УДК 004.056.53 DOI: 10.18698/2308-6033-2018-6-1771

Метод оценки показателя надежности программного обеспечения автоматизированной системы подготовки данных управления летательными аппаратами

© А.Г. Андреев[1], Г.В. Казаков[1], В.В. Корянов[2]

[1]ФГБУ «4 ЦНИИ» Министерства обороны России,
г. Королев, Московская обл., 141091, Россия

[2]МГТУ им. Н.Э. Баумана, Москва, 105005, Россия

Программное обеспечение — наиболее сложный компонент автоматизированной системы подготовки данных управления летательными аппаратами. Вероятной причиной прерывания процесса подготовки данных является наличие ошибок в программном обеспечении, для устранения которых может потребоваться продолжительное время. Заблаговременное устранение этих ошибок представляет собой исключительно важную задачу. Степень устранения ошибок определяется значением показателя надежности. Проблема оценки показателя надежности программного обеспечения до настоящего времени актуальна, поскольку общепринятой методики оценки этого показателя не существует. На основании результатов анализа существующих моделей надежности программного обеспечения показано, что ни одна из них не может быть использована для оценки показателя надежности программного обеспечения автоматизированной системы подготовки данных. Приведено доказательство корректности использования методов теории вероятностей для оценки рассматриваемого показателя. Доказана теорема о том, что между множеством вариантов входных данных и множеством вычислительных траекторий существует биекция (взаимно однозначное соответствие). Из теоремы следует возможность применения метода теории вероятностей при геометрической интерпретации вероятности появления события (ошибки) для оценки показателя надежности программного обеспечения. Использование этого метода является корректным, но требует бесконечных временны́х ресурсов, что делает его непригодным для практического применения.

Предложен принципиально иной метод оценки показателя надежности программного обеспечения, при использовании которого необходимо иметь документальное свидетельство уровня покрытия тестовыми вариантами всей области входных данных из допустимой области. Достоинство предлагаемого метода заключается в том, что он не требует каких-либо предположений, а исходные данные для оценки показателя надежности программного обеспечения имеют ясный физический смысл и могут быть получены на практике.

Ключевые слова: *вычислительная ветвь, вычислительная траектория, надежность, ошибка, программное обеспечение, эталон-результат*

Введение. Для заблаговременного устранения ошибок, допущенных в программном обеспечении (ПО) автоматизированной системы подготовки данных (АСПД), чрезвычайно важна оценка показателя надежности ПО. К сожалению, до настоящего времени общепринятой методики оценки этого показателя не существует.

Формальная модель программы. Угрозы информационной безопасности в значительной степени обусловлены возможностью воздействия на ПО. Корректность использования метода оценки влияния случайной или преднамеренной угрозы на правильность работы программы основывается на формальном представлении процесса функционирования и структуры программы.

Формализацию программного обеспечения АСПД представим в виде орграфа Г, размеченного над алфавитами B и U, где $B = B_1 \bigcup B_2$; $B_1 \bigcap B_2 = \varnothing$, здесь B_1 — операторный подалфавит, символами которого являются операторы; B_2 — логический подалфавит, символами которого являются логические символы; B — базис; U — алфавит, $U = \{0, 1\}$, где 0 — условие выполнено; 1 — условие не выполнено.

На множестве вершин графа Г определим отображение, сопоставляющее каждому оператору P_{1n} символ из подалфавита B_1, а каждому распознавателю P_{2m} — символ из подалфавита B_2. На множестве дуг графа Г определим отображение, сопоставляющее каждой дуге, выходящей из распознавателя, символ из алфавита U. Семантика σ базиса B определяется следующей совокупностью:

- множества Θ_σ состояний ξ_j программы;
- соответствия σ_y: $\Theta_\sigma \to \Theta_\sigma$, $y \in B_1$;
- соответствия σ_p: $\Theta_\sigma \to U$, $p \in B_2$.

Множество всех графов, размеченных над алфавитами B и U, обозначим Г*. Тогда пара (Γ, σ), где $\Gamma \in \Gamma^*$, является формальной моделью программы [1].

Если движение по графу Г завершается на его выходе, то программа (Γ, σ) останавливается на данном состоянии $\xi \in \Theta_\sigma$ с выходом $F(\Gamma, \sigma, \xi)$; в противном случае выход программы считается неопределенным.

Программа выдает правильный результат, если выполняется условие

$$\mathrm{abs}\left\{F^{\mathrm{тр}}(\Gamma, \sigma, \xi) - F(\Gamma, \sigma, \xi)\right\} \le \varepsilon, \qquad (1)$$

где abs — символ абсолютной величины; $F^{\mathrm{тр}}(\Gamma, \sigma, \xi)$ — требуемый выход программы (эталон-результат); ε — допустимое отклонение полученного результата от требуемого.

Программа выдает неправильный результат (или результат не выдается), если выполнены условия

$$(\mathrm{abs}\left\{F^{\mathrm{тр}}(\Gamma, \sigma, \xi) - F(\Gamma, \sigma, \xi)\right\} > \varepsilon). \qquad (2)$$

Пример формализованного представления программы в виде орграфа, размеченного над алфавитами B и U, показан на рис. 1.

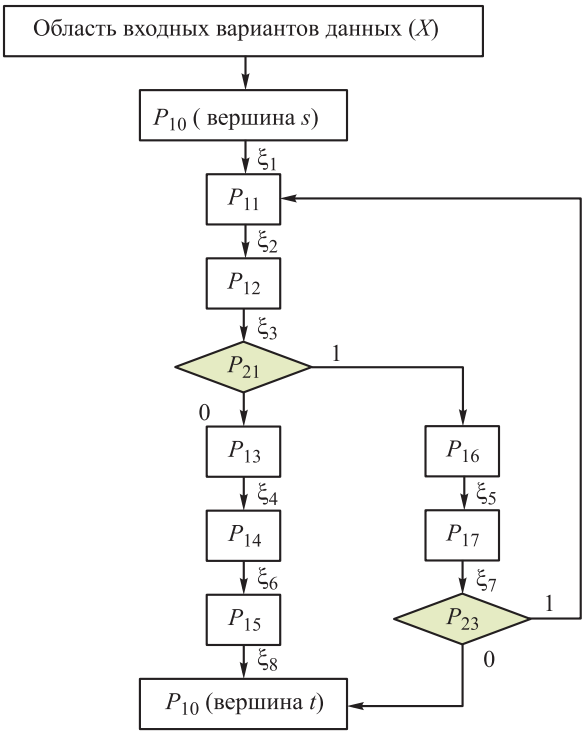

Рис. 1. Пример представления программы в виде орграфа Г

Представление структуры программы в виде орграфа, размеченного над алфавитами B и U, позволяет рассматривать ее в виде некоторой конечной совокупности подструктур. Действительно, как отмечено в работе [2], всякая программа П представима в виде последовательности операторов для любого варианта x_i входных данных:

