

шей СЧ, при которой получены рабочие температуры ЭРИ, приведенные в картах рабочих режимов (КРР); K_n – коэффициент нагрузки.

Надежностная макро модель СЧ (модель эксплуатационной интенсивности отказов) представляет собой дискретную трехмерную функцию, заданную ее значениями. Однако, в этом случае в базе данных придется хранить относительно большое число параметров макро модели (порядка двух тысяч) [6].

После некоторых преобразований, позволяющих представить функцию для расчета λ_j как функцию одного аргумента (температуры) с некоторым набором коэффициентов, получим:

$$\lambda_j = \lambda_{j0} \cdot k_{zj} \cdot r_j \cdot f_j\{b_{j1}, b_{j2}, \dots, b_{jM}, T\}, \quad (4)$$

где r_j – значение коэффициента, определенное для j -той координаты вектора K_n ; f_j – аппроксимирующая функция, вид которой определен для j -той координаты вектора K_n ; $b_{j,m}$ – m -тый коэффициент, определенный для j -той координаты вектора K_n ; T – температура среды, окружающей СЧ.

Для поиска вида функции f_j были использованы несколько способов аппроксимации. Наилучшие результаты, с точки зрения точности аппроксимации, были получены в результате сплайн-интерполяции. Однако, с точки зрения числа коэффициентов (M) функций (что в данном случае является критерием выбора вида функции, а точность аппроксимации – лишь ограничением), лучшей оказалась нелинейная аппроксимация экспоненциальной регрессией.

Таким образом, применение аппроксимации $f_j(T)$ позволяет существенно снизить объем численных значений, хранящихся в БД. Как следует из (4), точность макро модели определяется шагом дискретизации коэффициента нагрузки и погрешностью аппроксимирующей функции. Однако очевидно, что применение таких макро моделей в расчетах надежности РЭА в итоге дает значительно меньшую погрешность (см. рис. 1), чем в случае, когда используется общепринятое допущение о том, что для СЧ одного типоразмера $\lambda_j \approx const$ (не зависит ни от условий применения, ни от нагрузки).

Список литературы

1. Надежность ЭРИ: Справочник. // С. Ф. Прытков, В. М. Горбачева, А. А. Борисов и др. / Науч. рук. С. Ф. Прытков – М.: 22 ЦНИИ МО РФ, 2002. – 574 с.
2. Автоматизация проектных исследований надежности радиоэлектронной аппаратуры: Научное издание // Жаднов В. В., Кофанов Ю. Н., Малютин Н. В. и др. – М.: Изд-во Радио и связь. 2003. – 156 с.

ПОКАЗАТЕЛИ НАДЕЖНОСТИ ПРОГРАММНЫХ СРЕДСТВ

С. Н. Полесский, В. В. Жаднов (научный руководитель)
 Московский государственный институт электроники и математики
 (Технический университет)
 109028, Москва, Б. Трехсвятительский пер., д. 1-3/12, стр. 8
 E-mail: serg@asonika-k.ru

Интерес к оцениванию надежности программных средств (ПС) возник одновременно с появлением самих программ. Он был вызван естественным стремлением получить традиционную вероятностную оценку надежности технического устройства (ЭВМ), работа которого в основном и предназначалась для функционирования ПС.

Для обеспечения надежности ПС предложено множество подходов, включая организационные методы разработки, различные технологии и технологические программные средства, что требует, очевидно, привлечения значительных ресурсов. Однако отсутствие

общепризнанных критериев надежности не позволяет ответить на вопрос, насколько надежнее становится ПС при соблюдении предлагаемых процедур и технологий и в какой степени оправданы затраты. Таким образом, приоритет задачи оценки надежности должен быть выше приоритета задачи ее обеспечения, чего на самом деле не наблюдается.

Судя по имеющимся публикациям [1, 2], вопрос обеспечения надежности программ считается более важным, чем вопрос ее оценки. Многие авторы, не определяя номенклатуру показателей надежности ПС, пытаются их улучшить. Определим эти показатели.

Классическим определением надежности ПС является его свойство производить вычисления, сохраняя значения установленных эксплуатационных показателей в заданных пределах, соответствующих заданным режимам и условиях использования, обслуживания, ремонта (коррекции), хранения и копирования [2].

Наиболее распространенным показателем надежности ПС является вероятность правильного функционирования при определенных условиях. С ним непосредственно связан другой показатель – средняя наработка на отказ. Третий показатель, характеризующий надежность ПС, – это функция интенсивности отказов, с помощью которой определяется распределение отказов по какому-то аргументу, например, во времени.

