

## ПРОГНОЗИРОВАНИЕ ОЖИДАЕМОЙ НАДЁЖНОСТИ ПРИКЛАДНЫХ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ УЧЕБНОГО ПРОЦЕССА

Боровиков С.М., Цырельчук Н.И., Закривашевич М.Н., Цырельчук А.И.

Белорусский государственный университет информатики и радиоэлектроники», г. Минск, Беларусь, bsm@bsuir.by

Abstract. The authors propose a technique for predicting the expected reliability of applied software developed by programmers who received programming training in the framework of university academic disciplines.

Согласно ГОСТ 28806-90 [1] под надёжностью (англоязычный вариант термина – reliability) программного средства (ПС) понимают совокупность свойств, характеризующую способность ПС сохранять заданный уровень пригодности в заданных условиях в течение заданного интервала времени.

Наиболее приемлемыми показателями, характеризующими безотказность (завершённость, стабильность) программных средств, представляются показатели сходные с показателями безотказности технических систем, такие как интенсивность проявления ошибок (интенсивность отказов)  $\lambda_{\text{ПС}}$  или средняя наработка на проявление ошибки  $T_0$ . Эти показатели связаны между собой и, зная один из них, можно определить другой [2].

Надёжность ПС определяется качеством отладки программы, глубиной её тестирования. Целью тестирования является не тотальное обнаружение всех ошибок (это принципиально невозможно), а выявление наибольшего количества наиболее критичных ошибок.

При отладке ПС происходит локализация и устранение синтаксических ошибок и явных ошибок кодирования. В процессе же тестирования проверяется работоспособность программы, не содержащей явных ошибок.

Известные методы оценки надёжности прикладных ПС исходят из того, что имеются определённые данные о тестирования программных средств. Однако во многих случаях разработчиков программного обеспечения для учебных целей интересует ожидаемый уровень надёжности прикладных ПС ещё до написания их программного кода. Возникает вопрос, как оценить ожидаемый уровень надёжности ПС на этом этапе.

На начальном этапе проектирования прикладных учебных ПС в условиях ограниченности информации о ПС задачу по оценке их надёжности можно решить совместным использованием расчётного и экспертного методов [3].

Известно [4], что в ПС, прошедших тестирование, по различным оценкам содержится от 5 до 100 ошибок на 1000 строк программного кода.

Согласно [5] даже в ПС, которые прошли строгий контроль качества (Quality Assurance – QA) все равно содержатся ошибки, приблизительно 5 ошибок на 1000 строк кода. В ПС, которое прошло тестирование только на предмет работоспособности функциональных возможностей, что справедливо для учебного программного обеспечения, присутствует намного больше ошибок: около 50 ошибок на 1000 строк кода (рисунок 1).

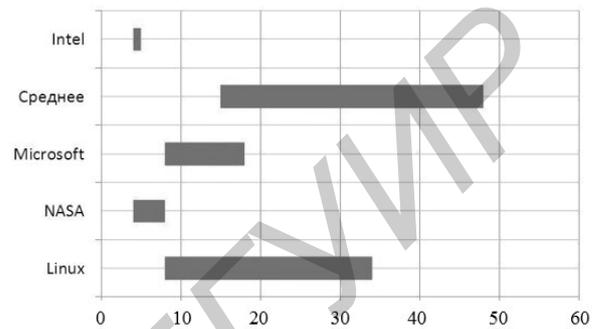


Рисунок 1 – Среднее число ошибок на 1000 строк кода для ПС, прошедших тестирование

В разработанной методике в качестве основы оценки ожидаемой надёжности прикладных ПС выбрана модель Шумана [7, 8], применяемая в мировой практике для оценки надёжности программ. Используются также взятые из отечественной и зарубежной печати обобщённые статистические данные об ожидаемом числе ошибок в разработанных прикладных ПС в зависимости от их объёма, динамики уменьшения числа ошибок от времени отладки (тестирования) и профессионального опыта программистов. Ниже описан предлагаемый способ оценки ожидаемой надёжности ПС.