$$\Pi(x_i) = P_1 \circ P_2 \circ ... \circ P_n \circ ... \circ P_N,$$

где P_n — операторы программы, $n \in \overline{1, N} = (1, 2, ..., N)$; N — множество всех вершин графа Г; «$\circ$» — символ вида соединения операторов в программе в соответствии с логикой ее функционирования σ.

При получении выражения для оценки показателя надежности ПО АСПД с учетом воздействия случайных угроз в виде допущенных в ПО ошибок наиболее целесообразно использовать геометрическую интерпретацию вероятности проявления ошибки в программе, поскольку в этом случае в принципе можно получить точную оценку показателя его надежности.

Введем необходимые определения.

Обозначим X множество входных вариантов данных, для преобразования которых программа (Γ, σ) имеет смысл. Введем определение допустимого множества входных данных для программы (Γ, σ).

Определение 1. Допустимым множеством для программы (Γ, σ) является такое подмножество $X_{\text{доп}} \subseteq X$ множества X, для каждого элемента которого отображение $F(\Gamma, \sigma, \xi)$ является всюду определенным, т. е. программа остановится на некотором состоянии $\xi \in \Theta_\sigma$ с выходом $Y = F(\Gamma, \sigma, \xi)$.

В общем случае один и тот же оператор программы P_n в процессе получения выходной величины для некоторого варианта входных данных $x_i \in X_{\text{доп}}$ может использоваться несколько раз. В символах P_n введем верхний индекс (индекс вхождения), который определяет, сколько раз использовался этот оператор для некоторого варианта входных данных $x_i \in X_{\text{доп}}$ (0 — не использовался, (l) — использовался l-й раз, $(+l)$ — повторное использование оператора l раз подряд). Формальную запись последовательности выполнения операторов программы для варианта входных данных $x_i \in X_{\text{доп}}$ обозначим B_σ^{xi}:

$$B_\sigma^{xi} = \pi_{xi}(P_n) = (P_1^{(1)} \circ P_2^{(1)} \circ \ldots \circ P_2^{(l)} \circ \ldots \circ P_n^{(+l)} \circ P_{n+1}^{(1)} \circ \ldots \circ P_N^{(1)}),$$

где $\pi_{xi}(\bullet)$ — операция упорядочения последовательности выполнения операторов P_n программы; N — число различных номеров операторов P_n, выполняемых при движении по графу от вершины s к вершине t, которое определяется семантикой программы σ при входном варианте данных x_i.

Граф Г представлен в виде упорядоченной пары $< B, V >$, где B — непустое множество вершин, а V — множество упорядоченных пар $\{(a, b)\} \subseteq B \times B$ дуг графа [3].

Доказательство возможности применения геометрической интерпретации вероятности для оценки показателя надежности ПО. Используя представление программы в виде орграфа, размеченного над алфавитами B и U, дадим определения подструктуры программы, вычислительной ветви программы (ВВП) и вычислительной траектории программы (ВТП).

Определение 2. Под структурой программы $\Pi_i = \Pi (x_i \in N_x)$, порожденной некоторым множеством N_x вариантов входных данных $N_x \in X_{\text{доп}}$ из допустимой области, называется частичный орграф $H_{xi} = (B_{xi}, V_{xi})$, $B_{xi} \subseteq B$, $V_{xi} \subseteq V$, содержащий путь s–t, вершинами которого служат операторы P_n $(x_i \in N_x \subseteq X_{\text{доп}})$.

Определение 3. Вычислительной ветвью программы называется одна и та же конечная последовательность B_σ^x операторов подструктуры программы H_{xi}, порожденная хотя бы одним вариантом входных данных $x_i \in X_{\text{доп}}$ $(i = \overline{1, I})$ из допустимой области.

Определение 4. Вычислительной траекторией B_σ^{xi} программы называется конечная последовательность

$$B_{\sigma}^{xi} = \pi_{xi}(P_n) = (P_1^{(1)} \circ \ldots \circ P_n^{(m)} \circ \ldots \circ P_{n+l+1}^{(+l)} \circ \ldots \circ P_N^{(k)})$$

операторов подструктуры программы H_{xi}, порожденная ровно одним вариантом входных данных x_i из допустимой области $X_{\text{доп}}$ ($x_i \in X_{\text{доп}}$).

Докажем, что к оценке показателя надежности программы в виде вероятности того, что программа выдаст правильный результат — см. (1) при заданном варианте входных данных из допустимой области, применима геометрическая интерпретация.

Введем следующие подмножества: подмножество ВТП B_{σ}^{xi}, для которых программа выдает правильный результат (1), и подмножество ВТП B_{σ}^{yj}, для которых программа выдает неправильный результат (2). Множество всех подмножеств B_{σ}^{xi} обозначим $\left\{B_{\sigma}^{xi}\right\}$, а множество всех подмножеств B_{σ}^{yj} — $\left\{B_{\sigma}^{yj}\right\}$. Множество всех ВТП программы обозначим $\left\{B_{\sigma}^{X\upsilon}\right\} = \left\{B_{\sigma}^{xi}\right\} \bigcup \left\{B_{\sigma}^{yj}\right\}$.

Для доказательства корректности применения геометрической интерпретации вероятности с целью оценки показателя надежности программы предварительно сформулируем и докажем следующие две леммы.

Лемма 1. Если заданы разбиение множества $X_{\text{доп}}$ на конечное число подмножеств:

$$X_{\text{доп}} = \left\{X_{(\text{доп})1}, X_{(\text{доп})2}, \ldots, X_{(\text{доп})L}\right\}$$

и бинарное отношение A_{φ} — «иметь правильный результат», определенное на множестве выходов программы, порождаемых множеством ВТП, то A_{φ} является отношением эквивалентности.

