

И. И. Губайдуллин, А. И. Фрид

ОЦЕНКА НАДЕЖНОСТИ ОДНОГО ТИПА ПРОГРАММ С ОПРЕДЕЛЕНИЕМ ПОГРЕШНОСТИ ПОЛУЧЕННОЙ ОЦЕНКИ

Рассматривается метод оценки надежности программного обеспечения, основанный на моделировании работы программы в зависимости от законов распределения исходных данных. Решается задача определения погрешности полученной оценки путем вычисления доверительных интервалов. Приводятся результаты применения метода для некоторых программ. *Программное обеспечение; программа; надежность; надежность программы; оценка программы; оценка надежности; моделирование программы*

ВВЕДЕНИЕ

Основным международным стандартом качества в области программного обеспечения (ПО) является стандарт ISO/IEC 9126. В российской системе стандартизации показатели качества программ устанавливают ГОСТы Р ИСО/МЭК 9126 и 28195 [1, 2]. В качестве оценочного элемента для измерения фактора «Надежность ПО» в работах [3–5] использовался показатель вероятности безотказной работы, рассчитываемый по формуле:

$$P = 1 - N_Q / N, \quad (1)$$

где N – количество запусков программы; N_Q – число зарегистрированных отказов. Этот показатель рассчитывается путем моделирования работы программы в зависимости от законов распределения исходных данных. Метод основан на получении информации об ошибке в процессе выполнения программы. Индикаторами ошибок служили исключительные ситуации и несоответствие выходных данных требованиям.

На сегодняшний день отсутствует универсальный метод оценки, позволяющий оценить надежность ПО как пользователю, так и создателю. Существующие модели [6] имеют следующие недостатки:

- необходима первоначальная статистика о качестве работы коллектива;

- исходные данные программы и непосредственно код программы не являются первоначальными данными самих моделей;

- число ошибок характеризует в большой степени не программу, а ее изготовителей и инструментарий;

- сложность применения моделей при сравнении надежности идентичных программ.

В рассмотренных моделях вероятность отказа ПО не учитывает влияние исходных данных. В связи с этим разработан и предложен такой метод оценки надежности ПО, в котором надежность оценивается, исходя из параметров законов распределения исходных данных. Метод оценки надежности программы можно сопоставить с методами стохастического тестирования [7].

Некоторые из известных моделей надежности ПО, помимо получения характеристики надежности, также имеют соответствующие им оценки. Например, такими моделями являются: экспоненциальная модель Джелинского–Моранды и модель Шумана, модель Липова, модель Вейбулла, модель роста надежности и др. [6]. Обычно инструментом оценки в таких моделях выступает метод максимального правдоподобия. В данной работе рассматривается метод оценки надежности ПО, основанный на результатах многочисленных экспериментов, и в качестве определения погрешности полученных значений вероятности безотказной работы программы используется метод доверительных интервалов. В качестве примера решается задача определения погрешности вероятности безотказной работы для системы уравнений, описывающих кинетику одной химической реакции [3–5].

Контактная информация: (347) 273-06-72

Работа поддержана грантом РФФИ № 08-08-00357 «Методы проектирования отказоустойчивых программно-аппаратных вычислительных комплексов для систем управления сложными техническими объектами с использованием средств искусственного интеллекта»

1. ЭКСПЕРИМЕНТАЛЬНЫЙ МЕТОД ОЦЕНКИ НАДЕЖНОСТИ ПО

Во многих задачах промышленности программы оперируют случайными входными данными. Идея оценки надежности таких программ заключается в моделировании работы программы с дальнейшей статистической обработкой полученных результатов, когда на ее вход поступают данные, поведение которых можно описать определенными законами распределения.

Охарактеризуем один из показателей надежности ПО как вероятность отказа при определенных распределениях исходных данных. Под отказом понимается отклонение поведения системы от предписанного, которое заключается в том, что система перестает выполнять предписанные ей функции [8].

