УДК 681.3.06

А. М. Андреев, Г. П. Можаров

# ИСПОЛЬЗОВАНИЕ РЕЗУЛЬТАТОВ ТЕОРИИ МАРТИНГАЛОВ ДЛЯ ОЦЕНКИ НАДЕЖНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОМПЬЮТЕРНЫХ СИСТЕМ И СЕТЕЙ

*Важность повышения надежности программного обеспечения обусловлена тем, что оно выполняет основные функции системного управления обработкой данных, и его отказы в работе могут существенно влиять на функционирование систем обработки данных в целом. Исследованы и разработаны подходы к математическому моделированию надежности программного обеспечения компьютерных систем и сетей, рассмотрены прогнозирование надежности, оценочные модели и измерение надежности программно-математического обеспечения. Для построения моделей надежности программного обеспечения использованы результаты теории мартингалов. Благодаря полученным условным распределениям (посредством решения задач фильтрации) можно охарактеризовать критерии надежности программного обеспечения.*

**E-mail: arkandreev@gmail.com**

***Ключевые слова***: *надежность и отказы программного обеспечения, компьютерные системы, критерии надежности, мартингалы в моделях ошибок программного обеспечения.*

Надежность — один из важнейших факторов, определяющих общую производительность и эффективность систем. Уже на стадии проектирования системы вопросам надежности должно уделяться пристальное внимание. В этот период, когда устанавливается первоначальная взаимозависимость между характеристиками системы, затратами и графиком выполнения работ, должны быть сформулированы и требования к надежности, так как именно они в значительной мере определяют реализуемость проекта и стоимость будущей системы.

Теория надежности как наука получила развитие применительно к сложным техническим системам. Необходимость и полезность контроля технических компонентов систем и систем в целом в целях проверки соответствия их текущих характеристик заданным доказаны практикой. В этой области выполнено значительное число работ по надежности применительно к техническим системам, разработано множество моделей обеспечения разумными методами надежности сложных систем и их технической готовности.

Эти модели в ряде случаев позволяют не только оценивать показатели надежности и готовности технических систем и их компонентов, но и дают возможность предсказывать значения этих показателей на основе накопленного опыта. Кроме того, ряд моделей позволяет на

основе накопленных данных высказывать предположения в отношении режимов работы, при которых наиболее часто проявляются отклонения от нормального функционирования, а также о применяемом подходе к восстановлению (ремонту) системы или ее компонентов после сбоя.

Вопросы анализа надежности сложных параллельных компьютерных систем (КС) всегда были в центре внимания проектирования перспективных КС и сетей (КСС). Параллельные КСС относятся к классу сложных разветвленных систем. Они состоят из элементов, узлов, центров и линий связи, образующих вполне определенные структуры. Поэтому условно в проблеме анализа надежности многопроцессорных вычислительных сетей (ВС) выделяют аспекты элементной и структурной (комбинаторной) надежности.

В современных работах по теории надежности недостаточно освещена надежность КСС, в связи с чем некоторые интересные и сильные в теоретическом и практическом отношении результаты, касающиеся непосредственно тематики комбинаторной надежности, не нашли отражения в работах авторов монографий и отдельные важные для теории надежности вопросы не попали в поле зрения авторов статей по теории надежности.

Критерии надежности представляют собой показатели, позволяющие оценить предпочтительность тех или иных решений при создании и эксплуатации системы по степени достижения основных целей с учетом затрат, при которых эти цели достигаются. При исследовании надежности основная цель заключается в разработке эффективных методов и обеспечении длительной работоспособности систем с заданными функциональными характеристиками. Для этого необходимо установить количественные показатели, характеризующие не только факт работоспособности системы, но и степень соответствия работоспособности основным требованиям, отраженным в технической документации.

Основная задача теории надежности на этапе технического проектирования — это помочь разработчику принять обоснованные решения, касающиеся выбора структуры КС, необходимости использования и мощности вводимой избыточности, построения оптимальной системы контроля и т.д.

