

Adaptive control methods

УДК 004.41:004.056

doi: 10.20998/2522-9052.2018.3.07

О. В. Коваленко

Центральноукраїнський національний технічний університет, Кропивницький, Україна

УДОСКОНАЛЕНИЙ МЕТОД УПРАВЛІННЯ РИЗИКАМИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ НАПІВМАРКОВСЬКОЇ МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ

В роботі проведені дослідження, які показали, що управління ризиком розробки програмного забезпечення включає систему заходів, здійснюваних як до прояву негативної події, так і після його реалізації. Однак превентивний аналіз і облік більшості можливих експлуатаційних помилок дозволить знизити фінансові та інші витрати в життєвому циклі розробки програмного забезпечення. Вже згадана задача управління ризиками розробки програмного забезпечення при певних обмеженнях на заходи з тестування якості та безпеки, сформульована у вигляді напівмарківської моделі прийняття рішень для керованого марківського процесу в безперервному часі і дисконтованими доходами або витратами. При цьому даний вид експлуатаційних ризиків ототожнюється з послідовно з'єднаними незалежними елементами, відновлюваними за кінцевий час. Оптимальна нерандомізована стаціонарна стратегія управління визначена за допомогою псевдобулевих методів бівалентного програмування, знаходячи всі рішення системи обмежень. Удосконалено метод управління ризиками розробки програмного забезпечення на основі напівмарківської моделі прийняття рішень для керованого марківського процесу в безперервному часі. Відмінною особливістю запропонованого методу є використання псевдобулевих методів бівалентного програмування з нелінійною цільовою функцією і лінійними обмеженнями для визначення оптимальної стратегії усунення експлуатаційних помилок. Проведені дослідження в роботі показали, що використані теоретичні положення в достатньому обсязі відображають стандарти та можливості сучасних методологій тестування програмного забезпечення. Як приклад розглянуті ситуації виникнення помилок безпеки програмного забезпечення (ПЗ), і визначена оптимальна стратегія управління для усунення зазначеної аномальної ситуації. Представлений в роботі метод доцільно використовувати не тільки при управлінні ризиками безпеки програмного забезпечення, але і при функціональному, навантажувальному, стресовому та інших видах тестування для запобігання можливих втрат.

Ключові слова: управління ризиками; напівмарківська модель; прийняття рішень; псевдобулеві методи бівалентного програмування.

Вступ

Проведені дослідження показали, що управління ризиком розробки ПЗ полягає в завчасному виявленні пов'язаних з ризиком фінансових, технічних, психологічних, і ін. небезпек, і вжиття заходів щодо зниження ризику шляхом цілеспрямованої зміни цих факторів з урахуванням ефективності заходів, що вживаються [1-3]. Управління ризиком розробки ПЗ включає систему заходів, здійснюваних як до прояву негативної події, так і після його реалізації. Однак, як показали дослідження, превентивний аналіз і облік більшості можливих експлуатаційних помилок дозволить знизити фінансові та ін. витрати в життєвому циклі розробки ПЗ.

Аналіз досліджень та постановка завдання. Ряд авторів [1, 2] під терміном "управління ризиком" розуміють розробку і обґрунтування оптимальних програм діяльності, покликаних ефективно реалізовувати рішення в галузі забезпечення безпеки. При цьому головним елементом такої діяльності є процес оптимального розподілу обмежених ресурсів з урахуванням характерних експлуатаційних, економічних і соціальних факторів.

Розглянуту задачу управління ризиками розробки ПЗ при певних обмеженнях на заходи з тестування якості та безпеки, сформулюємо у вигляді напівмарківської моделі прийняття рішень для керованого марківського процесу в безперервному

часі і дисконтованими доходами (з коефіцієнтом в нормальних умовах процесу створення ПЗ) або витратами (в умовах з відхиленнями від плану, пов'язаними з ігноруванням не виявлення вразливостей (помилки) безпеки). При цьому даний вид експлуатаційних ризиків ототожнюються з послідовно з'єднаними незалежними елементами, відновлюваними за кінцевий час.

