МЕТОДЫ РАСЧЁТА КОЛИЧЕСТВЕННЫХ ПОКАЗАТЕЛЕЙ НАДЁЖНОСТИ СЛОЖНЫХ ПРОГРАММНЫХ КОМПЛЕКСОВ НА СТАДИИ ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ

И. С. Новиков

Федеральное государственное учреждение «24 Центральный научно-исследовательский институт Министерства Обороны Российской Федерации»

ФГУ «24 ЦНИИ МО РФ», ул. Разводная, д. 17, Санкт-Петербург,198516.

УДК 681.3

Новиков И. С. **Методы расчёта количественных показателей надёжности сложных программных комплексов на стадии проектирования и разработки** // Труды СПИРАН. Вып. 6. — СПб.: Наука, 2008.

Аннотация. *Предложены методы расчёта количественных показателей надёжности специального программного обеспечения на стадии его проектирования и разработки. Для моделирования вычислительного процесса и расчёта количественных показателей его надёжности предлагается использовать математический аппарат полумарковских процессов. Моделирование производится в компьютерной математической системе MathCAD.* — Библ. 10 назв.

UDC 681.3

Novikov I. S. **Methods of calculation of quantity indicators of reliability of difficult program complexes on a design stage and development** // SPIIRAS Proceedings. Issue. 6. — SPb.: Nauka, 2008.

The summary. *Methods of calculation of quantity indicators of reliability of the special software at a stage of its designing and development are offered. For modelling computing process and calculation of quantity indicators of its reliability it is offered to use the mathematical device semi-Markov processes. Modelling is made in computer mathematical system MathCAD.* — Bibl. 10 items

1. Введение

В работе [1] математически строго доказано, что если при исправлении обнаруженной программной ошибки в отлаженной программе имеется конечная вероятность внесения новой программной ошибки, то никогда всех ошибок в программе не исправить. Поэтому проектирование программных комплексов проводится с учётом необходимости обеспечения безотказности вычислительных процессов при наличии оставшихся не обнаруженными при отладках и тестировании программных ошибках.

На этапах проектирования специального программного обеспечения (СПО) производится расчёт количественных показателей его надежности.

Под надёжностью программного обеспечения понимается свойство программ обеспечивать безотказность вычислительного процесса.

Под отказом вычислительного процесса понимается:

- наличие ошибок в результатах исполнения программы свыше допустимой величины;
- останов исполнения программы из-за ошибок как программных, так и инициируемых отказами и сбоями вычислительной техники;

- увеличение длительности исполнения программы свыше допустимого предела (зацикливание).

Под вычислительным процессом понимается процесс выполнения комплекса программ на определённом множестве вычислительных средств.

При наличии в программном комплексе СПО программных (программно-аппаратных) средств, обеспечивающих автоматическое выявление программных ошибок, их исправление и восстановление вычислительного процесса критерием отказа предлагается считать состояние отказа вычислительного процесса из-за проявления программной ошибки, если время автоматизированного восстановления вычислительного процесса превышает допустимую величину.

Количественные показатели надёжности программного обеспечения (КПН ПО) можно разбить на две группы. Первая группа показателей позволяет оценить безошибочность программ, вторая – безотказность вычислительного процесса в целом, учитывая отказы, вызванные как программными ошибками, так отказами и сбоями вычислительной техники. Перечень количественных показателей надёжности программного обеспечения приведён в табл. 1.

Таблица 1

Количественные показатели надёжности программного обеспечения

	Наименование показателей	**Обозначение**
Показатели для оценки безошибочности программы		
1	Математическое ожидание числа ошибок в программе	M_0
Показатели для оценки безотказности вычислительного процесса		
2	Вероятность безотказного исполнения вычислительного процесса в течение времени t	$P_{\text{би}}(t)$
3	Среднее время до отказа в исполнении вычислительного процесса	$T_{\text{п}}$

2. Метод предварительного расчёта КПН программ (на этапах эскизного проектирования)

На этапе эскизного проектирования программ и программных комплексов производится предварительный расчёт количественных показателей их надежности.

Вероятность безошибочного исполнения i программы рассчитывается по формуле

$$p_i = \int_0^\infty G_{\eta i}(t)\, d\, G_{\beta i}(t) = \int_0^\infty [1 - G_{\beta i}(t)]\, d\, G_{\eta i}(t), \qquad (1)$$

Где $G_{\eta i}(t)$ — функция распределения времени выполнения i программы $\tilde{t}_{\eta i}$; $G_{\beta i}(t)$ — функция распределения времени до проявления программной ошибки $\tilde{t}_{\beta i}$ в этой программе. Выражение (1) определяет вероятность того, что случайное

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

время выполнения i программы меньше случайного времени до проявления программной ошибки, то есть $t_{\eta i} < t_{\beta i}$.

Во многих работах [1, 2, 3, 4] показано, что функция распределения времени до проявления программной ошибки близка к экспоненциальному распределению с параметром, зависящим от числа оставшихся в программе ошибок.

В [2] для аппроксимации функции распределения нормированного времени ожидания до проявления программой ошибки предложено использовать гипергеометрический закон, который позволяет определить вероятность проявления программной ошибки в отдельной выборке из X команд (длительность выполнения программы в командах), принадлежащей конечному множеству N (объем программы в командах) и содержащей qN ошибочных команд; q — доля команд, могущих инициировать проявление программной ошибки.

Функция распределения нормированного времени до проявления программной ошибки в этом случае определяется формулой (2), полученной из гипергеометрического закона распределения времени ожидания [8]:

$$G_{qN}(x < X) = \sum_{x=0}^{X} \frac{\binom{N-x}{qN-1}}{\binom{N}{qN}} = \frac{qN[N(1-q)]!}{N!} \sum_{x=0}^{X} \prod_{i=1}^{qN-1} [N(1-q) - x + 1 - i] \qquad (2)$$

Математическое ожидание нормированного времени выполнения программы до проявления программной ошибки (в командах) и дисперсия определяются выражениями

$$M[T] = \frac{N+1}{qN+1}; \qquad (3)$$

$$D(T) = \frac{N(1-q)[N(1-q)-1]2}{(Nq+2)(Nq+1)} + \frac{3(N+1)}{qN+1} - \frac{(N+1)^2}{(qN+1)^2}. \qquad (4)$$

Распределение $G_{qN}(x < X)$ — дискретное (x — число выполненных операторов до первого проявления программной ошибки) и является случайной величиной. Каждый j оператор имеет конечное время выполнения τ_j, зависящее от производительности компьютера и языка, на котором написана программа. Если определить среднее время выполнения программы до первого проявления программной ошибки, например, по формуле $\bar{\tau} = \frac{\sum_{j=1}^{x} \tau_j}{x}$, то $x = \sum_{j=1}^{x} \frac{T_j}{\bar{T}}$ определяется как нормированное время.

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Для перехода к непрерывному закону распределения времени до проявления программной ошибки при исполнении i программы воспользуемся экспоненциальной аппроксимацией.

Аппроксимация гипергеометрического закона распределения времени ожидания проявления программной ошибки экспоненциальным распределением позволяет получить следующее выражение:

$$G_\beta(x < X) = 1 - e^{-\beta X}, \quad \beta = 1/M[T], \tag{5}$$

где $M[T]$ рассчитывается по формуле (3).