В КС компьютер, как часть системы, обычно выполняет функции управления и должен работать в реальном масштабе времени. В типовых КС компьютер выполняет бесконечный цикл, в котором сначала считываются сигналы и показания датчиков и сенсоров, затем затрачивается определенное время, чтобы вычислить или спланировать некоторый отклик или реакцию на воздействие, и в конце цикла компьютер

выполняет эту реакцию. Очевидно, что структура одного периода цикла управления может быть и гораздо сложнее. Однако укрупнение различных операций обычно сводится к описанным этапам. Надежность КС может определяться как последовательное соединение статистически независимых аппаратного и программного компонентов системы. Но наиболее подходящей мерой надежности является вероятность того, что система выполняет свою миссию или справляется с функциями управления в течение заданного времени при условии взаимодействия аппаратуры и программы. Надежность такой системы определяется надежностью аппаратной и программной частей в их взаимодействии. Отказы аппаратуры происходят вследствие многих причин: износа компонентов, сбоев, короткого замыкания, обрывов и т.п. Причинами отказов ПО являются: наличие ошибок в программе; использование неоптимальных и несовершенных алгоритмов, таких как эвристические, приближенные, численные; ограничения на функционирование в реальном масштабе времени.

Остановимся подробнее на некоторых факторах, влияющих на надежность компьютерной системы.

1. *Надежность элементов аппаратуры.* Компоненты аппаратной части КС имеют самые различные механизмы отказов. Некоторые из них могут быть вызваны воздействием программной части системы при чрезмерном стрессовом использовании каких-либо компонентов аппаратуры, особенно имеющих механические элементы функционирования. Примерами этого являются: длительное и непрерывное использование принтера, интенсивные чтение и запись на жесткий диск, частое изменение режимов работы дисплея. Таким образом, можно утверждать, что во многих случаях отказы ПО и техники компьютера являются событиями зависимыми.

2. *Влияние программы на надежность аппаратуры.* Например, рассмотрим двухфункциональную систему. Один план управления требует, чтобы обе функции были активными для расчета реакции на входные данные, в то время как другой требует, чтобы только одна функция была активной. Когда программа влияет на надежность аппаратуры таким образом, можно говорить только об условной статистической независимости аппаратной и программной частей.

3. *Отказы в программе.* Если программное обеспечение не модифицируется, то интенсивность его отказов остается постоянной вследствие оставшихся в ней необнаруженных ошибок.

4. *Внутренние отказы программы.* Такие отказы обусловлены фундаментальными ограничениями алгоритма, используемого в ПО. Например, использование эвристик может привести к случайным отказам, даже если в программе отсутствуют ошибки.

5. *Отказы, обусловленные ограничением на функционирование в реальном масштабе времени.* В рассматриваемых системах среда может изменяться динамически. Поэтому если время планирования или расчета отклика слишком велико, то к моменту выполнения отклика среда может быть уже измененной так сильно, что вычисленный или спланированный отклик не будет иметь требуемого эффекта. Эти отказы часто характеризуются неспособностью системы функционировать в ограничениях на реальное время.

Следует отметить, что перечисленные факторы влияют друг на друга. Например, имеется противоречие между устранением внутренних отказов и отказов, обусловленных ограничением на функционирование в реальном масштабе времени. Использование оптимального и высокоточного алгоритма решения какой-либо задачи требует значительных затрат времени для получения решения, что может привести к нарушению ограничений, связанных с реальным временем функционирования системы. В то же время, использование рационального (но не оптимального) или приближенного эвристического алгоритма позволит преодолеть проблему реального времени, однако может привести к внутренним ошибкам. Таким образом, анализ надежности КС является достаточно сложной задачей, на которую влияет большое число факторов. И очевидно, что ее решение необходимо выполнять поэтапно: рассмотреть в отдельности сначала аппаратную часть, затем программную и только после этого анализировать надежность с учетом взаимодействия всех компонентов.

На практике модели надежности могут применяться для выработки стратегии управления разработкой ПО, оценки соответствия ПО системным требованиям, определения эффекта от модернизации ПО и ввода ПО в эксплуатацию, оценки надежности готового программного продукта заказчиком и т.д.

Наиболее распространенная сфера применения моделей — это определение оптимальной продолжительности тестирования ПО.

Оценочные модели надежности — наиболее распространенный и развитый тип моделей. Они служат для априорного оценивания надежности по серии тестовых прогонов и обычно используются на этапе тестирования и отладки. По поведению программы в тестовой среде определяется вероятность отказа в операционной среде. Предполагается, что тестовая и операционная среды связаны известными соотношениями. К оценочным моделям относятся экспоненциальные и байесовские модели.

Экспоненциальные модели основаны на следующих предположениях: интенсивность отказов пропорциональна числу ошибок, оставшихся в программе; влияние всех ошибок на работу программы одинаково (интервалы времени между моментами обнаружения ошибок

независимы); обнаруженные ошибки исправляются и при этом не вносятся новые ошибки.

Однако эти модели имеют довольно простую структуру, малый объем выборок реального числа обнаруженных ошибок и большой разброс времени обнаружения последовательных ошибок при завершении отладки не позволяют построить высокоточные математические модели.