Оптимальну нерандомізовану стаціонарну стратегію управління визначимо за допомогою псевдобулевих методів бівалентного програмування, знаходячи всі рішення системи обмежень. Ці рішення визначаються на основі алгоритму перетину рішень окремих нерівностей-обмежень, запропонованого в роботі [4] для знаходження базисних рішень системи лінійних нерівностей з булевими змінними.

В таких умовах сформулюємо основну задачу. Нехай кожному стані $y \in S$, $\partial e S = \{0, 1, 2, \dots, N\}$ даної системи управління ризиками розробки ПЗ поставлено у відповідність кінцеву множину R_i рішень, елементи якої позначимо як $r = 1, 2, \dots, r_i$. Якщо система знаходиться в стані $i \in S$ і приймається рішення $r = R_i$, то її подальша поведінка визначається імовірнісним законом.

$$Y_{ij}^r(t) = P_{ij}^{(r)} F_{ij}^{(r)}(t), \quad j \in S, \quad (1)$$

де $P_{ij}^{(r)}$ – ймовірність переходу системи із стану в стан i ; $F_{ij}^{(r)}(t)$ – функція розподілу часу перебування системи в стані i при прийнятті рішення r та при умові, що наступний перехід відбудеться в стані j .

При цьому зробимо допущення, що виконані такі умови:

Стан $i = 0$ відповідає нормальному процесу розробки ПЗ, а $i \neq 0$ - ситуація помилки безпеки. Функції $F_{0j}^{(r)}(t)$ і $F_{j0}^{(r)}(t)$, $j \in \tilde{S} = S / \{0\}$, $r \in R_j$ разом зі своїми першими похідними безперервні при $t > 0$, за винятком кінцевого числа точок, і зростають відповідно до експоненціальним законом розподілу.

За одиницю часу перебування в стані i в разі прийняття рішення r витрачається в середньому $k_i^{(r)}$ коштів (при $i \neq 0$ число $k_i^{(r)}$ мінусове і дорівнює витратам системи за одиницю часу перебування в стані i за умови виходу з цього стану з урахуванням рішення r). Величини $|k_i^{(r)}|$ обмежені при всіх $i \in S$, $r \in R_i$ та ймовірності $P_i^{(r)}$ задовольняють співвідношенням:

$$\sum_{j \in S} P_i^{(r)} = 1, \quad i \in S, \quad r \in R_i, \quad P_{ij}^{(r)} \geq 0, \quad i, j \in S, \quad r \in R_i.$$

Таким чином, в кожному стані $i \in S$ є r_i рішень з кінцевої множини R_i . Вибір деякого рішення r з цієї множини R_i у відповідності $i \in S$ означає завдання величин $Y_{ij}^r(t)$, $P_{ij}^{(r)}$, $F_{ij}^{(r)}(t)$, $k_i^{(r)}$, $j \in S$.

При $i = 0$, $R_0 = \{0\}$, ймовірність $P_{0j}^{(r)} \neq 0$, $j \in S$ є ймовірністю переходу у стані j . Ймовірність $P_{0j}^{(r)} \neq 0$, $j \in S$ обчислюється на практиці як доля станів з помилками безпеки типу j в загальній сукупності вразливостей безпеки різних типів на основі даних передісторій процесу розробки ПЗ. В цьому випадку $F_{0j}^{(r)}(t)$ – функція розподілу часу тестової експлуатації ПЗ між виявленими помилками безпеки типу j .

При $i = 1, \dots, N$ для будь-якого $r \in R_i$, $P_{i0}^{(r)} = 1$, $P_{ij}^{(r)} = 0$, $j \neq 0$, функція $F_{i0}^{(r)}(t)$ це функція розподілу часу усунення вразливостей безпеки з використанням рішень r при помилці типу j .

За умови безперервності в часі досліджуваного процесу будемо користуватися переоцінкою експоненціального виду з нормою α , тобто якщо в певний момент часу витрати становлять якусь одиничну величину, то через час t ці витрати вже будуть $e^{-\alpha t}$ одиничних величин.

