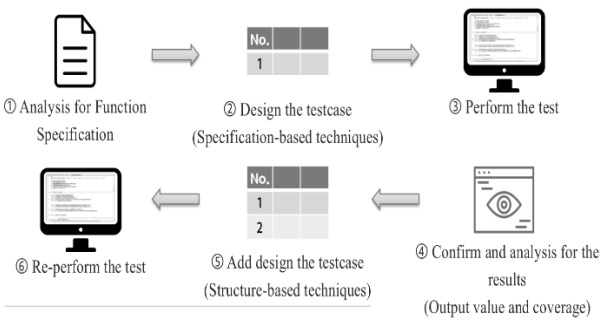

Fig. 1. Existing coverage measurement procedures

Details for the coverage measurement procedure shown in Fig.1 is as follows.

① Analysis the data for test environment and test design using function specification.
② Design the testcases using specification-based techniques.
③ Build up the test environment and perform the test using the testcases.
④ Confirm and analysis the output value and coverage using test tool.
⑤ Additional design the testcases for the partial coverage unmet using structure-based techniques.
⑥ Re-perform the test using additional testcases, and perform the No.④. (After achieving coverage targets until ④ ~ ⑥ repeated)

However, when performing the test for meeting the coverage may result in some code coverage cannot be achieved (E.g. defense code). In this case, after the content at issue, tester should verify the legitimacy and impact properties.

Existing test coverage measuring method described above, there are two main problems. First, for massive amounts of source code or source code having a high complexity, it takes a lot of time and cost of testing. Second, due to the variety frameworks, such as the use MFC, Photon Library, iLog, as a tool of white box testing, it is difficult to build a test environment.

So we were worried a way to solve the problems of the existing coverage measuring method, and as a result, devised the new coverage measuring method.

2.2 New test coverage measuring method

The basic concept of the new coverage measuring method for solving the two problems of the existing test coverage measuring method mentioned in Section 2.1, is generally a common Top-Down manner.

The introduction of Top-Down approach was made through "weapons systems software development and management manual" methodology of the DAPA (Defense Acquisition Program Administration). [2] In that manual, if there are constraints of the test environment are described that can be performed at a higher test. Through this, we first performs a higher GUI (Graphic User Interface) test considering the constraints of the environment, it was introduced as necessary to perform additional sub-test approach. Which after

performing a first test on the higher (Top) by performing sub-test at the lower (Down), it can be called Top-Down method.

Scenario of Coverage Top-Down Method for measurement is as follows.

1. Perform the GUI-based black box testing
2. Measure the test coverage by tool
3. Analysis the items that GUI test coverage is not possible to secure
4. Additional secure the remaining coverage through performing test based on white box testing tool

Then, it will be described in more detail through an example scenario above.

For example, “Button A” in the following Fig.2 is related with Function a1 and a2, “Button B” b1, b2, b3, and “Button C” c1, c2, c3, c4.

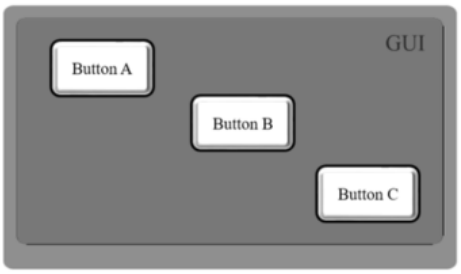

Fig. 2. GUI Example

And, calling the relationship between each function is the same as shown in the Fig.3 below.

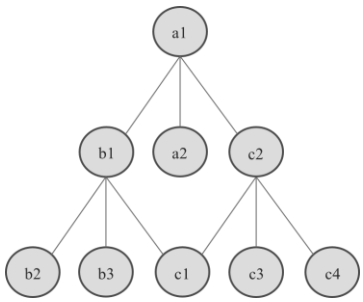

Fig. 3. Call-graph Example

Suppose a GUI-based black box testing utilizing the above example. Then, the test cases that can come out are as following Tab.1.

Test Case	Cover Function	Coverage
A Click	a1, a2	22.2%
B Click	b1, b2, b3	30.0%
A → B Click	a1, a2, b1, b2, b3	55.5%
A → B → C Click	a1, a2, b1, b2, b3, c1, c2, c3, c4	100%

Tab. 1. Example of testcases

As you can see from the table above, and can cover the functions associated with the GUI-based black box testing, a GUI-based coverage measurement tool it is possible to check the function of the coverage level. Coverage measurement method applied here, the coverage may be referred to as the method of the Top Level.