Любой блок программы можно представить в виде функционального элемента с определенными входами-выходами. На вход такого блока подаются входные величины, подчиняющиеся определенному закону распределения. Ошибкой в широком смысле будем считать невыполнение блоком своих функций. Для оценки надежности программы необходимо выделить множество возможных ошибок, которые можно ожидать в конкретной программе. Появление событий, приводящих к ошибкам, отслеживается определенными датчиками ошибок, заложенными в самом блоке. К таким событиям можно отнести арифметические ошибки (деление на ноль, выход за разрядную сетку), потерю точности, являющуюся следствием постепенных накопленных ошибок в связи с ограниченностью разрядной сетки, и др. После того, как выделены возможные ошибки и установлены датчики, начинается процесс оценки надежности блока. Этот этап можно сравнить со стохастическим тестированием, в котором исходные данные также распределены по определенному закону. После проверки достаточного количества прогонов, методом непосредственного подсчета делается заключение о вероятности появления ошибки в будущем. Метод оценки надежности состоит из 6 этапов:

- определение законов распределения исходных данных;
- выявление возможных ошибок в программе;
- моделирование работы программы, когда на ее вход поступают данные, подчиненные определенными законами распределения, расчет вероятности отказа;

- определение допустимых параметров законов распределения исходных данных, при которых программа будет обладать заданной надежностью;
- определение погрешности расчета надежности программы;
- получение результатов оценки надежности программы в пределах необходимой точности.

2. ОПРЕДЕЛЕНИЕ ПОГРЕШНОСТИ РАСЧЕТА НАДЕЖНОСТИ ПРОГРАММЫ МЕТОДОМ ДОВЕРИТЕЛЬНЫХ ИНТЕРВАЛОВ

Расчеты искомых величин, основанные на использовании метода моделирования, выполняются на ограниченном числе опытов и содержат элементы случайности. При оценке надежности программы искомой величиной является вероятность ее безотказной работы. Путем моделирования работы программы, изменяя параметры законов распределения исходных данных, получают так называемую точечную оценку надежности, выраженную одним числом. Замена искомой величины точечной оценкой может привести к значительной погрешности, поэтому требуется знать, с какой степенью уверенности можно ожидать, что эти погрешности не выйдут за известные пределы. Чтобы получить представление о точности расчетов, можно воспользоваться методом доверительных интервалов.

Предположим, что требуется оценить погрешность определения вероятности безотказной работы программы для фиксированных параметров законов распределения входных данных. Генерируя входные данные с заданными законами распределения, проведем ряд $q_1, q_2, \dots, q_i, \dots, q_n$ вычислений вероятности безотказной работы программы. Полученные n значений заменяются одним точечным значением – оценкой Q^* . Назначается некоторая большая вероятность P (например $P = 0,95$) такая, что событие с вероятностью P можно считать практически достоверным, и определяется такое значение u_f для которого

$$P(Q - Q^* < u_f) = P. \quad (2)$$

Это равенство означает, что с вероятностью P неизвестное значение параметра Q попадет в интервал

$$I_p = (Q^* - u_f; Q^* + u_f). \quad (3)$$

В данном случае величина Q^* не случайна, но случаен интервал I_p . Вероятность P принято называть доверительной вероятностью, а интервал I_p – доверительным интервалом [9].

Неизвестную величину u_f вычисляют на основе опытных данных $q_1, q_2, \dots, q_i, \dots, q_n$. В качестве оценки Q будем рассматривать математическое ожидание. В этом случае

$$u_f = \frac{t(P; (n-1))}{\sqrt{n}} s, \quad (4)$$

где $t(P; (n-1))$ – t -распределение с $n-1$ степенями свободы; n – количество наблюдений; s – оценка дисперсии, рассчитанная по формуле [10]:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (q_i - q_{cp})^2} = \sigma^+. \quad (5)$$

3. ПРИМЕРЫ ПРИМЕНЕНИЯ МЕТОДА К НЕКОТОРЫМ ПРОГРАММАМ

3.1. Применение метода к оценке надежности программы, реализующей решение квадратного уравнения

В разделе приведены результаты оценки надежности программы, реализующей решение квадратного уравнения $Ax + Bx + C = 0$. Более подробное описание метода для этой програм-

мы приводится в [3, 4]. Основным событием, приводящим к отказу в данной программе, является отрицательное подкоренное выражение при вычислении дискриминанта. Пусть Q – вероятность отказа программы; $P(B^2 - 4AC < 0)$ – вероятность события, заключающегося в том, что результат $(B^2 - 4AC < 0)$ окажется меньше нуля. Тогда $Q = P(B^2 - 4AC < 0)$.

