

Агрегированный нелинейный критерий оценки надежности программного обеспечения

С.И. Носков, И.В. Овсянников, А.П. Медведев

Иркутский государственный университет путей сообщения

Аннотация: В работе рассматривается способ построения нелинейной функции эффективности для оценки надежности программного обеспечения. Предложенный алгоритм основан на использовании информации о значениях критериев надежности, а также некоторых экспертных суждениях. Данный подход существенно отличается от предложенных ранее моделей оценки надежности программного обеспечения, которые основаны на вероятностных характеристиках. В предложенном методе помимо объективной информации учитываются субъективные экспертные оценки, что позволяет более гибко и адекватно оценить надежность программных продуктов.

Ключевые слова: надежность программного обеспечения, вероятностные модели, статистические модели, частные критерии эффективности, линейное программирование, векторная оптимизация, теория принятия решений.

Надежность – важная характеристика в оценке качества программного обеспечения, которая декларирована международной организацией по стандартизации (ISO). С момента появления первых программных продуктов и до настоящего времени разработано множество подходов для оценки их надёжности [1]. Так, начиная с середины 20-го века, начались активные исследования по методам оценки надежности программных систем. Одной из первых была модель, предложенная Дж. Мусой в 1975 году, она представляет собой фундаментальный подход к количественной оценке надёжности программного обеспечения. Опираясь на статистический анализ данных о сбоях, Муса разработал модель, основанную на предположении, что время возникновения ошибок подчиняется экспоненциальному распределению [2]. В работе [3] описана известная модель Джелинского – Моранды, в ней авторы предполагают, что количество ошибок в программном продукте фиксировано, но неизвестно, и что каждая исправленная ошибка увеличивает надежность программы, а вероятность возникновения ошибки уменьшается. Приведенные модели основаны на вероятностном подходе и редко

используются при промышленной разработке. В 1979 году в работе [4] Литтлвуд обратил внимание на то, что многие распространенные представления о надежности программного обеспечения могут быть вводящими в заблуждение или не отражать реального состояния дел, ведь для адекватной оценки надежности необходимо учитывать, как программа используется в реальных условиях.

В настоящее время подходы к оценке надежности несколько изменились, в частности, из-за развития искусственного интеллекта [5] (см., например, модель, основанную на использовании искусственных нейронных сетей [6]), появления статистических методов оценки [7]. В работе [8] предложен подход, основанный на функциональном подходе к оценке надежности программного обеспечения.

В данной работе предлагается новый подход к оценке надежности программных продуктов, основанный на алгоритме, описанном в работе [9]. Для решения задачи необходимо использовать информацию о значениях частных критериев надежности программного обеспечения, а также специальным образом сформированную экспертную информацию в виде матрицы парных сравнений оцениваемых программ между собой. Сформулированная задача сводится к задаче линейного программирования (ЛП), что позволяет, в частности, учитывать случаи противоречивых экспертных высказываний, используя приемы, описанные в [10].

Ранее авторами разработан программный комплекс [11], позволяющий вычислять коэффициенты линейной свертки частных критериев эффективности в автоматическом режиме. Так как в данной работе используется нелинейный вид функции эффективности, необходимо было предусмотреть усовершенствование этого комплекса. Новая его версия должна предполагать использование широкого набора арифметических

преобразований (логарифмических, показательных, степенных) для каждого локального критерия, который бы обеспечивал максимальную разрешающую способность системы ограничений сформированной задачи линейно-булева программирования (ЛБП), а в случае выявления ее несовместности – находить квазирешение задачи и степень противоречивости экспертных высказываний.

Существует большое количество численных характеристик, по которым можно оценить надежность программного обеспечения (см. например, [12]). Для построения функции эффективности в настоящей работе предлагается использовать следующие частные критерии надежности, представляющие собой метрики, предложенные Чидамбером и Кемерером [13]:

1. WMC (Weighted Methods per Class) – суммарная сложность всех методов данного класса.
2. DIT (Depth of Inheritance Tree) – глубина дерева наследования.
3. NOC (Number of Children) – количество непосредственных наследников.
4. CBO (Coupling Between Object Classes) – количество классов, с которым непосредственно связан исходный.
5. RFC (Response For a Class) – количество методов, которые могут быть вызваны методами исходного класса.
6. LCOM (Lack of Cohesion in Methods) – недостаток сцепления методов – количество методов, не имеющих общих переменных, за исключением тех, где общие переменные присутствуют.

