

УДК 004.46

Шевченко В.Л. д.т.н, с.н.с.;

Кірпічников Ю.А. к.т.н.;

Головченко О.В.;

Федорієнко В.А.;

Кондратенко Ю.В.

Центр воєнно-стратегічних досліджень Національного університету оборони України
імені Івана Черняхівського

Підходи щодо оцінки надійності програмної компоненти головної системи єдиного інформаційного середовища

Подходы к оценке надежности
программной компоненты
главной системы единой
информационной среды

Approaches to assessing the
reliability of the main software
component of a unified information
environment

Резюме. Стаття присвячена вивченню підходів щодо оцінки надійності програмної складової щодо створення єдиного інформаційного середовища.

Резюме. Стаття посвящена изучению подходов к оценке надёжности программной составляющей для создания единого информационного пространства.

Resume. The article is devoted to studying of approaches of the variants of the reliability to estimate on the software component for the creation of a single information environment.

Ключові слова: надійність програмного забезпечення, метод Монте-Карло, єдине інформаційне середовище.

Ключевые слова: надежность программного обеспечения, метод Монте-Карло, единая информационная среда.

Keywords: software reliability, the Monte Carlo method, a single information environment.

Постановка проблеми. У складних програмних системах надійність програмного забезпечення (ПЗ) відіграє провідну роль при визначенні ефективності інформаційних систем (ІС). Такими інформаційними системами можуть бути різні за складністю програмно-технічні та програмні комплекси. Для більш ефективного управління доцільно створення єдиного інформаційного середовища (ЄІС), що об'єднало б у собі подібні ІС, які є його елементами. Наприклад, це може бути автоматизована системи класу ERP, яка виконує роль головної системи для обміну даними в ЄІС; автономні ІС, поєднані між собою через засоби телекомунікаційних систем; окремі автоматизовані робочі місця (АРМ), що можуть бути реалізовані на персональній електронно-обчислювальній машині (ПЕОМ). Зазначені елементи ЄІС уже мають певне встановлене ПЗ. Але, для повного функціонування ЄІС необхідне подальше доопрацювання, розроблення, встановлення та супроводження спеціального

програмного забезпечення. Тому оцінка надійності програмної компоненти ЄІС (або ПЗ) є актуальною задачею.

За аналогією з надійністю технічних засобів під надійністю ПЗ розуміється властивість щодо виконання покладених функцій, зберігаючи свої характеристики у встановлених межах за певних умов експлуатації, що визначається його безвідмовністю і відновлюваністю [1]. Оцінка надійності ПЗ необхідна для вибору найкращого складу програмно-технічних засобів щодо створення єдиного інформаційного середовища. За умови використання складного програмно-технічного комплексу в єдиному інформаційному середовищі постає задача розрахунку надійності програмної та технічної компонент. Актуальними питаннями є розробка рекомендацій щодо оцінки і перевірки надійності програмного забезпечення проєктів інформатизації та розрахунок необхідної кількості фахівців щодо відновлення ПЗ.

Актуальними також є дослідження щодо підбору ПЗ для ЄІС та його відновлення до працездатного стану із врахуванням фактора надійності.

Актуальність проблеми надійності також підтверджує той факт, що при низькій надійності та нераціональності організації процесу експлуатації вартість використання програмно-технічного комплексу на порядок перевищує вартість його придбання.

Ступінь розробленості проблеми.

Поняття надійності інформаційних систем (програмно-технічного комплексу) розкрито у дослідженні [1] та визначено в державних стандартах України [2, 3]. Поняття оцінки надійності було висвітлено в роботах авторів [1, 4]. Напрямок теорії надійності щодо оцінки програмного забезпечення використовує той же понятійний апарат, що і для технічних засобів та може використовувати спільні методи. Для оцінки надійності програмної компоненти ЄІС можливе використання різноманітних методів. Одним із найбільш ефективних є метод Монте-Карло.

Метою статті є вивчення підходів до оцінки надійності програмної складової щодо створення єдиного інформаційного середовища.

Виклад основного матеріалу. Теорія надійності – це система понять, математичних методів та моделей, спрямованих на вирішення проблем прогнозування, оцінки та оптимізації показників надійності та ефективності процесу експлуатації [4]. Стаття присвячена дослідженню надійності відновлювальних об'єктів.

