

УДК 004.415.532.2

**АНАЛИЗ ПЛАВАЮЩИХ ДЕФЕКТОВ В ПРИКЛАДНОМ ПРОГРАММНОМ ОБЕСПЕЧЕНИИ С ИСПОЛЬЗОВАНИЕМ ВЕРОЯТНОСТНЫХ МЕТОДОВ****Петросян Г.С., Титов В.А.***ФГБОУ ВПО «Российский экономический университет имени Г.В. Плеханова»,  
Москва, e-mail: grantp@rambler.ru, vitov213@yandex.ru*

Настоящая статья посвящена анализу плавающих дефектов, что является актуальной задачей в области тестирования и обеспечения качества ПО. Дано определение плавающих дефектов, рассмотрены основные особенности и риски, связанные с управлением данными дефектами. В частности, описана классификация плавающих дефектов согласно инженерному стандарту IEEE 1044. В работе предложены две математические модели анализа плавающих дефектов. При построении данных моделей был использован аппарат теории вероятностей и математической статистики. Приведены конечные формулы для точечных и интервальных оценок следующих показателей: вероятность проявления дефекта, среднее и худшее число попыток до воспроизведения дефекта, среднее и худшее время до воспроизведения дефекта. На различных примерах из банковской сферы показано применение предложенных зависимостей. Представленные модели реализованы с использованием языка программирования R. Делается вывод о том, что предлагаемые модели могут облегчить работу тестировщиков, разработчиков и руководителей IT-проектов. Результаты статьи могут использоваться как в учебном процессе, так и в практической деятельности IT-компаний.

**Ключевые слова:** тестирование ПО, управление дефектами, плавающие дефекты, вероятность проявления дефекта, статистическое оценивание, язык программирования R

**ANALYSIS OF INTERMITTENT SOFTWARE BUGS USING PROBABILITY THEORY****Petrosyan G.S., Titov V.A.***Plekhanov Russian University of Economics, Moscow, e-mail: grantp@rambler.ru, vitov213@yandex.ru*

This article is dedicated to the analysis of intermittent defects, which is the actual problem in the field of software testing and quality assurance. The definition of intermittent defects is given, their main features and related risks are considered. Particularly, classification of intermittent software defects according to IEEE 1044 is considered. In the work two mathematical models of intermittent defects analysis were developed. These models are based on probability theory and mathematical statistics. The explicit formulae of point and interval estimation are provided for the following metrics: probability of defect reoccurring; mean and worst number of attempts to reproduce the defect; mean and worst time to reproduce the defect. These formulae are demonstrated with certain examples from the banking sphere. The models provided in the article are implemented by means of R programming language. It is concluded that proposed models can make easier the work of testers, developers and IT project managers. Results of the article can be used in academic affairs and in practical work of IT companies as well.

**Keywords:** software testing, defect management, intermittent defects, defect probability, statistical estimation, R programming language

В условиях интенсивного развития информационных технологий и усложнения программных комплексов в конце XX столетия (80–90-е гг.) сформировалась отдельная ветвь компьютерных наук – тестирование ПО. Одна из задач в тестировании ПО – это нахождение дефектов, то есть несоответствий реального и ожидаемого поведения ПО.

После того, как дефект найден, специалист по тестированию должен составить для него грамотное и однозначное описание. Указанный в отчёте об ошибке сценарий должен приводить к проявлению данной ошибки в 100% случаев.

Существует класс дефектов, для которых данная задача не является тривиальной. Пожалуй, каждый специалист по тестированию сталкивался с проблемой воспроизводимости: ошибка возникает, к примеру, 4 раза из 10. Такие ошибки принято называть «фантомными» или «плавающими».

Плавающим называется дефект, чёткие шаги для воспроизведения которого не очевидны либо отсутствуют.

Разработчик, который должен исправить тот или иной плавающий дефект, заинтересован в максимально подробном его описании. Так, например, фраза «Ошибка воспроизводится не всегда» представляет для разработчика меньшую ценность, чем «Для воспроизведения ошибки потребуется в среднем 5 попыток». В случае, когда описание дефекта является расплывчатым, существует риск того, что ответственный за исправление переведёт данный дефект в статус «Невозможно воспроизвести» (Cannot reproduce) или, что хуже, отложит исправление проблемы «на потом».

