

ПРОГНОЗИРОВАНИЕ ЧИСЛА ОТКАЗОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОСМИЧЕСКИХ АППАРАТОВ

А.А. Золотов
Э.Д. Нуруллаев

alexandrzolotov41@mail.ru
ned@mai.ru

Московский авиационный институт (национальный исследовательский университет),
Москва, Российская Федерация

Аннотация

Рассмотрены вопросы прогнозирования числа отказов программного обеспечения космических аппаратов, обусловленные ошибками в записи компьютерных программ и сбоями в их работе. Выполнена оценка оптимального количества циклов тестирования программного обеспечения космических аппаратов и спрогнозировано среднее число отказов программного обеспечения космических аппаратов, сохранившееся после устранения отказов при тестировании, что может привести к отказу в работе космических аппаратов. Работоспособность предлагаемой методики проиллюстрирована на примерах решения типовых задач

Ключевые слова

Надежность, программное обеспечение, число отказов, количество циклов тестирования

Поступила в редакцию 03.03.2016
© МГТУ им. Н.Э. Баумана, 2016

Развитие компьютерных технологий привело к широкому использованию их как в составе автоматизированных средств контроля, так и непосредственно для управления работой различных систем космических аппаратов (КА). Поэтому происходит рост объемов программного обеспечения (ПО) и ужесточаются требования к надежности компьютерных программ. Решение задачи надежности ПО усложняется, поскольку причины нештатных ситуаций становятся более разнообразными. В частности, они обуславливаются ошибками в записи компьютерных программ, сбоями в их работе. Работоспособность ПО оценивается по результатам тестирования, при котором определенное число ошибок устраняется. Тестирование состоит из ряда этапов: выполнение программы тестирования; обнаружение и устранение ошибки; продолжение выполнения программы тестирования. При тестировании выявляются отказы, возникающие на каждом цикле. Информация о выявленном числе отказов на различных циклах тестирования используется для оценки надежности ПО. Для решения этой задачи могут быть использованы различные модели прогнозирования надежности ПО. Далее рассмотрены некоторые из них [1–7].

Прогнозирование надежности ПО на основе аналитической модели. Существующие модели оценки надежности ПО основаны на использовании неоднородного пуассоновского процесса (НПП). При этом принимается, что мате-

математическое ожидание числа выявленных ошибок имеет экспоненциальный рост. Однако на практике, как правило, число выявленных ошибок подчиняется S-образной кривой. В данном разделе анализируется следующий вид зависимости, предложенной в работе [7]:

$$M(t) = a[1 - (1 + bt)e^{-bt}],$$

где $M(t)$ — математическое ожидание числа выявленных ошибок за время t ; a, b — коэффициенты аппроксимации; t — время функционирования устройства.

При решении задачи предполагается, что ошибки, выявляемые на каждом цикле тестирования, устраняются по мере их обнаружения. При этом новые ошибки не вводятся. В соответствии с исследуемой моделью надежность ПО оценивается по соотношению

$$P(t, s) = e^{-[M(t+s) - M(s)]},$$

где $P(t, s)$ — вероятность того, что на интервале $(s, s + t)$ не встретится никаких ошибок ПО при условии, что последняя ошибка выявлена и устранена в момент s .

Оценка параметров a и b исследуемой модели проводится по результатам тестирования. Предложенный в работе [7] подход использует только информацию о числе отказов на каждом цикле тестирования и не требует знания дополнительных сведений о конкретных особенностях ПО. Далее приведены статистические данные по результатам тестирования в виде значений y_k , которые характеризуют число ошибок ПО, выявленных к моменту времени t_k . Для нахождения искомых параметров используется метод максимального правдоподобия. В общем случае функция правдоподобия имеет вид

$$L = \prod_{i=1}^n f_i(\bar{\theta}, x_i),$$

где $\bar{\theta}$ — вектор искомых параметров; x_i — результаты тестирования.

В рассматриваемом случае $f_i(\bar{\theta}, x_i)$ характеризует вероятность выявления такого числа ошибок, которое реализовалось на каждом цикле тестирования. Согласно принятому предположению, эта вероятность подчиняется НПП с S-образным изменением математического ожидания. Таким образом, выражение для функции правдоподобия принимает вид

$$L = \prod_{k=1}^n \frac{(\Delta M_{k-1,k})^{\Delta k}}{\Delta k!} e^{-\Delta M_{k-1,k}},$$

где $\Delta M_k = M(k) - M(k-1)$, $\Delta k = y_k - y_{k-1}$.