Доказательство. Поскольку объединение всех классов разбиения совпадает с множеством $X_{\text{доп}}$, то любой элемент x_i, порождающий вычислительную траекторию B_{σ}^{xi}, входит в некоторый класс подмножеств $X_{(\text{доп})i}$ (i = 1, 2, …, l, …, L). Следовательно, имеет место отношение $x_i A_{\varphi} x_i$, т. е. отношение A_{φ} рефлексивно.

Если разные элементы x_i и x_j, порождающие ВТП B_{σ}^{xi} и B_{σ}^{xj}, входят в один и тот же класс подмножеств $X_{(\text{доп})i}$, то элементы x_j и x_i входят в этот же класс, т. е. из отношения $x_i A_{\varphi} x_j$ вытекает, что имеет место и отношение $x_j A_{\varphi} x_i$, откуда следует симметричность отношения A_{φ}.

Пусть выполнены отношения $x_i A_{\varphi} x_j$ и $x_j A_{\varphi} x_z$. Это означает, что для всех трех входных вариантов данных x_i, x_j, x_z, порождающих вычислительные траектории B_{σ}^{xi}, B_{σ}^{xj}, B_{σ}^{xz}, программа выдаст правиль-

ный результат. Следовательно, все они принадлежат одному и тому же подмножеству $X_{(\text{доп})i}$ и имеют место отношения $x_i A_\varphi x_j$, $x_j A_\varphi x_z$ и $x_i A_\varphi x_z$, т. е. отношение A_φ транзитивно.

Поскольку показано, что отношение A_φ рефлексивно, симметрично и транзитивно, то оно, по определению, является отношением эквивалентности, что и требовалось доказать.

Лемма 2. Пусть дано сюръективное отображение φ: $X_{\text{доп}} \to \left\{B_\sigma^{X_\upsilon}\right\}$ множества $X_{\text{доп}}$ на множество $\left\{B_\sigma^{X_\upsilon}\right\}$. Тогда для любых ВТП, для которых $\varphi(x_i) = \xi$ и $\varphi(y_j) = \eta$:

либо

$$(X_{\text{доп}})_\xi \bigcap (X_{\text{доп}})_\eta = \varnothing,$$

либо

$$(X_{\text{доп}})_\xi = (X_{\text{доп}})_\eta.$$

Здесь $(X_{\text{доп}})_\xi$ — множество всех элементов $x_i \in X_{\text{доп}}$, имеющих образ $\xi = \varphi(x_i)$; $(X_{\text{доп}})_\eta$ — множество всех элементов $y_j \in X_{\text{доп}}$, имеющих образ $\eta = \varphi(y_j)$.

Доказательство. Доказательство проведем от противного, т. е. допустим, что пересечение множеств $(X_{\text{доп}})_\xi \bigcap (X_{\text{доп}})_\eta \neq \varnothing$. Выберем какую-либо ВТП $B_\sigma^z(z)$, порожденную элементом z, принадлежащую пересечению множеств $B_\sigma^z(z) \in [(X_{\text{доп}})_\xi \bigcap (X_{\text{доп}})_\eta]$. Тогда выполнены отображения $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^z(z)$ и $B_\sigma^{yj}(y_j) A_\varphi B_\sigma^z(z)$ по определению множеств $(X_{\text{доп}})_\xi$ и $(X_{\text{доп}})_\eta$. Согласно лемме 1, из свойства симметричности отображения A_φ имеем отображение $B_\sigma^z(z) A_\varphi B_\sigma^{yj}(y_j)$, а из свойства транзитивности этого отображения имеем, что из отображений $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^z(z)$ и $B_\sigma^z(z) A_\varphi B_\sigma^{yj}(y_j)$ следует $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^{yj}(y_j)$.

Выберем произвольную ВТП $B_\sigma^\omega(\omega)$, порожденную элементом ω, $B_\sigma^\omega(\omega) \in (X_{\text{доп}})_\eta$. По определению отношения A_φ имеем $B_\sigma^{yj}(y_j) A_\varphi B_\sigma^\omega(\omega)$. Но из отображений $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^{yj}(y_j)$ и $B_\sigma^{yj}(y_j) A_\varphi B_\sigma^\omega(\omega)$ следует $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^\omega(\omega)$, т. е. $B_\sigma^\omega(\omega) \in (X_{\text{доп}})_\xi$. Итак, получено условие

$$(X_{\text{доп}})_\eta \subseteq (X_{\text{доп}})_\xi. \qquad (3)$$

Выберем произвольную ВТП $B_\sigma^\chi(\chi)$, порожденную элементом χ, $B_\sigma^\chi(\chi) \in (X_{\text{доп}})_\xi$. Для этого элемента выполнено отображение $B_\sigma^{xi}(x_i) A_\varphi B_\sigma^\chi(\chi)$. Из леммы 1 следует, что отношение A_φ является сим-

метричным и справедливо отображение $B_{\sigma}^{yj}(y_j)A_{\varphi}B_{\sigma}^{xi}(x_i)$. Но из отображений $B_{\sigma}^{yj}(y_j)A_{\varphi}B_{\sigma}^{xi}(x_i)$ и $B_{\sigma}^{xi}(x_i)A_{\varphi}B_{\sigma}^{\chi}(\chi)$ вытекает $B_{\sigma}^{yj}(y_j)A_{\varphi}B_{\sigma}^{\chi}(\chi)$. Значит, $B_{\sigma}^{\chi}(\chi)\in(X_{\text{доп}})_{\eta}$ и имеет место условие

$$(X_{\text{доп}})_{\xi}\subseteq(X_{\text{доп}})_{\eta}. \qquad (4)$$

Поскольку условия (3) и (4) выполняются одновременно, то $(X_{\text{доп}})_{\xi}\subseteq(X_{\text{доп}})_{\eta}$, что и требовалось доказать.

Теперь можно сформулировать и доказать следующую теорему.

Теорема. Пусть имеет место сюръективное отображение φ: $X_{\text{доп}}\to\left\{B_{\sigma}^{X\upsilon}\right\}$. Тогда существует отношение эквивалентности A_{φ} на множестве $X_{\text{доп}}$, такое, что между фактор-множеством $X_{\text{доп}}/A_{\varphi}$ множества $X_{\text{доп}}$ по отношению A_{φ} и множеством всех ВТП программы $\left\{B_{\sigma}^{X\upsilon}\right\}$ существует биекция.

