

Д. Федасюк, М. Сенів, П. Сердюк, Н. Мамроха
Національний університет “Львівська політехніка”,
кафедра програмного забезпечення

ПРОГРАМНИЙ МОДУЛЬ ПРОГНОЗУВАННЯ НАДІЙНОСТІ СИСТЕМИ ТЕПЛООВОГО ПРОЕКТУВАННЯ

© Федасюк Д., Сенів М., Сердюк П., Мамроха Н., 2008

Для прогнозування надійності програмного забезпечення (ПЗ) використано модель оцінювання надійності ПЗ, що ґрунтується на випадковому процесі Пуассона, яка визначає та дає змогу прогнозувати імовірність відмови ПЗ та його надійність на заданому етапі (ітерації) тестування. Розроблено програмний засіб автоматизованого тестування для верифікації системи теплового проектування.

For the forecasting of the reliability testing the model of software reliability estimation, which is based on the random Poisson process, has been used, which determines and allows to forecast the software fault probability and its reliability at the defined testing stage (iteration). The software environment of computer-aided testing for the verification of the thermal designing system has been developed.

Вступ

Існує багато конкуруючих поглядів на те, що називається надійністю ПЗ. Один з таких поглядів полягає в тому, що програма або коректна, або некоректна [1], тоді надійність ПЗ – бінарна за своєю природою: ідеальна програма має надійність, яка дорівнює одиниці, а неідеальна – яка дорівнює нулеві. Інший підхід ґрунтується на тестуванні ПЗ, причому процентне відношення вдалих тестів використовується для вимірювання надійності, тобто надійність ПЗ визначена як відносна частота вдалих виконань програми [2]. У [3, 4] надійність визначена як ймовірність безвідмовного функціонування ПЗ у специфікований час чи у специфікованому середовищі.

Модель надійності програмного забезпечення – це математична модель, побудована для оцінки залежності надійності програмного забезпечення від деяких певних параметрів, зокрема параметрами, пов'язаними з деякою гілкою програми на підмножині наборів вхідних даних, за допомогою яких ця гілка контролюється. Іншими такими параметрами є частота помилок, які дають змогу оцінити якість систем реального часу, що функціонують у безперервному режимі і водночас отримувати непряму інформацію про надійність ПЗ.

Розрізняють чотири типи моделей надійності [5]:

Аналітичні моделі [1,6,7] визначаються за такими критеріями, як:

- Визначення взаємозв'язків, зв'язаних з процедурою тестуванням ПЗ;
- Розроблення або вибір аналітичної моделі, яка ґрунтується на припущенні про процедуру тестування;
- Вибір параметрів моделей з використанням отриманих даних;
- Використання моделей – розрахунок кількості показників надійності за моделлю.

Динамічні моделі ґрунтуються на тому, що вихідні дані збираються в процесі тестування ПЗ протягом фіксованих або випадкових часових інтервалів. Кожен інтервал – це стадія, на якій виконується послідовність тестів і фіксується деяка кількість помилок.

До динамічних моделей можна віднести моделі: Шумана, Ла Падула, Джелінського – Моранди, Шика – Волвертона, Мауса, Перехідної ймовірності.

Статичні моделі відрізняються від динамічних тим, що в статичних моделях не враховується час появи помилок в процесі тестування. Ці моделі будуються на чіткому статистичному фундаменті. Сюди можна віднести такі моделі: Міллса, Ліпова, Коркорена, Нельсона.

Емпіричні моделі переважно ґрунтуються на аналізі структурних особливостей програмного засобу (або програми). Як вказувалося раніше, емпіричні моделі часто не дають кінцевих результатів показників надійності, проте їх використання на етапі проектування ПЗ корисно для прогнозування потрібних ресурсів тестування, уточнення планових термінів завершення.

Моделі оцінювання надійності ПЗ на основі розподілу Пуассона

У цій роботі ми використовуємо динамічні моделі оцінювання надійності ПЗ, що ґрунтуються на випадковому процесі Пуассона [8], який можна описати так: нехай $\{M(t), t \geq 0\}$ – неоднорідний пуассонівський процес з функцією інтенсивності $\lambda(t)$, який являє собою загальну кількість відмов ПЗ за час його використання t . Тоді кількість відмов ПЗ в інтервалі $(0, t)$ має розподіл Пуассона, тобто:

$$P\{M(t) = k\} = \frac{(\mu(t))^k}{k!} \exp(-\mu(t)) \quad k=0,1,\dots,N \text{ визначається емпірично, де}$$

$$\mu(t) = \int_0^t \lambda(u) du$$

середня кількість відмов ПЗ за час t .