Відповідно до ГОСТ 27.002-89 під *надійністю* розуміють властивість об'єкта зберігати певний термін у встановлених межах значення всіх параметрів, які характеризують здатність виконувати потрібні функції в заданих режимах і умовах використання, технічного обслуговування, ремонтів, зберігання та транспортування [5].

Основні причини відмови програмного забезпечення це помилки, приховані в програмі; спотворення вхідної інформації; невірні дії користувача; несправність апаратних засобів ІС, на якій реалізується обчислювальний процес.

1. Помилки, приховані в програмі. При розробці складного ПЗ можливе виникнення помилок, які не вдається виявити і ліквідувати в процесі налагодження. Серед помилок подібного роду можна виділити наступні характерні групи.

1.1. Помилки обчислень, пов'язані з некоректним записом або програмуванням математичних виразів, а також невірним перетворенням типів змінних.

1.2. Логічні помилки є причиною спотворення алгоритму розв'язання задачі (невірна передача управління, невірне завдання діапазону зміни параметра циклу, невірне умова тощо).

1.3. Помилки введення-виведення: управління введенням-виведенням, формуванням вихідних записів, визначення розміру записів тощо.

1.4. Помилки маніпулювання даними: невірне визначення числа елементів даних; невірні початкові значення, присвоєні даними; неправильне зазначення довжини операнда або імені змінної та інші помилки.

1.5. Помилки сумісності (несумісності) ПЗ з операційною системою або іншими прикладними програмами.

1.6. Помилки сполучень – невірна взаємодія ПЗ з іншими програмами, з системними програмами, пристроями ПЕОМ або вхідними даними.

2. Спотворення вхідної інформації виникає, коли вхідні дані не потрапляють у допустиму область значення змінних, що викликає невідповідність між вихідною інформацією і можливостями програми.

3. Невірні дії користувача пов'язані з неправильною інтерпретацією повідомлень, з неправильними діями користувача при роботі в діалоговому режимі. Часто ці помилки є наслідком неякісної програмної документації.

4. Несправності апаратних засобів ІС призводять до порушення ходу обробки інформації і, як наслідок, можуть спотворювати, як вихідні дані, так і саму програму.

Наслідком появи помилок у програмі є її *відмова*, яка веде до повного припинення виконання функцій програми або до короткочасного порушення ходу обробки інформації в ІС. Ступінь серйозності наслідків відмов ПЗ оцінюється співвідношенням між часом відновлення програми після відмови і динамічними характеристиками об'єктів, що використовують результати роботи цієї програми [6]. Для програмної компоненти можливо застосувати інше визначення – під *відмовою ПЗ* розуміється недопустиме відхилення характеристик функціонування цього забезпечення від запропонованих вимог, а саме зниження значень по певним параметрам.

Наведемо фактори, що визначають надійність інформаційних систем. Для побудови надійних ІС можливо використовувати різні види забезпечення: економічне, часове, організаційне, структурне, технологічне, експлуатаційне, соціальне, ергатичне, алгоритмічне, синтаксичне, семантичне.

Організаційне, економічне і часове забезпечення обумовлені необхідністю матеріальних і часових затрат, використовуються для підтримки достовірності результатів роботи ІС. Вони включають: правові та методичні аспекти функціонування ІС; нормативи достовірності інформації по функціональних підсистемах і етапах перетворення інформації; методики вибору та обґрунтування оптимальних структур, процесів і процедур перетворення інформації.

Структурне забезпечення ІС має забезпечувати надійність функціонування технічних комплексів і ергатичних ланок, а також ІС у цілому. Раціональна структура ІС, залежить від структури технологічного процесу перетворення інформації, визначає взаємозв'язок та резервування окремих функціональних ланок системи.

Надійність *технологічного* забезпечення пов'язана з вибором схемних і конструктивних рішень окремих засобів і технологічних комплексів системи, технологій і протоколів реалізації інформаційних процесів.

Експлуатаційне забезпечення пов'язано з вибором режимів роботи пристроїв, технологій обслуговування, профілактик і ремонтів.

До *соціального* забезпечення відносяться – створення здорової психологічної обстановки в колективі, підвищення відповідальності за виконану роботу, підвищення кваліфікації фахівців, моральної та матеріальної зацікавленості в правильності виконання роботи.