При использовании предложенной методики могут быть выбраны и другие критерии, которые удовлетворяют основным требованиям – критерий должен измеряться действительным числом, все критерии должны быть однородны. По указанной выше причине необходимо было существенно расширить функциональность разработанной ранее программы.

Для этого была создана новая, более технологичная программа, главное окно которой представлено на рис. 1. Теперь в меню программы вложена вкладка «Функции преобразования», позволяющее назначить широкий набор преобразований для локальных критериев эффективности. Также в меню программы встроены вкладки «Частные критерии» и «Экспертные оценки», позволяющие просматривать и изменять загруженные данные без необходимости повторной загрузки. В поле ввода «Минимальный вклад каждого критерия» учтена возможность задания минимального процентного вклада каждого критерия в финальную функцию эффективности.

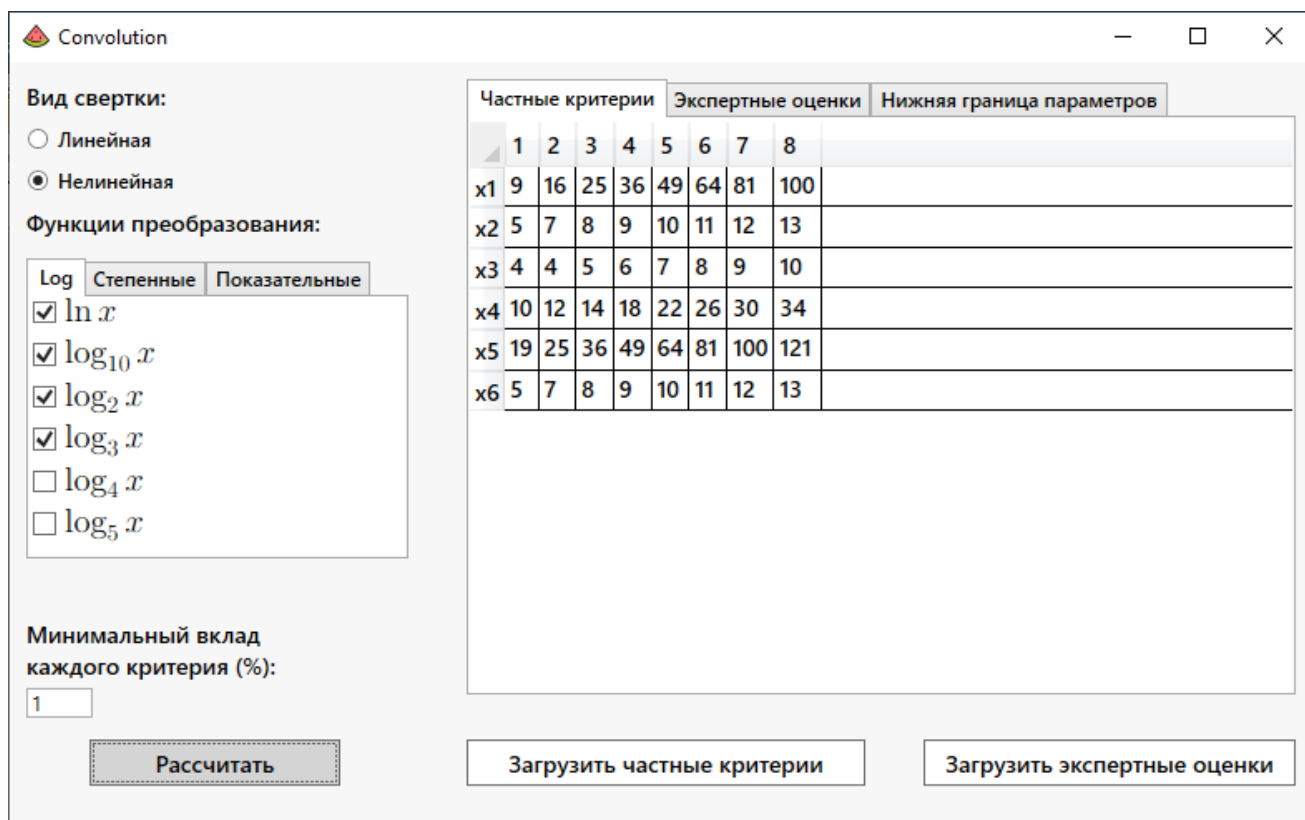


Рис. 1. – Главное окно программы

Для демонстрации работы разработанного программного комплекса и процесса построения функции эффективности сделаем следующее допущение: вместо оценки различных программ между собой сравним несколько версий одной программы.