На рис. 1 показаны результаты работы программы. Расчет величины Q для программы решения квадратного уравнения производился в среде MATLAB для различных параметров законов распределения входных величин.

Изучалось изменение величины Q при изменении математического ожидания (МО) входных величин в диапазоне $[-100, 100]$. МО величины B фиксировалось, а МО величин A и C изменялось с интервалом, равным 5. При этом дисперсия у всех входных величин была равна 100 и оставалась неизменной.

Видно, что при приближении $МО_B$ к нулю увеличивается площадь областей, в которых вероятность отказа близка к единице.

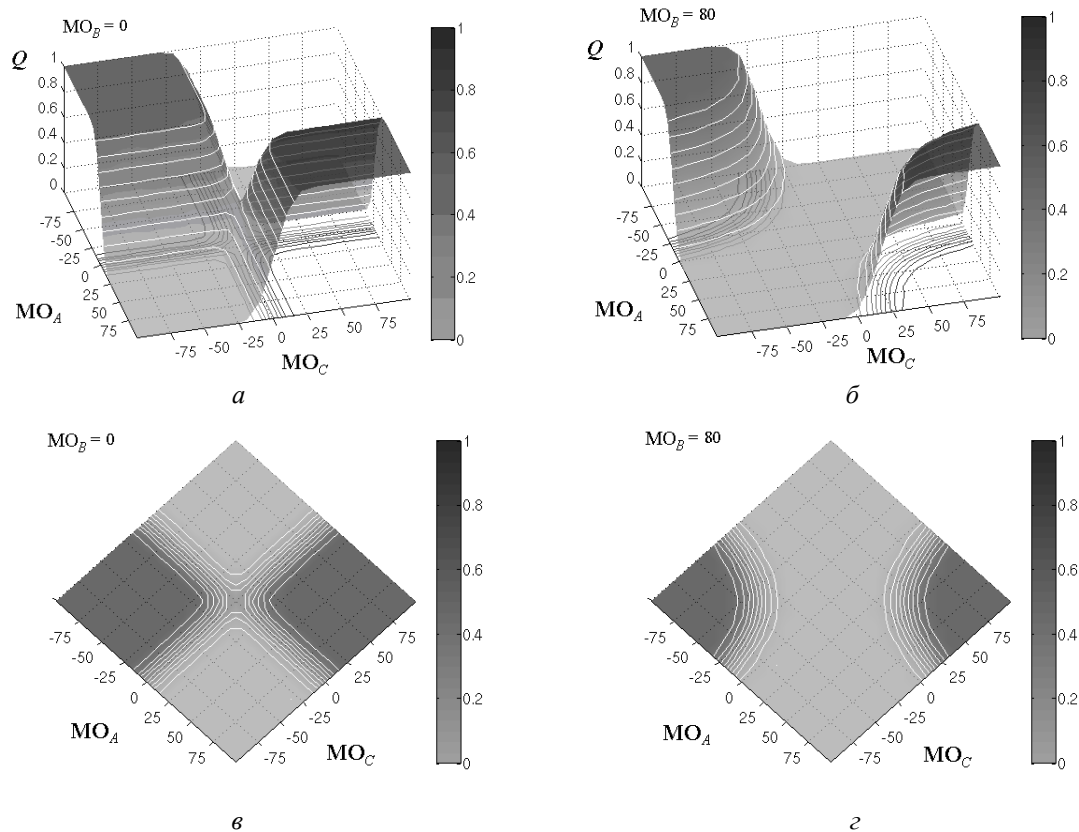


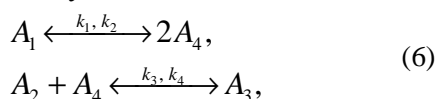
Рис. 1. Графики изменения величины Q от $МО_A$ и $МО_C$ при $МО_B = 0$ (а, в) и $МО_B = 80$ (б, г)

