

МОДЕЛЬ ДЛЯ ОЦЕНКИ НАДЁЖНОСТИ ПРИКЛАДНЫХ КОМПЬЮТЕРНЫХ ПРОГРАММ НА РАННИХ ЭТАПАХ ИХ ПРОЕКТИРОВАНИЯ

Дик С.С., Лэ В. Т., Клинов К.И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Боровиков С. М. – канд. техн. наук, доцент

Аннотация: в докладе предлагается модель прогнозирования ожидаемой надёжности прикладных компьютерных программ, планируемых к разработке. Приводятся исходные предпосылки и положения, используемые для получения модели.

Во многих случаях разработчики компьютерных информационных систем хотели бы знать ожидаемый уровень надёжности программных средств на раннем этапе их разработки, до написания кода на языке программирования. Для решения такой задачи надо располагать моделью надёжности.

В соответствии с ГОСТ 27.205-1997 [1] о проектной надёжности разрабатываемой компьютерной программы (программного средства) будем судить по интенсивности проявления скрытых ошибок, оставшихся в компьютерной программе после её разработки и тестирования. Эту интенсивность будем называть интенсивностью отказов компьютерной программы (обозначим как λ).

Предпосылки и положения, принятые во внимание при разработке модели надёжности компьютерных программ:

1. В качестве исходных данных рассматриваются:

- предполагаемый объём программы в строках кода;
- назначение прикладной компьютерной программы (сфера, область деятельности людей);
- быстродействие процессора используемого компьютера.

2. Значение интенсивности отказов λ прямо пропорционально числу оставшихся в программном средстве ошибок $N_{ош}$ [2, 3]:

$$\lambda \approx C_{пр} \cdot N_{ош},$$

где $C_{пр}$ – коэффициент пропорциональности, определяемый экспериментально.

20. Для прикладных программ различного функционального назначения известны коэффициенты трансформации $K_{тр}$ плотности ошибок F в интенсивность отказов λ [4] (таблица 1). Плотность ошибок F это отношение вида

$$F = \frac{\text{Количество возможных ошибок в компьютерной программе}}{\text{Число строк кода компьютерной программы}}$$

Таблица 1 – Значения коэффициентов для программ разного функционального назначения

Область применения компьютерной программы	Значение $K_{тр}$, строка кода/ошибок в час	Коэффициент C , 1/ошибка	Коэффициент K_{Σ}
Авиация	6,28	$1,74 \cdot 10^{-9}$	5,23
Мониторинг и обеспечение безопасности	1,2	$3,33 \cdot 10^{-10}$	1
Телекоммуникации, мобильные устройства	13,8	$3,83 \cdot 10^{-9}$	11,5
Управление производственными процессами	3,8	$1,06 \cdot 10^{-9}$	3,18
Автоматизированные системы управления	23	$6,39 \cdot 10^{-9}$	19,2
Разработка программ, моделирование, обучение	16,9	$4,69 \cdot 10^{-9}$	14,1
Среднее	10,6	$2,94 \cdot 10^{-9}$	8,83

Данные таблицы 1 соответствуют случаю выполнения программ на компьютере с 32-разрядным процессором, быстродействие: 10 миллионов операций в секунду (англоязычный вариант: 10 *million instructions per second*, кратко – 10 MIPS).

4. На интенсивность возникновения отказа λ влияют три характеристики эксплуатационной среды выполнения компьютерной программы:

- производительность компьютера (быстродействие процессора, пропускная способность);
- изменчивость входных данных и состояний управления компьютером;
- рабочая нагрузка, которую эксплуатационная среда будет оказывать на компьютерную программу при её использовании по назначению.

5. Используя значения коэффициента $K_{тр}$, определены средние значения коэффициента пропорциональности C для компьютерных программ различных областей применения (см. таблицу 1).

Представим коэффициент C в виде произведения

$$C = C_{Б} \cdot K_{вх} \cdot K_{н} = C_{Б} \cdot K_{\Sigma}, \quad (1)$$

а величины, входящие в выражение (1), будем называть как: C_B – базовое значение коэффициента проявления ошибок программы; $K_{вх}$ – коэффициент увеличения интенсивности отказов, обусловленный изменчивостью входных данных; $K_{н}$ – коэффициент увеличения интенсивности отказов, обусловленный рабочей нагрузкой на программу; K_{Σ} – суммарный коэффициент увеличения интенсивности отказов, обусловленный совместным действием изменчивости входных данных и рабочей нагрузки на компьютерную программу.