Как следствие, необходимы специальные методы составления правильных отчётов о таких ошибках. В силу стохастической природы рассматриваемого нами класса дефектов, разработка данных методов невозможна

без использования аппарата теории вероятностей и математической статистики.

В современной отечественной и зарубежной литературе по информационным технологиям и тестированию ПО слабо отражены принципы работы с плавающими дефектами, в частности, использование вероятностных методов в анализе подобных дефектов.

**Целью** данной работы является построение двух математических моделей анализа плавающих дефектов в прикладном программном обеспечении.

В стандарте IEEE 1044 (Standart Classification for Software Anomalies) наряду с такими атрибутами дефекта, как Критичность (Severity) и Приоритетность (Priority), можно видеть ещё один не менее важный атрибут – Вероятность (Probability) проявления данного дефекта [3]/

В документе также указаны возможные значения данного атрибута:

- Высокая (High) – значение вероятности более 70%.
- Средняя (Medium) – значение вероятности находится между 40 и 70%.
- Низкая (Low) – Значение вероятности менее 40%.

Однако в рассматриваемом инженерном стандарте не описываются способы оценки данной вероятности.

Перейдём к рассмотрению первой из двух моделей – к модели воспроизведения ошибки, основанной на схеме испытаний Бернулли. Основной задачей в рамках данной модели является оценка вероятности проявления дефекта для последующего указания в отчёте о данном дефекте.

Напомним, что схемой испытаний Бернулли принято называть серию последовательных испытаний, в которой любое испытание имеет только 2 исхода (например «дефект» и «отсутствие дефекта») и вероятности этих исходов не зависят от результатов предыдущих испытаний [1].

Дальнейшие расчёты будем производить в рамках следующего примера:

Рассмотрим тестирование системы интернет-банкинга. Иногда при попытке открыть web-форму для открытия депозита появляется сообщение об ошибке.

Попробуем воспроизвести дефект до 10-го его проявления.

Занесём результаты эксперимента в табл. 1, обозначив возникновение дефекта как «Fail» и его отсутствие как «-»:

**Таблица 1**

Статистика проявления дефекта в рамках примера с интернет-банкингом

№ п/п	Результат	№ п/п	Результат
1	-	17	-
2	-	18	Fail
3	Fail	19	-
4	-	20	-
5	-	21	-
6	-	22	-
7	Fail	23	-
8	Fail	24	-
9	Fail	25	Fail
10	-	26	-
11	Fail	27	-
12	-	28	-
13	-	29	-
14	Fail	30	Fail
15	-	31	Fail
16	-		

Назовём эмпирической частотой проявления дефекта отношение числа испытаний, в котором дефект проявился ( $d$ ) к общему кол-ву испытаний ( $n$ )

$$p^* = \frac{d}{n}. \tag{1}$$

Отсюда

$d = 10$  – число испытаний, в которых дефект проявился;

$n = 31$  – общее количество испытаний;

$$p^* = \frac{10}{31} = 0,323 \text{ – эмпирическая частота}$$

проявления дефекта.

Доказано, что эмпирическая частота  $p^*$  является несмещенной, состоятельной и эффективной оценкой вероятности дефекта [2].

Очевидно, что при последующем повторении эксперимента, который заключается в попытке воспроизвести дефект до 10-го его проявления, мы будем наблюдать другое значение  $p^*$ . Частота ошибки скорее всего будет отличаться от 0,323, даже если провести ровно 31 испытание.

Чтобы устранить этот недостаток, рассматривается задача построения доверительного интервала для вероятности дефекта.

В рамках схемы Бернулли доверительный интервал для вероятности имеет следующий вид [1]:

$$p^* - z_{\frac{1+\beta}{2}} \sqrt{\frac{p^*(1-p^*)}{n}} < p < p^* + z_{\frac{1+\beta}{2}} \sqrt{\frac{p^*(1-p^*)}{n}}. \tag{2}$$

Обозначения:

$z_\gamma$  – квантиль стандартного нормального распределения уровня  $\gamma$ ;

$\beta$  – доверительная вероятность;

$p^*$  – заранее рассчитанная эмпирическая частота дефекта;

$n$  – число испытаний, на базе которых построена оценка  $p^*$ .