После подстановки получим

$$L = \prod_{k=1}^n \frac{\{\Delta M_{k-1,k}\}^{(y_k - y_{k-1})}}{(y_k - y_{k-1})!} \prod_{k=1}^n e^{-\Delta M_{k-1,k}}.$$

Проводя логарифмирование, получаем

$$\ln L = \sum_{k=1}^n (y_k - y_{k-1}) \ln a + \sum_{k=1}^n (y_k - y_{k-1}) \ln \left\{ \Delta M_{k-1,k} / a \right\} - \sum_{k=1}^n \ln[(y_k - y_{k-1})!] - \sum_{k=1}^n \Delta M_{k-1,k},$$

где $\Delta M_{k-1,k} = a \left[(1 + bt_{k-1})e^{-bt_{k-1}} - (1 + bt_k)e^{-bt_k} \right]$.

Очевидно, что

$$\sum_{k=1}^n \Delta M_{k-1,k} = a \left[1 - (1 + bt_n)e^{-bt_n} \right].$$

Искомые параметры должны удовлетворять условию оптимальности:

$$\frac{\partial L}{\partial a} = 0, \quad \frac{\partial L}{\partial b} = 0.$$

Дифференцируя по параметру a , получаем

$$\frac{\partial \ln L}{\partial a} = \frac{1}{a} \sum_{k=1}^n (y_k - y_{k-1}) - \left[1 - (1 + bt_n)e^{-bt_n} \right] \equiv 0.$$

Отсюда имеем

$$\sum_{k=1}^n (y_k - y_{k-1}) = y_n = a \left[1 - (1 + bt_n)e^{-bt_n} \right].$$

Решая соотношение относительно a , приходим к искомой оценке:

$$a = \frac{y_n}{1 - (1 + bt_n)e^{-bt_n}}. \tag{1}$$

Дифференцируя логарифм функции правдоподобия по параметру b , получаем

$$\frac{\partial L}{\partial b} = \sum_{k=1}^n (y_k - y_{r-1}) \frac{(\Delta M_{k-1,k})'_b}{\Delta M_{k-1,k}} + a \left[(1 + bt_n)e^{-bt_n} \right]'_b,$$

где $(\Delta M_{k-1,k})'_b = b(t_k^2 e^{-bt_k} - t_{k-1}^2 e^{-bt_{k-1}})$; $\left[(1 + bt_n)e^{-bt_n} \right]'_b = -abt_n^2 e^{-bt_n}$.

Подставляя полученные соотношения в условие оптимальности, находим

$$b \sum_{k=1}^n (y_k - y_{r-1}) \frac{(t_k^2 e^{-bt_k} - t_{k-1}^2 e^{-bt_{k-1}})}{\left[(1 + bt_{k-1})e^{-bt_{k-1}} - (1 + bt_k)e^{-bt_k} \right]} = bat_n^2 e^{-bt_n}.$$

С учетом полученного соотношения для оценки параметра a окончательно имеем

$$y_n t_n^2 = \Phi(b),$$

где

$$\Phi(b) = \left\{ \sum_{k=1}^n (y_k - y_{r-1}) \frac{(t_k^2 e^{-bt_k} - t_{k-1}^2 e^{-bt_{k-1}})}{[(1 + bt_{k-1})e^{-bt_{k-1}} - (1 + bt_k)e^{-bt_k}]} \right\} e^{bt_n} [1 - (1 + bt_n)e^{-bt_n}].$$

Для иллюстрации работоспособности предлагаемой методики рассматриваем модельный пример, представленный в работе [7]. Предполагаем, что за 10 циклов тестирования была выявлена 31 ошибка. Далее приведены результаты по каждому циклу тестирования: $n_0 = 0$; $n_1 = 2$; $n_2 = 4$; $n_3 = 10$; $n_4 = 16$; $n_5 = 18$; $n_6 = 22$; $n_7 = 24$; $n_8 = 28$; $n_9 = 30$; $n_{10} = 31$, где n_i — накопленное число ошибок, соответствующее i -му циклу тестирования.

Искомые значения b оценены графически при выполнении условия (рис. 1)

$$y_n t_n^2 = \Phi(b). \tag{2}$$

Как следует из графика, условие (2) удовлетворяется при значении $b = 0,345$. Далее по соотношению (1) оценивают значение параметра a :

$$a = \frac{y_n}{[1 - (1 + bt_n)e^{-bt_n}]} = 36,1.$$

Зная a и b , можно найти распределение прогнозируемых значений числа ошибок $M(i)$ по циклам тестирования i :

$$M(i) = a [1 - (1 + bi)e^{-ib}].$$

Результаты расчета приведены на рис. 2.