Кроме этих трех показателей, взятых из общей теории надежности, в принципе возможны и другие, например, коэффициент готовности, количество оставшихся в программе первичных ошибок и т. п., необходимость введения которых определяется потребностями системы, использующей ПС.

Комплексы программ создаются, эксплуатируются и развиваются. Жизненный цикл программ охватывает весь процесс существования программы – от возникновения потребности в ней до полного прекращения использования этой программы вследствие ее морального старения или потери необходимости решения соответствующих задач.

В общем случае жизненный цикл программ состоит из трех фаз [3, 4, 5]:

- разработка ПС;
- эксплуатация (использование) ПС;
- сопровождение (продолжающаяся разработка) ПС.

За фазой разработки следуют сразу две фазы – эксплуатация и сопровождение, которые протекают, как правило, одновременно (рис. 1).

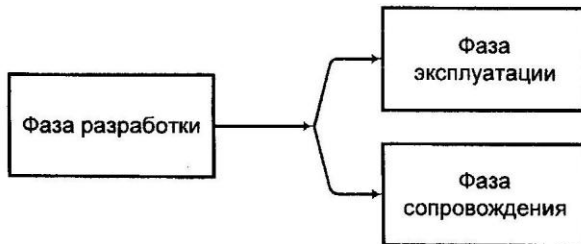


Рис. 1. Фазы жизненного цикла ПС

Для рассмотренных выше показателей надежности математических моделей не существует, но для конкретной практической задачи, например, расчет показателя надежности программного комплекса по известной надежности его составляющей, существует [1]. Мы и рассмотрим его.

Дан ПК, состоящий из M отдельных модулей, соединенных между собой. По структуре ПК строится стохастический граф, содержащий $M+2$ вершин. Вершина 0 озна-

чае исток, а вершина $M+1$ – сток графа. Каждый ПМ вызывается на решение с заданной вероятностью, исходя из цели функционирования или значений исходных данных.

В нашей же задаче должен выполняться принцип слабого звена, свойственный основному соединению элементов в теории надежности. Поэтому неправильно применять преобразования Лапласа плотностей распределений времени до ошибок в модулях. Необходимо вместо них поставить вероятности правильной работы соответствующих модулей, производящих вычисления на заданных временных интервалах. При этом мы предполагаем, что модули статистически независимы.

Рассмотрим матрицу:

$$G = G(t), t = (t_0, t_1, \dots, t_{M+1}),$$

элементами которой являются произведения:

$$p_{ij} P_i(t_i), i, j = (0, \dots, M+1),$$

где p_{ij} – вероятность перехода от i -го ПМ к j -му; $P_i(t_i)$ – вероятность безошибочного функционирования i -го ПМ в течение времени t_i .

Так как нулевая и $M+1$ -ая вершины фиктивные, то предполагаем, что время нахождения в них равно нулю, а вероятности безошибочной работы – единице. Введем понятие шага, подразумевая под ним единичный переход от одного ПМ к другому. Построим матрицу вида:

$$T = I + G(t) + G^2(t) + \dots = I(I - G(t))^{-1},$$

где I – единичная матрица.

Элемент матрицы T с номером $(0, M+1)$ представляет собой выражение для вероятности безошибочной работы всего ПК с учетом всех возможных последовательностей вызовов отдельных ПМ.

В соответствии с правилами вычисления значений элементов обратной матрицы, выражение для вероятности безошибочной работы ПК можно представить в виде:

$$Y(t) = Q(t) / R(t),$$

где $Q(t)$ – алгебраическое дополнение элемента с номером $(M+1, 0)$ матрицы $(I - G(t))$; $R(t)$ – главный определитель матрицы $(I - G(t))$.

Выполнив указанные преобразования, получим искомое выражение для вероятности безошибочного функционирования ПК с учетом задействования всех возможных маршрутов вычислений.

Получение более надежных ПС, в основном, зависит от выбора технологии проектирования. Если оценить безошибочность, правильность функционирования и взаимосвязанность входящих в ПС модулей еще на ранних стадиях, это приведет к улучшению качества ПС и, вследствие чего, понижению стоимости самого продукта.

Список литературы

1. Кузнецов В. В., Смагин В. А. Прямая и обратная задачи надежности сложных программных комплексов // Надежность и контроль качества. 1997. № 10. с. 56-62.
2. Майерс Г. Надежность программного обеспечения. Москва, Мир, 1980.
3. Липаев В. В. Тестирование программ. – М.: Радио и связь, 1986.
4. Липаев В. В. Качество программного обеспечения. – М.: Энергоиздат, 1981.
5. Липаев В. В. Надежность программных средств. – М: Синтер, 1998.