6. В качестве базового значения коэффициента проявления ошибок C_B примем значение коэффициента C для компьютерных программ, используемых для мониторинга и обеспечения безопасности. Пользуясь таблицей 1 (3-я строка), можно записать: $C_B = 3,33 \cdot 10^{-10}$ 1/ошибка. Для программ этого назначения суммарный коэффициент увеличения интенсивности отказов принят $K_{\Sigma} = 1,0$. Этот коэффициент собирательно учитывает увеличение интенсивности отказов компьютерной программы за счёт совместного действия изменчивости входных данных и рабочей нагрузки, которую оказывает эксплуатационная среда на компьютерную программу (см. таблицу 1).

7. Предполагаем, что процедура тестирования выполняется общепринятыми методами, и при этом её суммарная продолжительность составляет не менее 40 процентов общего времени, отводимого на разработку компьютерной программы. Согласно работе [4], экспериментальные данные показали, что в таких случаях интенсивность отказов компьютерных программ за счёт выполнения тестирования уменьшается примерно в семь раз относительно начальной интенсивности отказов λ_0 .

Результаты. Интенсивность отказов компьютерной программы до начала этапа тестирования может быть определена на основе модели Муса по формуле [5]

$$\lambda_0 = C \cdot N_{нач} \cdot V = C \cdot N_{нач} \cdot 3600R / (E_R \cdot L), \quad (2)$$

где C – коэффициент пропорциональности, некоторые авторы называют его коэффициентом проявления ошибок; V – скорость выполнения команд компьютерной программы, представляет собой возможное число прогонов компьютерной программы (команд программы) в течение одного часа; $N_{нач}$ – начальное число ошибок компьютерной программы (до начала тестирования); R – быстродействие процессора (производительность компьютера), размерность: операций в секунду; L – объём компьютерной программы в строках кода; E_R – коэффициент расширения кода (увеличения числа команд программы); множитель 3600 – коэффициент перевода быстродействия процессора в размерность операций в час.

Согласно [6] будем считать, что одна строка кода компьютерной программы транслируется в 10 машинных команд, т.е. $E_R = 10$.

С учётом изложенных предпосылок и положений модель для определения ожидаемой интенсивности отказов компьютерной программы, прошедшей тестирование, может быть представлена в виде

$$\lambda_{эксп}^{(i)} = 0,14 \cdot C_B \cdot K_{\Sigma}^{(i)} \cdot N_{нач}^{(i)} \cdot 3600R / 10L = 50,4 \cdot C_B \cdot R \cdot F_{нач}^{(i)} \cdot K_{\Sigma}^{(i)}, \quad (3)$$

где верхний индекс (i) указывает на то, что соответствующие характеристики относятся к компьютерным программам i -й области применения (см. таблицу 1); множитель 0,14 – коэффициент, учитывающий уменьшение интенсивности отказов в семь раз за счёт выполнения процедуры тестирования компьютерной программы ($1/7 \approx 0,14$); $F_{нач}$ – начальная плотность ошибок в компьютерной программе.

Значение $F_{нач}$ может быть найдено с использованием работы [3].

Список использованных источников:

- ГОСТ 27.205-1997. Надёжность в технике. Проектная оценка надёжности сложных систем с учётом технического и программного обеспечения и оперативного персонала. Основные положения. Минск : Госстандарт Республики Беларусь, 2005. 22 с.
- Модели оценки надёжности программных средств [Электронный ресурс]. – Режим доступа : <https://helpiks.org/4-73503.html> (дата обращения: 17.03.2020).
- Боровиков, С. М. Возможный подход к оценке надёжности прикладных программных средств для технологий Big Data / С. М. Боровиков, Лэ Ван Там, С. С. Дук // *BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня : сборник материалов V Международной научно-практической конференции*, Минск, 13–14 марта 2019 г. В 2 ч. Ч. 2 / Белорусский государственный университет информатики и радиоэлектроники; редкол. : В. А. Боеуш [и др.]. – Минск, 2019. – С. 77-83.
- Software reliability, measurement and testing guidebook for software reliability measurement and testing: RL-TR-92-52, Vol II (of two) Final technical report April 1992/ Science Applications International Corp. (SAIC), Research Triangle Institute (RTI). Rome Laboratory Air Force Systems Command Griffiss Air Force Base NY 13441-5700.
- Программирование и основы алгоритмизации исследование методов оценки и повышения надёжности программного обеспечения [Электронный ресурс]. – Режим доступа : <http://www.azovikdip.ru/index1.php> (дата обращения: 20.03.2020).
- Чуканов, В. О. Надёжность программного обеспечения и аппаратных средств систем передачи данных атомных электростанций : учеб. пособие. М.: МИФИ, 2008. 68 с.