Рассмотрим программный продукт, который в течение своего жизненного цикла претерпел изменения от версии 1.0 до версии 1.7. Каждая версия характеризуется следующими числовыми характеристиками (табл.1). К работе были привлечены квалифицированные эксперты, которые, основываясь на своем опыте в оценке надежности программного обеспечения, сформировали матрицу парных сравнений $A = \|a_{ij}\|, i, j = \overline{1,8}$ надежности версий программы между собой (табл. 2) по правилу:

$$a_{ij} = \begin{cases} 1, & \text{если } i - \text{ая версия надежнее } j - \text{ой} \\ -1, & \text{если } j - \text{ая версия надежнее } i - \text{ой} \\ 0, & \text{если надежность } i - \text{ой и } j - \text{ой равны равны.} \end{cases}$$

Результат выполнения расчетов по программе представлен на рис. 2 с тремя таблицами: таблица с коэффициентами нелинейной свертки, полученными в результате решения задачи ЛБП с выбранными преобразующими функциями, таблица с указанием степени несовместности, которая возникла при сравнении равных по надежности версий программы, т.е., когда $a_{ij} = 0$ и таблица со значениями несовместности пар версий программ, где первая версия из пары лучше, чем вторая, при этом $a_{ij} = 1$.

UnlinearResult

Вид свёртки: нелинейная

№	Коэф. свертки	Ф-ция
1	0,1924	$\log_2 x_1$
2	1,1123	$\log_{10} x_2$
3	0,3937	$\sqrt{x_3}$
4	0,2350	$\log_2 x_4$
5	0,1755	$\log_2 x_5$
6	25,3558	$\log_2 x_6$

Степень несовместности: Сохранить результат

Пара	Несовместность
(2, 1)	12,7620
(3, 1)	18,0725
(4, 1)	22,7865
(5, 1)	26,9901
(6, 1)	30,7850
(7, 1)	34,2448
(8, 1)	37,4246
(4, 2)	10,0246
(5, 2)	14,2282
(6, 2)	18,0230
(7, 2)	21,4828
(8, 2)	24,6626
(5, 3)	8,9176
(6, 3)	12,7125
(7, 3)	16,1722
(8, 3)	19,3520
(4, 6)	7,9985
(7, 4)	11,4582
(8, 4)	14,6380
(7, 5)	7,2547

Пара	Несовместность
(2, 3)	5,3106
(3, 4)	4,7140
(4, 5)	4,2036
(5, 6)	3,7949
(6, 7)	3,4598
(7, 8)	3,1798

Рис. 2. – Результаты расчетов по программе

Таблица № 1

Значения частных критериев надежности

Критерий	v.1.0	v. 1.1	v. 1.2	v. 1.3	v. 1.4	v. 1.5	v. 1.6	v. 1.7
WMC	9	16	25	36	49	64	81	100
DIT	5	7	8	9	10	11	12	13
NOC	4	4	5	6	7	8	9	10
CBO	10	12	14	18	22	26	30	34
RFC	19	25	36	49	64	81	100	121
LCOM	5	7	8	9	10	11	12	13

Таблица № 2

Матрица парных сравнений

	v. 1.0	v. 1.1	v. 1.2	v. 1.3	v. 1.4	v. 1.5	v. 1.6	v. 1.7
v. 1.0	0	-1	-1	-1	-1	-1	-1	-1
v. 1.1	1	0	0	-1	-1	-1	-1	-1
v. 1.2	1	0	0	0	-1	-1	-1	-1
v. 1.3	1	1	0	0	0	-1	-1	-1
v. 1.4	1	1	1	0	0	0	-1	-1
v. 1.5	1	1	1	1	0	0	0	-1
v. 1.6	1	1	1	1	1	0	0	0
v. 1.7	1	1	1	1	1	1	0	0

Итоговая функция эффективности приобрела вид:

$$y = 0.1924 \log_2 x_1 + 1.1123 \log_{10} x_2 + 0.3937 \sqrt{x_3} + 0.235 \log_2 x_4 + 0.1755 \log_2 x_5 + 25.3558 \log_2 x_6. \quad (1)$$

Таким образом, наиболее значимым критерием при оценке надежности оказался LCOM. Как следует из полученных показателей надежности (таблица 3), наиболее надежной следует признать версию 1.7, а наименее – версию 1.0. Все версии упорядочены по возрастанию агрегированного показателя надежности. Таким образом, с каждой новой версией программы ее надежность повышается.