Рассмотрим использование результатов теории мартингалов для построения моделей надежности ПО. Покажем, как мартингалы могут быть использованы в принятой модели ошибок и при решении проблем оценки характеристик надежности ПО. С помощью полученного в результате решения задач фильтрации условного распределения можно определить некоторые критерии надежности ПО.

Основная проблема состоит в том, чтобы определить соответствующий критерий надежности, алгоритм и методику для практического вычисления числа оставшихся ошибок в ПО (в течение периода тестирования). Однако, модели, используемые при моделировании ПО, имеют довольно простую структуру; малый объем выборок реального числа обнаруженных ошибок и большой разброс времени обнаружения последовательных ошибок при завершении отладки не позволяют построить высокоточные математические модели.

Применение теории мартингалов позволяет создать модель, надежности ПО, включающую в себя большинство существующих моделей. Покажем, как мартингалы могут быть использованы в принятой модели ошибок и при решении проблем оценки характеристик надежности ПО.

Исследуем общую модель, включающую в себя большинство существующих моделей. При вычислении критериев надежности и оценке параметров возникает задача, аналогичная задаче фильтрации, и тогда в качестве естественного инструмента решения используют теорию мартингалов [1]. Принимаем, что интенсивность ошибок постоянна и равна $v$. Тогда вероятность ошибки, встречающейся в интервале времени $(t, t + \Delta t)$, может быть записана в следующем виде:

$$P(t) = v\Delta t + O(\Delta t), \qquad (1)$$

считается, что любые две ошибки встречаются $s$-независимо одна за другой в данном интервале времени.

Интенсивность ошибок $v$ описывает одновременно случайность совокупности путей и расположения ошибки в структуре ПО. Например, ошибка в процедуре вывода будет иметь большее значение интенсивности $v$, в то время как ошибки программы, которые являются менее вероятными, будут иметь меньшее значение $v$. При этом возможные ошибки могут быть распределены по классам $k$ $(k \leqslant \infty)$

так, что одинаковое значение $v_j$ будет соответствовать всем ошибкам в классе $j$. Предполагая $s$-независимость ошибок, мы должны только рассмотреть число $X_{jt}$ остающихся ошибок в классе $j$ за время $t$. С учетом выражения (1) получаем, что вероятность возникновения ошибки $j$-го класса, встречающейся в интервале $(t, t+\Delta t)$, может быть записана уравнением вида

$$P(t) = v_j X_{jt} \Delta t + O(\Delta t). \tag{2}$$

Из выражения (2) следует, что мы можем моделировать для каждого класса $j$ процесс генерации ошибок $n_{jt}$ как процесс вычисления полумартингала:

$$dn_{jt} = \lambda_{jt} dt + dm_{jt}, \quad n_{j0} = 0, \tag{3}$$

где $\lambda_{jt} = v_j X_{jt}$ и $m_{jt}$ — независимые мартингалы [1]. Для более компактной формулировки определим следующие векторные процессы:

$$\mathbf{N}_t = [n_{1t}, \ldots, n_{Kt}]^{\mathrm{T}}, \quad \mathbf{X}_t = [X_{1t}, \ldots, X_{Kt}]^{\mathrm{T}},$$
$$\mathbf{M}_t = [m_{1t}, \ldots, m_{Kt}]^{\mathrm{T}}$$

и матрицу $\mathbf{A} = \operatorname{diag}(v_1, \ldots, v_K)$. Тогда, в соответствии с (3), можно получить уравнение ошибок

$$d\mathbf{N}_t = \mathbf{A}\mathbf{X}_t dt + d\mathbf{M}_t, \quad (\text{при } \mathbf{N}_0 = 0). \tag{4}$$

Уравнение (4) справедливо при соблюдении следующего условия:

$$n_t = \sum_{j=1}^{K} n_{jt}.$$

И тогда уравнение (4) можно преобразовать к виду

$$dn_t = \lambda_t dt + dm_t, \quad n_0 = 0,$$
$$\lambda_t \equiv \sum_{j=1}^{K} \lambda_{jt}, \quad m_t \equiv \sum_{j=1}^{K} m_{jt}.$$

Модель процесса ошибки должна объяснить механизм для исправления после обнаружения как случайных, так и детерминированных ошибок. Действительно, полагая, что $d\mathbf{X}_t = -d\mathbf{N}_t$, приходим к следующей модели обработки ошибки:

$$d\mathbf{X}_t = -\mathbf{A}\mathbf{X}_t dt - d\mathbf{M}_t. \tag{5}$$

Модель ошибки, описываемая уравнениями (4) и (5), более общая, чем другие существующие модели, например описываемые уравнением вида

$$\frac{dn}{d\tau} + bn = bN_0$$

(где $N_0$ — число ошибок в начале отладки ПО при $\tau = 0$; после отладки в течение времени $\tau$ осталось $n_0$ ошибок и устранено $n$ ошибок ($n_0 + n = N_0$)) и имеющие тот же самый механизм исправления ошибок [2–5].