Ергатичне забезпечення пов'язане з раціональною організацією роботи людини в системі. Це правильний розподіл функцій між людьми та технічними засобами, обов'язковість норм і стандартів роботи, оптимальність інтенсивності і ритмічності, побудова робочих місць відповідно до вимог ергономіки.

Надійність *алгоритмічного* забезпечення пов'язана із забезпеченням високої якості і безпомилковості алгоритмів і програм перетворення інформації, реалізації контролю достовірності інформації.

Інформаційне, синтаксичне і семантичне забезпечення повинне забезпечити спеціальну інформаційну надмірність, надмірність даних і смислової надмірність, що обумовлюють можливість проведення контролю достовірності інформації.

Особисті якості виконавців також відіграють важливу роль. Помилки осіб, що відповідають за працездатність ПЗ, вихід ІС зі штатного режиму експлуатації чинності випадкових або навмисних дій користувачів або

осіб, що відповідають за працездатність ПЗ – операторів (перевищення розрахункового числа запитів, надмірний обсяг оброблюваної інформації та інші не виправдані дії), неможливість або небажання обслуговуючого персоналу виконувати свої функції призводять до надзвичайно серйозних наслідків (тривале простоювання у роботі ІС, спотворення оброблюваної інформації та отримання невірних результатів, втрата інформації, збої в роботі програм та обладнання, відмови устаткування).

Таким чином, підтримка високої надійності роботи ІС у цілому є важливим і складним інженерно-технічним та соціально-організаційним завданням.

Щодо безпосередньої надійності ПЗ, то розв'язання будь-якої задачі, виконання будь-якої функції, покладеної на клієнтське автоматизоване робоче місце (АРМ), розгорнутому на базі ПЕОМ, що працює в мережі або локально, можливо при взаємодії апаратних і програмних засобів. При аналізі надійності виконання на АРМ заданих функцій слід розглядати комплекс апаратних і програмних засобів [6]. Але в даній статті для досліджуємо лише програмну компоненту.

Надійність ПЗ визначається його безвідмовністю і відновлюваністю. *Безвідмовність ПЗ* – це властивість зберігати працездатність для обробки інформації в ІС. Безвідмовність програмного забезпечення оцінюється ймовірність його роботи без відмов за певних умов зовнішнього середовища протягом заданого періоду спостереження. Певні умови зовнішнього середовища – це сукупність вхідних даних і стан ІС. Заданий період спостереження відповідає часу, необхідному для виконання на АРМ поставленої задачі.

Безвідмовність ПЗ може характеризуватися середнім часом виникнення відмов при функціонуванні програми. При цьому припускаємо, що апаратні засоби ЄІС знаходяться у справному стані. З точки зору надійності, принципова відмінність ПЗ від апаратних засобів полягає в тому, що програми не зношуються та їх вихід з ладу через поломку неможливий [7]. Отже, характеристики функціонування ПЗ залежать тільки від його якості, зумовлений процесом розробки. Це означає, що безвідмовність ПЗ визначається його коректністю і залежить від наявності в ньому помилок, внесених на етапі його створення. Крім того, прояв помилок ПЗ пов'язаний з деякими моментами часу витраченими на обробку даних, які програма не

спроможна коректно обробити. Тому вхідні дані певною мірою впливають на функціонування ПЗ.

Проведемо уточнення поняття зношеності ПЗ. Фізично програмна компонента не зношується, але її параметри з часом можуть знижуватися (під зношенням вважаємо втрату корисних властивостей програми). Наприклад, при використанні операційної системи Microsoft Windows XP у складних умовах експлуатації зростає ступінь відмов за рахунок некоректного реагування ПЗ на підвищення навантаження. У результаті цього здійснюється перепоповнення буферів пам'яті. При використанні ПЗ без виконання дій щодо його обслуговування спостерігається зниження значень за певними параметрами. Для виправлення ситуації повторно встановлюють ПЗ (інсталиують) на ПЕОМ. Щодо головної системи ЄС, то для підтримки у працездатному стані потрібно проводити деякі дії з обслуговування. Це може бути "розумне адміністрування". Воно було реалізоване на прикладі ПЗ SAP NetWeaver Business Intelligence 7.0 [8]. Тож, вважаємо, що зношеність ПЗ можлива і відновлення програмної компоненти теж, але більш детальний розгляд зношеності ПЗ виходить за межі дослідження.

