Новиков Н.Н., Титов А.В., Колесникова Ю.А., Виноградов С.А.

ИССЛЕДОВАНИЕ МОДЕЛЕЙ НАДЕЖНОСТИ ФУНКЦИОНИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННО-УПРАВЛЯЮЩИХ СИСТЕМ ПРИ N-ВЕРСНОМ ПРОГРАММИРОВАНИИ

Надежность программного обеспечения сложных информационно-управляющих систем (ИУС), как следует из [1,3], является одной из основных характеристик его качества. Именно от надежности функционирования программного обеспечения (ПО) сложных ИУС зависят полнота выполнения задач и безопасность человеческих жизней. Свидетельством этому является значительное число аварий как отечественных, так и зарубежных объектов повышенной опасности. По мере усложнения объектов управления, увеличения их функций проблема обеспечения требуемого уровня надежности ПО становится все острее.

В настоящее время известно четыре основных направления обеспечения надежности ПО [1,2]. Первое направление - предупреждение ошибок - основано на минимизации или исключении ошибок. Второе - обнаружение ошибок - реализуется за счет способности ПО самостоятельно выявлять ошибки. Третье - исправление ошибок и их последствий. Четвертое - обеспечение устойчивости к ошибкам – основано на уникальном свойстве ПО, заключающемся в способности ПО функционировать при наличии ошибки, т.е. его отказоустойчивости.

Одним из самых распространенных подходов к реализации программной отказоустойчивости является методология избыточности. В теории надежности известно несколько форм избыточности: структурная (аппаратная); информационная, нагрузочная, функциональная, временная, комбинированная.

А.Авиженисом [7,8] была предложена концепция мультиверсионного программирования как одного из способов программной избыточности. Он определил мультиверсионное программирование как независимую генерацию $N \geq 2$ функционально эквивалентных программ, что означает изолированное создание версий программ независимыми специалистами с использованием различных спецификаций, алгоритмов и языков программирования [5]. Данная концепция позволяет изменять или заменять отказавшую программу при работе другой без вреда для ИУС.

При практической реализации мультиверсионного программирования возникает задача исследования его эффективности для различных ситуаций, характеризуемых комбинированием аппаратной и программной избыточностей, наличием или отсутствием диагностических процедур, как основных, так и резервных программ, наличием или отсутствием восстановления работоспособного состояния ПО в случае их отказа. При этом для $N \geq 2$ функционально эквивалентных программ под отказом ПО понимается событие, заключающееся в отказе всех программ или если время подключения резерва после отказа основной программы оказалось больше допустимого.

Если в ПО предусматривается система диагностирования состояния программ, то, как правило, собственно ее надежность считается идеальной. В качестве показателя надежности ПО принимается вероятность нахождения системы в работоспособном состоянии, называемое нами в дальнейшем вероятностью безотказной работы, $P(t)$.

В качестве показателя эффективности применения мультиверсионного программирования для повышения надежности ПО предлагается использовать величину прироста вероятности безотказной работы по сравнению с одноверсионным ПО:

$$\Delta P_N(t) = P_N(t) - P_1(t) \quad (1)$$

Где $P_1(t)$ - вероятность безотказной работы одноверсионного ПО СУ; $P_N(t)$ - вероятность безотказной работы N - версионного ПО СУ;

и темп роста вероятности безотказной работы:

$$W_N(t) = \frac{\Delta P_N(t)}{P_N(t)} \quad (2)$$

где: $\Delta P_N(t)$ - величина прироста вероятности безотказной работы N - версионного ПО ИУС.

На вероятность безотказной работы N-версионного ПО сложной ИУС оказывают влияние следующие основные факторы: время эксплуатации;n стратегия восстановления; конфигурация структуры ПО; стратегия диагностирования.

При построении математических моделей надежности функционирования ПО ИУС были приняты следующие допущения:

отказы программ обнаруживаются системой контроля мгновенно и достоверно;

потоки отказов и восстановлений Пуассоновские;

во время переключения резервная программа не отказывает;

переключатель программ с основной на резервную не идеальный;

восстановление основной программы начинается только после подключения резервной;

система из отказового состояния не восстанавливается.

Математическая модель надежности функционирования двухверсионного ПО сложной ИУС при принятых допущениях включает:

1. Временную диаграмму, представленную на рис.1,поясняющую работу двух версий программ без восстановления аппаратной части сложной ИУС, но с восстановлением программной [4]. Предположим, что в начале функционирования сложной ИУС работает первая (основная) программа, вторая находится в ненагруженном резерве. В процессе отказов и восстановлений роль этих программ (как основной, так и резервной) может измениться, т.е. вторая программа может стать основной, а первая – резервной.