Решение уравнения (5) при известных значениях $N_s,\ \leqslant s \leqslant t$ или $n_s,\ \ 0 \leqslant s \leqslant t$, с одной стороны, служит для определения показателей надежности ПО — числа оставшихся ошибок в программе после отладки, среднего времени безотказной работы программы, с другой стороны, решение этого уравнения — это решение задачи фильтрации сигнала. Для некоторых случаев эта проблема строго сформулирована и решена далее.

*Случай одного отказа в программе.* В этом случае $(k = 1)$ наша модель (4), (5) может быть описана следующими уравнениями:

$$d\mathbf{n}_t = v\mathbf{X}_t dt + d\mathbf{m}_t,\ \ \mathbf{n}_0 = 0,$$

$$d\mathbf{X}_t = -v\mathbf{X}_t dt - d\mathbf{m}_t,\ \ \mathbf{X}_0 = 0.$$

Предположим, что $(\mathbf{X}_0, v)$ — случайная величина, и тогда задача фильтрации состоит в том, чтобы вычислить выражение для апостериорной функции распределения вероятностей $p_t$, значений $(X_t, v)$ при условии $\{n_s,\ \ 0 \leqslant s \leqslant t\}$. Решение этой задачи включает в себя и определение априорных детерминированных параметров функции распределения вероятностей $p_0$. Если они не известны, то они могут быть вычислены согласно правилу максимального правдоподобия.

Оценка максимального правдоподобия для функционала $\Lambda_t$ в этом случае вычисляется по формуле [5]:

$$\Lambda_t = \exp\left[-\int\limits_0^t \left(\widehat{\lambda}_s - 1\right) ds + \int\limits_0^t \log \widehat{\lambda}_{s^-} dn_s\right], \qquad (6)$$

где $\widehat{\lambda}_s$ является условным математическим ожиданием $\lambda_s$ данного процесса $n_\sigma,\ \ 0 \leqslant \sigma \leqslant s$: $\widehat{\lambda}_s = E\{\lambda_s | n_\sigma,\ 0 \leqslant \sigma \leqslant s\}$, которое может быть вычислено при условии известных значений $p_s$: $\widehat{\lambda}_s = \int xvp_s(x, v)\, d\mu(x, v)$.

Предположим, что наблюдается процесс $n_s\ \ (0 \leqslant s \leqslant t)$ и отказы произошли в моменты времени $t_j\ \ (j = 1, \ldots, n_t)$. Данные этих наблюдений получаются для условных выражений функции распределения вероятностей $p_t$ значений $(X_t, v)$:

$$p_t(x, v) = cp_0(x + n_t, v)\, \frac{(x + n_t)!}{x!} v^{n_t} \exp\left[-v\left(xt + \sum_j t_j\right)\right], \qquad (7)$$

где $c$ — такая величина, что $\int p_t(x, v)\, d\mu(x, v) = 1$.

*Специальные случаи* могут быть получены из (7) согласно различным выборам величины $p_0$.

а) Предположим, что $(\mathbf{X}_0, v)$ — $s$-независимые случайные величины, $\mathbf{X}_0 \sim \text{bin}\,(N,p)$, $v \sim \Gamma\,(\alpha,\beta)$. Тогда

$$p_0(x,v) = \frac{\binom{N}{x} p^x (1-p)^{N-x} \beta^\alpha v^{\alpha-1} e^{-\beta v}}{\Gamma\,(\alpha)},$$

и

$$p_t(x,v) = c_1 \left( \begin{array}{c} N-n_t \\ x \end{array} \right) p^x\,(1-p)^{N-n_t-x}\, v^{\alpha+n_t-1} e^{-v\left(\beta+xt+\sum\limits_j t_j\right)}. \quad (8)$$

Из выражения (8) следует, что $(X_t, v)$ не является $s$-независимым для любых $t > 0$ и распределение для $v$ — это смесь биномиального и гамма-распределений. Тогда условные средние значения вероятностей $p_t$ для $\mathbf{X}_t$ и $\lambda_t$ могут быть легко вычислены из выражения (8).