На рис. 1 показаны для конкретного примера типовые зависимости вероятности безошибочного исполнения программы от длительности исполнения программы для гипергеометрического закона ожидания времени проявления программной ошибки, рассчитанной по формуле

$$P_q(x > X) = 1 - G_{qN}(x < X)$$

и для экспоненциальной аппроксимации, рассчитанной по формуле

$$P_\beta(x > X) = 1 - G_\beta(x < X).$$

Рис. 1. Сравнение вероятности безошибочного исполнения программы от длительности исполнения программы для гипергеометрического закона ожидания и для его экспоненциальной аппроксимации.

Проверка по критерию согласия Колмогорова совпадения экспоненциальной аппроксимации функции распределения времени до проявления программной ошибки с гипергеометрической функцией распределения времени ожидания до проявления программной ошибки с тем же математическим ожиданием показала уровень согласия более 0.99. Это позволяет вполне обоснованно использовать

экспоненциальную аппроксимацию времени проявления программной ошибки с параметром $M(T)$ в реальных расчётах.

Определим функцию распределения времени до проявления программной ошибки i программы экспоненциальным законом

$$G_{\beta i}(t) = 1 - e^{-\beta_i \cdot t}, \quad \beta_i = \frac{k_i v}{M_i[T]} = \frac{k_i v (N_i q_i + 1)}{N_i + 1}, \tag{6}$$

где v — быстродействие ЭВМ оп/с.,

N_i — объём i программы в командах, $q_i = d_i / N_i$,

d_i — математическое ожидание числа программных ошибок в i программе,

k_i — коэффициент, учитывающий вероятность сочетания входных данных, при которых оператор — потенциальный носитель ошибок, вызовет ошибку результата в i программе.

Пусть n_i — длительность выполнения i программы (среднее число усредненных операторов, необходимых для выполнения i программы).

Тогда среднее время выполнения i программы определяется выражением

$$T_{\eta i} = \frac{n_i}{v}, \quad \eta_i = \frac{1}{T_{\eta i}}. \tag{7}$$

Рассмотрим два случая распределения времени выполнения программы: по экспоненциальному закону и по вырожденному

$$G_{\eta i}(t) = 1 - e^{-\eta_i t}, \qquad G_{\eta i}(t) = \begin{cases} 0, & t < T_{\eta i} \\ 1, & t \geq T_{\eta i} \end{cases}. \tag{8}$$

Подставляя выражения $G_{\eta i}(t)$ и $G_{\beta i}(t)$ с учетом (5) в формулу (1), можно получить для двух случаев распределения времени выполнения программы вероятность безошибочного исполнения i программы при однократном обращении.

$$P_i^1 = \frac{1}{1 + k_i n_i / M_i[T]} = \frac{1}{1 + \dfrac{k_i n_i (N_i q_i + 1)}{N_i + 1}}; \tag{9}$$

$$P_i^2 = e^{-\beta_i T_{\eta i}} = exp\left\{ \frac{k_i n_i (N_i q_i + 1)}{N_i + 1} \right\}. \tag{10}$$

Вероятность безошибочного функционирования всего программного комплекса в течение времени t рассчитывается по формуле при регулярном потоке обращений к программам

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

$$P(t) = \prod_{i=1}^{m} P_i^{a_i(t)} \qquad (11)$$

где $a_i(t)$ — число обращений к i программе в течение времени t.

При случайном пуассоновском потоке обращений

$$P(t) = exp\left\{-\sum_{i=1}^{m} a_i(1-P_i)t\right\} \qquad (12)$$

где P_i рассчитывается по одной из формул (9) или (10).

Число ожидаемых ошибок в программе можно определить по эмпирическим формулам.

В. В. Липаев [5] предлагает определять общее число ошибок d_i в i программе по формуле

$$d_i = 0{,}012 D_i$$

где D_i — общее число передач управления в i программе.

А. В. Александровым [6] предложена формула, базирующаяся на использовании метрик Холстеда [9]:

$$d_i = \frac{(H_{Ci} + H_{Ei})log(C_i + E_i)}{3000},$$

где H_{Ci} — суммарная частота использования операторов в i программе,

H_{Ei} — суммарная частота использования операндов в i программе,

C_i и E_i — количество различающихся операторов и операндов в i программе.

Л. Г. Осовецким [7] предложена похожая формула

$$d_i = \frac{v_i}{3000} 2 log_5 \left[\frac{v_i}{1300}\right],$$

где

$$v_i = N_i log_2 h_i, \quad h_i = C_i + E_i, \quad N_i = H_{Ci} + H_{Ei}.$$

Г. Майерс [3] предлагает определять для программ средних размеров число возможных ошибок в программе по формуле $d_i = 0.001 N_i \quad (q_i = 0.001)$.

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

По данным НПО «Центрпрограммсистем» для полностью отлаженных программ $k = 0.55 \cdot 10^{-8}$.

При принятых константах задача реализована в программе на ПЭВМ и выдает вполне удовлетворительные результаты.

3. Методы уточнённого расчёта КПН вычислительного процесса на этапе технического проектирования

3.1. Расчёт КПН линейных вычислительных процессов

Рассматривается последовательное выполнение n программных модулей. Время выполнения i модуля имеет в общем случае случайную величину, распределённую по закону $G_{\eta i}(t)$. Время проявления программной ошибки в i программном модуле имеет также случайный характер и распределено по закону $G_{\beta i}(t)$. В программе встроены программные средства для обнаружения ошибки, её исправления и восстановления вычислительного процесса. Случайное время исправления ошибки и восстановления процесса имеет распределение $G_{\mu i}(t)$. Проявлением отказа вычислительного процесса для всего программного комплекса будем считать превышение времени исправления ошибки и восстановления вычислительного процесса свыше допустимой величины, распределённой по закону $G_{gi}(t)$. Простейший вычислительный процесс (последовательное выполнение программных модулей) назовём линейным. Такой вычислительный процесс на практике встречается не часто, но для пояснения использования математического аппарата полумарковского процесса (ПМП) при построении функции распределения времени до отказа вычислительного процесса как пример вполне пригоден.

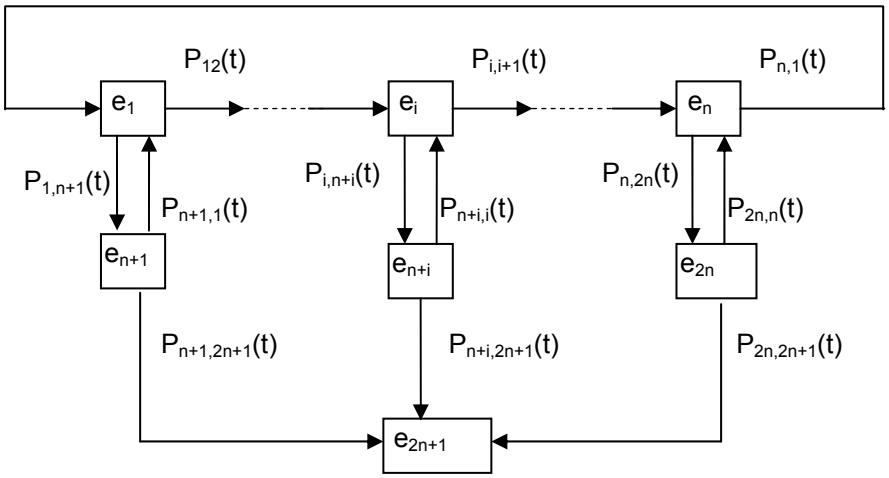

Рис. 2. Графосхема ПМП, описывающего вычислительный процесс до проявления программной ошибки (линейная структура вычислительного процесса).