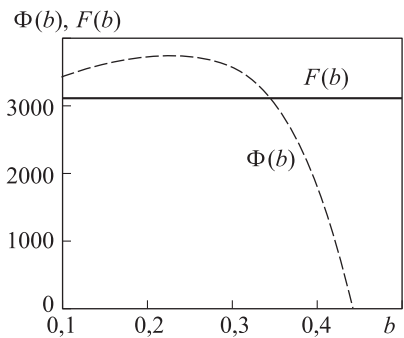


Рис. 1. Зависимости изменения функций $\Phi(b)$ (штриховая кривая) и $F(b)$ (сплошная) от параметра b

$i^T =$	0	1	2	3	4	5	6	7	8	9	
	0	0	1	2	3	4	5	6	7	8	9

$N(i)^T =$	0	1	2	3	4	5	6	7	8	9	
	0	0	2	4	10	16	18	22	24	28	30

$M(i)^T =$	0	1	2	3	4	5	6	7	8	9	
	0	0	1.7	5.5	10	14.5	18.6	22.1	25.1	27.5	29.46

Рис. 2. Распределение расчетных оценок $M(i)$ и реального накопления ошибок $N(i)$ по циклам тестирования i

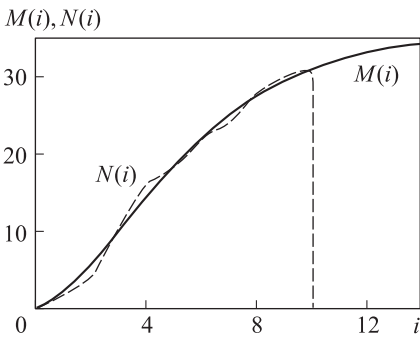


Рис. 3. Зависимости изменения сглаживающей $M(i)$ (сплошная кривая) и экспериментальной $N(i)$ (штриховая) кривых от количества циклов тестирования

Характер изменения математического ожидания числа ошибок $M(i)$ по циклам тестирования показан на рис. 3.

На графиках по оси абсцисс отложены циклы тестирования, по оси ординат — число выявленных ошибок. Реальное накопление ошибок представлено штриховой кривой $N(i)$. Обе кривые имеют слабо выраженную S-образную форму. Как видно на графике, аппроксимирующая кривая хорошо согласуется с опытными данными.

Прогнозирование надежности ПО на основе численной модели. Рассмотренная задача также может быть решена с помощью численной модели, позволяющей снять допущения аналитической модели. Для оценки числа отказов используем алгоритм, представленный в работе [5]. Программа расчетов по указанному алгоритму приведена на рис. 4. При расчетах принимают $\beta = 0$, что соответствует режиму «выжигания» отказов, задают значения исходных параметров $\lambda, \nu, M_{0,0}$. Искомые параметры, оцениваемые программой расчетов после n циклов тестирования, определяются треугольной матрицей $M(n)$, которая является функцией количества циклов тестирования. В этой матрице нулевая строка соответствует значению $M_{0,0}$, характеризующему исходное число анализируемых критических элементов системы, i -й строке и j -му столбцу матрицы соответствует элемент $M_{i,j}$, характеризующий число элементов, функционирующих на i -м цикле тестирования и замененных на новые на j -м цикле тестирования. Число ошибок на i -м цикле тестирования выводится отдельно как диагональный элемент матрицы $M(n)_{i,i}$. Для проведения расчетов по предлагаемой методике необходимо оценить параметры модели, согласующиеся с результатами тестирования. Расчеты параметров $\lambda, \nu, M_{0,0}$ выполняют по методу наименьших квадратов, согласно которому значения исходных параметров должны обеспечивать минимальную сумму квадратов отклонений зафиксированного числа ошибок от прогнозируемых на каждом цикле тестирования, т. е. минимум критериальной функции L :

$$L = \sum_k \left[\hat{M}_k - M_k(\lambda_0, \nu, M_{0,0}) \right]^2, \quad (3)$$

где \hat{M}_k — число ошибок, зафиксированных на k -м цикле тестирования; $M_k(\lambda_0, \nu, M_{0,0})$ — прогнозируемое число ошибок на k -м цикле тестирования, рассчитываемое по алгоритму, представленному ранее.

В дальнейшем, учитывая, что число ошибок на всех циклах пропорционально $M_{0,0}$, переходим к рассмотрению относительного числа ошибок:

$$\bar{M}_k = \frac{\hat{M}_k}{\hat{M}_1}, \tilde{M}_k = \frac{M_k}{M_1}.$$

Тогда критериальная функция зависит только от двух искомым параметров λ_0 , ν и принимает вид