Оскільки функція $\mu(t)$ характеризує ступінь підвищення надійності ПЗ, то її зазвичай називають функцією зростання надійності. Вона описує взаємозв'язок між кількістю виявлених відмов ПЗ і часом, витраченим на тестування та усунення дефектів.

Введемо такі позначення:

v – кількість дефектів у ПС на початку тестування (дорівнює кількості відмов, які можуть виникнути при $t \rightarrow \infty$), β – коефіцієнт пропорційності, що дорівнює швидкості виявлення одного дефекту.

До основних моделей надійності цього типу належать:

1) експоненційна, функція зростання надійності якої задається формулою

$$\mu(t) = v(1 - \exp(-\beta t)), \quad v, \beta > 0,$$

2) логарифмічна, функція зростання надійності для якої

$$\mu(t) = \frac{1}{\theta} \ln(\lambda_0 \theta t + 1), \quad \lambda_0, \theta > 0$$

де λ_0 – початкова інтенсивність відмов, θ – коефіцієнт, який характеризує зниження інтенсивності відмов з кожним усуненням дефектом.

3) сповільненого S -подібного зростання надійності, для якої

$$\mu(t) = v(1 - (1 + \beta t)\exp(-\beta t)), \quad v, \beta > 0$$

4) S -подібного зростання надійності з перегинами, яка має вигляд

$$\mu(t) = v(1 - \exp(-\beta t)) / (1 + \psi \exp(-\beta t)),$$

де ψ – параметр перегину, який визначається як: $\psi(r) = (1-r)/r$, $0 \leq r \leq 1$, де r – частота перегинів, що являє собою відношення виявлених дефектів до загальної їх кількості.

Предмет та інструмент тестування

На основі вищенаведеного підходу створено програмний модуль для тестування та прогнозування надійності програмного забезпечення, який дає змогу аналізувати тепловий розподіл у кристалі із плоскими джерелами тепла. Інтерфейс компонента тестування показано на рис. 1. Для автоматизованого тестування використано технологію TestUnit. Автоматизований тест містив випадкову генерацію вхідних даних задачі теплопровідності у заданих межах. Вхідні дані містили теплофізичні характеристики об'єкта, а саме теплопровідність, теплоємність та функцію джерела, дані граничних умов, а також дані конструкції. Теплофізичні характеристики вважалась сталими при тестуванні. Вхідні дані конструкції задавались геометричними розмірами кристала та джерел тепла, топологією джерел тепла у просторі.

Програмний модуль на основі отриманих від користувача вхідних даних дає змогу прогнозувати надійність ПЗ за допомогою моделі сповільненого S-подібного зростання надійності. Числові результати тестування можна перевести в графічну форму, а також зберегти у файл.

Тестування виявило програмні помилки, пов'язані із переповненням оперативної пам'яті при надто щільному розбитті області числовими методами; помилки, пов'язані із від'ємними чи близькими до нуля вхідними даними та інші.