1. Определяется прогнозное значение начального числа ошибок  $N_{\text{нач}}$ , содержащихся в ПС после его написания и устранения ошибок трансляции (компиляции, работы интерпретатора):

$$N_{\text{нач}} = \frac{131,3 \cdot S(1 - K_{\text{тр}})}{1000}, \quad (1)$$

где  $S$  – размер ПС в строках кода;  $K_{\text{тр}}$  – коэффициент, показывающий долю ошибок в ПС, устраняемых при его трансляции, по умолчанию принимают  $K_{\text{тр}} = 0,5$  [4].

2. Рассчитывается ожидаемое число ошибок, оставшихся в ПС после проведения тестирования  $N_{\text{п.тест}}$ :

$$N_{\text{п.тест}} = (1 - K_{\text{тест}})N_{\text{нач}}, \quad (2)$$

где  $K_{\text{тест}}$  – коэффициент, показывающий долю ошибок, выявляемых при тестировании; для профессиональных программистов согласно [6]  $K_{\text{тест}} \geq 0,5$ . Для программистов, получивших подготовку по программированию в рамках университетских учебных дисциплин, можно принять  $K_{\text{тест}} \approx 0,25 \dots 0,35$ .

3. Вся продолжительность процедуры тестирования ПС рассматривается как один этап тестирования ( $i = 1$ ). Для определения интенсивности проявления



ошибок используется моделью Шумана, согласно которой [6, 7]

$$\lambda_i = (N_{\text{нач}} - n_{i-1})C, \quad 1/\text{ч}, \quad (3)$$

где  $\lambda_i$  – интенсивность проявления ошибок (интенсивность отказов) ПС на  $i$ -м этапе тестирования ПС;  $n_{i-1}$  – число ошибок, исправленных к началу  $i$ -го этапа тестирования; в нашем случае  $i = 1$ , следовательно,  $n_{i-1} = n_0 = 0$ ;  $C$  – коэффициент пропорциональности (коэффициент связности), определяемый в данном случае как

$$C = \frac{n_{\text{тест}}}{N_{\text{нач}} - n_0} = \frac{N_{\text{нач}} - N_{\text{п.тест}}}{N_{\text{нач}} \cdot t_{\text{тест}}}, \quad 1/\text{ч}, \quad (4)$$

где  $t_{\text{тест}}$  – планируемое время тестирования ПС, представляющее собой суммарное время прогона ПС (выполнения на ЭВМ) на этапе тестирования;  $n_{\text{тест}}$  – прогнозируемое число обнаруживаемых и исправляемых ошибок при тестировании ( $n_{\text{тест}} = N_{\text{нач}} - N_{\text{п.тест}}$ ).

4. Уточняется время тестирования  $t_{\text{тест}}$ :

$$t_{\text{тест}} = n_{\text{дн}} t_{\text{дн}} K_{\text{ПС}}, \quad (5)$$

где  $n_{\text{дн}}$  – число рабочих дней, отводимое для тестирования ПС;  $t_{\text{дн}}$  – среднее время в день (в часах), затрачиваемое на выполнение тестирования;  $K_{\text{ПС}}$  – коэффициент, показывающий, какая часть времени в день в среднем используется программистом для прогона ПС (исполнения ПС на компьютере); по умолчанию можно принять  $K_{\text{ПС}} = 0,15$  (по данным объединённого института проблем информатики НАН Беларуси).

5. Определяется прогнозируемое значение интенсивности проявления ошибок (интенсивности отказов) ПС после процедуры тестирования и исправления ошибок, т. е. значение  $\lambda$ , соответствующее этапу эксплуатации:

$$\lambda_{\text{эсп}} = (N_{\text{нач}} - n_{\text{тест}})C = N_{\text{п.тест}} C, \quad (6)$$

где  $\lambda_{\text{эсп}}$  – интенсивность проявления ошибок (интенсивность отказов), соответствующая этапу эксплуатации ПС;  $N_{\text{п.тест}}$  – число ошибок, оставшихся в ПС после процедуры тестирования.