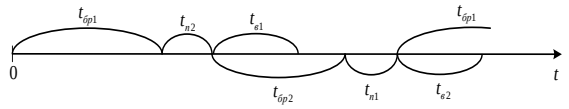

Рис.1. Временная диаграмма для двух версий программ

На диаграмме введены следующие обозначения:

$t_{бpi}$ - время безотказной работы i-й рабочей программы с постоянной интенсивностью отказов λ;

$t_{бpi\ рез}$ - время исправного состояния программы находящейся в резерве с постоянной интенсивностью отказов α;

t_{nj} -случайное время подключения резерва, имеющее экспоненциальное распределение с параметром ν;

$t_{\partial i}$ -допустимое время перерыва *i*-ой программы в работе без нарушения функционирования системы, имеющее экспоненциальное распределение с параметром δ.

t_{ei} - время восстановления *i*-й отказавшей программы с постоянной интенсивностью восстановления μ.

На временной диаграмме отсутствуют интервалы, соответствующие допустимому времени перерыва в работе, т.к. они не влияют на дальнейшую эволюцию процесса.

2. Граф состояний двухверсионного ПО сложной ИУС. В соответствии с [6] граф состояний содержит четыре работоспособных и два неработоспособных состояния (рис. 2):

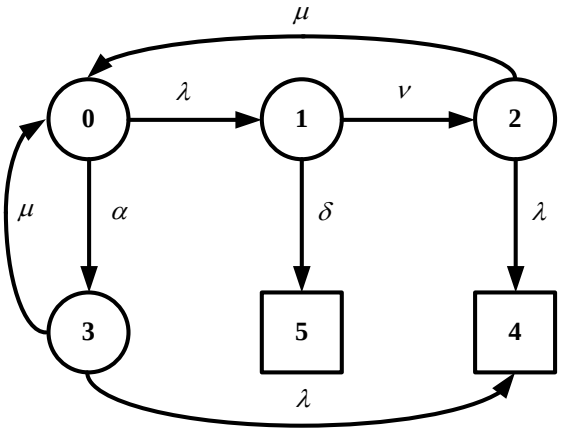

Рис.2. Граф состояний работы двухверсионного ПО сложной ИУС

0-обе программы работоспособны, работает основная;

1-основная программа отказала, происходит подключение резерва, но допустимое время перерыва не исчерпано;

2-резервная программа работоспособна и работает, основная отключена;

3-резервная программа отказала, при работающей основной, время перерыва отсутствует;

4-отказ сложной ИУС вследствие отказа всех программ;

5- отказ сложной ИУС вследствие несвоевременного подключения резерва.

3. Матрица переходных вероятностей системы:

$$\begin{bmatrix} -(\lambda+\alpha) & 0 & \mu & \mu & 0 & 0 \\ \lambda & -(\delta+\nu) & 0 & 0 & 0 & 0 \\ 0 & \nu & -(\lambda+\mu) & 0 & 0 & 0 \\ \alpha & 1 & 2 & -(\lambda+\mu) & 0 & 0 \\ 0 & 0 & \lambda & \lambda & 0 & 0 \\ 0 & \delta & 0 & 0 & 0 & 0 \end{bmatrix}$$

4. Система дифференциальных уравнений, описывающая работу ПО:

$$\begin{cases} \frac{dp_0(t)}{dt} = -p_0(t)\cdot(\lambda+\alpha)+\mu\cdot(p_2(t)+p_3(t)) \\ \frac{dp_1(t)}{dt} = \lambda p_0(t) - p_1(t)\cdot(\nu+\delta) \\ \frac{dp_2(t)}{dt} = \nu p_1(t) - p_2(t)\cdot(\lambda+\mu) \\ \frac{dp_3(t)}{dt} = \alpha p_0(t) - p_3(t)\cdot(\lambda+\mu) \\ \frac{dp_4(t)}{dt} = \lambda\cdot(p_3(t)+p_2(t)) \\ \frac{dp_5(t)}{dt} = \delta p_1(t) \end{cases} \quad (3)$$

где: p_i - вероятность нахождения ПО сложной ИУС в *i*-м состоянии

Начальные условия: $p_0(0)=1; p_i=0; i>0$. Вероятность отказа сложной ИУС имеет следующий вид:

$$Q_c(t) = p_4(t) + p_5(t) \quad (4)$$

Исследование эффективности применения мультиверсионного программирования на основе представленной выше модели, проводилось при следующих исходных данных (характерных для ПО сложных ИУС):

$\lambda = 1\cdot10^{-4}$ *(1/с)*; $\mu = 5\cdot10^{-4}$ *(1/с)*; $\delta = 0,9\cdot10^{-4}$ *(1/с)*; $\alpha = 7\cdot10^{-6}$ *(1/с)*; $\nu = 9\cdot10^{-4}$ *(1/с)*; $t = 10000$ *с*.