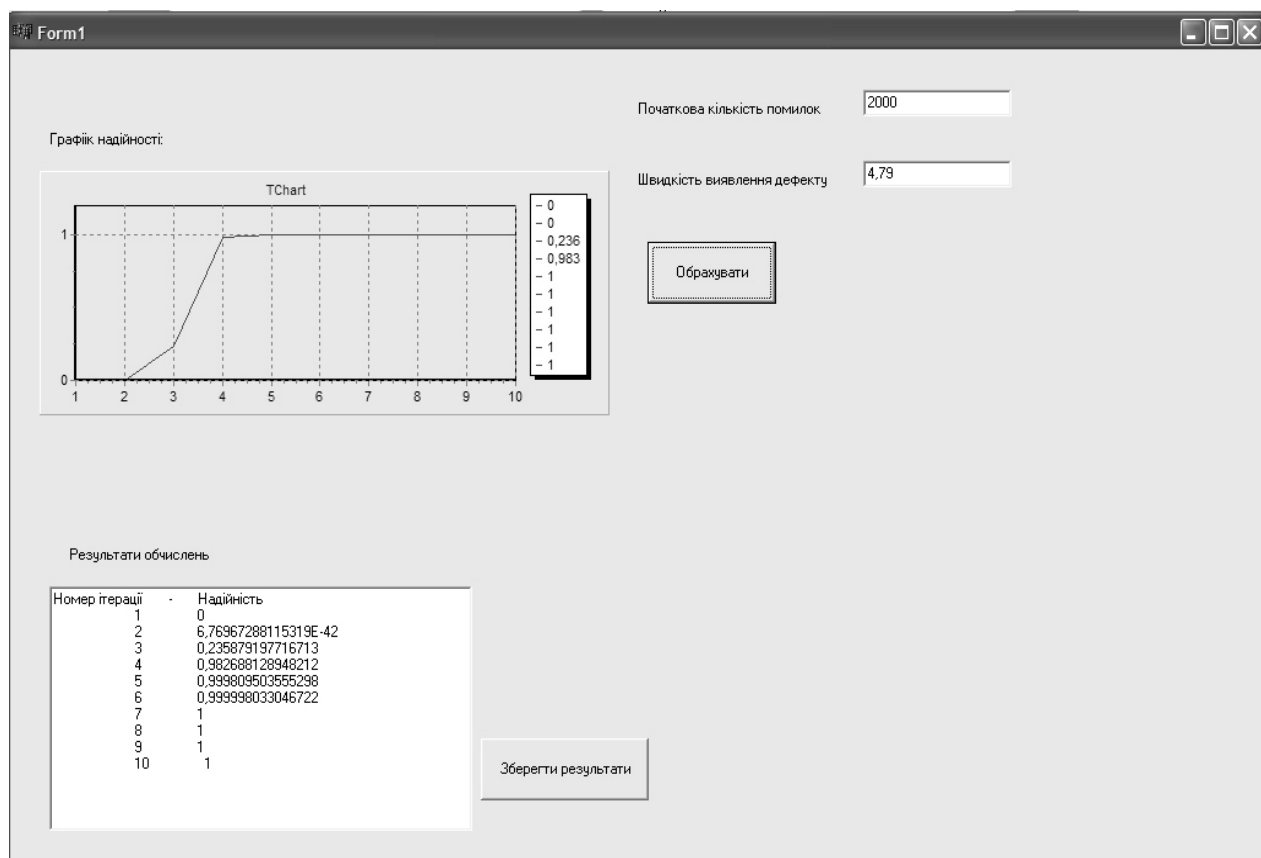


Рис. 1. Інтерфейс компонента тестування

Приклади обчислень та результати

У цій роботі верифіковано програмний продукт при таких параметрах: кількість тестів за ітерацію – 2000, загальна кількість ітерацій – 8.

Експериментальні дані, отримані при проведенні тестування на кожній ітерації

	Ітерація							
	1	2	3	4	5	6	7	8
Разом тестів	2000	2000	2000	2000	2000	2000	2000	2000
Разом помилок	2000	2000	1528	261	249	235	19	0
Кількість помилок 1 Типу EAccessViolation	2000	1035	1008	47	37	0	0	0
Кількість помилок 2 Типу EInvalidPointer	0	965	14	0	0	0	0	0
Кількість помилок 3 Типу EZeroDivide	0	0	99	8	13	11	19	0
Кількість помилок 4 Типу EStackOverflow	0	0	407	206	199	224	0	0

Програма для тестування на основі даних результатів кожної ітерації дає змогу визначити надійність програмного продукту який тестувався. У таблиці наведено експериментальні дані отримані при проведенні тестування на кожній ітерації.

Класи помилок, що виникли в процесі тестування: *EAccessViolation* – клас виключень (exceptions) для недопустимих помилок доступу до пам'яті; *EInvalidPointer* – клас виключень для помилок при операціях з вказівниками, що свідчить про некоректний вказівник на пам'ять; *EZeroDivide* – клас виключень для помилок при діленні на нуль; *EstackOverflow* – клас виключень для помилок переповнення стеку.

Згідно з даними, наведеними у таблиці, після третьої ітерації кількість помилок значно зменшується, а після шостої прямує до нуля. Табличні дані представлено на рис. 2 у вигляді графіка, де показано загальну кількість помилок, та окремо розподіл за типами помилок.

Відповідно до отриманих експериментальних даних визначається надійність програмного продукту.

За допомогою програми тестування на основі розподілу Пуассона, емпірично визначивши $\beta=4,79$ отримується графік зростання надійності, який максимально збігається з результатами, отриманими внаслідок експерименту.