$$A(\lambda_0, \nu) = \sum_k (\tilde{M}_k - \bar{M}_k)^2. \tag{4}$$

$\nu := 1.6$ $\lambda := 0.06$ $\beta := 0$ $r := 1$

```

M(n) :=
  M0,0 ← 34
  for m ∈ 0..n
    am ← 1 - e-λ·β[(m+1)ν-mν]
    for s ∈ 0..n
      bs ← 1 - e-λ[(s+1)ν-sν]
    for i ∈ 0..n - 1
      for j ∈ 0..n
        Mi+1,j ← [ [Mi,j(1 - bi-j) ] if i + 1 > j
                  [ [ ∑k=0i (Mi,k bi-k) ] ] otherwise
    for i ∈ r..n - 1
      for j ∈ r..n
        Mi+1,j ← [ [Mi,j(1 - ai-j) ] if i + 1 > j
                  [ [ ( ∑k=0r-1 Mi,k · bi-k ) + ( ∑k=ri Mi,k · ai-k ) ] ] otherwise
    for i ∈ 0..n
      for j ∈ 0..n
        Mi,j ← [ Mi,j if i ≥ j
                0 otherwise
  M
    
```

M(10) =

	0	1	2	3	4	5	6	7	8	9
0	34	0	0	0	0	0	0	0	0	0
1	32.02	1.98	0	0	0	0	0	0	0	0
2	28.346	1.98	3.674	0	0	0	0	0	0	0
3	24.008	1.98	3.674	4.338	0	0	0	0	0	0
4	19.589	1.98	3.674	4.338	4.419	0	0	0	0	0
5	15.462	1.98	3.674	4.338	4.419	4.127	0	0	0	0
6	11.84	1.98	3.674	4.338	4.419	4.127	3.622	0	0	0
7	8.815	1.98	3.674	4.338	4.419	4.127	3.622	3.025	0	0
8	6.391	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	0
9	4.519	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	1.872
10	3.12	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	1.872

Рис. 4. Программа расчетов

$$v := 1.85 \quad \lambda := 0.07 \quad \beta := 0 \quad r := 1$$

$$\begin{array}{l}
 M(n) := \left\{ \begin{array}{l}
 M_{0,0} \leftarrow 1 \\
 \text{for } m \in 0..n \\
 \quad a_m \leftarrow 1 - e^{-\lambda \cdot \beta \cdot [(m+1)^v - m^v]} \\
 \text{for } s \in 0..n \\
 \quad b_s \leftarrow 1 - e^{-\lambda \cdot [(s+1)^v - s^v]} \\
 \text{for } i \in 0..n-1 \\
 \quad \text{for } j \in 0..n \\
 \quad \quad M_{i+1,j} \leftarrow \begin{cases} [M_{i,j}(1 - b_{i-j})] & \text{if } i+1 > j \\ \left[\sum_{k=0}^i (M_{i,k} b_{i-k}) \right] & \text{otherwise} \end{cases} \\
 \text{for } i \in r..n-1 \\
 \quad \text{for } j \in r..n \\
 \quad \quad M_{i+1,j} \leftarrow \begin{cases} [M_{i,j}(1 - a_{i-j})] & \text{if } i+1 > j \\ \left[\left(\sum_{k=0}^{r-1} M_{i,k} \cdot b_{i-k} \right) + \left(\sum_{k=r}^i M_{i,k} \cdot a_{i-k} \right) \right] & \text{otherwise} \end{cases} \\
 \text{for } i \in 0..n \\
 \quad \text{for } j \in 0..n \\
 \quad \quad M_{i,j} \leftarrow \begin{cases} M_{i,j} & \text{if } i \geq j \\ 0 & \text{otherwise} \end{cases}
 \end{array} \right. \quad N := \begin{pmatrix} 1 \\ 1 \\ 1 \\ 3 \\ 3 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 0.5 \end{pmatrix} \\
 M \\
 A \leftarrow \sum_{k=1}^{10} \left[\left[(M(10))_{1,1} \right]^{-1} \cdot M(10)_{k,k} - N_k \right]^2 = \blacksquare \quad A := 6.056
 \end{array}$$

$$A(0.05) := 7.64 \quad A(0.06) := 6.013 \quad A(0.07) := 6.056$$

Рис. 5. Алгоритм и результаты расчета критериальной функции A для $v=1,85$ и различных значений λ

Значения λ_0 и v , обеспечивающие минимум этой функции, дают оценку искоемых параметров. Эти значения находят методом перебора. При этом значения критериальной функции рассчитывают для различных значений параметров λ_i и v_i . Согласно предлагаемой методике, для каждого фиксированного уровня v рассчитывают значения критериальной функции для различных значений λ_i , среди которых выбирают значение λ , обеспечивающее минимум критерия A . В дальнейшем такие же расчеты проводят для другого фиксированного уровня v . Допустим, при $v = v_r$ минимум достигается для значения $\lambda = \lambda_{r,s}$. Тогда, если для различных значений v_i выполняется условие $A(v_r) \leq A(v_i)$, то значение v_k и соответствующее ему значение $\lambda_{r,s}$ являются более предпочтительными. Точность по-