Доказательство. Пусть на множестве $X_{\text{доп}}$ введено отношение A_{φ} «иметь правильный результат». Следовательно, имеет место отображение $B_{\sigma}^{xi}A_{\varphi}B_{\sigma}^{xj}$ для двух разных вычислительных траекторий B_{σ}^{xi} и B_{σ}^{xj}, если $\varphi(B_{\sigma}^{xi})=\varphi(B_{\sigma}^{xj})$.

Обозначим как $(X_{\text{доп}})_{\xi}$ множество всех элементов x_i, порождающих ВТП $B_{\sigma}^{xi}(x_i)\in X_{\text{доп}}$, которые имеют образ $\xi=\varphi(B_{\sigma}^{xi})$, а как $(X_{\text{доп}})_{\eta}$ — множество всех элементов y_j, порождающих ВТП $B_{\sigma}^{yj}(y_j)\in X_{\text{доп}}$, которые имеют образ $\eta=\varphi(B_{\sigma}^{yj})$.

Множество всех подмножеств $\{(X_{\text{доп}})_{\xi}\}$ и $\{(X_{\text{доп}})_{\eta}\}$ покрывает множество $X_{\text{доп}}$, т. е. $\{(X_{\text{доп}})_{\lambda}\}=\bigcup\limits_{\lambda\in\{B_{\sigma}^{X\upsilon}\}}(X_{\text{доп}})_{\lambda}=X_{\text{доп}}$, $(\lambda=\xi\cup\eta),)$ так как любая ВТП для варианта входных данных из множества $X_{\text{доп}}$, в силу сюръекции, имеет образ.

Из леммы 2 следует, что при разных элементах $\varphi(B_{\sigma}^{xi})=\xi$ и $\varphi(B_{\sigma}^{yj})=\eta$ имеет место условие $(X_{\text{доп}})_{\xi}\cap(X_{\text{доп}})_{\eta}=\varnothing$.

Поскольку отображение φ сюръективно, то $(X_{\text{доп}})_{\xi}\neq\varnothing$ для любой ВТП $B_{\sigma}^{xi}(x_i)$, для которой $\xi=\varphi\{B_{\sigma}^{xi}(x_i)\}$.

Итак, множества $(X_{\text{доп}})_{\xi}$ образуют разбиение множества $X_{\text{доп}}$, а отношение A_{φ} есть эквивалентность, соответствующая этому разбиению. Из леммы 1 следует, что отображение $B_{\sigma}^{xi}(x_i)A_{\varphi}B_{\sigma}^{yj}(y_j)$ имеет место тогда и только тогда, когда ВТП $B_{\sigma}^{xi}(x_i)$ и $B_{\sigma}^{yj}(y_j)$ принадлежат общему множеству $(X_{\text{доп}})_{\xi}$. Отсюда следует, что между множеством

$$\left\{(X_{\text{доп}})_\lambda\right\} = \left\{(X_{\text{доп}})_\xi\right\} \cup \left\{(X_{\text{доп}})_\eta\right\}$$

и подмножествами $\left\{B_\sigma^{xi}\right\}$ и $\left\{B_\sigma^{yj}\right\}$ всех вычислительных траекторий $\left\{B_\sigma^{X\upsilon}\right\} = \left\{B_\sigma^{xi}\right\} \cup \left\{B_\sigma^{yj}\right\}$ существует биективное соответствие:

$$\psi: \quad \left\{(X_{\text{доп}})_\lambda\right\} \to \left\{B_\sigma^{X\upsilon}\right\}. \tag{5}$$

Теорема доказана.

Биективное отношение (5) определяет мозаичную модель входной области программы (Г, σ), представленную на рис. 2. Термин «мозаичная модель надежности ПО» введен Б.П. Пальчуном. В соответствии с этой моделью можно считать, что входные данные и порождаемые ими в силу соответствия (5) подмножества вычислительных траекторий $\left\{B_\sigma^{yj}\right\}$, для которых программа выдает неправильный результат, образуют некоторый «объем» в многомерном пространстве допустимых входных данных $X_{\text{доп}}$.

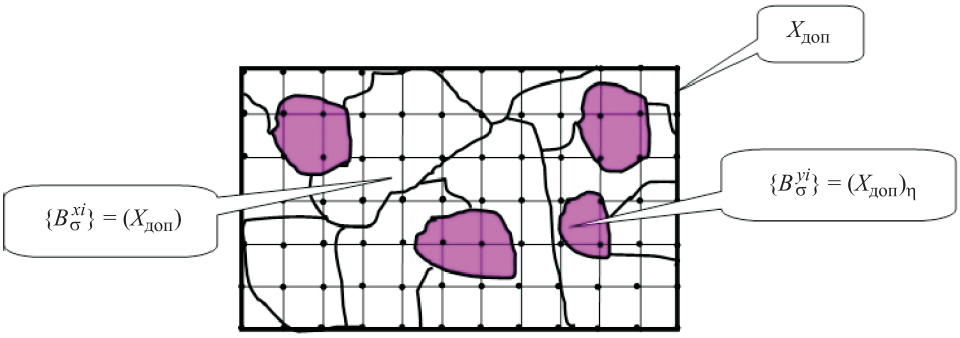

Рис. 2. Мозаичная модель входной области программы (Г, σ)

Используя геометрическую интерпретацию вероятности события и мозаичную модель входной области программы (Г, σ), можно получить выражение для оценки показателя надежности ПО в виде вероятности $P_{\text{ПО}}$ правильного функционирования программы на всем допустимом множестве входных вариантов данных $\left\{B_\sigma^{X\upsilon}\right\} = \left\{B_\sigma^{xi}\right\} \cup \left\{B_\sigma^{yj}\right\}$:

$$P_{\text{ПО}} = 1 - \frac{M\left[B_\sigma^{yj}\right]}{M\left[B_\sigma^{X\upsilon}\right]}, \tag{6}$$

где $M[B_\sigma^{yj}]$ — мощность множества всех вариантов входных данных, для которых программа (Г, σ) выдаст неправильный результат, — см. (2); $M[B_\sigma^{X\upsilon}]$ — мощность множества всех вариантов входных данных из допустимой области.

