Research of Software Reliability Designing and Modeling of Electronic Control Unit of Engine

Jian XIONG and Hong GU*

School of Control Science and Engineering, Dalian University of Technology, Dalian 600015, China

*Corresponding author

Keywords: Electronic control unit, Software reliability, Reliability testing, Reliability modeling.

Abstract. Only the hardware system of EMS cannot guarantee the reliability of ECU because it is constrained by system cost and other factors. So it is necessary to lean upon the reliability of the software system to complement and improve the reliability of engine. In this paper, we described the structure of the software of ECU of engine and the methods and processes of the software reliability. And then, we test the reliability of the software using the white-box testing method. During the verification, to check the path of the verifying process is the same as the real path by designing four pair of different examinations. The results of the exams show that the designing method satisfying the need of verifying the software reliability.

Introduction

With the development cost constraints, ECU (Electronic Control Unit) hardware often has been chosen with some low cost, small size, poor precision micro-control-unit. The hardware system platform constructed in this way is difficult to ensure the reliability of the whole system with high index. Therefore, when considering the reliability of the whole system, we should consider making up for the lack of reliability caused by the hardware system by improving the reliability of the software system. Engine control subsystem is the core of the system of the engine. Its reliability directly affects the reliability of the whole vehicle system. The importance of controlling the reliability of a system is even more so for small engine systems. It is more meaningful to study the reliability of a software system that forms the engine control system than its hardware system. Because, on the one hand with the microelectronics technology, micro-processing technology continues to improve, the reliability of the hardware system can be effectively protected. On the other hand, relative to the reliability of the hardware system, the reliability of the software system is more difficult to be effectively protected. Software errors and defects are often very subtle. Therefore, it is very important to study the reliability of engine control system software. Paper [1-6] introduced many practical software reliability design method in real development environment. Paper [7-10] presented some software reliability modeling technologies. Paper [11-12] introduced some methods of evaluation of the software reliability. Paper [13-15] introduced several software reliability test methods and technologies.

Software Reliability Definition

Reliability refers to the product ability which can successfully complete the provisions function within the prescribed time, under the conditions specified in. Software reliability is defined as the probability that a software product will not cause system failure within the specified time period under the specified conditions. From the definition which can be seen that the reliability of the software is related to two conditions, that is, the specified conditions and the prescribed time period. The reliability of the software is reflected in the runtime of the software system. However, the stages that determine the reliability of software are mainly focused on the software development life cycle of the software. The software life cycle is divided into requirements analysis, architecture design, code Programming, functional logic test, use and maintenance of five stages, of which the first four stages is called the development period, the last stage is in the actual use of the software [2].
Engine Control System Software Reliability

As mentioned above, although the reliability of the software is reflected in the operational phase, but it is to rely on the design of the development period to ensure that. Engine control software also follow this principle, its reliability is to be in the beginning of its design, and even hardware design, selection phase should begin to consider.

(1). Error-avoidance method: that does not allow the error to sneak into the software, does not allow the existence of the fault, the system reduces the probability of actual effect can be received.

(2). Fault tolerance: that allows some unexpected errors or failures occur through the software's own fault-tolerant design allows these errors or faults automatically eliminate the harm to minimize.

A. Software architecture for engine control system

Engine control software system is based on the engine control hardware system architecture, combined with functional needs to design and implementation. Therefore, the engine control software in its development phase of the demand analysis phase of the engine system to complete the task to be analyzed based on the formation of the software requirements document. In the software system architecture design stage, you need to combine engine control hardware architecture design, device selection and software platform for analysis and design information.

The main functions of the vehicle are described as follows: Vehicle Coordinator; Vehicle Motion; PT Powertrain; Electrical Supply System; Thermal System; Global Data.

The functions of the engine are as follows: Coordinator Engine which output of the engine state and shut-off of the engine in case of an error. ETS Engine Torque to determine the torques requisites, coordinates these and determines a resulting requisite on the torque conversion. Furthermore, information about the available torque is supplied. Engine data provides the engine on time, the engine off time and the temperature of the engine. Gas System combines the induction system, the exhaust air system and corresponding air model. Injection System CR provides functions for the formation of the injection characteristics and for generation of the injection pressure. Combustion system has the task of monitoring and influencing the conditions in the combustion chamber. Start system contains the functions starting cut-out, starting torque calculation starter control.