Тоді якщо k_i - витрата за одиницю часу, то сумарні витрати за час t має вигляд:

$$\int_0^t k_i e^{-\alpha \tau} d\tau = \frac{k_i}{\alpha} (1 - e^{-\alpha t}). \quad (2)$$

Позначимо i_n стан системи після n -го переходу, u_n - прийняте рішення, а τ_n - час перебування в цьому стані ($n = 0, 1, 2, \dots$), i_0 - початковий стан. Допустимо стратегію β для системи управління розробкою ПЗ визначимо як послідовність $\{\beta_0, \beta_1, \beta_2, \dots\}$, де $\beta_n(\bullet / z_n)$ - імовірнісна міра, зосереджена на функції обмеження $U(S)$ на прийнятті рішення (управління), які визначаються системою нерівностей:

$$\sum_{j \in S} c_{rj} x_{rj} \leq b_r, \quad r \in R = UR_j, \quad j \in S \quad (3)$$

і залежить від історії керованої системи до моменту n -го переходу $z_n = (i_0, u_0, \tau_0, \dots, i_{n-1}, u_{n-1}, \tau_{n-1}, i_n)$. Міра $\beta_n(\bullet / z_n)$ задає рандомізоване правило вибору рішення u_n на основі інформації z_n . Таку стратегію β можна назвати рандомізованою.

Стратегія β є марківською, якщо $\beta_n(\bullet / z_n) = \beta_n(\bullet / i_n)$, де $n = 0, 1, 2, \dots$. Марківська стратегія називається стаціонарною, якщо $\beta_n(\bullet / i_n) = \beta_n(\bullet / i_n)$. Щільність виміру такої стратегії при $i_n = i, u_n = r, (r \in R_i)$ позначимо $d_i^{(r)}$. Якщо стратегія β - марківська стаціонарна, то керований процес є напівмарківським.

Аналіз літератури показав, що найбільш популярна інформація про напівмарківські процеси і керовані напівмарківські моделі з додатковими витратами і дивідендами викладена в роботах [1-4].

Позначимо через $g_i(t, \alpha, \beta)$ сумарні витрати системи, керованої відповідно до стратегії β , з нормою переоцінки α , за час t життєвого циклу розробки ПЗ. Обов'язковою умовою є те, що процес починається в момент $t = 0$ зі стану i . Через $v_i(t, \alpha, \beta) = g_i(t, \alpha, \beta) / t$ позначимо сумарні середні витрати системи за час t за тих самих умов.

Нехай c_{rj} - витрати, пов'язані з реалізацією заходу r в разі події порушення безпеки ПЗ j і x_{rj} - булева змінна: $x_{rj} = 1$, якщо r застосовується при події j , $x_{rj} = 0$ в іншому випадку.

Припустимо, що загальний обсяг коштів, відпущених для усунення недоліків безпеки ПЗ (заходи типу r) обмежений константою b_r , тобто виконується нерівність (3).

Якщо витрати c_{rj} дозволяють виконати кожне з обмежень (3), то реалізована на підставі (3) система визначає в просторі \mathfrak{R}^d , $d = \dim R$, деяку кінцеву множину дискретних точок. Тоді відповідно до

робіт [5, 6] існує нерандомізована стаціонарна стратегія β^* , звана β - оптимальною, яка мінімізує сумарні середні витрати при довільній стратегії β і нормі переоцінки α ($\alpha > 0$). При цьому $v(\alpha, \beta) \in (N+1) \times 1$ -мірний вектор $(v_0(\alpha, \beta), \dots, v_N(\alpha, \beta))$, де

$$v_i(\alpha, \beta) = \lim_{t \rightarrow \infty} v_i(t, \alpha, \beta), i \in S. \quad (4)$$

Необхідно знайти α - оптимальну нерандомізовану марківську стаціонарну стратегію β^* , яка мінімізує сумарні середні витрати $v(\alpha, \beta)$ при довільному початковому розподілі процесу

$$y = (y_0, y_1, \dots, y_N), \quad (5)$$

$$\sum_{i \in S} y_i = 1, y_i \geq 0, i \in S. \quad (6)$$

Не зменшуючи спільності, в якості початкового розподілу візьмемо вектор $y = (1, 0, \dots, 0)$, тобто початковий стан системи. На основі напівмарківської моделі прийняття рішень це завдання приведемо до еквівалентної задачі бівалентного програмування з використанням псевдодобулевих методів.