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Графосхема динамики состояний ПМП, описывающего процесс проявления программной ошибки, показана на рис. 2.

ПМП, описывающий функционирование линейного вычислительного процесса до отказа, имеет $2n+1$ состояние.

Состояния e_i ($i=1...n$) определяются двумя событиями — исполнением программных модулей $1,2,...,i,...n$ и возможным проявлением программной ошибки.

Состояния e_{n+i} ($i=1...n$) определяются исправлением выявленной программной ошибки внедрёнными в программу специальными программными средствами восстановления вычислительного процесса.

Состояние e_{2n+1} определяется отказом вычислительного процесса.

В состоянии e_i ПМП будет находиться случайное время $\tilde{t}_i = min(\tilde{t}_{\eta i}, \tilde{t}_{\beta i})$, где $\tilde{t}_{\eta i}$ — случайное время выполнения i программного модуля, распределённое по закону $G_{\eta i}(t)$; $\tilde{t}_{\beta i}$ — случайное время проявления программной ошибки i модуля, распределённое по закону $G_{\beta i}(t)$. Функция распределения времени пребывания ПМП в состоянии e_i, очевидно, определяется выражением

$$F_i(t)=1-[1-G_{\eta i}(t)]\cdot[1-G_{\beta i}(t)].$$

Из состояния e_i ПМП может перейти в состояние e_{i+1}, если i программный модуль закончит процесс вычисления раньше, чем произойдёт проявление программной ошибки, то есть будет выполнено неравенство $\tilde{t}_{\eta i} < \tilde{t}_{\beta i}$. Вероятность перехода ПМП в состояние e_{i+1} после пребывания в состоянии e_i время t определяется по формуле

$$P_{i,i+1}(t) = \int_0^t [1 - G_{\beta i}(u)]dG_{\eta i}(u).$$

Если программная ошибка проявится раньше, чем закончится процесс вычисления i программного модуля, то есть $\tilde{t}_{\beta i} < \tilde{t}_{\eta i}$, то ПМП перейдёт из состояния e_i в состояние e_{n+i}. Это означает, что вычислительный процесс в i программном модуле попадает в состояние восстановления. Вероятность перехода ПМП в состояние e_{n+i} после пребывания в состоянии e_i время t определяется выражением

$$P_{i,n+i}(t) = \int_0^t [1 - G_{\eta i}(u)]dG_{\beta i}(u),$$

В состоянии e_{n+i} ПМП будет находиться случайное время $\tilde{t}_{n+i} = min(\tilde{t}_{\mu i}, \tilde{t}_{g})$, где $\tilde{t}_{\mu i}$ — случайная величина времени восстановления вычислительного процесса,

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

распределённая по закону $G_{\mu i}(t)$; $\tilde{t}_{\text{g}}$ — в общем случае случайная величина допустимого времени задержки вычислительного процесса, распределённая по закону $G_{\text{g}}(t)$.

Функция распределения времени пребывания ПМП в состоянии e_{n+i}, очевидно, определяется выражением

$$F_{n+i}(t) = 1 - [1 - G_{\mu i}(t)] \cdot [1 - G_{\text{g}}(t)].$$

Из состояния e_{n+i} ПМП может перейти в состояние e_i, если восстановление вычислительного процесса в i программном модуле закончится раньше допустимого времени восстановления, то есть будет выполнено неравенство $\tilde{t}_{\mu i} < \tilde{t}_{\text{g}}$. Вероятность перехода ПМП в состояние e_i после пребывания в состоянии e_{n+i} время t определяется по формуле

$$P_{n+i,i}(t) = \int_0^t [1 - G_{\text{g}}(u)] dG_{\mu i}(u).$$

Если случайное время восстановления вычислительного процесса i программного модуля превзойдёт допустимое время, то есть $\tilde{t}_{\text{g}} < \tilde{t}_{\mu i}$, то произойдет отказ вычислительного процесса и ПМП перейдёт из состояния e_{n+i} в состояние e_{2n+1}. Вероятность перехода ПМП в состояние e_{2n+1} после пребывания в состоянии e_{n+i} время t определяется выражением

$$P_{n+i,2n+1}(t) = \int_0^t [1 - G_{\mu i}(u)] dG_{\text{g}}(u).$$

По графосхеме динамики состояний ПМП пишется система интегральных и алгебраических уравнений. Правила построение систем уравнений хорошо изложены в [10].

Система интегральных уравнений ПМП, описывающая функционирование линейного вычислительного процесса до отказа, имеет следующий вид:

$$\begin{aligned}
&\psi_i(t) = \int_0^t \psi_{i+1}(t-u) dP_{i,i+1}(u) + \int_0^t \psi_{n+i}(t-u) dP_{i,n+i}(u), \quad i = 1 \ldots n-1; \\
&\psi_n(t) = \int_0^t \psi_1(t-u) dP_{n,1}(u) + \int_0^t \psi_{n+i}(t-u) dP_{n,n+i}(u); \\
&\psi_{n+i}(t) = \int_0^t \psi_i(t-u) dP_{n+i,i}(u) + P_{n+i,2n+1}(t), \quad i = 1 \ldots n,
\end{aligned} \tag{13}$$

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

где $\psi_i(t)$ — функция распределения времени пребывания ПМП в подмножестве состояний $E_1\{e_i,\ i=1...2n\}$ до первого перехода в состояние e_{2n+1} из другого подмножества $E_2\{e_{2n+1}\}$, если процесс начался в состоянии e_i подмножества состояний E_1.

Решение системы уравнений (13) относительно $\psi_1(t)$ позволяет получить функцию распределения времени до появления отказа вычислительного процесса.

Система интегральных уравнений (13) решается численно [2] или аналитически с помощью преобразований Лапласа–Стильтьеса [10].

Вероятность безотказного исполнения вычислительного процесса в течение времени t рассчитывается по формуле (14)

$$P_{\text{би}}(t) = 1 - \psi_1(t). \tag{14}$$

Математическое ожидание времени до отказа в вычислительном процессе определяется из решения алгебраической системы уравнений ПМП (15) относительно τ_1

$$\begin{aligned} &\tau_i = \varsigma_i + p_{i,i+1}\tau_{i+1} + p_{i,n+i}\tau_{n+i}, \quad p_{n,n+1}\tau_{n+1} \equiv p_{n,1}\tau_1 \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ &\tau_{n+i} = \varsigma_{n+i} + p_{n+i,i}\tau_i, \quad i = 1...n. \end{aligned} \tag{15}$$

где τ_i — математическое ожидание времени пребывания ПМП в подмножестве состояний ПМП $E_1\{e_i,\ i=1...2n\}$ до первого перехода в состояние e_{2n+1} из другого подмножества $E_2\{e_{2n+1}\}$, если процесс начался в состоянии e_i подмножества состояний E_1. Исходными данными для решения системы алгебраических уравнений (15) являются средние времена пребывания ПМП в каждом e_i состоянии ς_i и вероятности переходов из состояния e_i в состояние e_j — p_{ij}, которые рассчитываются по формулам:

$$\begin{aligned} &\varsigma_i = \int_0^\infty [1 - F_i(t)]dt, \\ &F_i(t) = 1 - [1 - G_{\eta i}(t)][1 - G_{\beta i}(t)], \\ &F_{n+i}(t) = 1 - [1 - G_{\mu i}(t)][1 - G_{\text{g}}(t)], \\ &p_{i,j} = P_{i,j}(t = \infty). \end{aligned}$$