Выражение (6) определяет точное значение оценки $P_{\text{ПО}}$.

Порядок мощности множества допустимых вариантов входных данных для ПО вычислительного типа АСПД в первом приближении можно оценить как $10^{30}\ldots10^{54}$, что является «практической бесконечностью». Мощность множества $M[B_{\sigma}^{yj}]$ составляет единицы, поскольку количество вычислительных траекторий (ветвей), дефекты в которых приводят к невыполнению ПО возложенных на него функций (отказам ПО), незначительно. Таким образом, значение $P_{\text{ПО}}$, вычисляемое по точной формуле (6), практически равно единице.

С формальной точки зрения ПО осуществляет некоторое преобразование F варианта входных данных из допустимой области $X_{\text{доп}}$. Преобразование F определяется алгоритмом функционирования ПО, который зависит как от варианта входных данных $x \in X_{\text{доп}}$ из допустимой области, так и от параметров состояния программы S (вычислительного процесса), образующих некоторую область Z. Выход ПО Y является сложной функцией от допустимой области входных данных $X \subseteq X_{\text{доп}}$ и от параметров состояния:

$$Y = F\left(\Gamma, \sigma, \xi\right) = F\left\{X, f\left(S\left(x\right)\right)\right\},$$

где $S \in \Theta_{\sigma}$ — множество состояний программы, определяемое входными вариантами данных $x \in X_{\text{доп}}$ и ее семантикой σ; f — функция перехода программы из одного состояния в другое.

Из изложенного материала следует, что теорию вероятностей в геометрической интерпретации можно использовать для оценки показателя надежности ПО. Метод оценки указанного показателя, сводящийся к получению выражения (6), теоретически позволяет найти точную оценку этого показателя, но имеет один существенный недостаток, который делает его непригодным для практического применения. Чтобы получить числовое значение мощности множества дефектных зон для всей области допустимых входных данных ПО АСПД, потребуется практически бесконечный ресурс в виде практически бесконечного времени тестирования, если проверять выдачу правильного результата для каждого варианта входных данных из допустимой области. Поскольку использовать показатель надежности ПО АСПД $P_{\text{ПО}}$, определяемый выражением (6), для практических целей не представляется возможным, целесообразно проанализировать возможность применения моделей надежности ПО.

Анализ существующих моделей надежности ПО. Существует множество определений понятия надежности ПО, которые приведены, например, в работах [4–8]. Наиболее корректное определение надежности ПО дал Г. Майерс в работе [4]. Но даже это определение имеет ряд недостатков, которые сводятся к следующему. Во-первых,

это определение не надежности ПО, а его показателя, во-вторых, не определено понятие отказа ПО, в-третьих, не ясно, по какой причине показатель надежности привязан ко времени, а не к входным данным, в-четвертых, стоимость ошибки (дефекта) в ПО определяется объектом, в котором это ПО используется.

Дадим более корректное определение надежности ПО.

Определение 5. Надежность ПО — это его свойство выдавать правильный результат (1), если варианты входных данных принадлежат допустимой области.

Если отказ ПО определить как выполнение условий (2), то первое из этих условий остается неопределенным, так как необходимо знать заранее или определить вид желаемого выхода программы $F^*(\Gamma, \sigma, \xi)$. Для простых программ вычислительного типа значение $F^*(\Gamma, \sigma, \xi)$ может быть известно заранее (например, для программы, вычисляющей значения $\sin x$).

Для программ со сложной логической структурой зачастую бывает не ясно, чему должно равняться результирующее числовое значение программы. В этом случае необходимы специальные исследования, чтобы определить желаемый выход ПО для всего множества входных вариантов данных из допустимой области.

Из всех неизвестных параметров моделей надежности ПО самым важным является число имеющихся в нем ошибок. Согласно работе [9], любая программа объемом более 6000 символов (операторов и операндов) содержит не менее двух ошибок.

Самой известной моделью надежности ПО является модель, разработанная Z. Jelinski и P.B. Moranda [5]. Однако, поскольку в этой модели используется теория надежности аппаратуры, она признана некорректной.

При практическом применении изложенных моделей надежности программ возникают трудности, которые в основном связаны с невозможностью получения исходных данных или с явным невыполнением предположений, которые лежат в основе модели. Например, предполагается, что после обнаружения ошибки она мгновенно устраняется. На практике после устранения обнаруженной ошибки программу тестируют, чтобы выявить факт невнесения новых ошибок.

Из приведенного анализа моделей надежности ПО ясно, что до настоящего времени модели надежности, пригодной для практического применения, не существует. Из этого вывода следует, что метод оценки надежности ПО должен строиться не на использовании какой-либо модели надежности, а на результатах испытаний программы при определенном количестве тестовых вариантов входных данных и известных методах теории вероятностей и математической статистики. Такой подход применим тогда и только тогда, когда заранее известен правильный результат (1).

Метод оценки надежности ПО АСПД. Этот метод базируется на основных положениях нормативных документов [10–12], результатах исследований [13–15] и определяет порядок оценки надежности ПО АСПД на всех этапах его жизненного цикла.

Введение понятия эталона-результата диктуется необходимостью оценить правильность функционирования программы. Некоторые специалисты считают, что можно создать эталонную программу, результат ее функционирования принять за эталон и по нему ориентировать программы аналогичного назначения. Выше было показано, что создать безошибочную программу, имеющую сложную логическую структуру, объективно практически невозможно. Безошибочным является только эталон-результат, определяемый следующим образом.

Определение 6. Эталон-результат программы — ее желаемый выход, известный заблаговременно и не связанный с алгоритмами и языковыми средствами реализации программы.

Получить эталон-результат можно не для любой программы. Например, для программ, вычисляющих сложные математические выражения, результат которых невозможно знать априори, определить эталон-результат не представляется возможным.

Модель процесса оценивания показателя надежности ПО, представленная на рис. 3, соответствует документу [10] применительно к оценке показателя надежности ПО АСПД.

Для оценки показателя надежности ПО АСПД необходимо выполнение следующих требований.