иска искомых параметров определяется шагом перебора параметров λ_i и ν_i . Для повышения точности уменьшается шаг. Алгоритм расчетов и исходные данные приведены на рис. 5 и 6. Зафиксированные значения \bar{M}_k представлены на рисунках в виде матрицы-столбца N . Как следует из рис. 5, для $\nu=1,85$ минимум $A = 6,103$ достигается при $\lambda = 0,06$. Соответственно для $\lambda = 0,06$ минимум $A = 4,109$ достигается при $\nu = 1,6$ (см. рис. 6). Таким образом, в рассматриваемом случае минимум критериальной функции (4) достигается при $\nu=1,6$, которому соответствует $\lambda = 0,06$. Величина $M_{0,0}$ оценивается по изложенному методу наименьших квадратов для найденных оптимальных значений $\lambda = 0,06$ и $\nu = 1,6$. При этом минимум критериальной функции (4) находится методом перебора по $M_{0,0}$. Результаты расчета для рассмотренного примера приведены на рис. 8.

$$\lambda := 0.06 \quad \nu := 1.55 \quad \beta := 0 \quad r := 1$$

```

M(n) :=
  M0,0 ← 1
  for m ∈ 0..n
    am ← 1 - e-λ·β[(m+1)ν-mν]
  for s ∈ 0..n
    bs ← 1 - e-λ[(s+1)ν-sν]
  for i ∈ 0..n-1
    for j ∈ 0..n
      Mi+1,j ←
        [Mi,j(1 - bi-j)] if i + 1 > j
        [
          [
            ∑k=0i (Mi,k bi-k)
          ]
        ] otherwise
  for i ∈ r..n-1
    for j ∈ r..n
      Mi+1,j ←
        [Mi,j(1 - ai-j)] if i + 1 > j
        [
          [
            ∑k=0r-1 Mi,k · bi-k
          ] + [
            ∑k=ri Mi,k · ai-k
          ]
        ] otherwise
  for i ∈ 0..n
    for j ∈ 0..n
      Mi,j ←
        Mi,j if i ≥ j
        0 otherwise
  M
  
```

$$N := \begin{pmatrix} 1 \\ 1 \\ 1 \\ 3 \\ 3 \\ 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 0.5 \end{pmatrix}$$

$$A \leftarrow \sum_{k=1}^{10} \left[\left[(M(10))_{1,1} \right]^{-1} \cdot M(10)_{k,k} - N_k \right]^2 = \bullet \quad A \leftarrow 4.218$$

$$A(1.55) := 4.218 \quad A(1.6) := 4.109 \quad A(1.65) := 4.15$$

Рис. 6. Алгоритм и результаты расчета критериальной функции A для $\lambda = 0,06$ и различных значений ν

$v := 1.6$ $\lambda := 0.06$ $\beta := 0$ $r := 1$

```

M(n) :=
  M0,0 ← 34
  for m ∈ 0..n
    am ← 1 - e-λ·β[(m+1)v-mv]
  for s ∈ 0..n
    bs ← 1 - e-λ[(s+1)v-sv]
  for i ∈ 0..n-1
    for j ∈ 0..n
      Mi+1,j ←
        [Mi,j(1 - bi-j)] if i + 1 > j
        [
          [
            [
              ∑k=0i (Mi,k bi-k)
            ]
          ]
        ] otherwise
  for i ∈ r..n-1
    for j ∈ r..n
      Mi+1,j ←
        [Mi,j(1 - ai-j)] if i + 1 > j
        [
          [
            [
              ∑k=0r-1 Mi,k · bi-k
            ]
          ]
          +
          [
            [
              ∑k=ri Mi,k · ai-k
            ]
          ]
        ] otherwise
  for i ∈ 0..n
    for j ∈ 0..n
      Mi,j ←
        Mi,j if i ≥ j
        0 otherwise
  M
  
```

N := $\begin{pmatrix} 30 \\ 2 \\ 2 \\ 6 \\ 6 \\ 2 \\ 4 \\ 2 \\ 4 \\ 2 \\ 1 \end{pmatrix}$

$$L \leftarrow \sum_{k=1}^{10} [(M(17))_{k,k} - N_k]^2 = \blacksquare \quad L := 16.447$$

L(33) := 16.602 L(34) := 16.447 L(35) := 16.475

Рис. 7. Алгоритм и результаты расчета параметра $M_{0,0}$

Как следует из расчета, минимум критериальной функции $L = 16,447$ достигается на 34-м цикле тестирования, что соответствует значению $M_{0,0} = 34$. Расчеты числа ошибок по циклам тестирования для найденных значений $\lambda_0, v, M_{0,0}$ приведены на рис. 8.