При $n = 1$ модель упрощается и системы уравнений ПМП будут иметь вид:

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

$$\psi_1(t) = \int_0^t \psi_2(t-u)dP_{12}(u) + \int_0^t \psi_1(t-u)dP_{11}(u),$$
$$\psi_2(t) = \int_0^t \psi_1(t-u)dP_{11}(u) + P_{23}(t); \qquad (16)$$

$$\tau_1 = \varsigma_1 + p_{12}\tau_2 + p_{11}\tau_1,$$
$$\tau_2 = \varsigma_2 + p_{21}\tau_1. \qquad (17)$$

Из (17) можно получить

$$\tau_1 = \frac{\varsigma_1 + p_{12}\varsigma_2}{1 - p_{12}p_{21} - p_{11}}. \qquad (18)$$

Положим:

$$G_\eta(t) = \begin{cases} 0, & t < T_\eta \\ 1, & t \ge T_\eta \end{cases}, \qquad G_\beta(t) = 1 - e^{-\beta t};$$
$$G_g(t) = \begin{cases} 0, & t < T_g \\ 1, & t \ge T_g \end{cases}, \qquad G_\mu(t) = 1 - e^{-\mu t}, \qquad (19)$$

где T_η – среднее время выполнения программы;

β – параметр функции распределения времени до появления программной ошибки, $T_\beta = \frac{1}{\beta}$ – среднее время до появления программной ошибки;

μ – параметр функции распределения времени восстановления вычислительного процесса, $T_\mu = \frac{1}{\mu}$ – среднее время восстановления вычислительного процесса;

T_g — допустимое время восстановления вычислительного процесса. Тогда

$$p_{11} = e^{-\beta T_\eta}, \quad p_{12} = 1 - e^{-\beta T_\eta}, \quad p_{21} = 1 - e^{-\mu T_g},$$
$$\varsigma_1 = T_\beta(1 - e^{-\beta T_\eta}), \quad \varsigma_2 = T_\mu(1 - e^{-\mu T_g}). \qquad (20)$$

Подставляя выражения (20) в уравнение (18), можно получить расчётную оценку математического ожидания времени (в дальнейшем изложении – среднее время) до отказа вычислительного процесса, с учётом возможности исправления

программной ошибки и восстановления вычислительного процесса при условии, что время восстановления не превосходит заданного значения T_g:

$$T_п = \tau_1 = \frac{T_\beta + T_\mu(1 - e^{-\mu T_g})}{e^{-\mu T_g}}. \qquad (21)$$

На рис. 3 для конкретного примера показаны типовые зависимости среднего времени до отказа вычислительного процесса T_o от среднего времени восстановления вычислительного процесса T_μ при различных значениях допустимого времени восстановления T_g, рассчитанные по формуле (21).

Рис. 3. Зависимости среднего времени до отказа вычислительного процесса $T_п$ от среднего времени восстановления вычислительного процесса T_μ при различных значениях допустимого времени восстановления T_g

Вероятность безотказного исполнения вычислительного процесса в течение времени t в обоснованных случаях можно рассчитывать по приближенной формуле (22)

$$P_{б\,и} \approx e^{-\frac{t}{\tau_1}}, \qquad (22)$$

где τ_1 определяется из (21).

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

3.2. Расчёт КПН сложных ветвящихся вычислительных процессов с циклами

Расчёт количественных показателей надёжности программного обеспечения, если известен обобщённый алгоритм функционирования комплекса программ, достаточно полно отражающий вычислительный процесс, производится путём моделирования функционирования процесса вычислений. В модели учитываются возможные проявления программных ошибок в модулях программы, их устранение и восстановление процесса. Учитывается появление события, приводящее к отказу вычислительного процесса, если случайное время восстановления вычислительного процесса $\tilde{t}_{\mu}$ превысит допустимую величину $\tilde{t}_{g}$. При построении математической модели функционирования вычислительного процесса используется математический аппарат полумарковских процессов.

3.2.1. Исходные данные для моделирования

Исходными данными для моделирования являются:

- число режимов функционирования,
- укрупнённая блок-схема обобщённого алгоритма вычислительного процесса, обеспечивающая режимы функционирования изделия,
- укрупнённая блок-схема алгоритма вычислительного процесса каждого режима, связывающая программные модули между собой,
- для каждого i программного модуля (ПМ), входящего в алгоритм режима: объём программного модуля в условных операторах и в килобайтах N_i, средняя длительность выполнения программного модуля (среднее число выполняемых операторов) n_i, необходимая для завершения вычисления модуля.

Предлагаемый метод расчёта КПН вычислительного процесса целесообразно рассматривать на примере.

Для примера блок-схема алгоритма режима функционирования (или обобщённый алгоритм) имеет вид, представленный на рис. 4. На блок-схеме алгоритма представлены номера программных модулей (ПМ), например для i модуля i ПМ: их объём в килобайтах N_i, и средняя длительность их выполнения (в операторах) n_i. Стрелками показаны последовательность выполнения модулей. Причём, если управление от i модуля в зависимости от некоторых условий может передаваться другому j модулю из некоторого множества модулей $b_j \in B(b_l, l = 1...m)$, то должны быть заданы доли числа обращений к каждому j ПМ — r_{ij} (вероятность обращения к соседнему модулю). Причём должно выполняться условие $\sum_{j=1}^{m} r_{ij} = 1$.

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

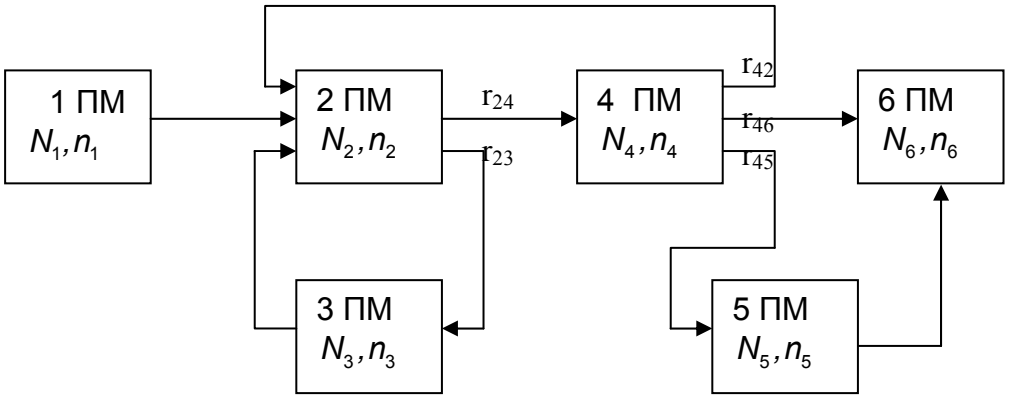

Рис. 4. Блок-схема алгоритма функционирования вычислительного процесса с циклами и разветвлениями.

Исходные данные для расчёта количественных показателей надёжности программного обеспечения (ПО) целесообразно представить в виде следующей табл. 2.