Из графиков, представленных на рис. 1, можно определить МО A , B и C , при которых программа будет обладать заданным уровнем надежности. Например, если требуемый уровень надежности составляет 0,9, то $МО_A$ должно быть больше +10, при этом $МО_B = \pm 40$ и $МО_C < -10$.

3.2. Применение метода к оценке надежности программы, моделирующей решение прикладной задачи химической промышленности

Рассматриваемый метод также был применен для оценки надежности программы, реализующей решение задачи по определению кинетических параметров химической реакции. Программа включает в себя оптимизационные алгоритмы и алгоритмы решения системы обыкновенных нелинейных дифференциальных уравнений (СОНДУ). Необходимость проведения исследования такого типа алгоритмов явилась следствием частых остановов при выполнении программы, с помощью которой решается задача поиска скоростных коэффициентов химической реакции.

Цель решения задачи – определить скоростные коэффициенты каждой стадии многостадийной химической реакции. Проводится исследование определенной реакции, в процессе которой через заданные промежутки времени фиксируют выход некоторых веществ, причем выход остальных веществ не наблюдается. В качестве примера рассмотрена модель реакции, имеющей следующий вид:



где A_j – определенные вещества (конкретный химический состав каждого вещества в данной работе не приводится). Выход веществ A_1 , A_3 – наблюдаем, остальные – нет; k_i (c^{-1}) – скоростной коэффициент i -й стадии.

Для расчета скоростей, согласно теории химической кинетики, используются следующие формулы:

$$\begin{aligned} w_1 &= k_1 x_1 - k_3 x_4^2, \\ w_2 &= k_2 x_2 x_4 - k_4 x_3, \end{aligned} \quad (7)$$

где w_i (c^{-1}) – скорость i -й стадии; x_i – мольные доли i -го вещества в конкретный момент времени.

Исходными данными для программы являются: схема превращений, т. е. математическая модель реакции, первоначальные мольные доли веществ и экспериментальные данные, полу-

ченные в процессе проведения реакции через фиксированные промежутки времени. После ввода исходных данных начинается основная работа программы – поиск скоростных коэффициентов. При этом используются три основных блока:

- блок, реализующий алгоритм оптимизации;
- блок решения СОНДУ методом Рунге–Кутты 4-го порядка точности [10];
- блок сравнения теоретических и экспериментальных данных.

В начале работы оптимизационный алгоритм (ОА) выбирает первоначальный вектор скоростных коэффициентов.

Подставив этот вектор в СОНДУ, находим решение, из которого извлекается информация о количественном соотношении выхода веществ. Найденные процентные доли веществ, являющиеся численным решением, сравниваются с данными, полученными в процессе эксперимента. Цель ОА заключается в подборе таких скоростных коэффициентов, при которых разность между теоретическими и экспериментальными данными будет удовлетворять некоторой наперед заданной величине. Разница между этими данными минимизируется методом наименьших квадратов. Суть этого метода заключается в том, что через ряд экспериментальных точек проводят такую зависимость $F(x, k_0, k_1, \dots, k_n)$, сумма квадратов отклонений которой от экспериментальных значений при соответствующих варьируемых значениях k_0, k_1, \dots, k_n минимальна:

$$\zeta = \min \left(\sum_i^m (y_i - F(x_i, [k]))^2 \right), \quad (8)$$

где y_i – ряд экспериментальных точек, определенных для соответствующих интервалов времени x_i ; m – количество экспериментальных точек; $[k]$ – вектор скоростных коэффициентов; $F(x_i, [k])$ – функция теоретических значений при определенном наборе k .