Математическое ожидание $L(i)$ суммарного числа ошибок по циклам тестирования представлено на рис. 9.

Результаты расчетов приведены на рис. 10. На этом же графике для сравнения представлено изменение прогнозируемого числа ошибок $M(i)$ от количества циклов тестирования, полученного по модели НПП с S-образным законом изменения математического ожидания числа ошибок.

$$v := 1.6 \quad \lambda := 0.06 \quad \beta := 0 \quad r := 1$$

```

M(n) :=
  M0,0 ← 34
  for m ∈ 0..n
    am ← 1 - e-λ·β[(m+1)v-mv]
  for s ∈ 0..n
    bs ← 1 - e-λ[(s+1)v-sv]
  for i ∈ 0..n - 1
    for j ∈ 0..n
      Mi+1,j ←
        [ Mi,j(1 - bi-j) ] if i + 1 > j
        [ [ ∑k=0i (Mi,k bi-k) ] ] otherwise
  for i ∈ r..n - 1
    for j ∈ r..n
      Mi+1,j ←
        [ Mi,j(1 - ai-j) ] if i + 1 > j
        [ [ ( ∑k=0r-1 Mi,k · bi-k ) + ( ∑k=ri Mi,k · ai-k ) ] ] otherwise
  for i ∈ 0..n
    for j ∈ 0..n
      Mi,j ←
        Mi,j if i ≥ j
        0 otherwise
  M
    
```

M(10) =

	0	1	2	3	4	5	6	7	8	9
0	34	0	0	0	0	0	0	0	0	0
1	32.02	1.98	0	0	0	0	0	0	0	0
2	28.346	1.98	3.674	0	0	0	0	0	0	0
3	24.008	1.98	3.674	4.338	0	0	0	0	0	0
4	19.589	1.98	3.674	4.338	4.419	0	0	0	0	0
5	15.462	1.98	3.674	4.338	4.419	4.127	0	0	0	0
6	11.84	1.98	3.674	4.338	4.419	4.127	3.622	0	0	0
7	8.815	1.98	3.674	4.338	4.419	4.127	3.622	3.025	0	0
8	6.391	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	0
9	4.519	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	1.872
10	3.12	1.98	3.674	4.338	4.419	4.127	3.622	3.025	2.424	1.872

Рис. 8. Алгоритм и результаты расчета числа ошибок по циклам тестирования

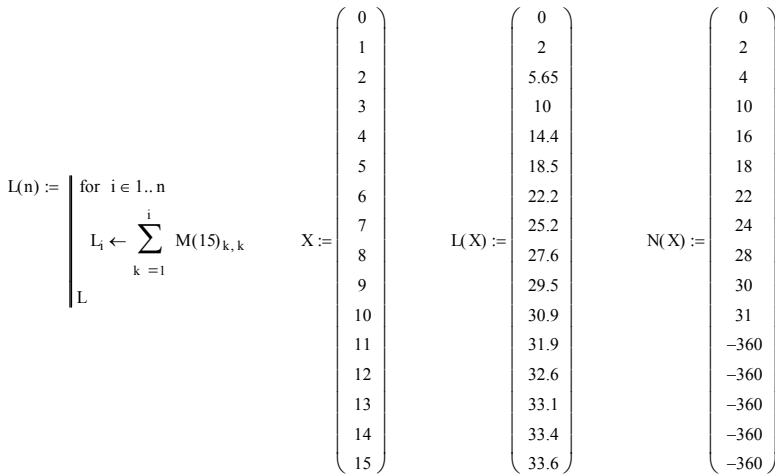


Рис. 9. Распределение зафиксированных $N(X)$ и прогнозируемых $L(X)$ чисел ошибок по количеству циклов тестирования

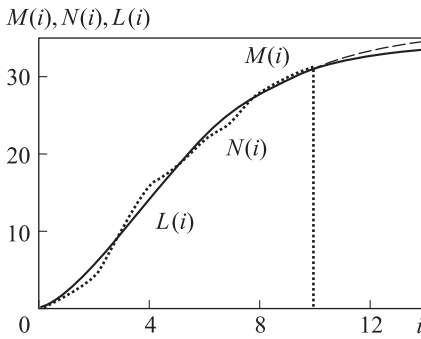


Рис. 10. Изменение экспериментальной $N(i)$ (точечная кривая) и сглаживающих кривых $L(i)$ (сплошная линия) и $M(i)$ (штриховая линия) от количества циклов тестирования

Как следует из графиков, результаты расчетов по обеим моделям хорошо согласуются с экспериментальными данными и друг с другом. В то же время надо заметить, что использование численной модели позволяет расширить круг задач, решаемых при прогнозировании надежности ПО. В частности, предлагаемая методика может быть использована для доработок различного характера и разных моментов начала доработок, что требует уточнения параметра β :

- $\beta = 1$ — сохранение ошибок по мере обнаружения;
- $\beta = 0$ — полное устранение ошибок по мере обнаружения («выжигание» отказов);
- $0 < \beta < 1$ — частичное устранение ошибок по мере обнаружения;
- $\beta > 1$ — ввод новых ошибок.