Таблица 2

Исходные данные для расчета КПН ПО

№ п/п	Наименование программного модуля	Объём модуля N_i Кбайт	Средняя длительность выполнения модуля n_i операторов	Условные вероятности переходов r_{ij} из i модуля в j
1	1ПМ (название)	2500	1300	$r_{12} = 1$
2	2ПМ (название)	120	100	$r_{24} = 0.7$, $r_{23} = 0.3$
3	3ПМ (название)	250	220	$r_{32} = 1$
4	4ПМ (название)	150	100	$r_{42} = 0.2, r_{12} = 0.1,$ $r_{12} = 0.7$
5	5ПМ (название)	110	100	$r_{56} = 1$
6	6ПМ (название)	1100	1000	

Примечание. Цифровые значения в таблице приведены для примера.

3.2.2. Построение полумарковской модели для определения функции распределения и среднего времени до отказа вычислительного процесса

По блок-схеме алгоритма функционирования программного обеспечения составляется графосхема полумарковского процесса, описывающая процесс возможного возникновения программной ошибки при выполнении программ, её обна-

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

ружение и возможное восстановление вычислительного процесса или появление отказа в вычислении. Граф-схема полумарковского процесса (ПМП), описывающего функционирование вычислительного процесса до его отказа (блок-схема которого показана на рис. 4) приведена на рис. 5.

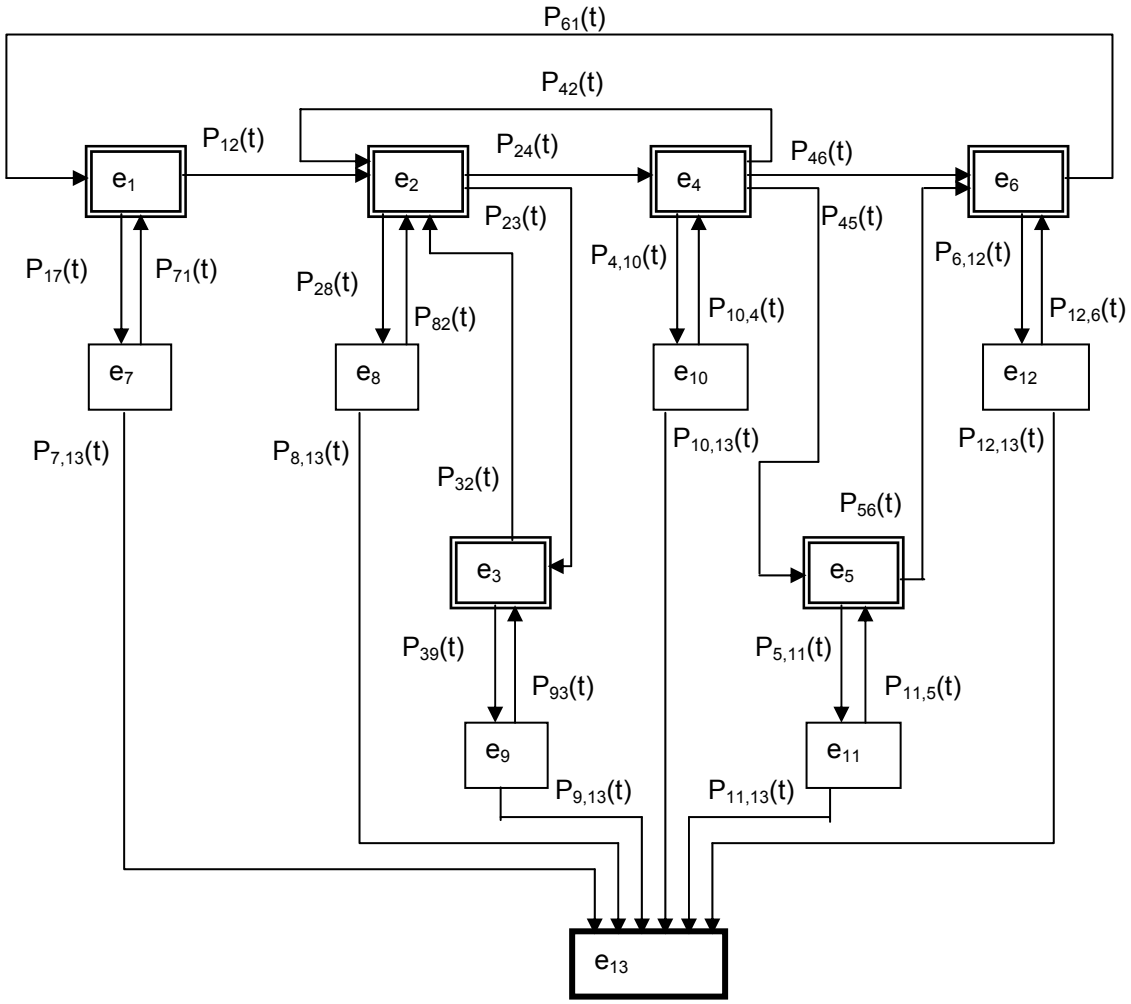

Рисунок 5. Граф-схема полумарковского процесса, описывающего вычислительный процесс с циклами и разветвлением.

Состояния ПМП e_1, e_2, e_3, e_4, e_5, e_6 определяют выполнение программных модулей 1ПМ, 2ПМ, 3ПМ, 4ПМ, 5ПМ, 6ПМ. Состояния ПМП e_7, e_8, e_9, e_{10}, e_{11}, e_{12} определяют появление ошибки вычислений и возможное восстановление вычислительного процесса программных модулей 1ПМ, 2ПМ, 3ПМ, 4ПМ, 5ПМ, 6ПМ программными средствами. В состояние e_{13} ПМП переходит в том случае, если длительность процесса восстановления вычислительного процесса больше допустимого времени.

Состояние e_{13} — состояние отказа вычислительного процесса.

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Система интегральных уравнений ПМП, описывающая процесс функционирования алгоритма вычислений до появления его отказа, представлена ниже (23):

$$
\begin{aligned}
&\psi_1(t) = \int_0^t \psi_2(t-u)dP_{12}(u) + \int_0^t \psi_7(t-u)dP_{17}(u),\\
&\psi_2(t) = \int_0^t \psi_3(t-u)dP_{23}(u) + \int_0^t \psi_4(t-u)dP_{24}(u) + \int_0^t \psi_8(t-u)dP_{28}(u),\\
&\psi_3(t) = \int_0^t \psi_2(t-u)dP_{32}(u) + \int_0^t \psi_9(t-u)dP_{39}(u),\\
&\psi_4(t) = \int_0^t \psi_2(t-u)dP_{42}(u) + \int_0^t \psi_5(t-u)dP_{45}(u) +\\
&+\int_0^t \psi_6(t-u)dP_{46}(u) + \int_0^t \psi_{10}(t-u)dP_{4,10}(u),\\
&\psi_5(t) = \int_0^t \psi_6(t-u)dP_{56}(u) + \int_0^t \psi_{11}(t-u)dP_{5,11}(u),\\
&\psi_6(t) = \int_0^t \psi_1(t-u)dP_{61}(u) + \int_0^t \psi_{12}(t-u)dP_{6,12}(u),\\
&\psi_i(t) = \int_0^t \psi_{i-6}(t-u)dP_{i,i-6}(u) + P_{i,13}(t), \quad i = 7...12.
\end{aligned}
\tag{23}
$$