Таблица № 3

Оценки надежности

Версия программы	Агрегированный показатель надежности
------------------	--------------------------------------

v. 1.0	62.58
v. 1.1	73.33
v. 1.2	80.65
v. 1.3	85.36
v. 1.4	89.57
v. 1.5	93.36
v. 1.6	96.82
v. 1.7	99.99

Важно отметить, что полученные результаты отражают в себе субъективные оценки экспертов, которые в данном случае оказались противоречивыми. Такая ситуация является скорее правилом, чем исключением.

Приведем некоторые выявленные противоречивые экспертные высказывания и степень этой противоречивости (табл. 4).

Таблица № 4

Степень противоречивости экспертных оценок

Пары при $a_{ij} = 0$	Степень противоречивости	Пары при $a_{ij} = 1$	Степень противоречивости
(2, 1)	12,7620	(2, 3)	5,3106
(3, 1)	18,0725	(3, 4)	4,7140
(4, 1)	22,7865	(4, 5)	4,2036
(5, 1)	26,9901	(5, 6)	3,7949
(6, 1)	30,7850		

(7, 1)	34,2448		
(8, 1)	37,4246		
(4, 2)	10,0246		
(5, 2)	14,2282		
(6, 2)	18,0230		
(7, 2)	21,4828		
(8, 2)	24,6626		
(5, 3)	8,9176		
(6, 3)	12,7125		
(7, 3)	16,1722		

Полученная функция эффективности для оценки надежности программного обеспечения может быть успешно использована для сравнения надежности других, уже реальных, программ.

Литература

1. Чернов, А. В., Паращенко И.Г. Классификация моделей надежности программного обеспечения // Инженерный вестник Дона, 2012, № 4-2. URL: ivdon.ru/ru/magazine/archive/n4p2y2012/1319/.
2. Musa, J.D. A theory of software reliability and its application, IEEE Transactions on Software Engineering, 1975, Vol. SE-1, No. 3, pp. 312-327.
3. Jelinski Z., Moranda P., "Software reliability research", Statistical Computer Performance Evaluation, 1972, p. 465-484. URL: doi.org/10.1016/b978-0-12-266950-7.50028-1/.
4. Littlewood B., "How to measure software reliability and how not to", IEEE Transactions on Reliability, 1979, vol. R-28, no. 2, pp. 103-110. URL: doi.org/10.1109/tr.1979.5220510/.
5. Sheptunov S.A., Larionov M.V., Suhanova N.V., Salakhov M.R., Solomentsev Y.M., Kabak I.S., "Simulating reliability of the robotic system

software on the basis of artificial intelligence," 2016 IEEE Conference on Quality Management, Transport and Information Security, Information Technologies (IT&MQ&IS), Nalchik, Russia, 2016, pp. 193-197. DOI: 10.1109/ITMQIS.2016.7751956.

6. Bhuyan M., Mohapatra D., Sethi S., "Software reliability prediction using fuzzy min-max algorithm and recurrent neural network approach", International Journal of Electrical and Computer Engineering (IJECE), 2016, vol. 6, no. 4, p. 1929. URL: doi.org/10.11591/ijece.v6i4.pp1929-1938

7. Bal P., Jena N., Mohapatra D., "Software reliability prediction based on ensemble models", Proceeding of International Conference on Intelligent Communication, Control and Devices, 2016, pp. 895-902. URL: doi.org/10.1007/978-981-10-1708-7_105/.

8. Степович-Цветкова, Г. С. Оценка надежности программного обеспечения посредством применения функционального подхода // Инженерный вестник Дона, 2015, № 3. URL: ivdon.ru/ru/magazine/archive/n3y2015/3206/.

