

УДК 004.519.217

Д.А. Маевський, канд. техн. наук, доц., Одес. нац. політехн. ун-т

## ОСНОВЫ ДИСКРЕТНОЙ ДИНАМИКИ ПРОГРАММНЫХ СИСТЕМ

*Д.А. Маевський. Основи дискретної динаміки програмних систем.* Розглянуто побудову дискретних динамічних моделей виявлення дефектів в програмних системах. На основі теорії динаміки програмних систем виконано дискретизацію за часом і кількістю дефектів. При побудові прийнято припущення, за яким при виправленні виявленого дефекту нові дефекти не вносяться. Це припущення притаманне більшості відомих моделей надійності програмного забезпечення. Отримано математичні вирази для розрахунку кількості виявлених програмних дефектів і виконано їх верифікацію. Показано збіг результатів розрахунків по дискретній і безперервній моделях.

*Ключові слова:* інформаційні системи, динаміка програмних систем, динамічні системи дискретного часу, надійність програмного забезпечення.

*Д.А. Маевский. Основы дискретной динамики программных систем.* Рассмотрено построение дискретных динамических моделей выявления дефектов в программных системах. На основе теории динамики программных систем выполнена дискретизация по времени и количеству дефектов. При построении принято допущение о том, что при исправлении выявленного дефекта новые дефекты не вносятся. Это допущение свойственно большинству известных моделей надежности программного обеспечения. Получены математические выражения для расчета количества выявленных программных дефектов и выполнена их верификация. Показано совпадение результатов расчетов по дискретной и непрерывной моделям.

*Ключевые слова:* информационные системы, динамика программных систем, динамические системы дискретного времени, надежность программного обеспечения.

*Д.А. Maevsky. Basics of discrete dynamics of software systems.* The construction of discrete dynamic models for detection of defects in the software systems is considered. On the basis of theory of the dynamics of software systems, discretization is performed on time and on the amount of defects. In the course of the construction an assumption peculiar to the majority of well-known models of software reliability is accepted — in correcting of the revealed defect new defects are not brought in. Mathematical expressions for calculating the amount of the revealed software defects are obtained, and their verification is performed. The coincidence of calculation results on discrete and continuous models is shown.

*Keywords:* information systems, dynamics of software systems, dynamic systems of discrete time, software reliability.

Теория надежности программного обеспечения как самостоятельный раздел общей теории надежности существует уже более тридцати лет. За это время разработано более двадцати моделей надежности программного обеспечения (МНПО), призванных описать процессы выявления программных дефектов во времени. Такое разнообразие подходов к описанию одного и того же процесса вызвано тем, что ни одна из этих моделей не может с одинаковой точностью формализовать закономерности выявления дефектов в программных системах (ПС) разных разработчиков [1]. Зачастую, модель, хорошо зарекомендовавшая себя на продуктах одной компании, является неприменимой для описания процесса выявления дефектов в программных продуктах другой. Это привело к тому, что некоторые крупные фирмы создают и используют свои, узкоспециализированные модели. Примером может служить так называемая “модель фирмы IBM” [3], о назначении которой можно судить по ее названию.

Сложившаяся ситуация свидетельствует о том, что современная теория надежности ПО имеет ряд недостатков. Так, известный специалист, директор института надежности и анализа рисков университета им. Джорджа Вашингтона (США) Нозер Сингпурвалла (Nozer Singpurwalla),

вообще задается вопросом: “Отказы программного обеспечения: существуют ли они?” [3]. Известный специалист в области теории надежности И.А. Ушаков (Сан-Диего, США) основным недостатком видит в том, что теория надежности программных систем полностью унаследовала концептуальный базис и математический аппарат теории надежности технических систем [4]. При этом выявление дефектов рассматривается как случайный процесс. Однако, некоторые специалисты в области надежности обосновывают невозможность такого подхода. В частности, утверждается, что процесс выявления дефектов не удовлетворяет определению случайного процесса [4]. Основным в этом определении является то, что при повторении одних и тех же условий случайное событие может не наступить. Для программного обеспечения это требование не выполняется: если в программе выявлен дефект, то вызываемая им ошибка будет повторяться всякий раз при обращении к программе с тем же набором входных данных. Кроме того, эта ошибка будет выявлена во всех копиях программного продукта у всех пользователей. В теории надежности технических устройств, напротив, считается, что возникновение одинакового отказа в нескольких одинаковых устройствах — явление случайное, причем его вероятность уменьшается с ростом числа устройств. Это свидетельствует о том, что для создания теории надежности программного обеспечения нужно использовать иной, отличный от теории вероятностей, концептуальный базис.