В системе интегральных уравнений $\psi_i(t)$ — функция распределения времени пребывания ПМП в подмножестве состояний $E_1\{e_1 \ldots e_{12}\}$ до его перехода в состояние e_{13} подмножества $E_2\{e_{13}\}$, если процесс начался в состоянии $e_i \in E_1$. Вероятности переходов ПМП из состояния e_i в любое другое состояние e_j определяются по формулам:

- для состояний ПМП $e_1 \ldots e_6$

$$P_{ij}(t) = r_{ij}\int_0^t [1 - G_{\beta i}(u)]dG_{\eta i}(u),$$

если программная ошибка не проявилась в i модуле;

$$P_{i,6+i}(t) = \int_0^t [1 - G_{\eta i}(u)]dG_{\beta i}(u),$$

если программная ошибка проявилась в i модуле;

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

- для состояний ПМП $e_7 \ldots e_{12}$

$$P_{i,i-6}(t) = \int_0^t [1 - G_g(u)] dG_{\mu i}(u),$$

если программная ошибка будет исправлена и вычислительный процесс в i модуле будет восстановлен за допустимое время;

$$P_{i,13}(t) = \int_0^t [1 - G_{\mu i}(u)] dG_g(u),$$

если программная ошибка в i модуле не будет исправлена за допустимое время и произойдёт отказ вычислительного процесса.

Время выполнения i модуля имеет в общем случае случайную величину, распределённую по закону $G_{\eta i}(t)$, время проявления программной ошибки в i программном модуле имеет также случайный характер и распределено по закону $G_{\beta i}(t)$; r_{ij} — вероятность после безошибочного исполнения i модуля передачи вычислений j модулю. Если из i модуля вычисления передаются только в следующий за ним модуль, то $r_{ij} = 1$, $G_{\mu i}(t)$ — функция распределения времени восстановления вычислительного процесса, $G_g(t)$ — функция распределения допустимого времени для восстановления вычислительного процесса.

Решение системы интегральных уравнений относительно $\psi_1(t)$ позволяет определить функцию распределения времени до отказа вычислительного процесса. Способы решения системы интегральных уравнений ПМП изложены в [2,10]. Вероятность безотказного исполнения вычислительного процесса в течение времени t определяется по формуле $P_{би}(t) = 1 - \psi_1(t)$.

Расчёт среднего времени до появления до отказа вычислительного процесса производится путём решения системы алгебраических уравнений ПМП. Для рассматриваемого примера эта система уравнений имеет следующий вид

$$\begin{aligned}
\tau_1 &= \varsigma_1 + p_{12}\tau_2 + p_{17}\tau_7, \\
\tau_2 &= \varsigma_2 + p_{23}\tau_3 + p_{24}\tau_4 + p_{28}\tau_8, \\
\tau_3 &= \varsigma_3 + p_{32}\tau_2 + p_{39}\tau_9, \\
\tau_4 &= \varsigma_4 + p_{42}\tau_2 + p_{45}\tau_5 + p_{46}\tau_6 + p_{4,10}\tau_{10}, \\
\tau_5 &= \varsigma_5 + p_{56}\tau_6 + p_{5,11}\tau_{11}, \\
\tau_6 &= \varsigma_6 + p_{61}\tau_1 + p_{6,12}\tau_{12}, \\
\tau_i &= \varsigma_i + p_{i,i-6}\tau_{i-6}, \quad i = 7 \ldots 12,
\end{aligned} \tag{24}$$

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

где τ_i — среднее время пребывания ПМП в подмножестве состояний $E_1\{e_1 \ldots e_{12}\}$ до его перехода в состояние e_{13} из подмножества $E_2\{e_{13}\}$, если процесс начался в состоянии $e_i \in E_1$;

$$p_{ij} = P_{ij}(t = \infty) \cdot \tag{25}$$

- вероятности переходов ПМП из состояния e_i в состояние e_j;

$$\varsigma_i = \int_0^\infty [1 - F_i(t)]dt,$$
$$\text{где} \quad F_i(t) = 1 - [1 - G_{\eta i}(t)] \cdot [1 - G_{\beta i}(t)], \quad F_i(t) = 1 - [1 - G_{\mu i}] \cdot [1 - G_g(t)] \tag{26}$$

– средние времена и функции распределения времени пребывания ПМП в состояниях e_i ($i = 1 \div 6$ в первом случае и $i = 7...12$ во втором случае).

Решение системы уравнений (24) относительно τ_1 позволяет получить среднее время до отказа вычислительного процесса.

В обоснованных случаях допускается приближённый расчёт вероятности безотказного выполнения вычислительного процесса по формуле экспоненциальной аппроксимации

$$P_{\text{б и}}(t) \approx e^{-\frac{t}{\tau_1}}$$

где τ_1 рассчитывается при решении системы уравнений (24).

3.2.3. Определение функций распределения времени совершения событий, описывающих функционирование вычислительного процесса до отказа, вероятностей переходов из состояния в состояние и средних времён пребывания ПМП в каждом состоянии

В предположении гипергеометрической функции распределения ожидания появления первой программной ошибки математическое ожидание нормированного времени до её появления определяется выражением (3)

$$M = \frac{N + 1}{q\,N + 1}$$

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

где N — объем программы в операторах, содержащей qN ошибочных операторов,

Выше показана возможность аппроксимации функции распределения времени проявления программной ошибки экспоненциальным законом. Тогда функция распределения нормированного времени до проявления программной ошибки в i программном модуле будет иметь вид

$$G_{\beta i}(t) = 1 - e^{-\beta_i t}, \quad \beta_i = \frac{k_i}{M[T]_i} = \frac{k_i (N_i q_i + 1)}{N_i + 1} \tag{27}$$

где k_i — коэффициент, учитывающий вероятность сочетания исходных данных, при которых усредненный оператор – потенциальный носитель ошибок вызовет ошибку результата.

Среднее нормированное время выполнения i программы определяется выражением $T_{\eta i} = 1/\eta_i = n_i$ (среднее число усредненных операторов, необходимых для выполнения i программы).

В примере приняты функции распределения допустимого времени восстановления вычислительного процесса $G_{gi}(t)$ в каждом модуле как вырожденные, а функции распределения времени выполнения программного модуля $G_{\eta i}(t)$, проявления программной ошибки $G_{\beta i}(t)$ и восстановления вычислительного процесса $G_{\mu i}(t)$ как экспоненциальные

$$G_{\eta i}(t) = 1 - e^{-\eta_i t}, \qquad G_{\beta i}(t) = 1 - e^{-\beta_i t};$$
$$G_{gi}(t) = \begin{cases} 0, & t < T_g \\ 1, & t \geq T_g \end{cases}, \qquad G_{\mu i}(t) = 1 - e^{-\mu_i t}. \tag{28}$$

Тогда вероятности переходов из состояния в состояние после пребывания ПМП в предыдущем состоянии за время t определяются выражениями

$$P_{i,i+1}(t) = \frac{\eta_i}{\eta_i + \beta_i}[1 - e^{-(\eta_i + \beta_i)\cdot t}], \quad P_{i,n+i}(t) = \frac{\beta_i}{\eta_i + \beta_i}[1 - e^{-(\eta_i + \beta_i)\cdot t}];$$
$$P_{n+i,i}(t) = \begin{cases} 1 - e^{-\mu t}, & t < T_g \\ 1 - e^{-\mu T_g}, & t \geq T_g \end{cases}, \quad P_{n+i,2n+1}(t) = \begin{cases} 0, & t < T_g \\ e^{-\mu T_g}, & t = T_g \\ 0, & t > T_g \end{cases}; \tag{29}$$

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Подстановка полученных выражений вероятностей переходов (29) в системы интегральных уравнений (23) позволяет решать эти уравнения относительно $\psi_1(t)$ – функции распределения времени до отказа вычислительного процесса.

