

УДК 004.4

ДОВГАЛЬ Володимир Володимирович, доцент, кандидат технічних наук

## АНАЛІТИЧНА ОЦІНКА НАДІЙНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОНТРОЛЕРІВ АСУ ТП

*Запорізька державна інженерна академія*

Визначено поняття «надійність програмного забезпечення» та «помилка у програмі». Розглянуто властивості програмних помилок. Проаналізовано групи можливих помилок у програмному забезпеченні та наслідки їх проявлення. Розглянуто існуючі методи оцінки надійності програмного забезпечення. Запропоновано методика оцінки якості налагодження програмного забезпечення контролера, що базується на відомій статистичній моделі Міллса.

Ключові слова: надійність, відмова програми, помилка у програмі, критерії надійності, моделі надійності, випробування, тестування, оцінка ступеня налагодження програми

*Вступ.* У монографії [1] надійність програмного забезпечення визначена як ймовірність його роботи без відмов протягом певного періоду часу, яку розраховано з урахуванням вартості для користувача кожної відмови. Слово «ймовірність» у визначенні означає ймовірність того, що користувач не вводить до системи деякий конкретний набір даних, який призводить до її відмови. Відмова програми відбувається як результат прояви помилки. Програма містить помилку, якщо її поведінка не відповідає специфікації (невідповідність функціональним або не функціональним вимогам). Помилки у програмному забезпеченні виникають як наслідок неправильного перекладу програмістом інформації з одного подання на інше.

Помилки у програмі приховані від спостерігача, тобто можна спостерігати не самі помилки, а результати їх прояви – відмови. Надійність програми пов'язана із частотою прояви помилок, а не з їх кількістю. Різні помилки мають різну частоту прояви. Відмова може бути наслідком не однієї, а відразу декількох помилок, які спроможні компенсувати одна одну. Тому після виправлення помилки інтенсивність відмов програми може збільшитися.

Помилки у програмах поділяють на дві групи [2]:

– функціональні помилки, тобто порушення програмної специфікації, що призводять до погіршення функціональності програмного забезпечення (придатності, точності).

– не функціональні помилки, тобто порушення правил використання мови програмування, неправильне використання бібліотечних функцій тощо, що призводить до зниження надійності та погіршення функціональності (захищеності) програмного забезпечення.

Помилки у програмах виявляють та виправляють на етапах налагодження і випробування програмного забезпечення. Налагодження програм здійснюють розробники програмного забезпечення, а випробування виконують незалежні експерти. Деякі своєчасно не виявлені функціональні помилки можуть проявитися через кілька років після передавання АСУ ТП до промислової експлуатації. Тому дуже важливо якісно виконати випробування програмного забезпечення АСУ ТП та оцінити його надійність.

Надійність програми за часом не змінюється. Для програмного забезпечення, на відміну від апаратної системи, відсутньою є фаза припрацювання та зносу. Корекція програми аналогічна внесенню змінювань до конструкції апаратного пристрою, тобто виходить нове програмне забезпечення з іншими показниками надійності.

Сама проблема надійності програмного забезпечення має два аспекти [3]: забезпечення надійності та оцінка (вимірювання) надійності.

Більша частина з наявної літератури [4-6] присвячена першому аспекту. Питання оцінки надійності комп'ютерних програм опрацьовано недостатньо. Надійність програми, яка працює в АСУ ТП, є не менш важливою ніж такі традиційні її характеристики, як час виконання або необхідний обсяг оперативної пам'яті. Проте загальноприйнята кількісна міра надійності програм дотепер не існує.

Відсутність загальновизнаних критеріїв надійності не дозволяє відповісти на питання, наскільки надійніше стає програмне забезпечення під час дотримання запропонованих процедур і технологій розробки та в якій мірі є виправданими витрати. Таким чином, пріоритет задачі оцінки надійності повинен бути вище пріоритету задачі забезпечення надійності [7].

Моделі надійності програмних засобів піділяють на аналітичні та емпіричні [8]. Аналітичні моделі дають можливість розрахувати

кількісні показники надійності, ґрунтуючись на даних про поведінку програми під час тестування. Емпіричні моделі базуються на аналізі структурних особливостей програм.