Вероятности безотказной работы определялись по следующим формулам:

-для одноверсионного ПО:

$$P_1(t) = p_0(t) \quad (5)$$

-для двухверсионного ПО:

$$P_2(t) = p_0(t) + p_1(t) + p_2(t) + p_3(t) \quad (6)$$

Темп роста вероятности безотказной работы двухверсионного ПО сложной ИУС определим по формуле:

$$W_2(t) = \frac{P_2(t) - P_1(t)}{P_2(t)}\cdot 100\% \quad (7)$$

Моделирование проводилось при помощи математического пакета Mathcad 11. Ниже, на рис.3, приведен график темпа роста вероятности безотказной работы двухверсионного ПО с восстановлением и без восстановления программ.

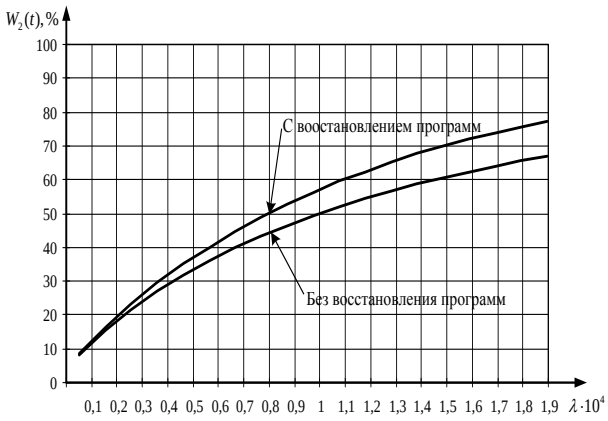

Рис.3. Темпы роста вероятности безотказной работы двухверсионного ПО с восстановлением и без восстановления программ

Таким образом, двухверсионное ПО дает существенный прирост вероятности безотказной работы, причем, чем больше интенсивность отказа программ, тем выше прирост вероятности безотказной работы. Например, при $\lambda = 1,5 \cdot 10^{-4}$ прирост составляет $\Delta P_2(t) = 0,3436$ для двухверсионного ПО без восстановления программ и $\Delta P_2(t) = 0,5172$ - с восстановлением программ. Темпы же роста вероятности безотказной работы составят соответственно 60% и 69%. В свою очередь, применение восстановления программ дает менее существенный прирост в вероятности безотказной работы и менее значимое увеличение темпа роста для двухверсионного ПО сложной ИУС, в сравнении с отсутствием восстановления, причем, чем ниже интенсивность отказа программ, тем ниже и прирост и темп роста вероятности безотказной работы. Дальнейшие исследования подтвердили зависимость прироста вероятности безотказной работы и ее темпа роста от числа версий ПО. Чем больше версий ПО, тем менее существенен прирост вероятности работоспособного состояния и ее темпа роста.

ЛИТЕРАТУРЫ

1. Боэм Б., Дж. Браун, Каспар Х и др. Характеристики качества программного обеспечения. - М.: Мир, 1981.

2. Гецци К., Джазайери М., Мандриоли Д. Основы инженерии программного обеспечения. 2-е изд.: Пер. с англ. – СПб.: БХВ - Петербург, 2005. – 832с.

3. Липаев В.В Системное проектирование сложных программных средств для информационных систем. М: СИНТЕГ, 2002, - 268с.

4. Половко А.М., Гуров С.В. Основы теории надежности. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург,2006.

5. Соммервилл Иан. Инженерия программного обеспечения. 6-е издание : Пер. с англ.- М: Издательский дом «Вильямс», 2002.

6. Черкесов Г.Н. Надежность аппаратно-программных комплексов. Учебное пособие. – СПб.:Питер,2005. – 479 с.

7. Avizienis A. The N-Version approach to fault-tolerant software/IEEE Trans. on Software Engineering. – Vol.SE-11, №12, December, 1985. – P.1491-1501.

8. Avizienis A. A methodology of N-version programming. – In: Software Engineering Tolerance (M.R.Lyu, ed.). – Chichester: John Wiley and Sons, 1995. – P. 23-69.