9. Носков, С. И. Построение нелинейной свертки частных критериев эффективности в задаче векторной оптимизации // Вестник ВГУ. Серия: Системный анализ и информационные технологии, (4), 30-36. 2021.

10. Носков С.И. Технология моделирования объектов с нестабильным функционированием и неопределенностью в данных. Иркутск. Облформпечать. 1996. 320 с.

11. Носков С.И., Овсянников И.В. Свидетельство о регистрации № 2024615237. Программа построения линейной функции эффективности в задаче многокритериальной оптимизации // 2024. URL: fips.ru/EGD/6960a8d8-0ea4-4621-b2b0-d30a4630f6ff/.

12. Haindl P., Plösch R., "Value- oriented quality metrics in software development: practical relevance from a software engineering perspective", IET Software, 2021, vol. 16, no. 2, pp. 167-184. URL: doi.org/10.1049/sfw2.12051

13. Chidamber, S. R. A Metrics Suite for Object Oriented Design / S. R. Chidamber, C. F. Kemerer // IEEE Transactions on Software Engineering. – 1994. – V. 20. – №6. – pp. 476–493.

References

1. A. V. Chernov, I. G. Parashchenko, Inzhenernyj vestnik Dona, 2012, № 4-2. URL: ivdon.ru/ru/magazine/archive/n4p2y2012/1319/.

2. J.D. Musa, IEEE Transactions on Software Engineering, 1975, Vol. SE-1, No. 3, pp. 312-327.

3. Z. Jelinski and P. Moranda, Statistical Computer Performance Evaluation, 1972, pp. 465-484. URL: doi.org/10.1016/b978-0-12-266950-7.50028-1/.

4. B. Littlewood, IEEE Transactions on Reliability, 1979, vol. R-28, no. 2, pp. 103-110. URL: doi.org/10.1109/tr.1979.5220510/.

5. S. A. Sheptunov, M. V. Larionov, N. V. Suhanova, M. R. Salakhov, Y. M. Solomentsev, I. S. Kabak, 2016 IEEE Conference on Quality Management, Transport and Information Security, Information Technologies (IT&MQ&IS), Nalchik, Russia, 2016, pp. 193-197. DOI: [10.1109/ITMQIS.2016.7751956/](https://doi.org/10.1109/ITMQIS.2016.7751956/).

6. M. Bhuyan, D. Mohapatra, S. Sethi, International Journal of Electrical and Computer Engineering (IJECE), 2016, vol. 6, no. 4, p. 1929. URL: doi.org/10.11591/ijece.v6i4.pp1929-1938/.

7. P. Bal, N. Jena, D. Mohapatra, Proceeding of International Conference on Intelligent Communication, Control and Devices, 2016, pp. 895-902. URL: doi.org/10.1007/978-981-10-1708-7_105/.

8. G. S. Stepovich-Tsvetkova, Inzhenernyj vestnik Dona, 2015, № 3, URL: ivdon.ru/ru/magazine/archive/n3y2015/3206/.



9. S.I. Noskov, I.V. Ovsyannikov, Vestnik VGU. Seriya: Sistemnyy analiz i informatsionn·yye tekhnologii(4), pp. 30-36. 2021.
10. S.I Noskov, Tekhnologiya modelirovaniya ob"yektov s nestabil'nyy funktsionirovaniyem i neopredelennost'yu v dannykh [Technology for modeling objects with unstable functioning and data uncertainty], Irkutsk. Oblinformpechat, 1996. 320 p.
11. S. I. Noskov, I. V. Ovsyannikov, Svidetel'stvo № 2024615237 [Certificate No. 2024615237]. Programma postroyeniya lineynoy funktsii effektivnosti v zadache mnogokriterial'noy optimizatsii [Software for implementing a linear convolution function in vector optimization problems], 2024. URL: fips.ru/EGD/6960a8d8-0ea4-4621-b2b0-d30a4630f6ff/.
12. P. Haindl, R. Plösch, IET Software, 2021, vol. 16, no. 2, pp. 167-184. URL: doi.org/10.1049/sfw2.12051/.
13. S. R. Chidamber, C. F. Kemerer, IEEE Transactions on Software Engineering, 1994, V. 20, №6, pp. 476–493.

Дата поступления: 14.03.2024

Дата публикации: 22.04.2024