1. Установленные потребности в ПО определяются основными функциями ПО АСПД и его характеристиками, которые задаются в ТЗ на разработку ПО АСПД, равно как и требуемые значения показателя надежности ПО $\left(P_{\text{ПО}}^{\text{тр}}\right)$ как вероятности выдавать правильный результат и доверительной вероятности $\beta_{\text{тр}}$, на основе которых определяется требуемое число его испытаний $N_{\text{тр}}$.

2. Спецификации требований к надежности ПО АСПД включают:

- алгоритмы функционирования каждого из компонентов ПО и ПО АСПД в целом;
- эталоны-результаты, определяющие требуемый выход каждого из компонентов ПО и ПО АСПД в целом, которые позволяют определить правильность выдачи результата;
- допустимые пределы $\varepsilon \geq 0$ отличия выхода каждого из компонентов ПО и ПО АСПД в целом от соответствующих эталонов-результатов;
- язык программирования, который используется для написания исходных текстов программ ПО АСПД;
- свидетельство о покрытии допустимой области входных данных тестовыми вариантами разработчика.

Рис. 3. Модель процесса оценивания надежности ПО АСПД

3. Выбор метрики производится на основании положения, состоящего в том, что показатель надежности ПО АСПД вычисляется только по результатам его тестирования на вариантах входных данных из допустимой области $X_{\text{доп}}$. Метрикой является величина ε в выражении (1).

4. В качестве промежуточной продукции рассматриваются отдельные компоненты ПО АСПД, а в качестве продукции — ПО в целом, но еще не прошедшее этапы тестирования и документальное оформление результатов тестирования.

5. Под измерениями понимается результат испытаний компонентов ПО или ПО АСПД в целом, который имеет вид x_k ($k = 1, 2, \ldots, K$). Таким образом, можно считать, что испытания ПО АСПД соответствуют схеме Бернулли.

6. Выделяют всего два уровня ранжирования с использованием полученной оценки показателя надежности ПО АСПД $P_{\text{ПО}}$:

$$P_{\text{ПО}} \geq P_{\text{ПО}}^{\text{тр}} \left(\text{при } \beta = \beta_{\text{тр}}\right),$$

$$P_{\text{ПО}} < P^{\text{тр}}.$$

7. Ранжирование означает отнесение полученной оценки показателя надежности ПО АСПД к одному из двух установленных уровней ранжирования.

8. Критерий оценки определяют следующим образом. Если ПО АСПД прошло все необходимые этапы тестирования на документально зафиксированных тестовых вариантах разработчика и полученное значение его показателя надежности не меньше заданного в ТЗ, то ПО признается пригодным к сдаче в эксплуатацию в составе АСПД, в противном случае разработчик продолжает его тестирование и устраняет выявленные ошибки.

9. Оценку показателя качества ПО проводят в два этапа. На первом этапе разработчик составляет тестовые варианты (в количестве, определенном в ТЗ значениями $P_{\text{ПО}}^{\text{тр}}$ и $\beta_{\text{тр}}$) и проводит испытания ПО АСПД на этих вариантах. По результатам тестирования компонентов ПО или ПО АСПД в целом разработчик вычисляет границы доверительного интервала (p_1, p_2), в которых заключена неизвестная оценка p^* показателя надежности ПО АСПД, при K испытаниях по формулам [16]

$$p_1 = \frac{p^* + 0,5t_\beta^2 / K - t_\beta \sqrt{p^*(1-p^*)/K + 0,25(t_\beta^2 / K^2)}}{1 + (t_\beta^2 / K)},$$

$$p_2 = \frac{p^* + 0,5t_\beta^2 / K + t_\beta \sqrt{p^*(1-p^*)/K + 0,25(t_\beta^2 / K^2)}}{1 + (t_\beta^2 / K)}.$$

Доверительный интервал $I_\beta = (p_1, p_2)$. Если границы слишком велики, то анализируются причины получения отрицательных исходов испытаний и выполняется их устранение. После этого проводится серия испытаний компонентов ПО и ПО АСПД в целом и вновь вычисляются границы доверительного интервала. Процедура продолжается до тех пор, пока оценка показателя надежности ПО АСПД не будет соответствовать ТЗ.

На втором этапе ПО АСПД считается надежным, показатель надежности имеет требуемые значения. В связи с этим на втором этапе определяют не оценку верхней границы показателя надежности ПО АСПД p_2, а требуемое число бездефектных испытаний $N_{\text{бдф}}^{\text{тр}}$ [16]:

$$N_{\text{бдф}}^{\text{тр}} = -\frac{\ln(1-\beta_{\text{тр}})}{p_2}.$$

Если достигнуто требуемое значение и не выявлены ошибки, можно считать, что разработанное ПО АСПД действительно обладает требуемой надежностью.

Важно иметь в виду, что оценка показателя надежности ПО АСПД p_2 не отражает относительного числа не выявленных в ПО ошибок. Например, ПО может содержать 100 ошибок. Если проведено большое число испытаний (например, 1 000 000) и ни одна из этих ошибок не проявилась, то по формуле (6) будет получено значение показателя надежности p_2 = 0,0001. Все имеющиеся в ПО ошибки можно выявить только в том случае, если тестирование ПО АСПД проводить по всей допустимой области, мощность которой составляет примерно 10^{30}. Но это будет детерминированный вариант оценки числа имеющихся в ПО ошибок, а не вероятность их проявления. Поскольку провести полное тестирование ПО АСПД не представляется возможным, утверждать, что показатель p_2 отражает число оставшихся в ПО ошибок, нельзя. Этот показатель отражает лишь вероятность того, что при каком-либо варианте входных данных ошибки в ПО не проявятся.

Заключение. В теории надежности используется порядка десяти различных моделей надежности ПО. В связи с этим проведен анализ существующих моделей надежности ПО и показано, что ни одна из них не может быть использована для оценки показателя надежности ПО АСПД. Модели надежности ПО имеют значительные методические ошибки, связанные с алгоритмами вычисления показателя надежности ПО, и требуют таких исходных данных, достоверные значения которых получить на практике невозможно.