В роботі [2] розглянуті різні методи оцінки надійності програмного забезпечення: статичні, архітектурні, емпіричні та динамічні. Статичні методи засновано на аналізі особливостей технологічного процесу проектування програми (розробка специфікацій, розробка архітектури програми та написання початкового тексту програми). Архітектурні методи засновано на аналізі архітектури системи та можуть використовувати як динамічні, так і статичні підходи. Емпіричні методи використовують інформацію про процес проектування.

На окрему увагу заслуговують динамічні методи, які базуються на використанні результатів виконання програми. Їх перевага полягає в тому, що вони дозволяють отримувати абсолютні показники надійності програми. Недоліками є: висока трудомісткість підготовки початкової інформації для тестування програми, використання припущень, що спрощують, про взаємні впливи програмних помилок, а також сильна залежність точності прогнозів від якості й обсягу початкової інформації.

Так, модель Бернуллі дозволяє оцінити надійність програми за результатами тестування та враховує те, що кожний запуск програми має два результати: правильний і неправильний. Після запусків відома кількість правильних і неправильних результатів, за якими оцінюють надійність програми.

Модель Міллса [1,9] базується на внесенні штучних помилок до програми. Тестування виконують доти, доки не будуть знайдені всі внесені помилки. Під час тестування також можуть бути виявлені помилки, які були присутніми у програмі до внесення штучних помилок. Така модель дозволяє передбачити кількість помилок, присутніх у програмі до початку тестування та визначити достовірність прогнозу.

*Постановка задачі.* Розробити інженерну методику експериментальної оцінки ступеня налагодженості програмного забезпечення контролерів АСУ ТП, що заснована на статистичній моделі Міллса, яку можна використовувати для оцінки надійності програм.

*Головний матеріал дослідження.* Об'єктом випробувань, що виконують з метою оцінки надійності, є програмне забезпечення контролера, який функціонує в складі АСУ ТП. Методика випробувань полягає в тому, що до випробуваної програми незалежними експертами, без

участі розробників програми, вносяться штучні семантичні помилки, які не можуть бути виявлені компілятором інструментальної системи програмування. Для випробувань використовують тестові приклади, що розроблено незалежними експертами, у тому числі представниками приймальної комісії. Випробування виконують з використанням симулятора контролера, наявного в складі сучасних «Soft Logic» інструментальних систем програмування контролерів. Під час випробувань фіксують кількість виявлених штучно внесених помилок і кількість виявлених помилок, які були присутні у програмі до внесення штучних помилок (власні помилки програміста).

Випробування програмного забезпечення здійснюють у такому порядку:

1. До випробуваної програми штучно вносять  $S$  помилок, які визначають, виходячи з вимог замовника до ймовірності відсутності помилок у програмі (довірчої ймовірності), за формулою (1):

$$S = \frac{P_0}{1 - P_0}, \quad (1)$$

де  $S$  – кількість помилок, які вносять штучно;  $P_0$  – довірна ймовірність.

2. Програму зі штучно внесеними  $S$  помилками піддають тестуванню. Помилки, що виявлено у процесі тестування, поділяють на внесені  $s$  та власні  $n$  і виправляють їх.

3. Програма вважається такою, що витримала випробування, якщо під час їх проведення не виявлено власні помилки ( $n = 0$ ) за умови, що виявлено всі штучно внесені помилки ( $s = S$ ).

4. Якщо виявлено не всі внесені помилки ( $s < S$ ), то роблять висновок, що тести, які використовують під час випробувань не забезпечують необхідну достовірність їх результатів. Здійснюють доопрацювання тестів і виконують повторні випробування.

5. Якщо виявлено деяку кількість власних помилок ( $n > 0$ ), то за формулою (2) оцінюють початкову кількість власних помилок у програмі, яку випробовують. При цьому виходять з того, що ймовірності виявлення власних і внесених помилок є рівними.

$$N = \frac{n \cdot S}{s}, \quad (2)$$

де  $N$  – кількість власних помилок у програмі, яку випробовують;  $S$  – кількість штучно внесених помилок;  $s$  – кількість виявлених штучно внесених помилок;  $n$  – кількість виявлених власних помилок.