б) Если выбираем распределение Пуассона вместо биномиального распределения, то получаем вместо (8) выражение

$$p_t\,(x,v) = c\frac{\mu^x}{x!} e^{-\mu} v^{\alpha+n_t} e^{-v\left(\beta+xt+\sum\limits_j t_j\right)} \quad \text{при} \quad N \to \infty \quad \text{и} \quad Np \to \mu.$$

в) Предположим, что $v$ принято в качестве детерминированной величины и, следовательно, значения $v_0$ и $\mathbf{X}_0$ имеют априорное пуассоновское распределение. Тогда из уравнения (7) следует, что $\widehat{\lambda}_t = v_0 \widehat{\mathbf{X}}_t = v_0 \mu e^{-v_0 t}$ независимо от $n_t$. В результате получается модель, подобная приведенной в работе [3].

г) Предположим, что вместо совокупности программ есть одна программа. Тогда $\mathbf{X}_0$ может быть представлен как детерминированная величина (при этом $p = 1$ для случая а)). Тогда из уравнения (8) следует, что плотность распределения $X_t$ вырождается в величину $N - n_t$, и, следовательно, остается лишь функция плотности распределения для $\nu$, являющаяся гамма-функцией: $\Gamma\left[\alpha + n_t,\ \beta + (N-n_t)t + \sum\limits_j t_j\right]$.

Тогда

$$\hat{\lambda}_t = \frac{(N-n_t)(\alpha+n_t)}{\beta + (N-n_t)t + \sum\limits_j t_j}.$$

Используя этот результат вычисляют критерий надежности, состоящий из функции плотности вероятностей для последующего отказа. Поскольку дана определенная величина $\nu$, то плотность для $\theta_t$ принимает значение $\nu X_t e^{-\nu X_t \theta}$, $\theta > 0$, т.е. можно получить выражение для функции плотности вероятностей $\theta_t$ для следующего отказа, учитыва-

ющего предысторию процесса отказа:

$$\int_0^\infty v X_t e^{-v X_t \theta} p_t(v)\, dv = \frac{(\alpha + n_t)(N - n_t)\left(\beta + (N - n_t)\, t + \sum_j t_j\right)^{n_t+\alpha}}{\left(\beta + (N - n_t)(t + \theta) + \sum_j t_j\right)^{n_t+\alpha+1}}. \tag{9}$$

Из (9) следует, что если условное среднее $\theta_t$ существует, то тогда

$$\widehat{\theta}_t = \frac{\beta + (N - n_t)\, t + \sum_j t_j}{(n_t + \alpha - 1)(N - n_t)}.$$

Во всех этих случаях оценки максимального правдоподобия параметров в априорных распределениях (если они неизвестны) могут быть получены, подходящим выбором и подстановкой выражения для $\widehat{\lambda}_t$ в уравнение (6).

**Выводы.** Мартингалы могут быть полезны в принятой модели ошибок и процессов отказа при разных обстоятельствах; при решении задач оценки и вычислении критериев надежности. Этот подход теоретически обобщает и придает объединяющую роль структуре модели ошибок, которая не только содержит известные результаты, но также позволяет получать новые результаты.

## СПИСОК ЛИТЕРАТУРЫ


1. Дуб Дж. Л. Вероятностные процессы. – М.: ИЛ, 1956. – 605 с.
2. Липаев В. В. Надежность программного обеспечения АСУ. – М.: Энергоиздат, 1981. – 240 с.
3. Колозезный Э. А., Бужинский В. А., Динеев В. Г., Ковригин М. И., Колозезный А. Э., Можаров Г. П. Независимая экспертиза — основа сертификации программно-математического обеспечения // Тез. докл. 3-й Междунар. науч.-технической конференции “Космонавтика. Радиоэлектроника. Геоинформатика”. Рязан. гос. радиотехн. акад. Рязань. 2000.
4. Колозезный Э. А., Бужинский В. А., Динеев В. Г., Ковригин М. И., Колозезный А. Э., Можаров Г. П. Независимая экспертиза — основа сертификации программно-математического обеспечения изделий ракетно-космической техники // Космонавтика и ракетостроение. – 2001. – Вып. 24. – С. 154–162.
5. Ачильдиев В. М., Вязов С. М., Динеев В. Г., Ковригин М. И., Колозезный Э. А., Можаров Г. П., Теплова И. В., Цуцаева Т. В. Системы управления средств выведения космических аппаратов и их сертификация: Учеб. пособие для студентов специальности 210500 / Под ред. проф. Э.А. Колозезного. – М.: МГУЛ, 2002. – 158 с.


Статья поступила в редакцию 15.12.2011