B. Engine control system software reliability design

The reliability of the engine control system software is affected by the conditions of use and the environment in which it is used. In addition, the engine control system belongs to the embedded system, in considering its software reliability, should be embedded system software reliability related rules combined consideration.
According to the software architecture in the previous section, the reliability design of the engine control system software should also be based on the principle of modularity, according to the characteristics of the various subsystems. For example, fuel injection pulse width control and injection current acquisition and processing module requires a higher degree of accuracy data, the engine speed data requires a higher real-time, while the calibration data link module requires a high degree of confidentiality and so on. These are all things that need to be considered for reliability of engine control software. Due to the large size of engine control software and the large number of modules, we only describe the reliability design of modular software based on the reliability design process of the engine fuel injection current data acquisition and processing module.

The engine fuel injection current data acquisition and processing module firstly collects the real-time operating data acquired by them according to the sampling rate of each sensor, and then uses the data fusion algorithm to obtain reliable high-precision real-time status data [4]. The module requires high accuracy of the data and should pay particular attention to this when considering its reliability issues. The reliability of the design process can be carried out according to the following steps (initialization of the module by the system initialization module). Figure 3 shows the reliability of the design flow chart.

**Engine Control System Software Reliability Testing**

Software reliability testing refers to a series of software-oriented testing activities to test whether software meets the specified reliability requirements in a simulation environment simulating a real environment. For the engine control system software, the reliability of the test is more self-evident.

In order to test the software reliability of the engine control system, a semi-physical simulation environment was established as a platform to test the reliability of the engine control software by using the white-box test method combined with the correction condition / decision coverage standard. The reason for choosing a correction condition / decision coverage criterion is it has the advantages of fewer test cases, high target code coverage, and higher sensitivity to operands, especially missing or redundant operands. In addition, because each step of the engine control system is subject to a number of conditions, the branch decision of the program will be a complex Boolean expression which combined with multiple conditions, while covering the standard with the correction conditions/ decisions may cause errors in the program does not have to design a large number of test cases.

![Figure 3. Injection current sample flow.](image-url)
The following section 2.2 to run the data acquisition and processing module as an example to illustrate the reliability of engine control software testing. Figure 4 is obtained after abstracting Figure 3 for a test case for the construction of the flow chart.

Comparing Figure 3 and Figure 4, it can be seen that the English letters A, B, C, D, E, G, and H indicate the judging of program branches, where each decision is a combination of a series of conditions. Numbers indicate executable statements, lowercase English letters indicate the path to the program, and letters T and F indicate logical and logical falsifications. With the above illustration and description, the next step you can extract the decision of the program branch point in Figure 4, and design test cases based on test standards, and finally get the corresponding output. Table 1 shows the design of several test cases, and gives the corresponding output.

From the above test cases and the corresponding results can be seen that the rational design of test cases can ensure that every possible path will be tested. The choice of these paths corresponds to only one test case, that is, the test path changes with a single test case, and the entry and exit of the program module are invoked. Thus, also shows that if the test case design is reasonable, the correction conditions / decision coverage of this method can be fully and effectively used for engine control system software reliability testing. The other module tests are similar. System-wide software testing is the module as a whole (internal no longer test), still use this method for testing.

Table 1. Reliability test case table.

<table>
<thead>
<tr>
<th>Test case</th>
<th>Design logic (ABCDEGH)</th>
<th>Corresponding path</th>
<th>Corresponding result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>107 (01101011)</td>
<td>acehjnpqstvwz</td>
<td>After real-time acquisition, real-time fusion data</td>
</tr>
<tr>
<td>2</td>
<td>89 (01011001)</td>
<td>acdfxz</td>
<td>uses the most recent valid data as output</td>
</tr>
<tr>
<td>3</td>
<td>109 (01101101)</td>
<td>acehjkmoz</td>
<td>Partial sensor real-time data without fusion processed as output</td>
</tr>
<tr>
<td>4</td>
<td>244 (11100100)</td>
<td>acehinoz</td>
<td>Partial sensor real-time data without fusion processed as output</td>
</tr>
</tbody>
</table>