6. Висувають гіпотезу, що у програмі залишилося не більше ніж  $k = N - n$  власних помилок. Обчислюють нову кількість штучно внесених помилок за формулою (3) і повторюють дії, починаючи з п. 2.

$$S' = \frac{P_o \cdot (1+k)}{1-P_o}, \quad (3)$$

де  $S'$  – кількість штучно внесених помилок для повторного тестування програми;  $k$  – кількість власних помилок, які залишилися у програмі після попереднього тестування.

7. У разі, якщо за повторним тестуванням програми виявлено  $n' \leq k$  власних помилок і всі штучно внесені помилки ( $s' = S'$ ), довірчу ймовірність того, що гіпотеза, висунута у п. 6, є вірною, визначають за формулою (4):

$$P'_o = \begin{cases} 1, & \text{якщо } n' > k \\ \frac{S'}{S' + k + 1}, & \text{якщо } n' \leq k \end{cases}, \quad (4)$$

де  $P'_o$  – довірна ймовірність;  $n'$  – кількість власних помилок, виявлених під час повторного тестування програми;  $k$  – кількість власних помилок, що залишилися у програмі перед повторним тестуванням;  $S'$  – кількість штучно внесених до програми помилок перед повторним тестуванням.

8. За результатами повторного тестування роблять висновок про ступінь налагодження програми та необхідність її доопрацювання у разі, якщо  $n' > k$ , залежно від ступеня впливу помилок, які залишилися у програмі, на результати її виконання.

*Приклад використання розробленої методики.* Потрібно з довірчою ймовірністю  $P_o = 0,96$  оцінити кількість помилок у програмі контролера до початку тестування та ступінь налагодженості програми.

За формулою (1) визначають кількість штучних помилок, які потрібно внести до програми:  $S = 0,96 / (1-0,96) = 24$ . Виконують тестування програми.

Під час тестування виявлено дев'ятнадцять штучно внесених помилок і чотири помилки, що допущено програмістом (власні помилки).

За формулою (2) оцінюють початкову кількість власних помилок у програмі, яку випробовують:  $N = 4 \cdot (24 / 19) = 5$ .

Висунуто гіпотезу, що у програмі залишилося не більше ніж  $k = 5 - 4 = 1$  власних помилок. Виправляють виявлені чотири власні помилки програміста.

Обчислюють за формулою (3) нову кількість штучних помилок, які потрібно внести:  $S' = 0,96 \cdot (1 + 1) / (1-0,96) = 48$ . Допрацьовують тести та виконують повторне тестування програми.

В результаті повторного тестування виявлено сорок вісім штучно внесених помилок ( $S' = 48$ ) і одну помилку, що внесено програмістом ( $n' = 1$ ). Виправляють помилку програміста.

За формулою (4) оцінюють ступінь налагодженості програми контролера (для випадку, коли  $n' \leq k$ :  $P'_o = 48 / (48 + 1 + 1) = 0,96$ , що становить 96 %).

*Висновки.* Для оцінки надійності програмного забезпечення контролерів АСУ ТП запропоновано інженерну методику визначення ступеня налагодження програми, яка заснована на відомій статистичній моделі Міллса. Методика дозволяє апріорно визначити кількість штучних помилок, які потрібно внести до програми перед випробуванням, що залежить від довірчої ймовірності до результатів випробувань програми, та розрахувати ступінь налагодженості програми контролера після її випробувань.