Одним із проявів безвідмовності є *стійкість функціонування*, під якою розуміється здатність ПЗ обмежувати наслідки власних помилок і несприятливих впливів зовнішнього середовища або протистояти їм. Стійкість ПЗ зазвичай забезпечується за допомогою введення різних форм надмірності, що дозволяють мати дублюючі модулі програм, альтернативні програми для одних і тих же завдань, здійснювати контроль за процесом виконання програм.

Показники надійності для програмної складової ЄС (крім показника відновлення) за умови випадкового характеру відмови, набувають рис показників, які використовуються для технічних засобів:

- ймовірність безвідмовної роботи програми $p(t)$, що представляє собою ймовірність того, що помилки програми не з'являться на інтервалі часу $(0, t)$;
- ймовірність відмови програми $q(t)$ або ймовірність події відмови ПО до моменту часу t ;
- інтенсивність відмов програми $\lambda(t)$;
- середнє напрацювання програми на відмову T , що є математичним очікуванням часового інтервалу між послідовними відмовами.

При визначенні характеристик надійності ПЗ (за умови його відновлення) будемо враховувати той факт, що помилки, які виникають при роботі програм усуваються, та їх

кількість зменшується. Тому, їх інтенсивність знижується, а напрацювання програми на відмову збільшується. У зв'язку з такими припущеннями у літературі розглядається кілька моделей надійності ПЗ [1]: модель з дискретно-понижувальною частотою появи помилок, модель з дискретним збільшенням напрацювання на відмову або помилку, експоненціальна модель надійності.

У моделі з дискретно-понижуючою частотою появи помилок (рис. 1) інтенсивність відмов програми $\lambda(t)$ є постійною величиною до виявлення виниклої помилки. Після її усунення значення $\lambda(t)$ зменшується й інтенсивність відмов знову стає константою. Модель передбачає, що між $\lambda(t)$ і числом помилок, що залишилися в програмі існує залежність $\lambda(t) = K \cdot (M - i) = \lambda_i$, де M – невідоме початкове число помилок; i – число виявлених помилок, залежне від часу t ; K – деяка константа. Щільність розподілу часу виявлення i -й помилки t_i визначається співвідношенням $\lambda(t_i) = \lambda_1 e^{-t_i \lambda_1}$.

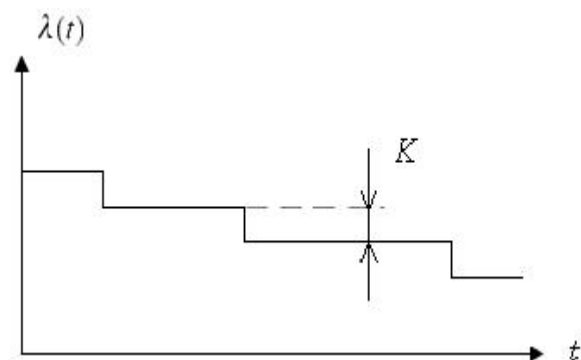


Рис. 1. Зміна інтенсивності відмов програми від часу напрацювання

Значення параметрів K і M можна оцінити на підставі послідовності спостереження інтервалів між моментами виявлення помилок.

На практиці умови розглянутої моделі нерідко не дотримуються, а саме:

- не завжди при усуненні помилки інтенсивність відмов зменшується на одну і ту ж величину K , оскільки різні помилки мають різний вплив на хід виконання програми;
- досить часто виникають ситуації, при яких усунення одних помилок призводить до появи нових;
- не завжди вдається усунути причину помилки і програму продовжують використовувати, оскільки при інших вихідних даних помилка може себе і не проявити.

Для цієї моделі за основним допущенням є прийняте збільшення часу напрацювання на відмову. Це є припущення про те, що відмови і помилки програми на початку експлуатації

виникають часто. У міру налагодження програми таких помилок стає менше, а час напрацювання на відмову після ліквідації чергової відмови збільшується (Рис. 2).