Метою роботи є удосконалення методу управління ризиками розробки програмного забезпечення на основі напівмарківської моделі прийняття рішень для керованого марківського процесу в безперервну часі.

Оптимізаційна стратегія напівмарківських моделей прийняття рішень

Ймовірності переходів розглянутого, для системи розробки ПЗ, напівмарківського процесу прийняття рішень в моменти стрибків з стану i в стан j при прийнятті рішення $r \in R_i$ визначається стохастичною $(N+1) \times (N+1)$ матрицею $P^{(r)} = \{p_{ij}^{(r)}\}$, яка задає вкладений ланцюг Маркова. Елементи $p_{ij}^{(r)}$ при будь-яких $i, j \in S$ і $r \in R_i$ дозволяють визначати за формулою (1) спільну ймовірність $Q_{ij}^{(r)}(t)$ того, що тривалість перебування в стані i не перевищує час t зі стану i при $r \in R_i$ процес переходить в стан j з ймовірністю $p_{ij}^{(r)}$. Функції $Q_{ij}^{(r)}(t)$ в (1) задовольняють умовам

$$Q_{ij}^{(r)}(0) = 0, i, j \in S, r \in R_i, \quad (7)$$

$$\sum_{j \in S} Q_{ij}^{(r)}(\infty) = \sum_{j \in S} p_{ij}^{(r)} = 1, i \in S, r \in R_i. \quad (8)$$

За допомогою матриці $Q_{ij}^{(r)}(t) = \{Q_{ij}^{(r)}(t)\}$ перехідних розподілів, визначимо функцію

$$H_i^{(r)}(t) = \sum_{j \in S} Q_{ij}^{(r)}(t), i \in S, r \in R_i, \quad (9)$$

яка є функцією розподілу часу перебування процесу в стані i при прийнятті рішення $r \in R_i$.

Випадковий процес $(Z_t), t \geq 0$ зі значеннями $Z_t = i$, якщо в момент t система знаходиться в стані i , являється напівмарківським, і задається величинами $N, y, Q_{ij}^{(r)}(t), i, j \in S, r \in R_i$.

Напівмарківський процес називається регулярним, якщо за кінцевий проміжок часу він з ймовірністю $p_p = 1$ перейде в будь-який стан не більше кінцевого числа раз. Таким чином, регулярний напівмарківський процес за кінцевий проміжок часу завжди здійснює лише кінцеве число переходів. Далі в розділі будемо розглядати тільки регулярні напівмарківські процеси.

У разі одноелементних множин рішень R_i в результаті стандартних для теорії відновлення [7] міркувань отримуємо наступне рівняння відновлення

$$i \in S, v_i(t) = (1 - H_i(t)) \cdot (k_i/\alpha) \cdot (1 - e^{-\alpha t}) + \sum_{j \in S} \int_0^t ((k_i/\alpha)(1 - e^{-\alpha t}) + e^{-\alpha t} v_j(t - \tau)) dQ_{ij}(\tau),$$

де $v_i(t)$ - короткий запис сумарних середніх витрат $v_i(t, \alpha, \beta)$ за час t .

У випадку кінцевих множин R_i рівняння відновлення з урахуванням ймовірностей $d_i^{(r)}$ прийняття рішень r в стані i запишемо у вигляді

$$i \in S, v_i(t) = \sum_{r \in R_i} d_i^r (1 - H_i^{(r)}(t)) \frac{k_i^{(r)}}{\alpha} (1 - e^{-\alpha t}) + \sum_{j \in S} \sum_{r \in R_i} \int_0^t d_i^r \left(\frac{k_i^{(r)}}{\alpha} (1 - e^{-\alpha t}) + e^{-\alpha t} v_j(t - \tau) \right) dQ_{ij}^{(r)}(\tau), \quad (10)$$

де $k_i^{(r)}$ - витрата системи за одиницю часу перебування в стані i при вирішенні $r \in R_i$; $v_j(t)$ - сумарні середні витрати з урахуванням переоцінки (2), за умови, що процес починається при $t = 0$ зі стану j .