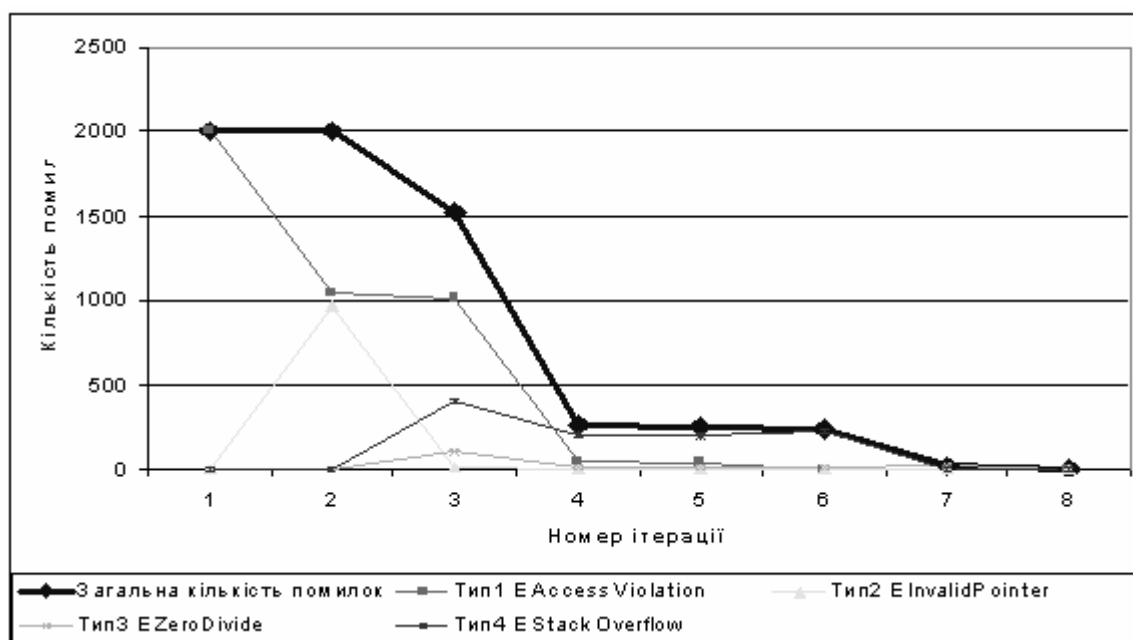


Рис. 2. Загальна кількість помилок та розподіл помилок за типами

Порівняння експериментальної та програмно визначеної надійності представлено на рис. 3. Як видно з графіків, після четвертої ітерації тестування надійність програмного продукту прямує до 1, що дає змогу характеризувати програмний продукт як надійний.



Рис. 3. Графік зростання надійності програмного продукту

Висновки

У статті проаналізовано результати роботи програмного продукту для автоматизації тестування надійності ПЗ, що ґрунтується на випадковому процесі Пуассона, яка визначає та дає змогу прогнозувати кількість відмов ПЗ залежно від затрачених ресурсів на тестування програми. Для оцінювання надійності на основі розподілу Пуассона вибрано модель сповільненого S-подібного зростання. Ця модель відповідає практиці проведення тестування ПЗ і найточніше – реально отриманим результатам. Обрана форма кривої зростання надійності (S-подібна) пояснюється так: спочатку йде крива навчання, потім тестова команда ознайомлюється з ПЗ, і крива починає швидко зростати, поки нарешті не переходить в третю фазу – так, що несправності все важче виявити. Для системи теплового проектування емпірично визначено параметр моделі Пуассона β . Показано, що програмний модуль дає змогу прогнозувати зростання надійності ПЗ, що значно скорочує затрати на тестування.

1. Шураков В.В. Надежность программного обеспечения систем обработки данных. – М.: Статистика, 1981. – С. 215.
2. Gokhale S.S., Marinos P.N., Trivedi K.S. Important Milestones in Software Reliability Modelin, In Proc. of Software Engineering and Knowledge Engineering, 1996.
3. Handbook of Software Reliability Engineering, Editor M. Lyu, McGraw Hill, 1996.
4. Laprie J.C. Dependability – its attributes, impairments and means. In Predictably Dependable Computing Systems, chapter I, Basic Concepts, pages 3-24, Springer Verlag, 1995.
5. Казарин О.В. Теория и практика защиты программ. – М., 2004. – С.450.
6. Гласс Р. Руководство по надежному программированию: Пер. с англ. – М.: Финансы и статистика, 1982. – С. 280.
7. Липаев В.В. Тестирование программ. – М.: Радио и связь, 1986 – С. 292 с.
8. Коротун Т.М. Моделі і методи інженерії тестування програмних систем: Автореф. дис... канд. техн. наук. – К., 2005. – С. 21.