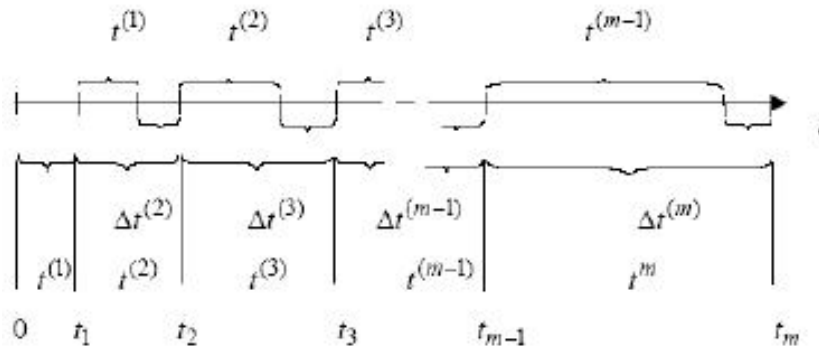


Рис. 2. Шкала інтервалів часу напрацювання на відмову ПЗ

На шкалі позначені величини: $t_1, t_2, t_3, K, t_{m-1}, t_m$ - випадкові моменти часу виникнення першої, другої, третьої і так далі - m -ої відмови; $t^{(1)}, t^{(2)}, t^{(3)}, K, t^{(m-1)}, t^{(m)}$ - випадкові інтервали, $\Delta t^{(1)}, \Delta t^{(3)}, \Delta t^{(m-1)}, \Delta t^{(m)}$ - час між сусідніми відмовами програми.

Нехай перша помилка, яка з'явилася у результаті відмови програми, сталася у випадковий момент часу t_1 і була усунена. Напрацювання до першої відмови та виникнення помилки рівний інтервалу часу $t^{(1)}$. Друга помилка виникла в момент часу t_2 . Напрацювання до другої помилки визначається інтервалом $t^{(2)}$. Відповідно до припущення, цей інтервал більше, ніж $t^{(1)}$, оскільки після перезапуску програма пропрацювала час до першої ліквідованої помилки, продовжила роботу до нової, другої помилки. Отже, інтервал $t^{(2)}$ можна представити у вигляді $t^{(2)} = t^{(1)} + \Delta t^{(2)}$, де $\Delta t^{(2)}$ - доповнення інтервалу до величини $t^{(2)}$. Узагальнення до будь-якого i -го інтервалу, може бути записане, як $t^{(i)} = t^{(i-1)} + \Delta t^{(i)}$.

Враховуючи, що від моменту часу $t_0 \in [0, t_1]$ не виявлено жодної помилки програми і в силу того, що інтервал $t^{(1)}$ порівняно невеликий, оскільки помилки програми спочатку її експлуатації відбуваються досить часто, можна уявити інтервал $t^{(1)}$, як $\Delta t^{(1)}$. Тоді, з урахуванням цієї заміни, вираз для t_m набуде вигляду:

$$t_m = m\Delta t^{(1)} + (m-1)\Delta t^{(2)} + (m-2)\Delta t^{(3)} + L + 2\Delta t^{(m-1)} + \Delta t^{(m)}$$

Основним припущенням цієї експоненціальної моделі є відповідний характер зміни числа помилок у програмі в часі. Прогноз надійності програми проводиться на підставі даних, одержуваних під час її тестування.

Основними параметрами моделі є: τ - сумарний час функціонування від початку тестування (з усуненням виявлених помилок) до моменту оцінки надійності; M - число помилок, наявних у програмі перед початком тестування; $m(\tau)$ - кінцеве число виправлених помилок; $m_0(\tau)$ - число помилок, які залишилися.

Доведено [4], що число помилок у програмі в кожен момент часу має розподіл Пуассона

$$P_m = \frac{a^m}{m!} e^{-a}$$

де $m = 0, 1, 2, \dots$ - число подій, a - параметр закону Пуассона. Якщо випадкова величина у найпростішому потоці подій за час τ деяке ціле позитивне значення m , то ця величина розподілена за законом Пуассона. Параметр цього розподілу змінюється після розподілу чергової помилки. Інтенсивність відмов вважається безперервною функцією, пропорційної числу помилок, які залишилися. З урахуванням уведених параметрів і припущень очевидно, що

$$\lambda(\tau) = C \cdot m_0(\tau),$$

де C - коефіцієнт пропорційності, що враховує швидкодію АРМ і число команд у програмі. Тому, для визначення надійності програмного забезпечення беруть до уваги найпростіші потоки відмов - пуассонівські, які мають такі властивості, як стаціонарність, ординарність та відсутність послідовності [1, 6].