Для всех вариантов параметра β уточнение параметра r в диапазоне $1, 2, \dots, n-1$ предполагает, что доработки начинают выполнять после $1, 2, \dots, n-1$ циклов тестирования. В рассмотренном примере для сравнения аналитического и численного методов принято $\beta = 0$.

Преимущество предлагаемой методики также заключается в том, что она основывается только на анализе формирования отказов при тестировании и не

использует никаких гипотез относительно вида аппроксимирующей кривой. Приведенные результаты позволяют прогнозировать среднее число отказов, сохранившееся после устранения отказов при тестировании, проявление которых может привести к отказу при эксплуатации, а также оценивать оптимальное количество циклов тестирования. Очевидно, оптимальное количество циклов тестирования должно соответствовать минимуму суммарных затрат на реализацию целевой космической программы. В общем случае суммарные затраты определяются стоимостью тестирования и величиной ущерба при отказах:

$$C_{\Sigma} = cn + c_{\text{ущ}} (2M_{0,0} - \text{tr}(M(n))),$$

где c — затраты на одно тестирование; $c_{\text{ущ}}$ — ущерб при одном отказе; n — количество циклов тестирования; $\text{tr}(M(n))$ — след матрицы $M(n)$.

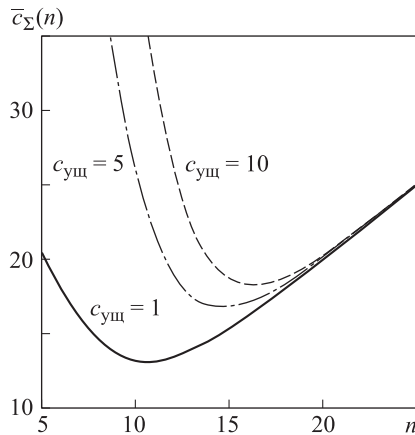
Соответственно для безразмерной стоимости \bar{C}_{Σ} выражение принимает вид

$$\bar{C}_{\Sigma} = n + \bar{c}_{\text{ущ}} (2M_{0,0} - \text{tr}(M(n))),$$

где $\bar{c}_{\text{ущ}} = c_{\text{ущ}}/c$ — относительный ущерб при отказе; $\bar{C}_{\Sigma} = C_{\Sigma}/c$.

Значения матрицы $M(n)$ рассчитываются по программе, приведенной на рис. 8. Результаты оптимизации приведены на рис. 11. Расчеты проводятся для $c_{\text{ущ}} = 1; 5; 10$. Как видно на графике, с увеличением относительного ущерба при от-

Рис. 11. Изменение безразмерных суммарных затрат $c_{\text{ущ}} = 1$ (сплошная кривая), $c_{\text{ущ}} = 5$ (штрихпунктирная), $c_{\text{ущ}} = 10$ (штриховая) от количества циклов тестирования



казе $\bar{c}_{\text{ущ}}$ оптимальное количество циклов тестирования r возрастает, что объясняется снижением ущерба ввиду сокращения источников отказов, не выявленных при тестировании.

Выводы. 1. Разработана численная модель прогнозирования числа отказов ПО космических аппаратов.

2. Разработан метод прогнозирования числа невыявленных отказов программного обеспечения космических аппаратов.

3. Разработан метод оценки оптимального количества циклов тестирования программного обеспечения космических аппаратов.

4. Предлагаемая численная модель в отличие от аналитической позволяет более адекватно отразить реальный процесс функционирования программного обеспечения космических аппаратов.

5. Предлагаемая методика в сочетании с существующими методами позволяет получить более полноценное представление о надежности программного обеспечения космических аппаратов.

6. Работоспособность предлагаемой методики проиллюстрирована на примерах решения типовых задач с использованием программ расчетов в системе MathCAD.