### Бібліографічний список

1. **Майерс, Г.** Надежность программного обеспечения: перевод с англ. Ю. Ю. Галимова. [Текст] / Г. Майерс. – М. : Мир, 1980. – 360 с.
2. **Моисеев, М.** Формальные методы обеспечения качества ПО [Электронный ресурс]. Методы оценки надежности ПО – Режим доступа: <http://kspt.icc.spbstu.ru/media/files/2010/course/softwarequality/lec2.pdf> – Дата доступа: вересень 2017. – Назва з екрана.
3. **Открытые системы. СУБД** [Электронный ресурс] С. Г. Романюк. Оценка надежности программного обеспечения – Режим доступа: <https://www.osp.ru/os/1994/04/178540/> – Дата доступа: вересень 2017. – Назва з екрана.
4. **Гласс, Р.** Руководство по надежному программированию: перевод с англ. Ю. П. Кондранина, В. М. Рабиновича [Текст] / Р. Глас. – М. : Финансы и статистика, 1982 – 256 с.
5. **Липаев, В. В.** Тестирование программ. [Текст] / В. В. Липаев. – М. : Радио и связь, 1986 – 296 с.
6. **Штрик, А. А.** Структурное проектирование надежных программ встроенных ЭВМ. [Текст] / А. А. Штрик, Л. Г. Осовецкий, И. Г. Мессих. – Л. : Машиностроение, Ленингр. Отделение, 1989 – 296 с.
7. **Благодатских, В. А.** Стандартизация разработки программных средств: Учеб. пособие под ред. О. С. Разумова. [Текст] / В. А. Благодатских, В. А. Волнин, К. Ф. Посакалов. – М. : Финансы и статистика, 2005. – 288 с. – ISBN 5-279-02657-3.

8. **Василенко, Н. В.**, Макаров В. А. Модели оценки надежности программного обеспечения [Электронный ресурс] – Режим доступа: <https://cyberleninka.ru/article/n/modeli-otsenki-nadezhnosti-programmnogo-obespecheniya> – Дата доступа: вересень 2017. – Назва з екрана.
9. **StudFiles**. [Электронный ресурс] Статистическая модель Миллса. – Режим доступа: <https://studfiles.net/preview/6196491/page:6> – Дата доступа: вересень 2017. – Назва з екрана.

**ДОВГАЛЬ Владимир Владимирович**, кандидат технических наук, доцент кафедры автоматизированного управления технологическими процессами, Запорожская государственная инженерная академия (Запорожье, Украина). E-mail: [ur4qqj@ukr.net](mailto:ur4qqj@ukr.net)

### АНАЛИТИЧЕСКАЯ ОЦЕНКА НАДЕЖНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОНТРОЛЕРОВ АСУ ТП

Определены понятия «надежность программного обеспечения» и «ошибка в программе». Рассмотрены свойства программных ошибок. Проанализированы группы возможных ошибок в программном обеспечении и последствия их проявления. Рассмотрены существующие методы оценки надежности программного обеспечения. Предложена методика оценки качества отладки программного обеспечения контроллера АСУ ТП, основанная на известной статистической модели Миллса.

Ключевые слова: надежность, отказ программы, ошибка в программе, критерии надежности, модели надежности, испытания, тестирование, оценка степени отладки программы

**DOVGAL' Volodymyr**, Candidate of Technical Sciences, Associate Professor of Department of Automated Control by Flow Processes, Zaporizhska State Engineering Academy (Zaporizhzhya, Ukraine). E-mail: [ur4qqj@ukr.net](mailto:ur4qqj@ukr.net)

### ANALYTICAL ASSESSMENT FOR SOFTWARE RELIABILITY OF INDUSTRIAL CONTROL CONTROLLERS

The concepts of «software reliability» and «bug in the program» are defined. The properties of program errors are considered. Two groups of possible errors in software and the consequences of their manifestation are analyzed. Existing methods for evaluating the reliability of software are considered: dynamic, static, architectural and empirical. A technique for estimating the debugging quality of the control system software is proposed, based on the well-known Mills statistical model.

Key words: reliability, program failure, program error, reliability criteria, reliability models, tests, testing, evaluation of the debugging degree of the program.

Стаття надійшла до редакції 06.10.2017 р.  
Рецензент, проф. А.М. Ніколаєнко

Текст даної статті знаходиться на сайті ЗДІА в розділі Наука  
<http://www.zgia.zp.ua>