Продемонстрируем использование формулы (2) при  $p^* = 0,323$ ;  $n = 31$  и  $\beta = 0,95$ :

$$0,323 - z_{\frac{1+0,95}{2}} \sqrt{\frac{0,323(1-0,323)}{31}} < p < 0,323 + z_{\frac{1+0,95}{2}} \sqrt{\frac{0,323(1-0,323)}{31}}.$$

Для расчёта квантиля стандартного нормального распределения можно воспользоваться специальными таблицами или, что проще, функцией «qnorm()» в R.

$$z_{\frac{1+0,95}{2}} = z_{0,975} = 1,960.$$

Таким образом, получим, что истинное значение вероятности  $p$  находится внутри интервала (0,158; 0,487).

Возможна и другая запись полученного результата:  $p = 0,323 \pm 0,165$ .

Следует также отметить, что доверительный интервал расширяется при уменьшении  $n$ . Отсюда следует, что при небольшом количестве испытаний расчёт доверительного интервала не имеет практической значимости.

Таким образом, приведены конечные формулы для точечной и для интервальной оценок вероятности проявления дефекта.

В рамках того же примера несложно рассчитать среднее количество попыток, которое было предпринято для того, чтобы наблюдать дефект. Данная величина может быть рассчитана как простая средняя арифметическая (табл. 2):

$$\bar{x} = \frac{31}{10} = 3,1.$$

Вычислим теперь число попыток воспроизвести дефект, которое не будет превышено с заданной наперёд вероятностью  $\alpha$ .

Доказано [1], что такое число попыток может быть найдено по следующей формуле:

$$q_\alpha = \left\lceil \log \left( \frac{1}{1-\alpha} \right) \right\rceil. \quad (3)$$

В данной формуле:

$\alpha$  – заданная вероятность;

$[y]$  – целая часть числа  $y$ .

Вернёмся к примеру с интернет-банком.

$$q_{0,95} = \left\lceil \log \left( \frac{1}{1-0,995} \right) \right\rceil = \lceil 7,692 \rceil = 7.$$

**Таблица 2**

Расчёт среднего количества попыток для воспроизведения дефекта в примере с интернет-банкингом

Номер серии испытаний	Кол-во попыток для воспроизведения
1	3
2	4
3	1
4	1
5	2
6	3
7	4
8	7
9	5
10	1
$\Sigma$	31

Таким образом, с вероятностью 95% для воспроизведения дефекта понадобится не более 7 попыток.

Итак, теперь в отчёт об ошибке в интернет-банке может добавиться следующая информация:

«Вероятность дефекта  $\approx 32,3 \pm 16,5\%$  (низкая).

Среднее количество попыток для воспроизведения дефекта – 3,1.

С вероятностью 95% для воспроизведения дефекта понадобится не более 7 попыток».

Не стоит забывать и о другой информации, которая поможет воспроизвести и локализовать дефект: логи, HTTP-запросы, и т.д. Подобные уточнения являются необходимым условием любого грамотного отчёта о плавающей ошибке.

Полученные результаты также могут быть использованы для того, чтобы убедить руководителя проекта по разработке интернет-банка в необходимости исправления дефекта:

«Ожидается, что в течение следующего месяца 15 000 пользователей попытаются произвести открытие депозита онлайн, при этом средний размер одного депозита составит 600 руб. В этих условиях наиболее вероятные потери от дефекта будут равны  $2\,907\,000 \pm 1\,485\,000$  руб.».

Данные цифры были рассчитаны исходя из вероятности дефекта:

$$2\,907\,000 = 15\,000 \cdot 600 \cdot 0,323;$$

$$1\,485\,000 = 15\,000 \cdot 600 \cdot 0,165.$$

Рассмотрим недостатки данной модели. Описанная выше модель основывается на двух предположениях, которые могут не выполняться в реальных условиях:

- Считается, что каждое последующее испытание не зависит от исходов предыдущих.
- Считается, что испытания независимы от прочих факторов (например, нагрузки на систему).

