

## НАДЁЖНОСТЬ РАЗРАБАТЫВАЕМЫХ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

Боровиков С.М., Дик С.С., Лэ Ван Там, Клинов К.И.

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

Abstract. The attractiveness of applied computer programs for distance learning systems is largely determined by the reliability of these programs. An approach to assessing the expected level of reliability of computer programs developed for distance learning systems by employees of higher educational institutions involving graduate students, undergraduates and students is described.

Эффективность систем дистанционного обучения во многом определяется надёжностью используемых программных средств (ПС). Ограничение уровня пригодности ПС является следствием наличия скрытых дефектов (говорят также ошибок), внесённых в компьютерные программы и содержание ПС в процессе постановки и решения задачи их создания. Ошибка компьютерной программы – это невыявленная погрешность или искажение кода, которые при определённом наборе входных данных в ходе выполнения этой программы могут вызвать её отказ или привести к неверным результатам, что рассматривается также как отказ.

В системах дистанционного обучения частое проявление ошибок компьютерных программ будет раздражать обучаемого, снизит эффективность восприятия учебного материала, а иногда может подорвать доверие к дистанционным системам получения образования. Многие учебные компьютерные программы для дистанционного обучения разрабатываются работниками учебных заведений. И, несомненно, им хотелось бы знать хотя бы примерный уровень надёжности будущих компьютерных программ. Ниже излагаются подходы, позволяющие сделать это.

Согласно ГОСТ 27.205-97, о проектной надёжности разрабатываемого программного средства (компьютерной программы) будем судить по интенсивности проявления скрытых дефектов, обусловленных ошибками его проектирования.

С учётом работы [1] и технического отчёта [2] для оценки ожидаемой надёжности прикладного ПС, прошедшего тестирование, получена модель

$$\lambda_{\text{н.тест}} = C \cdot F \cdot L_{\text{KLOC}} \cdot K_{\text{тест}}(S_1, S_2, S_3), \quad (1)$$

где  $\lambda_{\text{н.тест}}$  – ожидаемая интенсивность проявления дефектов программного средства, прошедшего тестирование;  $C$  – коэффициент пропорциональности;  $F$  – ожидаемая плотность оставшихся дефектов, приходющаяся на 1000 строк кода компьютерной программы после устранения в ней ошибок программирования (до выполнения тестирования);  $L_{\text{KLOC}}$  – объём текста кода компьютерной программы в тысячах строк кода;  $K_{\text{тест}}(S_1, S_2, S_3)$  – поправочный коэффициент уменьшения интенсивности проявления дефектов ПС за счёт выполнения его тестирования.

Коэффициент  $K_{\text{тест}}(S_1, S_2, S_3)$  предлагается определять с учётом используемых методов тестирования (метрика  $S_1$ ), времени тестирования (метрика  $S_2$ ), опыта и квалификации тестируемых (метрика  $S_3$ ).

Коэффициент  $C$  показывает, как оставшиеся в ПС скрытые дефекты «трансформируются» в ин-

тенсивность проявления ошибок при использовании компьютерной программы по назначению. Размерность коэффициента – 1/ошибок в час. Известно [2], что в некоторых моделях, используемых в зарубежной практике, изменчивость входных данных является основным фактором, определяющим эксплуатационную надёжность программного обеспечения. В модели (1) влияние изменчивости входных данных на интенсивность проявления ошибок компьютерной программы включено в коэффициент  $C$ . По мнению авторов, коэффициент  $C$  модели (1) в общем случае можно записать в виде

$$C = C_b \cdot K_{\text{вх.дан}},$$

где  $C_b$  – базовое значение коэффициента, характерное для компьютерных программ данной отрасли и/или программ, выполняемых на компьютерах определённого класса;  $K_{\text{вх.дан}}$  – коэффициент, характеризующий степень изменчивости входных данных, используемых компьютерной программой.

Согласно [3] коэффициент проявления ошибок  $C$  изменяется в диапазоне  $(1,4 \dots 10,6) \cdot 10^{-7}$ . В качестве базового (усреднённого) значения коэффициента  $C_b$  может быть принято типовое значение этого коэффициента ( $C_b \approx 4,2 \cdot 10^{-7}$  1/ошибок в час). Значения коэффициента  $K_{\text{вх.дан}}$  определены на примере приложений Windows (таблица 1), используя данные работы [5].

Таблица 1 – Коэффициент  $K_{\text{вх.дан}}$

Степень изменчивости входных данных	Примерное значение коэффициента $K_{\text{вх.дан}}$	Примеры приложений
Слабая	1...3	Excel, Word
Заметная	3...15	Power Point
Сильная	15...50	Access
Очень сильная	Более 50	Publisher

Значение прогнозируемой плотности дефектов  $F$  выражения (1) предлагается получать с помощью модели RL-92-52 [2], основанной на использовании метрик программного обеспечения:

$$F = A \cdot D \cdot S, \quad (2)$$

где  $A$  – средняя или базовая плотность ошибок для программных средств, используемых в данной сфере деятельности (отрасли);  $D$  – метрика, учитывающая производственную среду разработки программного обеспечения (особенность организации, разрабатывающей ПС; квалификацию программистов);  $S$  – метрика, учитывающая характеристики программного средства.