Величини $v_i(\alpha, \beta)$ з виразу (4) можна записати у вигляді $v_i(\alpha)$, і для цього рівняння скористатися основними положеннями рівняння (інтеграла) Лапласа-Стілтєса. Відповідно роботам [3, 8] для будь-якої функції $F(t)$, похідна $F'(t)$ якої є функцією-оригіналом, що задовольняє нерівності $F'(t) < Ce^{\alpha t}$ для всіх $t < 0$, при всіх комплексних s , коли $\text{Re } s > \alpha$ існує функція

$$F^*(s) = L_s^* \langle F(t) \rangle = \int_0^\infty e^{-st} dF(t), \quad (11)$$

тобто функція e^{-st} при $\text{Re } s > \alpha$ інтегрована по функції $F(t)$. Функцію $F^*(s)$ називають перетворення Лапласа-Стілтґєса функції $F(t)$.

З виразів (8) і (9) випливає, що $H_i^{(r)}(\infty) = 1, i \in S, r \in R_i$, тому перша сума в виразі 10 при $t \rightarrow \infty$ зводиться в нуль. Інтегруючи по частинах вираз (11) для $L_s^* \langle F(t) \rangle$, отримуємо

$$sL_s^* \langle F(t) \rangle = L_s^* \langle F(t) \rangle - F(0), \quad (12)$$

де $F(s) = L_s \langle F(t) \rangle = \int_0^\infty e^{-st} F(t) dt$ є перетворення Лапласа функції $F(t)$. Із (12) при $s \neq 0$ знаходимо

$$L_s \langle F(t) \rangle = \frac{1}{s} (L_s^* \langle F(t) \rangle - F(0)). \quad (13)$$

Інтегруємо по частинам з урахуванням виразу (9) та знаходимо

$$\begin{aligned} & \sum_j \int_0^t (1 - e^{-\alpha t}) dQ_{ij}^{(r)}(\tau) = \\ & = (1 - e^{-\alpha t}) \sum_j dQ_{ij}^{(r)}(\tau) \Big|_0^t - \sum_j \alpha \int_0^t e^{-\alpha t} H_i(\tau) d\tau. \end{aligned} \quad (14)$$

Проводячи перетворення, переходячи у виразі (14) до межі $t \rightarrow \infty$ та застосовуючи (13) для $s = \alpha, (\alpha > 0)$, з урахуванням співвідношень (7) і (8):

$$\begin{aligned} \sum_j \int_0^t (1 - e^{-\alpha t}) dQ_{ij}^{(r)}(\tau) &= (1 - \alpha) L_{s=\alpha} \langle H_i^{(r)}(\tau) \rangle = \\ &= 1 - L_{s=\alpha}^* \langle H_i^{(r)}(\tau) \rangle = 1 - h_i^{(r)}(\alpha), \end{aligned} \quad (15)$$

де $h_i^{(r)}(\alpha) = L_{s=\alpha}^* \langle H_i^{(r)}(t) \rangle$. Застосовуючи до функції

Фі $\Phi_i^{(r)}(t) = \int_0^t e^{-\alpha t} v_j(t - \tau) dQ_{ij}^{(r)}(\tau)$ теорему про

граничний перехід в інтегралі по параметру, від якого залежать границі інтегрування та підінтегральна функція [8], при $t \rightarrow \infty$ отримуємо

$$\Phi_i^{(r)}(\infty) = \int_0^\infty e^{-\alpha t} v_j(\alpha) dQ_{ij}^{(r)}(\tau) = v_j(\alpha) q_{ij}^{(r)}(\alpha), \quad (16)$$

де $q_{ij}^{(r)}(\alpha) = L_{s=\alpha}^* \langle Q_{ij}^{(r)}(\alpha) \rangle$.