Теперь перейдем к рассмотрению простейшей модели времени до воспроизведения дефекта. В некоторых случаях такие показатели, как «вероятность дефекта» и «среднее число попыток» оказываются неприменимыми.

В качестве примера такой ситуации рассмотрим автоматизированную систему «Оповещения», которая отправляет клиентам банка SMS-сообщения об операциях по счетам и картам. Пусть по факту снятия средств в банкомате клиент получает необходимое оповещение, а спустя некоторый промежуток времени происходит ошибочная отправка дублирующего SMS. Причём этот промежуток времени иногда может составлять 8 мин, а иногда – 2 часа.

Данный дефект также можно отнести к классу плавающих.

В связи с этим разумно ввести такую характеристику, как «среднее время до воспроизведения дефекта».

Для моделирования времени будем использовать экспоненциальное распределение вероятностей.

В рамках примера с самопроизвольным переходом сделаем 20 попыток воспроизвести дефект (табл. 3):

**Таблица 3**

Статистика проявления дефекта в рамках примера с системой «Оповещения»

№ п/п	Время до проявления дефекта (мин)	№ п/п	Время до проявления дефекта (мин)
1	8,277	11	9,610
2	59,425	12	22,343
3	55,909	13	23,166
4	17,696	14	3,355
5	35,295	15	133,107
6	35,111	16	8,831
7	36,915	17	21,335
8	11,664	18	57,676
9	5,078	19	57,901
10	21,851	20	3,679
Σ	628,226		

Тогда

$$\bar{x} = \frac{628,226}{20} = 31,411 - \text{среднее время до}$$

проявления дефекта.

Время, которое при попытке воспроизвести дефект не будет превышено с заданной наперед вероятностью  $\alpha$ , может быть вычислено как квантиль экспоненциального распределения уровня  $\alpha$  [1]:

$$x_\alpha = -\bar{x} \ln(1 - \alpha). \quad (4)$$

Так, например:

$$x_{0,95} = -31,411 \cdot \ln(1 - 0,95) = 94,100.$$

Для среднего значения времени до воспроизведения дефекта также можно построить доверительный интервал [4]:

$$\frac{2n\bar{x}}{\chi_{1-\beta}^2(2n)} < m < \frac{2n\bar{x}}{\chi_{1+\beta}^2(2n)}. \quad (5)$$

Здесь  $\chi_\gamma^2(k)$  – это квантиль распределения хи-квадрата с  $k$  степенями свободы уровня  $\gamma$ ;  $\beta$  – доверительная вероятность;  $\bar{x}$  – заранее рассчитанное среднее значение;  $n$  – число испытаний, на базе которых рассчитано среднее значение  $\bar{x}$ ;  $m$  – «истинное значение среднего времени».

Воспользуемся формулой (5):

$$\frac{2 \cdot 20 \cdot 31,411}{\chi_{1-0,95}^2(2 \cdot 20)} < m < \frac{2 \cdot 20 \cdot 31,411}{\chi_{1+0,95}^2(2 \cdot 20)};$$

$$\chi_{1-0,95}^2(2 \cdot 20) = \chi_{0,025}^2(40) = 59,342;$$

$$\chi_{1+0,95}^2(2 \cdot 20) = \chi_{0,975}^2(40) = 24,433.$$

Итого,

$$21,173 < m < 51,424.$$

Перейдём к подведению итогов. В отчёт о рассматриваемой ошибке можно добавить следующую информацию:

«Ожидаемое время до отправки дублирующего сообщения  $\approx 21,173$  мин – 51,424 мин.

С вероятностью 95% время до ошибочной отправки SMS не превысит 94,100 мин»

Рассмотрим недостатки второй модели. Данная модель также базируется на предположении о том, что испытания не зависят от прочих факторов (например, нагрузки на систему).

Отметим также, что время до воспроизведения дефекта может быть более точно смоделировано с помощью гамма-распределения. Рассмотрение данного метода выходит за рамки настоящей статьи в силу трудоёмкости расчётов, связанных с оценкой параметров гамма-распределения.