Подстановка выражений (28) в формулы (26) позволяет получить средние времена пребывания ПМП в каждом i состоянии ς_i; учёт $t=\infty$ в выражениях вероятностей переходов $P_{ij}(t)$ (29) позволяет получить вероятности p_{ij} для подстановки в систему уравнений (24). Тогда коэффициенты и свободные члены системы алгебраических уравнений (24) будут иметь вид:

$$p_{i,i+1}=\frac{\eta_i}{\eta_i+\beta_i}\quad,\quad p_{i,n+i}=\frac{\beta_i}{\eta_i+\beta_i},\quad p_{n+i,i}=1-e^{-\mu T_g}$$
$$\varsigma_i=\frac{1}{\eta_i+\beta_i},\quad i=1\ldots n,\quad \varsigma_i=T_\mu(1-e^{-\mu T_g}),\quad i=n+1\ldots 2n. \qquad (30)$$

Учитывая выражения для β_i из (27) и $T_{\eta i}$ и условную вероятность r_{ij} из формул (30), можно окончательно получить:

$$p_{ij}=r_{ij}\left(\frac{1}{1+k_i n_i/M_i}\right)=r_{ij}\left[\frac{1}{1+\dfrac{k_i n_i(N_i q_i+1)}{N_i+1}}\right],$$
$$p_{i,n+i}=1-\left[\frac{1}{1+\dfrac{k_i n_i(N_i q_i+1)}{N_i+1}}\right],\quad i=1\ldots m, \qquad (31)$$
$$\varsigma_i=\frac{1}{\dfrac{1}{n_i}+\dfrac{k_i(N_i q+1)}{N_i+1}}$$

где k_i — коэффициент, учитывающий вероятность сочетания исходных данных, при которых усредненная команда — потенциальный носитель ошибок, вызовет ошибку результата. Для хорошо отлаженных программ среднего размера (1000–3000 килобайт) $k=5.5\cdot10^{-10}$; m – число уравнений ПМП. Для данного примера $m=12$.

Для учёта масштабирования, необходимо T_μ и T_g умножить на коэффициент перевода s времени (заданного в минутах) в нормированное время (заданное в операторах), а затем полученное решение в операторах перевести снова в единицы времени (часы).

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

3.2.4. Расчёт количественных показателей надёжности вычислительного процесса

Решение системы интегральных уравнений ПМП (23) относительно $\psi_1(t)$ позволяет определить функцию распределения времени до отказа вычислительного процесса. Способы решения интегральных уравнений ПМП приведены в [2]. Вероятность безотказного исполнения вычислительного процесса в течение времени t определяется по формуле $P_{\text{би}}(t) = 1 - \psi_1(t)$.

Среднее время до отказа в вычислительном процессе определяется из решения алгебраической системы уравнений ПМП (24) относительно τ_1.

Решение целесообразно выполнять на математической системе MathCAD матричным способом. Переписав систему уравнений (24) в виде (32):

$$\begin{aligned}
&-\tau_1 + p_{12}\tau_2 + p_{17}\tau_7 = -\varsigma_1,\\
&-\tau_2 + p_{23}\tau_3 + p_{24}\tau_4 + p_{28}\tau_8 = -\varsigma_2,\\
&-\tau_3 + p_{32}\tau_2 + p_{39}\tau_9 = -\varsigma_3,\\
&-\tau_4 + p_{42}\tau_2 + p_{45}\tau_5 + p_{46}\tau_6 + p_{4,10}\tau_{10} = -\varsigma_4,\\
&-\tau_5 + p_{56}\tau_6 + p_{5,11}\tau_{11} = -\varsigma_5,\\
&-\tau_6 + p_{61}\tau_1 + p_{6,12}\tau_{12} = -\varsigma_6,\\
&-\tau_i + p_{i,i-6}\tau_{i-6} = -\varsigma_i, \quad i = 7...12.
\end{aligned} \tag{32}$$

можно получить $\tau \cdot D = \varsigma$, откуда матричное решение будет иметь вид

$$\tau = D^{-1} \cdot \varsigma \tag{33}$$

где τ – вектор-столбец решений системы уравнений (32),

D^{-1} – обратная матрица определителя D системы уравнений (32),

ς – вектор-столбец свободных членов системы уравнений.

Численное значение первого элемента вектора решений τ определяет среднее время до отказа вычислительного процесса.

Допускается приближённый расчёт вероятности безотказной работы программных средств изделия по формуле экспоненциальной аппроксимации

$$P_{\text{би}}(t) \approx e^{-\frac{t}{\tau_1}}$$

где τ_1 рассчитывается при решении системы уравнений (32).

Ниже на рабочих страницах компьютерной математической системы MathCAD приведён пример уточнённого расчёта количественных показателей надёжности вычислительного процесса, алгоритм которого представлен на рисунке 2. Рассчитывается среднее время до отказа вычислительного процесса с последующей аппроксимацией функции распределения времени до отказа экспоненциальным распределением.

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Расчёт количественных показателей надёжности программного обеспечения изделия

Обозначения:

Ξ - число программных модулей в комплексе ПС

ID - массив исходных данных (N -объём программных модулей, n -средняя длительность вычисления модуля - число исполняемых операторов в модуле)

q - доля ошибочных операторов в программе

$T\mu$ - среднее время восстановления вычислительного процесса

Tg - допустимое время восстановления вычислительного процесса

ζ - массив средних времён пребывания ПМП в каждом e_i состоянии (вектор свободных членов системы уравнений)

p - массив вероятностей переходов ПМП из e_i состояния в e_j при безошибочном выполнении программного модуля

r - массив вероятностей переходов ПМП из e_i состояния в e_{10+i} при проявлении программной ошибки

$pn(T\mu, Tg)$ - вероятность перехода ПМП из e_{m+i} состояния в e_i

D - матрица вероятностей переходов ПМП из состояния e_i в состоянии e_j (определитель системы уравнений)

v оп/сек - производительность компьютра

Исходные данные для расчёта

ORIGIN := 1

$\Xi := 6$

$q := 0.001$

$i := 1 .. 6$

$c := \frac{1024}{32}$ Коэффициент перевода килобайты в 32 разрядные операторы

$c := 32$

$$ID := \begin{pmatrix} 2500 \cdot c & 1300 \\ 120 \cdot c & 100 \\ 250 \cdot c & 220 \\ 150 \cdot c & 100 \\ 110 \cdot c & 100 \\ 1100 \cdot c & 1000 \end{pmatrix}$$

$N_i := ID_{i,1}$ $n_i := ID_{i,2}$

$r_{24} := 0.7$ $r_{23} := 0.3$

$r_{42} := 0.2$ $r_{45} := 0.1$ $r_{46} := 0.7$

$v := 1000000 \ \frac{опер}{сек}$

Согласование единиц исходных данных

$T\mu=6$ м - среднее время восстановления вычислительног процесса в минутах

$Tg=12$ м - допустимое время задержки в востановлении вычислительного процесса

Перевод минут в число операторов (масштабирование)