Переходячи у виразі (10) до границі при $t \rightarrow \infty$, з урахуванням (15) и (16) отримуємо:

$$v_i(t) = \sum_{r \in R_i} d_i^{(r)} \left(\zeta_i^{(r)}(\alpha) \right) + \sum_{j \in S} q_{ij}^{(r)}(\alpha) v_j(\alpha), \quad (17)$$

$$\zeta_i^{(r)}(\alpha) = \left(k_i^{(r)} / \alpha \right) \cdot \left(1 - h_i^{(r)}(\alpha) \right). \quad (18)$$

Нехай $\zeta_i(\alpha) = \sum_{r \in R_i} d_i^r \left(\rho_i^{(r)}(\alpha) \right)$ і

$\mathfrak{Z}(\alpha) = \left(\zeta_0(\alpha), \dots, \zeta_N(\alpha) \right)^T, \wp(\alpha) = \left(v_0(\alpha), \dots, v_N(\alpha) \right)^T$ (T – символ транспонування матриці). Тоді

$$\wp(\alpha) = \mathfrak{Z}_0(\alpha) + q(\alpha) \wp(\alpha), \quad (19)$$

де $q(\alpha) = \{q_{ij}(\alpha)\}, q_{ij}(\alpha) = \sum_{r \in R_i} d_i^r \left(q_{ij}^{(r)}(\alpha) \right)$.

Із виразу (19) знайдемо

$$\wp(\alpha) = \{I - q(\alpha)\}^{-1} \mathfrak{Z}_0(\alpha). \quad (20)$$

Даний вираз правильний, тому що при $\alpha > 0$ матриця $\{I - q(\alpha)\}$ – невинроджена, I – одинична матриця розміру $(N \times 1) \times (N \times 1)$.

Помноживши дві частини рівняння (19) зліва на вектор y , отримаємо таке:

$$y v(\alpha) = \sum_{i \in S} \sum_{j \in \tilde{S}} \sum_{r \in R_i} y_i \mu_{ij}(\alpha) \zeta_j^{(r)}(\alpha) d_i^{(r)},$$

$$\{I - q(\alpha)\}^{-1} = \{\mu_{ij}(\alpha)\}. \quad (21)$$

Величини $\mu_{ij}(\alpha)$ залежать від $d_i^{(r)}, r \in R_i, i \in S$, так як елементи матриці $\{I - q(\alpha)\}$ можна виразити через $d_i^{(r)}, r \in R_i, i \in S$.

Нехай $\{d_i^{(r)}\} (r \in R_i)$ – нерандомізована марківська стаціонарна стратегія системи розробки ПЗ у стані j. $d_j^{(r)} \in \{0, 1\}, \sum_{j \in S} d_j^{(r)} = 1, i, x_{00} = 1,$

$x_{rj} = d_j^{(r)}, r \in R_i, j \in \tilde{S}$. Мінімізація витрат (вираз 21) призводить до наступної задачі оптимізації для мулевих змінних $X = \{x_{rj}\}, r \in R_i, j \in \tilde{S}$:

$$f(\alpha, X) = \sum_{i \in S} \sum_{j \in \tilde{S}} \sum_{r \in R_i} y_i \mu_{ij}(\alpha, X) \zeta_j^{(r)} x_{rj} \rightarrow \min, \quad (22)$$

$$\text{при} \quad \sum_{r \in R_i} x_{rj} = 1, j \in \tilde{S}, \quad (23)$$

$$\sum_{j \in \tilde{S}} c_{rj} x_{rj} \leq b_r, r \in R_i, j \in \tilde{S}, \quad (24)$$

$$x_{rj} \in \{0, 1\}, j \in \tilde{S}, r \in R_i. \quad (25)$$

Побудова оптимальної нерандомізованої марківської стаціонарної стратегії

Позначимо системи (24), (25) як систему С. Вона являється системою псевдобулевих нерівнос-

тей. Підключивши далі додаткову умову (23) позначимо систему як \tilde{C} , а допустимі рішення r -ої нерівності системи C як $X_r^{(k)} = \{x_{r1}^{(k)}, \dots, x_{rN}^{(k)}\}$, $k=1, \dots, k_r$.