Согласно документу [2], для компьютерных программ, используемых в обучении, усреднённое значение базовой плотности ошибок  $A$  модели (2) составляет 12,3 ошибок на каждую тысячу строк кода (KLOC) компьютерной программы. Метрика  $D$  может быть представлена в виде:

$$D = K_{\text{орг}} \cdot K_{\text{кв.прог}}, \quad (3)$$

а для определения метрики  $S$  предлагается уточнённая модель вида:

$$S = K_{\text{нов}} \cdot K_{\text{слож}} \cdot K_{\text{С.Р}} \cdot K_{\text{мод}}, \quad (4)$$

где нижние индексы при поправочных коэффициентах  $K$  выражений (3) и (4) указывают на факторы, влияние которых принимается во внимание при расчёте метрик  $D$  и  $S$ . Пояснение поправочных коэффициентов вида  $K$ , их экстраполированные значения и/или рекомендации по их получению приводятся в [1].

В таблице 2 указаны значения коэффициентов, полученные с помощью экспертных оценок и/или экстраполяцией с использованием документа [4] и гипотезы о том, что число возможных ошибок в компьютерной программе прямо пропорционально затратам труда на её разработку.

Таблица 2 – Коэффициенты моделей (3) и (4)

Коэфф.	Учитываемый фактор	Значение
$K_{\text{орг}}$	Особенность организации, разрабатывающей ПС	0,5...2,0
$K_{\text{кв.прог}}$	Квалификация и опыт программиста	0,7...2,5*
$K_{\text{нов}}$	Степень новизны ПС	0,63...1,58
$K_{\text{слож}}$	Категория сложности ПС	1...1,47
$K_{\text{С.Р}}$	Средства разработки (язык программирования, ОС, компьютерная сеть)	0,16...1,3
$K_{\text{мод}}$	Степень использования стандартных модулей в ПС	0,55...1,0
$K_{\text{вх.дан}}$	Степень изменчивости входных данных	1...200*
Примечание. * – получено методом экспертной оценки		

Отметим, что значения, выбираемые для поправочных коэффициентов  $K_i$  по таблице 1, следует использовать в случаях, когда методом экспертной оценки затруднительно получить более достоверные значения этих коэффициентов.

Примерное обобщённое значение коэффициента  $K_{\text{тест}}(S_1, S_2, S_3)$  в модели (1) может быть получено с учётом следующего. В процессе тестирования специалистами, имеющими достаточный опыт тестирования компьютерных программ, выявляется не менее 50% ошибок, содержащихся в ПС. Выполненные за рубежом исследования [2] показали, что во многих случаях при календарной продолжительности тестирования, составляющей не менее 40% от времени, отводимого на разработку ПС, примерно до 35-40% ошибок остаются невыявленными. С учётом этого можно принять:

$$K_{\text{тест}}(S_1, S_2, S_3) \approx 0,43.$$

Пример применения разработанного подхода. Разрабатываемое прикладное ПС будет включать ориентировочно 20 тысяч строк программного кода. Определим прогнозное значение наработки на проявление ошибки  $T_{\text{о.пс}}$ , которое будет иметь место для ПС после завершения этапа его тестирования.

Будем предполагать, что время проявления ошибок ПС при использовании его по назначению описывается по экспоненциальному закону. Тогда:

$$T_{\text{о.пс}} = 1 / \lambda_{\text{п.тест}}$$

Принимая во внимание особенности учреждения образования, квалификацию программистов и характеристики разрабатываемого учебного прикладного ПС, с учётом таблиц 1 и 2, а также рекомендаций работы [1] выбраны следующие значения коэффициентов моделей (3) и (4):  $K_{\text{орг}} = 2,0$  (неспециализированная ИТ-организация);  $K_{\text{кв.прог}} = 2,5$  (магистранты, студенты);  $K_{\text{нов}} = 1,0$  (новое ПС, не имеющее аналогов; для действующих компьютеров и освоенной ОС);  $K_{\text{слож}} = 1,26$  (ПС, включающее моделирование объектов и процессов; интерактивный доступ; связь с другими ПС),  $K_{\text{С.Р}} = 1,2$  (процедурный язык высокого уровня, функционирование ПС в локальной сети);  $K_{\text{мод}} = 0,9$  (степень использования стандартных модулей до 20 %);  $K_{\text{вх.дан}} = 20,3$  [2] (сильная изменчивость входных данных).

Получено:

$$\lambda_{\text{п.тест}} = 4,2 \cdot 10^{-7} \cdot 20,3 \cdot 12,3 \cdot 2,0 \cdot 2,5 \cdot 1,0 \cdot 1,26 \cdot 1,2 \cdot 0,9 \cdot 20 \cdot 0,43 = 0,0061 \text{ ч}^{-1}, \text{ что соответствует } T_{\text{о.пс}} = 164 \text{ ч и значению вероятности } 0,976 \text{ отсутствия проявления ошибки ПС за время непрерывного использования компьютерной программы в течение 4 ч.}$$

## Литература

- Боровиков, С. М. Возможный подход к оценке надёжности прикладных программных средств для технологий Big Data / С.М. Боровиков, Ван Там Лэ, С.С. Дик // BIG DATA и анализ высокого уровня : сб. материалов V Междунар. науч.-практ. конф. (Республика Беларусь, Минск, 13–14 марта 2019 года). В 2 ч. Ч. 2. – Минск : БГУИР, 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). Rome Laboratory Air Force Systems Command Griffiss Air Force Base NY 13441-5700.
- Модели оценки надёжности программных средств [Электронный ресурс]. – Режим доступа: <https://helpiks.org/4-73503.html>. (Дата обращения 16.11.2019).
- Укрупнённые нормы затрат труда на разработку программного обеспечения [Электронный ресурс]. – Режим доступа : <http://www.levonevski.net/pravo/norm2013/num28/d28024.html>. (Дата обращения: 16.11.2019).
- Measuring Reliability of Software Products [Электронный ресурс]. – Режим доступа : <http://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-2004-145.pdf>. (Дата обращения: 16.11.2019).