Стаціонарність потоків відмов являє собою ймовірність попадання того чи іншого числа подій на ділянку часу довжиною Δt залежить тільки від довжини ділянки і не залежить від місця положення на осі t . Таким

потокам відповідає експоненціальний закон розподілу інтервалу часу між відмовами. Вважають, що відмови розподілені на осі часу в процесі експлуатації з однаковою середньою щільністю λ .

Ординарність визначається тим, що ймовірність виникнення двох або більше відмов системи в деякий момент часу t незрівнянно мала по відношенню до ймовірності однієї відмови. Це означає практично, що одночасно в системі відмови більше двох елементів бути не може.

Відсутність послідовності визначається тим, що поява відмови в момент t не залежить від того, скільки відмов і в які моменти часу вони виникали до моменту t .

Для показників надійності використовують ймовірнісну та статистичну форми подання у відповідних термінах [4]. Наприклад, основними показниками безвідмовності відповідно до ГОСТ 27.002 є ймовірність безвідмовної роботи, інтенсивність відмов і середній наробіток об'єктів до відмови [5].

Відповідно до Держстандарту України [3] рекомендовано вибирати такі види функцій розподілу для описування випадкових величин наробітку до відмови: експоненціальний розподіл, логарифмічно-нормальний розподіл, розподіл Вейбула, дифузійно-монотонний розподіл (DM-розподіл), дифузійно-немонотонний розподіл (DN-розподіл).

Як альтернатива методам щодо визначення розподілів пропонуються методи Монте-Карло.

Узагалі, методи Монте-Карло – це загальна назва групи числових методів, оснований на одержанні великої кількості реалізацій стохастичного (випадкового) процесу, який формується у той спосіб, щоб його ймовірнісні характеристики збігалися з аналогічними величинами задачі, яку потрібно розв'язати.

Суть класичного методу Монте-Карло для визначення оцінки полягає в необхідності знаходження значення a деякої досліджуваної величини [7]. Слід зазначити, що метод Монте-Карло для визначення надійності ПЗ найкраще використовувати разом з ймовірнісними моделями, де набір даних проводиться по випадковим даними за правилами розподілу Пуассона [10]. Теорія цього методу вказує на те, що найбільш правильно підібрати випадкову величину, і як знайти її можливі значення. Для цього, розробляються способи зменшення дисперсії випадкових величин, у результаті чого зменшується помилка. Однак даний метод є апроксимаційним, і не дозволяє отримати точне значення надійності.

Для реалізації обчислень методом Монте-Карло при кожному визначенні дуги і вершин графа (рис.3) генерується випадкове число у діапазоні від 0 до 1. Після цього перевіряється нерівність: $1 - p_{(c)} > p$, де p_c – випадкове число, а p – ймовірність появи дуги і вершини графу.

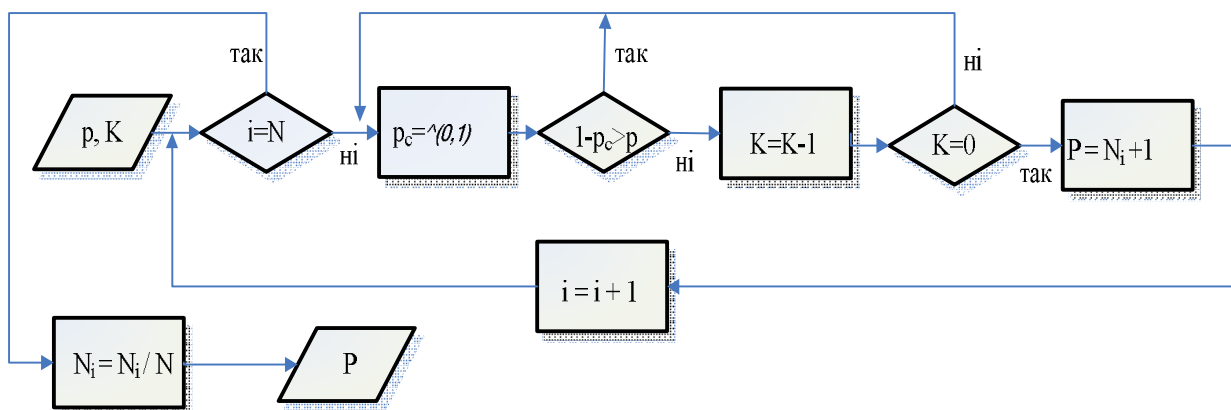


Рис. 3. Загальний алгоритм для оцінки надійності ПЗ за методом Монте-Карло

Шляхи, в які входять дуги і вершини для яких рівність виконується, відкидаються. Якщо в ході ітерації всі шляхи відкинуті, то відмова (помилка) на з'являється і ця ітерація вважається неуспішною. Якщо в ході ітерації залишився хоча б один шлях, то випробування вважається успішним. Ймовірність визначається виразом: $P = N_1 / N$ [11], де N – загальна кількість

проведених випробувань, N_1 – кількість випробувань, що закінчилися успішно.