В теории динамики программных систем (ДПС) предложен иной, не вероятностный, а детерминированный подход к проблеме теории надежности ПО [5, 6]. В его основу положено представление о потоках дефектов, основанное на теории неравновесных процессов и теории переноса. Верификация построенной на базе теории ДПС модели надежности показала в среднем в 2,5 раза более точные результаты, чем существующие модели. Точность оценивания при этом практически не зависит от типа программной системы [5], что может свидетельствовать в пользу ее универсальности.

Теория ДПС основана на следующих допущениях:

- ПС является открытой неравновесной системой, взаимодействующей со своей предметной областью по законам неравновесных процессов;
- ПС характеризуется специальной функцией состояния  $f(t)$  — количеством находящихся в ней дефектов;
- исчезновение и появление дефектов в ПС является результатом совместного действия прямого (выходного) и обратного (входного) потоков дефектов;
- интенсивность потока пропорциональна количеству дефектов, образующих данный поток;
- все дефекты равноправны и участвуют в образовании потоков одинаковым образом, независимо от причины возникновения, локализации, типа дефекта и возможных последствий его проявления;
- функция состояния  $f(t)$  является дифференцируемой на всей области ее определения.

Последнее допущение фактически предполагает наличие в ПС не целого, а дробного количества дефектов. Однако физическая природа дефектов такова, что в любой ПС их количество может быть лишь целым числом. Введение принципа непрерывности в теорию ДПС — только компенсация за использование математического аппарата дифференциальных уравнений при оценивании надежности. Здесь следует отметить, что все известные МНПО также используют непрерывную модель процесса выявления дефектов, и вопрос их дискретизации раньше не рассматривался [1]. Поэтому актуальной является задача построения и исследования дискретной модели на базе теории ДПС, в которой снято требование непрерывности функции состояния.

Любая непрерывная динамическая система может быть дискретизована по одному из двух параметров: времени либо значению функции состояния. Поэтому целесообразно рассмотрение двух дискретных моделей процесса выявления дефектов в программном обеспечении — по времени и по количеству дефектов. Здесь рассматривается частный случай такого процесса, когда при исправлении выявленного дефекта вторичные дефекты не вносятся.

### Динамика программных систем дискретного времени

При временной дискретизации процесса выявления дефектов считается, что наблюдение состояния системы производится через одинаковые промежутки времени  $\Delta t = 1$ . Использование дискретного временного шага автоматически подразумевает появление дробного количества дефектов. В силу положенных в основу ДПС допущений интенсивность потока дефектов  $\frac{\Delta f}{\Delta t}$  должна быть пропорциональна количеству дефектов  $f_k$ , содержащихся в системе в момент времени  $t_k$ ,

$$\frac{\Delta f}{\Delta t} = -A f_k. \quad (1)$$

Здесь принято, что изменение количества дефектов  $\Delta f = f_{k+1} - f_k$ , где  $f_{k+1}$  – количество дефектов в момент времени  $t_{k+1}$ . С учетом того, что  $\Delta t = 1$ , выражение (1) можно переписать как  $f_{k+1} - f_k = -A f_k$  или

$$f_{k+1} = f_k(1 - A). \quad (2)$$

Уравнение (2) позволяет рассчитать количество дефектов в конце каждого единичного интервала времени по их известному количеству в начале интервала. Однако наибольший практический интерес представляет другая зависимость, связывающая количество дефектов в конце каждого интервала с начальным значением  $F_0$  при  $t = 0$ . С учетом начального условия  $f_0 = F_0$ :

- в конце первого интервала  $f_1 = F_0(1 - A)$ ;
- в конце второго интервала  $f_2 = f_1(1 - A) = F_0(1 - A)^2$ .