Функционирование рассматриваемого алгоритма зависит от выбора первоначального вектора k , т. е. от выбора начальной точки:

$$\{k_1, k_2, k_3, k_4\}. \quad (9)$$

В связи с этим вероятность отказа программы будет рассчитываться в зависимости от выбора начальной точки. Примем, что все элементы вектора (9) распределены по нормальному закону. В данном примере среднее квадратическое отклонение для всех элементов принято равным 1. Математическое ожидание (МО) эле-

ментов было расположено в интервале от 1 до 11 с шагом 2. Вероятность отказа программы зависит от МО элементов вектора (9). Из фиксированного набора

$$\{MO(k_1), MO(k_2), MO(k_3), MO(k_4)\}. \quad (10)$$

генерировалась последовательность из 50 различных векторов (9) и в результате получалось 50 различных значений критерия ζ , представляющих собой некий результат работы программы. В качестве отказа принято событие А, при котором заданная величина $\zeta > \zeta_0$. Вероятность этого события определялась методом непосредственного подсчета. В данном исследовании выбрано 225 фиксированных векторов (10). При этом на каждый вектор приходится по 50 запусков программы, получается 50 векторов (9) и соответствующих им решений ζ . Таким образом, получена табл. 1, содержащая 11250 строк, в которой вектору (10) сопоставлено определенное значение ζ .

В данном случае из физических соображений ζ_0 принято равным 120. Для каждой группы

из 50 экспериментов, в которых постоянный вектор (9), вычисляется вероятность отказа программы Q .

В результате моделирования для конкретного вектора (10) было определено значение вероятности отказа. Для наглядности математические ожидания k_3 и k_4 были зафиксированы значениями 11 и 1, соответственно. В этом случае изменение вероятности безотказной работы программы P в зависимости от МО k_1 и k_2 можно представить графически (рис. 2).

Области, в которых алгоритмический отказ наименее вероятен, обозначены темным цветом. Для обеспечения высокой надежности программы необходимо выбирать МО k_1 и k_2 из интервалов [1; 6] и [2; 5], соответственно.

Моделируя работу программы в зависимости от распределения исходных данных, получили области, в которых вероятность безотказной работы программы для расчета скоростных коэффициентов составляет более 0,7.

Таблица 1

Исходные данные для расчета вероятности отказа Q

№ запуска	Фиксированный вектор МО				Начальный вектор (9)				Результирующий вектор				ζ
	МО k_1	МО k_2	МО k_3	МО k_4	k_1	k_2	k_3	k_4	k_1	k_2	kr_3	kr_4	
1	1,0	3,0	3,0	1,0	1,533	2,992	1,628	1,849	0,770	7,110	478,138	0,000	139,548
2	1,0	3,0	3,0	1,0	2,033	3,858	2,314	1,964	0,803	6,514	435,127	0,000	138,924
...
49	1,0	3,0	3,0	1,0	1,285	3,620	3,888	1,138	0,429	3,000	3,000	0,000	72,669
50	1,0	3,0	3,0	1,0	1,826	2,712	3,172	0,000	0,429	3,000	3,000	0,000	72,668
51	1,0	3,0	5,0	1,0	0,000	2,687	4,849	0,413	0,429	3,000	3,000	0,000	72,668
...
11250	9,0	11,0	11,0	9,0	7,948	10,815	13,482	10,964	0,376	11,000	11,000	0,013	180,889

Таблица 2

Результаты оценки

№	q_1	q_2	q_3	q_4	q_5	q_{cp}	s_k	$P = 0,80$			$P = 0,95$		
								u_f	$x(u)$	$x^{(0)}$	u_f	$x(u)$	$x^{(0)}$
1	0,5	0,7	0,6	0,6	0,7	0,620	0,084	0,0574	0,5626	0,6774	0,1039	0,5161	0,7239
2	0,1	0,1	0,3	0,2	0,2	0,180	0,084	0,0574	0,1226	0,2374	0,1039	0,0761	0,2839
...
225	0,8	0,7	0,5	0,6	0,6	0,640	0,114	0,0782	0,5618	0,7182	0,1416	0,4984	0,7816