Для обчислення певної функції приналежності при застосуванні методу Монте-Карло необхідно оцінити похибку одержуваного результату. Переваги методу Монте-Карло полягають у тому, що алгоритм може працювати довгий або короткий період часу з пропорційною зміною рівня точності.

Висновок. У статті вивчені підходи до оцінки надійності програмної складової щодо створення єдиного інформаційного середовища. Визначені основні причини відмови ПЗ та запропонований загальний алгоритм, який полягає у використанні методу Монте-Карло для оцінки надійності програмної складової ЄС.

Подальші дослідження слід присвятити розробці моделі надійності програмної компоненти ЄС за допомогою методу Монте-Карло.

СПИСОК ЛІТЕРАТУРИ

1. Надёжность информационных систем: учебное пособие / Ю. Ю. Громов, О. Г. Иванова, Н. Г. Мосягина, К. А. Набатов. – Тамбов : ГОУ ВПО ТГТУ, 2010. – 160 с.
2. Надійність техніки. Терміни та визначення (ІЕС 50 (191)) : ДСТУ 2860-94. – [Чинний від 01.01.1996]. – К. : Держстандарт України, 1996. – 34 с. – (Національні стандарти України).
3. Надійність техніки. Методи розрахунку показників надійності: ДСТУ 2862-94. – [Чинний від 01.01.1996]. – К. : Держстандарт України, 1996. – 39 с. – (Національні стандарти України).
4. Соловйов В. І. Основи теорії надійності і експлуатації авіаційних систем / Соловйов В. І. – К. : КІ ВПС, 2000. – 248 с. – (Курс лекцій).
5. Надёжность в технике. Основные понятия. Термины и определения : ГОСТ 27.002–89. – [Чинний від 01.07.1990]. – М. : Изд-во стандартов, 1990. – 24 с.
6. Курлаев С. А. Курс: Компьютерное моделирование : (електронні курси НГТУ) [Електронний ресурс] / Курлаев С. А., Волкова В. М., Гульятеева Т.А. – Новосибирск : Новосибирский гос. техн. ун-т, 2013. – (Инст. дистанционного навчання) – Режим доступа : <http://dispace.edu.nstu.ru/didesk/course/show/3293>.
7. Ермаков А. А. Основы надежности информационных систем / Ермаков А. А. – Иркутск. : ИРГУПС, 2006. – 151 с. – (Учебное пособие).
8. Федорієнко В. А. Аналіз переваженості аналітичної системи / В. А. Федорієнко, Н. О. Федорієнко, О. А. Кошлань // К: Збірник наукових праць ЦВСД НУОУ. – 2011. – №1(42) – С. 27–37.
9. Alex F Bielajew. Fundamentals of the Monte Carlo method for neutral and charged particle transport [Електронний ресурс] / Alex F Bielajew. – Michigan : The University of Michigan, 2001. – 348 с. – Режим доступа : <http://www-personal.umich.edu/~bielajew/MCBook/book.pdf>.
10. Перегуда А. И. Оценка показателей надежности автоматизированного технологического комплекса «Объект защиты – система безопасности» с нечеткими параметрами методом Монте-Карло [Електронний ресурс] / А. И. Перегуда, Д. А. Тимашов. // Информационные технологии – 2008. – №10. – С.7–15. – Режим доступа до журн. http://library.mephi.ru/data/scientific-sessions/2010/fulltext_t5/4-7-1.doc.
11. Бережной В. В. Программная реализация методов расчета структурной надежности информационно-вычислительных сетей [Електронний ресурс] / В. В. Бережной, В. В. Княжецкий. // Русская цивилизация: наука, образование, общество – 2009. – №3. – С.4. – Режим доступа до журн. : <http://publ.uchis-online.ru/files/cd2.pdf>.