$s := 60 \cdot v$

$Tg := 12 \cdot s$ $T\mu := 6 \cdot s$

$T\mu = 3.6 \times 10^8$ $Tg = 7.2 \times 10^8$ операций

Листинг 1

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Объём прогаммных модулей в операторах

$$N = \begin{pmatrix} 8 \times 10^4 \\ 3.84 \times 10^3 \\ 8 \times 10^3 \\ 4.8 \times 10^3 \\ 3.52 \times 10^3 \\ 3.52 \times 10^4 \end{pmatrix}$$

Средняя длительность вычислений каждого модуля в операторах

$$n = \begin{pmatrix} 1.3 \times 10^3 \\ 100 \\ 220 \\ 100 \\ 100 \\ 1 \times 10^3 \end{pmatrix}$$

Формирование численных значений вероятностей переходов ПМП

$$p_i := \frac{1}{1 + \frac{k \cdot (N_i \cdot q + 1) \cdot n_i}{N_i + 1}} \qquad k := 5.5 \cdot 10^{-10} \qquad w_i := 1 - p_i$$

$$p = \begin{pmatrix} 0.99999999927607 \\ 0.9999999999307 \\ 0.99999999986389 \\ 0.99999999993356 \\ 0.9999999999294 \\ 0.99999999943439 \end{pmatrix} \qquad w = \begin{pmatrix} 7.2392847 \times 10^{-10} \\ 6.9304784 \times 10^{-11} \\ 1.3610801 \times 10^{-10} \\ 6.6444406 \times 10^{-11} \\ 7.0604855 \times 10^{-11} \\ 5.6560889 \times 10^{-10} \end{pmatrix}$$

$pn(T\mu, Tg) := 1 - e^{\frac{-Tg}{T\mu}}$ Вероятности переходов из состояния e_{6+i} в состояние e_i при восстановления процесса за время не более Tg

$pn(T\mu, Tg) = 0.865$

Листинг 2

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Формирование определителя системы уравнений - матрицы D

$i := 1 .. 12 \qquad j := 1 .. 12$

$$D_{i,j} := \begin{cases} (-1) & \text{if } i = j \\ (w_i) & \text{if } j = i + 6 \\ 1 - e^{\frac{-Tg}{T\mu}} & \text{if } i > 6 \wedge j = i - 6 \\ 0 & \text{otherwise} \end{cases}$$

$D_{1,2} := p_1 \qquad D_{2,3} := r_{23} \cdot p_2 \qquad D_{2,4} := r_{24} \cdot p_2$

$D_{3,2} := p_3 \qquad D_{4,2} := r_{42} \cdot p_4 \qquad D_{4,5} := r_{45} \cdot p_4$

$D_{4,6} := r_{46} \cdot p_4 \qquad D_{5,6} := p_5 \qquad D_{6,1} := p_6$

D =

	1	2	3
1	-1	0.999999999	0
2	0	-1	0.3
3	0	1	-1
4	0	0.2	0
5	0	0	0
6	0.999999999	0	0
7	0.864664717	0	0
8	0	0.864664717	0
9	0	0	0.864664717
10	0	0	0
11	0	0	0
12	0	0	0

Формирование вектор столбца ζ свободных членов системы уравнений (средние времена пребывания ПМП в каждом состоянии)

$$\zeta n(T\mu, Tg) := T\mu \cdot \left(1 - e^{\frac{-Tg}{T\mu}}\right) \qquad \zeta n(T\mu, Tg) = 3.113 \times 10^8$$

$$\zeta_i := \begin{cases} \left[T\mu \cdot \left(1 - e^{\frac{-Tg}{T\mu}}\right)\right] & \text{if } i > 6 \\ \dfrac{1}{\dfrac{i}{n_i} + \dfrac{(q \cdot N_i + 1) \cdot k}{N_i + 1}} & \text{otherwise} \end{cases}$$

$\zeta_i := -1 \cdot \zeta_i \qquad \zeta =$

	1
1	$-1.3 \cdot 10^3$
2	-50
3	-73.333
4	-25
5	-20
6	-166.667
7	$-3.113 \cdot 10^8$
8	$-3.113 \cdot 10^8$
9	$-3.113 \cdot 10^8$
10	$-3.113 \cdot 10^8$
11	$-3.113 \cdot 10^8$
12	$-3.113 \cdot 10^8$

Листинг 3

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Решение системы уравнений ПМП матричным методом

$$\tau := D^{-1} \cdot \zeta$$

Результат решения системы уравнений ПМП

$\tau =$

	1
1	$7.62967 \cdot 10^{12}$
2	$7.62967 \cdot 10^{12}$
3	$7.62967 \cdot 10^{12}$
4	$7.62967 \cdot 10^{12}$
5	$7.62967 \cdot 10^{12}$
6	$7.62967 \cdot 10^{12}$
7	$6.59741 \cdot 10^{12}$
8	$6.59741 \cdot 10^{12}$
9	$6.59741 \cdot 10^{12}$
10	$6.59741 \cdot 10^{12}$
11	$6.59741 \cdot 10^{12}$
12	$6.59741 \cdot 10^{12}$

$\tau_1 = 7.63 \times 10^{12}$ операций

Перевод в часы

$$Tпс := \frac{\tau_1}{3600 \cdot 1000000}$$

Средняя наработка на отказ вычислительного процесса (программных средств) - Тпс

$Tпс = 2.119 \times 10^{3}$ час

Вероятность безотказного функционирования вычислительного процесса

$$P(t, Tпс) := e^{\frac{-t}{Tпс}}$$

$P(16, Tпс) = 0.992478948$

Листинг 4

Труды СПИИРАН. 2008. Вып. 6. ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн)
SPIIRAS Proceedings. 2008. Issue 6. ISSN 2078-9181 (print), ISSN 2078-9599 (online)
www.proceedings.spiiras.nw.ru

Литература

1. *Новиков И. С.* Теоретические основы надежности автоматизированных систем управления. Учебное пособие. Петродворец.: ВВМУРЭ им А. С. Попова, 1989. 258 с.
2. *Новиков И. С.* Математические модели для обоснования количественных показателей надёжности программного обеспечения, методы их оценки на этапах проектирования и испытаний // Научно-технический сборник 8. СПб.: МО, 1997.
3. *Майерс Г.* Надежность программного обеспечения. М.: Мир, 1980. 360 с.
4. *Тайер Т., Липов М., Нельсон Э*. Надежность программного обеспечения. М.: Мир,1981. 325 с.
5. *Липаев В. В.* Надежность программного обеспечения АСУ. М.: Энергия, 1981. 240 с.
6. *Тоценко В. Г., Александров А. В., Парамонов Н. Б.* Корректность, устойчивость, точность программного обеспечения. Киев: Наукова думка, 1990. 196 с.
7. *Штрик А. А., Осовецкий Л. Г., Мессих И. Г.* Структурное проектирование надежных программ встроенных ЭВМ. Л.: Машиностроение, 1989. 296 с.
8. *Уилкс С*. Математическая статистика. М.: Наука, 1967. 632 с.
9. *Холстед М. Х.* Начала науки о программах. М.: Финансы и статистика, 1981. 128 с.
10. *Броди С. М., Власенко О. Н., Марченко Б. Г.* Расчёт и планирование испытаний систем на надёжность. Киев: Наукова думка, 1970. 192 с.