**Engine Control System Software Reliability Modeling**

Reliability models are important tools for evaluating system reliability. There are many models of software systems, but there is not a universally accepted model. Software reliability models are divided into deterministic and probabilistic models. Deterministic model is a statistical model to determine the quality of software. Probability model is the model of software failure and discharge as a probability event. Common probabilistic models include fault sowing model, failure rate model, curve fitting model, reliability growth model, Markov structural model, etc. [6-7]. In this paper, we establish the exponential reliability growth model of engine control system software. Below the
software test results in Section 3 as the test data, and based on this software reliability model. The reliability model of the engine control software is based on the software system architecture shown in Figure 2. The overall model of the software system is composed of each subsystem model with the corresponding weighted values that express the importance of the subsystems model; Using non-homogeneous Poisson process model to describe the modeling environment to select MATLAB software [8-9].

Take the engine data acquisition and processing module as an example to illustrate the construction process of the analysis model. Engine data acquisition and processing module were written in C language, about 1500 lines of code. Perform the software test according to the test method in Section 3, obtaining a total of 20 groups. Table 2 shows the measured data sheet of the module. The reference model of running data acquisition and processing module is selected as exponential software reliability growth model. The mean function can be expressed as:

\[ p(t) = m (1 - e^{-nt}), \quad (m > 0, \ n > 0). \]

The parameters \( m \) and \( n \) can be solved by the maximum likelihood method. The results obtained are as follows: \( m = 27.5392, \ n = 0.1106 \) (detailed solution process can refer to [5]).

The total number of errors at the test time is following:

<table>
<thead>
<tr>
<th>Time</th>
<th>Errors</th>
<th>Total errors</th>
<th>Time</th>
<th>Errors</th>
<th>Total errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>11</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3</td>
<td>13</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>14</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>6</td>
<td>15</td>
<td>1</td>
<td>17</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>7</td>
<td>16</td>
<td>2</td>
<td>19</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>7</td>
<td>17</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>8</td>
<td>18</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>9</td>
<td>19</td>
<td>1</td>
<td>21</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>9</td>
<td>20</td>
<td>0</td>
<td>21</td>
</tr>
</tbody>
</table>

Then the likelihood function of the mean function \( p(t) \) is:

\[
\hat{K}(t) = 27.5392 \times (1 - e^{-0.1106t})
\]

The weighting factor is calculated according to the following formula:

\[
\hat{\lambda}_i = \frac{\delta_i}{\sum_{i=1}^p \delta_i}, \quad (i = 1, 2, ..., p)
\]

Where, \( \delta_i = t_i t, \quad (i = 1, 2, ..., p) \), \( t_i \) represents the total number of system failures caused by the \( i \)-th subsystem failure. \( t \) indicates the total number of system failures. The overall model of the engine control software system is a multi-layered architecture composed of subsystems in series and in parallel. According to the reliability model representation of the series structure and the parallel structure, the overall evaluation model of the overall reliability of the system software can be expressed as:

\[
K = \prod_{i=1}^r \sum_{j=1}^s \hat{\lambda}_i K_{ij} (i = 1, 2, ..., r; \ j = 1, 2, ..., s)
\]

The above expression of the software reliability model of the engine control system takes fully into account the internal structure of the software and the interlink age between the various modules and their respective share in the total software system. Such a model helps to reflect the detailed information that affects the reliability of the software system and facilitates the assessment of reliability [10].
Conclusions

Because of the cost, body structure constraints, the hardware of the vehicle engine control system used in the system reliability cannot guarantee the reliability of the whole machine. In this paper, the reliability of the engine control system software is studied, the reliability of the engine control system software is designed, and reliability of the software is tested by the white box test method. Finally, the exponential reliability growth model of the engine control system software is established. The research in this paper is helpful to the research on the safety and reliability of the engine control system of automobile engine, and also provides some foundation for the research on the reliability evaluation of the software. In addition, the choice of MATLAB as a modeling environment helps to analyze the software test data, which provides powerful data analysis and processing functions to make the calculation and analysis easier.

References