Приведено доказательство того, что для оценки показателя надежности ПО АСПД использование методов теории вероятностей является корректным. Доказательство основано на представлении программы в виде вычислительных ветвей и траекторий, для которых даны соответствующие определения. Доказана теорема о том, что между множеством вариантов входных данных ПО из допустимой области и множеством вычислительных траекторий существует биекция (взаимно однозначное соответствие). Из этой теоремы следует математически строгая возможность применения для оценки показателя надежности ПО АСПД метода теории вероятностей в геометрической интерпретации вероятности появления события — ошибки в ПО АСПД. Показано, что использование метода теории вероятностей, основанного на геометрической интерпретации вероятности проявления ошибки в ПО АСПД, в принципе является корректным, но этот метод требует практически бесконечных временны́х ресур-

сов, что делает его непригодным для получения оценки показателя надежности ПО АСПД.

В связи с указанными недостатками моделей надежности ПО в работе предложен принципиально другой метод оценки показателя надежности ПО АСПД, который не имеет указанных выше недостатков и пригоден для практического применения. Предлагаемый метод основан на использовании основных положений стандартов, в том числе и международных, для оценки показателя надежности ПО АСПД. Метод включает два этапа: этап, связанный с оценкой доверительных границ, и этап, связанный с подтверждением требуемого значения показателя надежности при определенном числе бездефектных испытаний.

ЛИТЕРАТУРА

[1] Подловченко Р.И. Иерархия моделей программ. *Программирование*, 1981, № 2, с. 3–14.

[2] Летичевский А.А. Функциональная эквивалентность дискретных преобразований информации. *Кибернетика*, 1969, № 2, с. 5–16.

[3] Евстигнеев В.А. *Применение теории графов в программировании*. Москва, Наука, 1985, 352 с.

[4] Майерс Г. *Надежность программного обеспечения*. Москва, Мир, 1980, 360 с.

[5] Jelinski Z., Moranda P.B. *Software Reliability Research*. Statistical Computer Performance Evaluation. New York, Academic Press, 1972, pp. 465–484.

[6] *Характеристики качества программного обеспечения*. Москва, Мир, 1981, 208 с.

[7] Казаков Г.В. Критерии анализа связанных выборок при испытаниях программного обеспечения АСУ. *Двойные технологии*, 2014, № 3, с. 59–63.

[8] Mills H.D. *On the Statistical Validation of Computer Programs*, FSC-72-6015. IBM Federal Systems Division. Gaithersburg, Maryland, 1972.

[9] Холстед М.Х. *Начала науки о программах*. Москва, Финансы и статистика, 1981, 128 с.

[10] *ГОСТ Р ИСО/МЭК 9126–93. Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению.* Москва, Госстандарт России, 2004, 9 с.

[11] *ГОСТ Р ИСО/МЭК 15408–3–2013. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Компоненты доверия к безопасности*. Москва, Стандартинформ, 2014, 267 с.

[12] *ГОСТ 28195–89. Оценка качества программных средств. Общие положения.* Москва, Издательство стандартов, 2001, 30 с.

[13] Галактионов В.С., Знак В.А., Знак Н.Е., Казаков Г.В., Котяшев Н.Н., Сидоров А.В. О принципах испытания программного обеспечения АСУ двойного назначения с гибкими показателями эффективности. *Стратегическая стабильность*, 2009, № 3, с. 59–66.

[14] Казаков Г.В., Знак В.А., Данилин С.Б. Об одном подходе к формированию рационального множества тестовых вариантов на основе метода факторного анализа. *Труды МИТ*, 2015, т. 15, ч. 1, с. 114–119.

[15] Казаков Г.В. Метод оценки показателя надежности специального программного обеспечения комплексов средств подготовки данных по результатам испытаний на этапе разработки. *Труды МИТ*, 2015, т. 15, ч. 1, с. 102–113.

[16] Бордюков М.М., Галактионов В.С., Знак В.А., Знак Н.Е., Казаков Г.В., Сидоров А.В. Гарантированное оценивание конечного фазового состояния управляемых систем на заданном множестве достижимости. *Двойные технологии*, 2009, № 4, с. 34–38.

Статья поступила в редакцию 21.03.2018

Ссылку на эту статью просим оформлять следующим образом:

Андреев А.Г., Казаков Г.В., Корянов В.В. Метод оценки показателя надежности программного обеспечения автоматизированной системы подготовки данных управления летательными аппаратами. *Инженерный журнал: наука и инновации*, 2018, вып. 6. http://dx.doi.org/10.18698/2308-6033-2018-6-1771

Статья подготовлена по материалам доклада, представленного на XLII Академических чтениях по космонавтике, посвященных памяти академика С.П. Королёва и других выдающихся отечественных ученых — пионеров освоения космического пространства, Москва, 23–26 января 2018 года

Андреев Анатолий Георгиевич — канд. техн. наук, старший научный сотрудник ФГБУ «4 ЦНИИ» Министерства обороны России. Автор более 70 работ в области надежности автоматизированных систем управления. e-mail: kgv.64@mail.ru

Казаков Геннадий Викторович — канд. техн. наук, доцент, начальник управления ФГБУ «4 ЦНИИ» Министерства обороны России, почетный работник науки и техники Российской Федерации. Автор более 70 работ в области надежности автоматизированных систем управления. e-mail: kgv.64@mail.ru

Корянов Всеволод Владимирович — канд. техн. наук, доцент, первый заместитель заведующего кафедрой «Динамика и управление полетом ракет и космических аппаратов» МГТУ им. Н.Э. Баумана. Автор более 40 публикаций.
e-mail: vkoryanov@bmstu.ru

Method for estimating the reliability index of automated system software for aircraft control data preparation

© A.G. Andreev[1], G.V. Kazakov[1], V.V. Koryanov[2]

[1]Federal State Budgetary Institution 4th Central Research Institute of the Ministry of Defence of the Russian Federation, Korolev, Moscow region, 141091, Russia
[2]Bauman Moscow State Technical University, Moscow, 105005, Russia

The most complicated component of automated system for aircraft control data preparation is software. The likely reason for the interruption of the data preparation process is the presence of errors in the software, which may take a long time to resolve. Early elimination of these errors is an extremely important task. The degree of error elimination is determined by the value of the reliability index. The problem of assessment of the software reliability index is up to date, since there is no generally accepted technique for estimating this indicator. Based on the results of the analysis of existing software reliability models, it is shown that none of them can be used to estimate the software reliability index of an automated data preparation system. The proof of the correctness of using the methods of probability theory for the estimation of the indicator under consideration is presented. The theorem about a bijection (one-to-one correspondence) between the set of variants of input data and the set computational trajectories is proved. The possibility of applying the method of probability theory in the geometric interpretation of the probability of occurrence of an event (error) for estimating the software reliability index follows from the theorem. The use of this method is correct, but it requires infinite time resources, which makes it unsuitable for practical application. A fundamentally different method for estimating the software reliability index is proposed. Using this method it is necessary to have a documentary evidence of the coverage level by test variants of the entire input data area from the permissible region. The advantage of the proposed method is that it does not require any assumptions, and the initial data for estimating the software reliability index have a clear physical meaning and can be obtained in practice.