ЛИТЕРАТУРА

1. Александровская Л.Н., Афанасьев А.П., Лисов А.А. Современные методы обеспечения безотказности сложных технических систем. М.: Логос, 2001. 208 с.
2. Байхельт Ф., Франкен П. Надежность и техническое обслуживание. М.: Радио и связь, 1988. 392 с.
3. Галеев А.Г., Золотов А.А., Перминов А.Н., Родченко В.В. Эксплуатация стартовых комплексов ракетно-космических систем. М.: Изд-во МАИ, 2007. 347 с.
4. Гнеденко Б.В. Математические методы в теории надежности. М.: Наука, 1965. 524 с.
5. Золотов А.А. Прикладные методы обеспечения работоспособности технических систем на этапе их эксплуатации // Оборонный комплекс – научно-техническому прогрессу России. 2010. № 1. С. 17–25.
6. Золотов А.А., Матвеев Ю.А., Шавич С.К. Методы оценки и обеспечения безопасности ракетно-космической техники при разработке. М.: Изд-во МАИ, 2002. 67 с.
7. Калошин А.М. Наземная отработка космических аппаратов. М.: КомКнига, 2005. 176 с.

Золотов Александр Алексеевич — д-р техн. наук, профессор Московского авиационного института (национального исследовательского университета) (Российская Федерация, 125993, Москва, Волоколамское шоссе, д. 4).

Нуруллаев Эльмар Джаннаталиевич — аспирант Московского авиационного института (национального исследовательского университета) (Российская Федерация, 125993, Москва, Волоколамское шоссе, д. 4).

Просьба ссылаться на эту статью следующим образом:

Золотов А.А., Нуруллаев Э.Д. Прогнозирование числа отказов программного обеспечения космических аппаратов // Вестник МГТУ им. Н.Э. Баумана. Сер. Машиностроение. 2016. № 5. С. 24–37. DOI: 10.18698/0236-3941-2016-5-24-37

FORECASTING THE NUMBER OF SPACECRAFT SOFTWARE FAILURES

A.A. Zolotov
E.D. Nurullaev

alexandrzolotov41@mail.ru
ned@mai.ru

Moscow Aviation Institute (National Research University), Moscow, Russian Federation

Abstract

Spacecraft software failures depend on software defects and errors. This paper offers a part of a solution for a software reliability problem by forecasting the failures. Using the experimental data we calculated the optimal quantity of software testing cycles. We used numerical algorithm to make a forecast of software reliability. Our forecast method is based on software failure formation analysis. It predicts the average quantity of software failures taking into account that the defects are eliminated after testing. The examples of typical cases provided in this article show that applying our method results in effective forecasting

Keywords

Software, reliability, software failure, quantity of testing cycles

REFERENCES

- [1] Aleksandrovskaya L.N., Afanas'ev A.P., Lisov A.A. *Sovremennye metody obespecheniya bezotkaznosti slozhnykh tekhnicheskikh sistem* [Modern methods of non-failure operation ensuring of complex technical systems]. Moscow, Logos Publ., 2001. 208 p.
- [2] Baihelt F., Franken P. *Reliability and maintenance*. Moscow, Radio i svyaz' Publ., 1988.
- [3] Galeev A.G., Zolotov A.A., Perminov A.N., Rodchenko V.V. *Ekspluatatsiya startovykh kompleksov raketno-kosmicheskikh sistem* [Operation of space-rocket systems starting complexes]. Moscow, MAI Publ., 2007. 347 p.
- [4] Gnedenko B.V. *Matematicheskie metody v teorii nadezhnosti* [Mathematical methods in theory of reliability]. Moscow, Nauka Publ., 1965. 524 p.
- [5] Zolotov A.A. Applied methods of ensuring operability of technical systems at operational stage. *Defensive complex — to scientifically technical progress of Russia*, 2010, no. 1, pp. 17–25 (in Russ.).
- [6] Zolotov A.A., Matveev Yu.A., Shaevich S.K. *Metody otsenki i obespecheniya bezopasnosti raketno-kosmicheskoy tekhniki pri razrabotke* [Assessment and providing methods of space-rocket equipment at developing]. Moscow, MAI Publ, 2002. 67 p.
- [7] Kaloshin A.M. *Nazemnaya otrabotka kosmicheskikh apparatov* [Spacecrafts ground workout]. Moscow, KomKniga Publ., 2005. 176 p.

Zolotov A.A. — Dr. Sci. (Eng.) Professor of Moscow Aviation Institute (National Research University) (Volokolamskoe shosse 4, Moscow, 125993 Russian Federation).

Nurullaev E.D. — post-graduate student of Moscow Aviation Institute (National Research University) (Volokolamskoe shosse 4, Moscow, 125993 Russian Federation).

Please cite this article in English as:

Zolotov A.A., Nurullaev E.D. Forecasting the Number of Spacecraft Software Failures. *Vestn. Mosk. Gos. Tekh. Univ. im. N.E. Baumana, Mashinostr.* [Herald of the Bauman Moscow State Tech. Univ., Mech. Eng.], 2016, no. 5, pp. 24–37. DOI: 10.18698/0236-3941-2016-5-24-37