6. С учётом выражения (4) для величины  $C$  формула для расчёта прогнозируемого значения  $\lambda_{\text{эсп}}$  может быть представлена в виде

$$\lambda_{\text{эсп}} = \frac{N_{\text{п.тест}} (N_{\text{нач}} - N_{\text{п.тест}})}{N_{\text{нач}} \cdot t_{\text{тест}}}, \quad 1/\text{ч}. \quad (7)$$

7. Прогнозируемое значение средней наработки на проявление ошибки  $T_{0,\text{ПС}}$  (средней наработки на отказ) для этапа эксплуатации ПС находится в предположении экспоненциального распределения времени работы ПС до проявления ошибки (использования ПС по функциональному назначению до проявления ошибки), ч:

$$T_{0,\text{ПС}} = \frac{1}{\lambda_{\text{эсп}}} = \frac{1}{N_{\text{п.тест}} \cdot C} = \frac{N_{\text{нач}} \cdot t_{\text{тест}}}{N_{\text{п.тест}} (N_{\text{нач}} - N_{\text{п.тест}})}. \quad (8)$$

*Пример.* Разрабатываемое учебное ПС предположительно будет содержать 3 000 строк кода.

Время, отводимое на тестирование ПС одним программистом, определено в размере десяти недель (шестидневные рабочие недели). Среднее время работы программиста в день составляет 6 ч. Требуется определить среднюю наработку на проявление ошибки  $T_{0,\text{ПС}}$  для начального этапа эксплуатации ПС.

*Решение.*

1. По формуле (1) определяем прогнозируемое значение числа ошибок  $N_{\text{нач}}$ , содержащихся в ПС после написания кода и устранения ошибок трансляции:  $N_{\text{нач}} \approx 197$  ошибок.

2. Ожидаемое число ошибок, оставшихся в ПС после проведения тестирования  $N_{\text{п.тест}}$ , находим по выражению (2). Принимаем  $K_{\text{тест}} = 0,3$ . Получим

$$N_{\text{п.тест}} \approx 138 \text{ ошибок.}$$

3. По выражению (5) определяем прогнозируемое время тестирования ПС на ЭВМ:

$$t_{\text{тест}} = (6 \cdot 10) \cdot 6 \cdot 0,15 = 54 \text{ ч.}$$

4. По формуле (4) получаем значение коэффициента  $C$  с учётом того, что  $n_{\text{тест}} = N_{\text{нач}} - N_{\text{п.тест}} = 59$  ошибок,  $n_0 = 0$ :

$$C = 59 / (197 \cdot 54) \approx 0,00555 \text{ 1/ч.}$$

5. По выражению (8) определяем среднюю наработку на проявление ошибки  $T_{0,\text{ПС}}$  для начального этапа эксплуатации ПС:

$$T_{0,\text{ПС}} \approx 1,3 \text{ ч.}$$

### Литература

- ГОСТ 28806-90. Качество программных средств. Термины и определения. – М. : Изд-во стандартов, 1990.
- ГОСТ 27.002-89. Надёжность в технике. Основные понятия, термины и определения. – М. : Изд-во стандартов, 1990.
- ГОСТ 28195-99. Оценка качества программных средств. Общие положения. – Изд-во стандартов, 2000.
- Методы обеспечения аппаратно-программной надёжности вычислительных систем / Д. т. н., проф. Чуканов В.О., к.т.н., доц. Гуров В.В. [Электронный ресурс]. – 2017. – Режим доступа: [http://www.mcst.ru/files/5357ec/dd0cd8/50af39/000000/seminar\\_metody\\_obespecheniya\\_apparatno-programmnoy\\_nadezhnosti\\_vychislitelnyh\\_sistem.pdf](http://www.mcst.ru/files/5357ec/dd0cd8/50af39/000000/seminar_metody_obespecheniya_apparatno-programmnoy_nadezhnosti_vychislitelnyh_sistem.pdf).
- Программное обеспечение – источник всех проблем. [Электронный ресурс]. – 2017. – Режим доступа: <http://www.williamspublishing.com/PDF/5-8459-0785-3/part1.pdf>.
- Чуканов, В. О. Надёжность программного обеспечения и аппаратных средств систем передачи данных атомных электростанций : учеб. пособие / В. О. Чуканов. – М. : МИФИ, 2008. – 168 с.
- Shooman, M. L. Software engineering: Reliability, Development and Management / M. L. Shooman. – McGraw-Hill, International. Book Co, 1983.