Keywords: *computational branch, computational path, reliability, error, software, reference result*

REFERENCES

[1] Podlovchenko R.I. *Programmirovanie — Programming and Computer Software*, 1981, no. 2, pp. 3–14.

[2] Letichevsky A.A. *Kibernetika — Cybernetics*, no. 2, 1969, pp. 5–16.

[3] Yevstigneyev V.A. *Primenenie teorii grafov v programmirovanii* [Application of the graph theory in programming]. Moscow, Nauka Publ., 1985, 352 p.

[4] Myers G.J. *Software Reliability: Principles and Practices*. New York, John Wiley Publ., 1976, 275 p. [In Russ.: Myers G. Reliability of the software. Moscow, Mir Publ., 1980, 360 p.].

[5] Jelinski Z., Moranda P.B. Software Reliability Research. In: *Statistical Computer Performance Evaluation.* Freiberger W., ed. New York, Academic Press Publ., 1972, pp. 465–484.

[6] Boehm B.W., Brown J.R., Kaspar H., Lipow M., MacLeod G.J., Merritt M.J. *TRW Series on Software Technology. Volume 1: Characteristics of Software Quality.* Amsterdam, North-Holland Publ., 1978 [In Russ.: Boehm B.W., Brown J.R., Kas-

par H. i dr. Kharakteristiki kachestva programmnogo obespecheniya. Moscow, Mir Publ., 1981, 208 p.].

[7] Kazakov G. V. *Dvoynye tekhnologii* [Double-Purpose Technology]. 2014, no. 3, pp. 59–63.

[8] Mills H. D. *On the Statistical Validation of Computer Programs. FSC-72-6015,* Gaithersburg, Md., IBM Federal Systems Div. Publ., 1972.

[9] Halstead M.H. *Elements of Software Science.* Amsterdam, Elsevier North-Holland, Inc. Publ., 1977, 128 p. [In Rus.: Halstead M.H. Nachala nauki o programmakh. Moscow, Finansy i statistika Publ., 1981, 128 p.].

[10] *GOST P ISO/IEC 9126-93. Informatsionnaya tekhnologiya. Otsenka programmnoy produktsii. Kharakteristiki kachestva i rukovodstva po ikh primeneniu* [RF standard P ISO/IEC 9126-93. Information technology. Assessment of program production. Characteristics of quality and a manual on their application]. Moscow, Gosstandart Russii Publ., 2004, 9 p.

[11] GOST P ISO/IEC 15408-3–2013. *Informatsionnaya tekhnologiya. Metody i sredstva obespecheniya bezopasnosti. Kriterii otsenki bezopasnosti informatsionnykh tekhnologiy. Chast 3. Komponenty doveriya k bezopasnosti* [RF standard GOST P ISO/IEC 15408-3–2013. Information technology. Methods and means of ensuring security. Criteria for assessing the security of information technology. Part 3. Components of trust in security]. Moscow, Standartinform Publ., 2014, 267 p.

[12] *GOST 28195-89. Otsenka kachestva programmnykh sredstv. Obshchie polozheniya* [USSR State standard. Software quality assessment. General provisions]. Moscow, Izdatelstvo standartov Publ., 2001, 30 p.

[13] Galaktionov V.S., Znak V.A., Znak N.E., Kazakov G.V., Kotyashev N.N., Sidorov A.V. *Strategicheskaya stabilnost — Strategic stability*, 2009, no. 3, pp. 59–66.

[14] Kazakov G.V., Znak V.A., Danilin S.B. Ob odnom podkhode k formirovaniu ratsionalnogo mnozhestva testovykh variantov na osnove metoda faktornogo analiza [On an approach to the formation of a rational set of test options based on the factor analysis method]. In: *Trudy Moskovskogo instituta teplotekhniki* [Proceedings of Moscow Institute of Heat Engineering]. Moscow, MIT Publ., 2015, vol. 15, part 1, pp. 114–119.

[15] Kazakov G. V. Metod otsenki pokazatelya nadezhnosti spetsialnogo programmnogo obespecheniya komplekcov sredstv podgotovki dannykh po rezultatam ispytaniy na etape razrabotki [Method for estimating the reliability index of special software for data-preparation facilities based on test results at the development stage]. In: *Trudy Moskovskogo instiyuta teplotekhniki* [Proceedings of Moscow Institute of Heat Engineering]. Moscow, MIT Publ., 2015, vol. 15, part 1, pp. 102–113.

[16] Bordukov M.M., Galaktionov V.S., Znak V.A., Znak N.E., Kazakov G.V., Sidorov A.V. *Dvoynye tekhnologii* [Double-Purpose Technology]. 2009, no. 4, pp. 34–38.

Andreev A.G. (b. 1941), Cand. Sc. (Eng.), Senior Research Fellow, Federal State Budgetary Institution 4th Central Research Institute of the Ministry of Defence of the Russian Federation. Author of over 70 research publications in the field of automated control system reliability. e-mail: kgv.64@mail.ru

Kazakov G.V. (b. 1964), Cand. Sc. (Eng.), Assoc. Professor, Head of the division, Federal State Budgetary Institution 4th Central Research Institute of the Ministry of Defence

of the Russian Federation, honorary worker of science and technology of the Russian Federation. Author of over 70 research publications in the field of automated control system reliability. e-mail: kgv.64@mail.ru

Koryanov V.V. (b. 1982) graduated from Bauman Moscow State Technical University in 2006. Cand. Sc. (Eng.), Assoc. Professor, Department of Space Flight Dynamics and Control, Bauman Moscow State Technical University. Author of over 40 research publications in the field of ballistics modelling and dynamics of spacecraft and descent vehicle motion. e-mail: vkoryanov@bmstu.ru