Тогда общую формулу для расчета количества дефектов в конце  $k$ -го интервала времени можно записать как

$$f_k = F_0(1 - A)^k. \quad (3)$$

Для сравнения, в таблице и на рис. 1 приведены результаты моделирования реального процесса выявления дефектов по формуле (3) в сравнении с результатами для непрерывной системы, которые могут быть получены в виде

$$f(t) = F_0 e^{-At}. \quad (4)$$

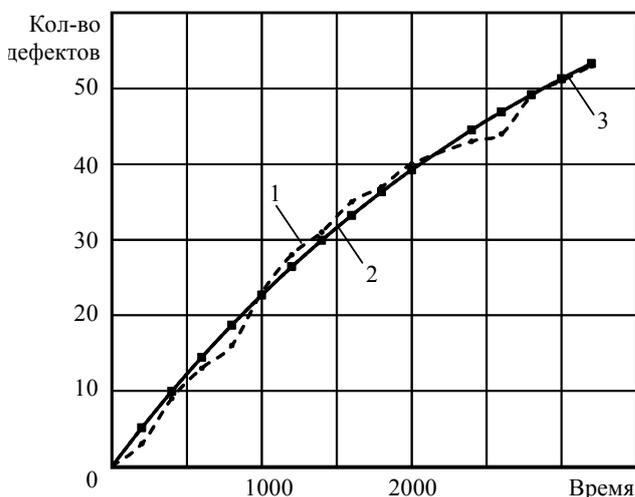


Рис. 1. Результаты моделирования.  
Модель с дискретным временем:  
1 — эксперимент; 2 — расчет непрерывный;  
3 — расчет дискретной модели

Оценка точности модели  
с дискретным временем

$t$	$f(t)$ эксп.	$f(t)$ расч. непр.	$f(t)$ расч. дискр.
0	0	0	0
200	3	5,1	5,1
400	9	9,9	9,9
600	13	14,4	14,5
800	16	18,7	18,6
1000	23	22,6	22,7
1200	28	26,5	26,4
1400	31	29,9	29,9
1600	35	33,2	33,2
1800	37	36,3	36,3
2000	40	39,2	39,2
2400	43	44,5	44,5
2600	44	46,9	46,9
2800	49	49,1	49,2
3000	51	51,2	51,3
3200	53	53,2	53,3

Данные для моделирования взяты из приложения, файл crs1.dat [1]. Как видно, непрерывная модель и модель с дискретным временем показывают практически одинаковые результаты, которые хорошо согласуются с экспериментальными. Максимальное отклонение расчетного и экспериментального результатов составило три дефекта или 5 % от общего количества дефектов.

#### Динамика программных систем с дискретным количеством дефектов

Дискретизация по количеству выявленных дефектов, в силу физической природы исследуемого процесса, является наиболее естественной. Дискретная модель процесса выявления дефектов в этом случае также может быть построена на базе тех же допущений теории ДПС. Приращение времени в этом случае составляет

$$\Delta t = t_k - t_{k-1}, \quad (5)$$

где  $t_k$  — время, при котором в ПС оставалось  $k$  дефектов, а  $t_{k-1}$  — время, при котором их становится на единицу меньше.

Поэтому выражение (1) может быть переписано как  $\frac{1}{\Delta t} = -Ak$ , или, с учетом (5) как

$t_k - t_{k-1} = -\frac{1}{Ak}$ , откуда окончательно

$$t_{k-1} = t_k + \frac{1}{Ak}. \quad (6)$$

Формула (6) позволяет вычислить время, при котором в программной системе остается  $k-1$  дефект, если известно время, когда их было на единицу больше. Но для практических расчетов нужно получить общую зависимость, которая связывала бы время возникновения дефекта с его порядковым номером  $k$ . Ее можно получить, используя начальные условия – в начальный момент времени в системе оставалось  $F_0$  дефектов. С учетом этого по формуле (6) определяются моменты времени, в которые количество дефектов в системе будет последовательно уменьшаться на единицу:

— в системе оставалось  $F_0$  дефектов при  $t = 0$ ;

— в системе оставался  $F_0 - 1$  дефект при  $t = 0 + \frac{1}{AF_0}$ ;

— в системе оставалось  $F_0 - 2$  дефекта при  $t = 0 + \frac{1}{AF_0} + \frac{1}{A(F_0 - 1)}$ .

На основании этого не трудно заметить, что общая формула для момента времени, при котором в системе остается  $F_0 - k$  дефектов, будет

$$t_k = \frac{1}{A} \sum_{i=0}^{k-1} \frac{1}{F_0 - i}. \quad (7)$$

Следует обратить внимание, что если в какой-то момент времени в программной системе оставалось  $F_0 - k$  дефектов, то к этому моменту из системы были вынесены, т. е. выявлены и ликвидированы,  $k$  дефектов. Поэтому формула (7) позволяет рассчитать время, при котором в программной системе будет выявлено  $k$  дефектов.