Using the actual coverage tool measurement screen is as shown below Fig.4. The Fig.4 shows the results indicated by utilizing the GUI coverage measurement tool called COVER. The results shown in the Fig.4 is the only one black box testing performed on the GUI. As you can see in the Fig.4, only through GUI test can check for statement / branch code coverage, and you can see the cover portion and the not covered portion of the actual source code. [3]

Fig. 4. Coverage measurement screen (Using COVER)

But, only GUI test is not always possible to achieve 100% as shown above Fig.4. Because they exist functions that can not be tested on the GUI. Therefore, to test these functions, existing white box testing carry out and meet the coverage. The method applied here can be called Down Level of Top-Down approach. Thus, the new method is a combined GUI-based coverage measurements and White Box based coverage measurements.

2.3 Measuring case

To verify the efficiency of the new measuring method was applied to the GUI software of the QNX OS environment.

COVER tools has been used for test of GUI software from QNX OS environment, the test environment is as below Fig.5.

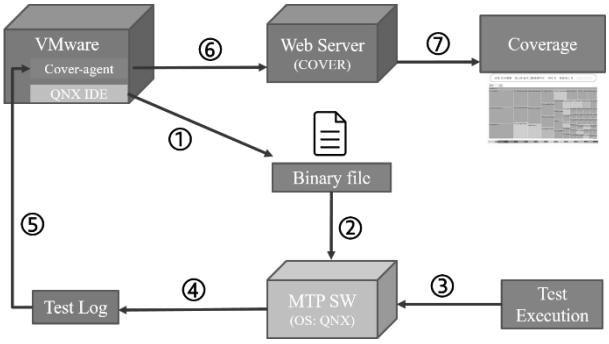

Fig. 5. Coverage measurement screen

First, ① the tester generates a binary file, the probe is inserted through Cover-agent, ② and execute the generated binary file in QNX environment. When the GUI is activated, ③ perform a test on the GUI. ④ If the test is performed when generating a log, at this time, ⑤ copy the generated log files to the Cover-agent. In ⑥ Web Server collects the file automatically, ⑦ displays coverage on the web.

Results of the test with the above method, the cost was reduced by 43% compared to existing methods. In addition, through the results, it can be seen that when increasing test coverage on the GUI, it can be seen that further savings are possible.

The table below is a comparison of the result of executing the new method and the existing method.

	Existing method	New method
The total number of lines tested	9212	9212
1 months testing the number of available lines	7000	68000
Coverage category	Branch : 100% (CT : 100%)	Branch : 100% (COVER : 67.3%, CT : 32.7%)
M/M for coverage 100% attainment	1.3 MM	0.53 MM

Tab. 2. Comparison existing method and new method

The calculation results for the above are as follows.

Actual test date to 9212 Line (using COVER)
➔ 2 day (Coverage achieved 67.3%)

Lines can be tested (1MM basis)
- Using COVER
➔ 9212[1] * 0.673[2] / 2[3] = 62000
- Using CT[4]
➔ 9212[1] * 0.327[5] / 9[6] = 6694

Note)
1) number of line
2) coverage of GUI test using COVER
3) day for GUI test
4) white box testing support tool [4]
5) coverage of white box test using CT
6) day for white box test

As can be seen through the above result, it is possible to obtain a considerable efficiency in terms of coverage achieved when using Top-Down method presented in this paper compared to existing method.

3. Conclusions

In this paper, we checked the problems such as limit and the low efficiency of the existing test coverage measuring method on the MMI software using in nuclear power instrumentation and control systems, and it proposed a new test coverage measuring method as a solution for this. If you apply a new method of Top-Down approach, can mitigate all of the problems of existing test coverage measurement methods and possible coverage achievement of the desired objectives.

Of course, it is still necessary to secure more cases, and the methodology should be systematization based on the cases.

Thus, if later the efficient and reliable are ensured through the application in many cases, as well as nuclear power instrumentation and control, may be used to ensure code coverage of software of the many areas where the GUI is utilized.

REFERENCES

[1] Regulatory guidelines: digital computer software unit testing used in safety systems, KINS/RG-N08.18.
[2] Decree: Weapon System Software Development and Maintenance Manual, 2014-1.
[3] Code coverage measurement tool, reference site: www.suresofttech.co.kr.