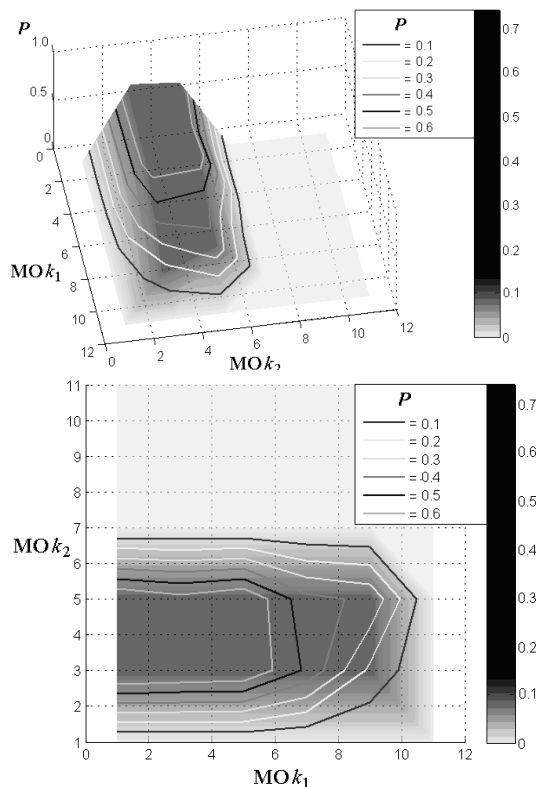


Рис. 2. Вид функции $P(MOk_1, MOk_2)$ при $MOk_3 = 11$ и $MOk_4 = 1$

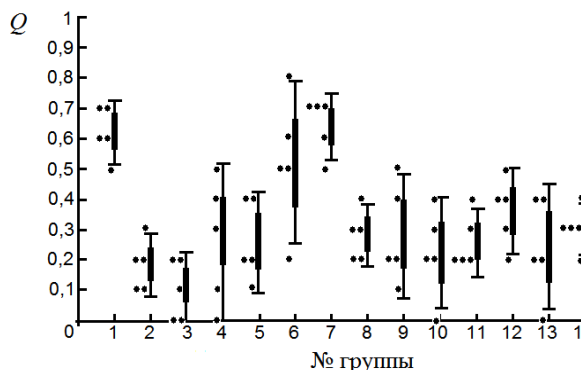


Рис. 3. Графическое представление полученных результатов

3.3. Определение погрешности полученной оценки

Определим погрешность полученной оценки методом доверительных интервалов. Найдем доверительные интервалы, при которых Q накрывается с вероятностями $P_1 = 0,80$ и $P_2 = 0,95$. При этом воспользуемся точным методом построения доверительных интервалов при неизвестной дисперсии [9, 11].

Определение погрешности производится для каждой группы из 50 экспериментов. В каждой такой группе извлекается ряд эквивалентных выборок. Пусть проведено m наблюдений ($0 < m < 50$) величины ζ_0 . В каждом m -м наблюдении вычисляется вероятность отказа Q .

Результаты этих вычислений обозначим через q_1, q_2, \dots, q_n . Будем рассматривать их как выборку из нормально распределенной генеральной совокупности.

Среднее значение этих n измеренных значений q_{cp} является хорошей оценкой математического ожидания.

Общая формула расчета вероятности P , с которой накрывается значение Q в интервале $[q_{cp} - u_f, q_{cp} + u_f]$, имеет вид:

$$P(q_{cp} - u_f \leq Q \leq q_{cp} + u_f) = P. \quad (11)$$

По условию задачи, допустимым приемлемым отклонением u_f для $P = 0,95$ является 0,300; для $P = 0,85$ — $u_f = 0,200$.

Пусть $m = 10$. Тогда из 50 наблюдений получим $n = 50/10 = 5$ значений q . Доверительные границы вычисляются по формуле (4), в данном случае $t(80; 4) = 1,5332$; $t(95; 4) = 2,7764$.