Таблица 4

## Стандартные функции R

Функция в R	Описание
qnorm(вероятность, mean = 0, sd = 1)	Квантиль стандартного нормального распределения
qchisq(вероятность, степени свободы, lower.tail = FALSE)	Квантиль распределения хи-квадрата
log(число)	Натуральный логарифм
log(число, основание)	Логарифм с произвольным основанием
sqrt(число)	Корень
floor(число)	Целая часть

Таблица 5

## Использование стандартных функций R

Выражение	Формула в R
$\left\lceil \log_{\left(\frac{1-1}{3,1}\right)}(1-0,95) \right\rceil$	floor(log(1 - 0.95, 1 - 1/(3.1)))
$-31,411 \cdot \ln(1 - 0,95)$	-31.411*log(1 - 0.95)
$0,323 + z_{\frac{1+0,95}{2}} \sqrt{\frac{0,323(1-0,323)}{31}}$	0.323 + qnorm((1 + 0.95)/2, mean = 0, sd = 1)*sqrt(0.323*(1 - 0.323)/31)
$\frac{2 \cdot 20 \cdot 31,411}{\chi_{1-0,95}^2(2 \cdot 20)}$	(2*20*31.411)/qchisq((1 - 0.95)/2, 2*20, lower.tail = FALSE)

Для расчётов целесообразно воспользоваться статистическим пакетом R [5]. В табл. 4 приведены некоторые стандартные функции R.

В табл. 5 продемонстрируем использование функций R для решения наших задач (табл. 5).

В заключение необходимо упомянуть о риске ложного отчёта о плавающем дефекте. Перед тем как утверждать, что тот или иной дефект является плавающим, специалист по тестированию обязан потратить время и попытаться найти условия, при которых дефект возникает постоянно.

Стоит иметь в виду некоторые стандартные ситуации проявления плавающих ошибок: интернет работает с перебоями или блокирует определённые порты, определённый функционал также может блокировать антивирус и т.д.

Таким образом, рассмотрены два математических метода, которые удобно использовать для анализа некоторых плавающих дефектов.

Информация, полученная при помощи данных инструментов, несомненно, является необходимой как для лица, ответственного за исправление дефекта, так и для руководителя IT-проекта, в рамках которого возник тот или иной дефект.

## Список литературы

1. Вентцель Е.С., Овчаров Л.А. Теория вероятностей и ее инженерные приложения. – М.: Высшая Школа, 2000.
2. Смирнов Н.В., Дунин-Барковский И.В. Курс Теории Вероятностей и математической статистики для технических приложений. – М.: Наука, 1969.
3. [IEEE1044] IEEE Standard Classification for Software Anomalies. – URL: [http://www.ctestlabs.org/neoacm/1044\\_2009.pdf](http://www.ctestlabs.org/neoacm/1044_2009.pdf).
4. Ross, Sheldon M. (2009). Introduction to probability and statistics for engineers and scientists (4th ed.). Associated Press, – P. 267.
5. The R Manuals [Электронный ресурс]. – Режим доступа: <https://cran.r-project.org/manuals.html> (дата обращения: 10.06.2016).

## References

1. Ventcel E.S., Ovcharov L.A. Teorija verojatnostej i ee inzhenernye prilozhenija. M.: Vysshaja Shkola, 2000.
2. Smimov N.V., Dunin-Barkovskij I.V. Kurs Teorii Verojatnostej i matematicheskoj statistiki dlja tehniceskij prilozhenij. M.: Nauka, 1969.
3. [IEEE1044] IEEE Standard Classification for Software Anomalies. – URL: [http://www.ctestlabs.org/neoacm/1044\\_2009.pdf](http://www.ctestlabs.org/neoacm/1044_2009.pdf).
4. Ross, Sheldon M. (2009). Introduction to probability and statistics for engineers and scientists (4th ed.). Associated Press, pp. 267.
5. The R Manuals [Jelektronnyj resurs]. Rezhim dostupa: <https://cran.r-project.org/manuals.html> (data obrashhenija: 10.06.2016).