В случае непрерывной модели программной системы время, при котором в ней остается  $f$  дефектов, может быть вычислено как [6]

$$t = -\frac{1}{A} \ln \left( \frac{f}{F_0} \right). \quad (8)$$

Для сравнения результатов моделирования для дискретной и непрерывной системы использованы те же данные, что и в предыдущем случае. Результаты расчетов по формулам (7), (8) и экспериментальная кривая приведены на рис. 2.

Как видно, расчетные значения непрерывной и дискретной модели практически совпадают. Максимальное отклонение составило 30 единиц времени или не более 10 % общего времени наблюдений. Однако, с ростом количества дефектов эта ошибка увеличивается, причем дискретная модель систематически показывает большие значения времени. Значительные отклонения расчетных кривых от экспериментальных могут быть объяснены использованием индивидуальных наблюдений, в которых неизбежны случайные ошибки. Исследована природа этих ошибок и выяснены пути их уменьшения [6].

#### Выводы

Выполненные в работе исследования позволяют сделать вывод о том, что дискретные и непрерывные модели выявления дефектов в программных системах обеспечивают практически одинаковую точность моделирования в случае моделей дискретного времени. При дискретизации модели по количеству дефектов возникает систематическая ошибка из-за накопления погрешности дискретизации. Поэтому, в случае использования дискретных моделей, актуальной становится дальнейшее развитие методов учета этой погрешности и уменьшения ошибки моделирования.

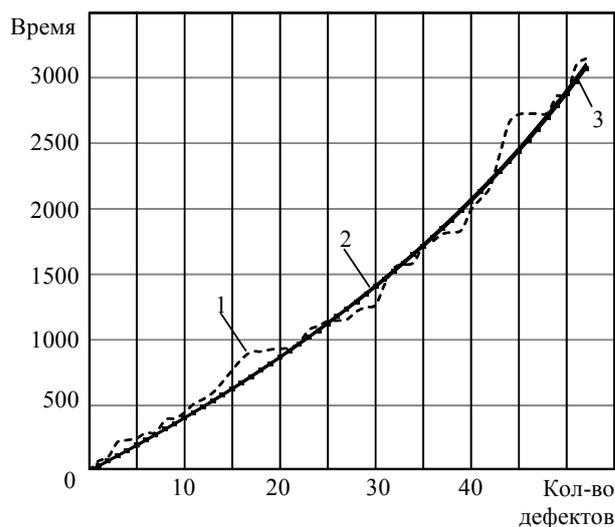


Рис. 2. Результаты моделирования.  
Модель с дискретным количеством дефектов:  
1 — эксперимент; 2 — расчет непрерывный;  
3 — расчет дискретной модели

#### Литература (References)

1. Lyu, M.R. Handbook of Software Reliability Engineering. / M.R. Lyu. — London: McGraw-Hill, 1996. — 805 p.
2. Brooks, W.D. Analysis of discrete reliability models / W.D. Brooks, R.W. Montley // Technical Report RADC-TR-80-84. — New York: Rome Air Development Center, 1980.
3. Singpurwalla, N.D. The failure rate of software: does it exist? / N.D. Singpurwalla // IEEE Transactions on Reliability. — 1995. — Vol. 44, № 3. — P. 463 — 469.
4. Ushakov, I. Reliability theory. History & current state in bibliographies [Electronic resource] / I. Ushakov // Reliability: Theory & Applications. — 2012. — Vol. 7, № 1(24). — [http://gnedenko-forum.org/Journal/2012/012012/RTA\\_1\\_2012-01.pdf](http://gnedenko-forum.org/Journal/2012/012012/RTA_1_2012-01.pdf) — 01.11.2012.
5. Maevsky, D.A. Software reliability. Non probabilistic approach [Electronic resource] / D. A. Maevsky, H. D. Maevskaya, A.A. Leonov // Reliability: Theory & Applications. — 2012. — Vol. 7, № 3(26) — P. 8 — 20. Access mode: [http://www.gnedenko-forum.org/Journal/2012\\_3.html](http://www.gnedenko-forum.org/Journal/2012_3.html) — 01.11.2012.
6. Maevsky, D.A. Fundamentals of software stability theory [Electronic resource] / D. A. Maevsky, // Reliability: Theory & Applications. — 2012. — Vol. 7, № 4(27) — P. 31 — 40. — [http://www.gnedenko-forum.org/Journal/2012/RTA\\_4\\_2012.pdf](http://www.gnedenko-forum.org/Journal/2012/RTA_4_2012.pdf) — 01.11.2012.

Рецензент д-р техн. наук, проф. Одес. нац. политехн. ун-та Антощук С.Г.

Поступила в редакцию 15 сентября 2012 г.