Результаты оценки для $P = 0,80$ и $P = 0,95$ представлены в табл. 2.

Полученные результаты можно представить графически (рис. 3).

На рисунке серыми точками обозначены значения каждой выборки, т.е. значения вероятности отказа q_i , где i — номер группы от 1 до n . Для наглядности эти значения находятся левее значения группы по оси абсцисс. Выделенная часть линии соответствует доверительному интервалу для $P = 0,85$, вся линия — для $P = 0,95$.

ВЫВОДЫ

1. Предложен экспериментальный метод определения вероятности безотказной работы программ, оперирующих исходными данными, законы распределения которых известны. Моделируя работу программ, можно получить области значения входных данных, в которых вероятность безотказной работы программы не менее некоторой заданной величины.

2. Показано применение метода доверительных интервалов для определения погрешности вероятности безотказной работы программ.

3. Применение метода проиллюстрировано на примере прикладной программы, моделирующей решение задачи по определению кинетических параметров химической реакции. Для этой программы получена интервальная оценка определения вероятности отказа Q . Для рассмотренного примера максимальный интервал, в который можно с вероятностью $P=0,80$ гарантировать попадание величины вероятности отказа программы, составил $(Q - 0,1775; Q + 0,1775)$, для $P = 0,95$ — $(Q - 0,3214; Q + 0,3214)$. В связи с большими отклонениями МО вероятности отказа можно сделать вывод

о необходимости увеличения числа экспериментов для получения более точных результатов.

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ Р ИСО/МЭК 9126-93. Оценка программной продукции. Характеристики качества и руководства по их применению. Госстандарт России. М.: Изд-во стандартов, 1994.

2. ГОСТ 28195-89. Оценка качества программных средств. Общие положения. Государственный комитет СССР по стандартам. М.: Изд-во стандартов, 1989.

3. **Губайдуллин И. И., Фрид А. И.** Методика оценки надежности программного обеспечения в зависимости от распределения исходных данных // Естественные и технические науки. 2007. № 3. С. 194–197.

4. **Frid A. I., Gubaydullin I. I.** Reliability Estimation of the Program Solving One Task of Chemical Kinetics // Proc. of 10th International Workshop on Computer Science and Information Technologies (CSIT'2008). 2008. Vol. 1. P. 184–187.

5. **Губайдуллин И. И.** Оценка точности расчета вероятности отказа одной программы // Актуальные проблемы в науке и технике: Сб. ст. третьей всероссийск. зимн. шк.-семинара аспирантов и молодых ученых, Т. 1. Уфа: Диалог, 2008. С. 287–293.

6. **Черкесов Г. Н.** Надежность аппаратно-программных комплексов: Учеб. пособие. СПб.: Питер, 2005.

7. **Коварцев А. Н., Логвинов А. Л.** Стохастическое тестирование программных модулей. Метод независимых испытаний // Телекоммуникации. 2004. № 4.

8. **Лаврищева Е. М., Рожнов А. М.** Концепция аналитической оценки характеристик качества программных компонентов // Пятая международн. науч.-практ. конф. по программированию УкрПрог. Украина, 2004.

9. **Вентцель Е. С.** Теория вероятностей: Учеб. для вузов. М.: Высш. шк., 1999.

10. **Джонсон К.** Численные методы в химии. М.: Мир, 1983. 504 с.

11. **Тойберт П.** Оценка точности результатов измерений. М.: Энергоатомиздат, 1988.

ОБ АВТОРАХ



Губайдуллин Ильдар Ирекович, аспирант. Магистр техники и технологии по информатике и вычисл. технике (УГАТУ, 2007)



Фрид Аркадий Исаакович, проф. каф. вычисл. техники и защиты инф. Дипл. инж.-электромех. (УАИ, 1968). Д-р техн. наук по управ. в техн. системах (УГАТУ, 2000). Иссл. в обл. управ. сложн. сист. в условиях неопределенности.