Для побудови рішення системи \tilde{C} при відомих допустимих рішеннях кожної нерівності (24) застосуємо наступний підхід. Розв'язки системи \tilde{C} знаходяться у вигляді $Z = \{s_j\}$, $j=1, \dots, N$, де s_j – множина номерів r , для яких допустиме рівняння $x_{rj}=1$. Рішення знаходяться за m кроків, де m – число обмежень (24). В початковому стані кожна із множин $s_j^{(0)}$ вектора $Z^{(0)}$ включає всі можливі значення $r \in R_j$. На r -му кроці відбувається перетин вектору $Z^{(r-1)}$ з одним із рішень r -ої нерівності. Допускаючи, що r -ій нерівності відповідає $r=r_1$, а також, що α_j являється j -м елементом допустимого рішення даної нерівності, $\alpha_j \in \{0, 1, \phi\}$, де ϕ – невизначений параметр з множини $\{0, 1\}$, можна сформулювати такі правила для r -го кроку алгоритму побудови рішень системи \tilde{C} :

1. Якщо α_j не фіксоване, то $s_j^{(r)} = s_j^{(r-1)}$.
2. Якщо $\alpha_j = 1$, то при $r_1 \in s^{(r-1)}$ допускаємо $s^{(r)} = \{r_1\}$, а при $r_1 \notin s^{(r-1)}$ допускаємо $s^{(r)} = \emptyset$.
3. Якщо $\alpha_j = 0$ то $s^{(r)} = s^{(r-1)} / \{r_1\}$. При цьому пересічення сімейств рішень здійснюється з врахуванням додаткових обмежень (23).

На m -му кроці $Z^{(m)} = \{\alpha_1^{(m)}, \dots, \alpha_N^{(m)}\}$, кожна компонента $\alpha_j^{(m)}$ якого є одноелементною множиною $\{r\}$, $r \in R$, $R = \{1, \dots, m\}$ і отже, $Z^{(m)}$ є рішення системи \tilde{C} . Виняток становлять випадки, коли \tilde{C} є певним набором чисел r з множини R . В цьому випадку за допомогою поєднання елементів багатозначних компонент з вектора $Z^{(m)}$ можна отримати кілька рішень системи \tilde{C} . Потім в результаті знаходимо сукупність всіх рішень системи \tilde{C} , з яких вибираємо оптимальне рішення, що доставляє мінімум цільової функції $f(\alpha, X)$. Це рішення може знаходитися різними відомими методами лінійного програмування, або просто шляхом безпосереднього порівняння значень $f(\alpha, X)$ при визначенні X системи \tilde{C} .

Чисельна реалізація викладеного методу управління ризиками безпеки індустрії програмного забезпечення для напівмарківської моделі прийняття рішень при аномальних ситуаціях безпеки представлена нижче.

Рекомендації по використанню удосконаленого методу

Розглянемо дві можливі ситуації помилок безпеки ПЗ: ситуація 1 - наявність функціональних помилок ПЗ (функціональні можливості ПЗ, не описані або не відповідають описаним у документації, при використанні яких можливе порушення конфіденційності, доступності або цілісності оброблюваної інформації); ситуація 2 - наявність не функціональних помилок ПЗ (надлишкові ФО і ІВ, переповнення буфера, витоку пам'яті, помилки типів даних, помилки покажчиків та ін.). Альтернативні заходи (спрощено) у разі першої аномальної ситуації такі:

- відгук ПЗ (для усунення помилок);
- декомпозиція програми і оцінка метрик складності;
- сигнатурний аналіз.

У разі наявності не функціональних помилок ПЗ альтернативні заходи наступні:

- відгук ПЗ (для усунення помилок);
- аналіз покажчиків, залежностей за даними та інтерв'яльний аналіз.

Припустимо, що в обох випадках перший захід (відгук ПЗ) здійснюється за один і той же час $T = 1$ у.е. Тому, з огляду на низьку ціну відповідних функцій тестування в порівнянні зі збитками просте ПЗ, можна вважати перші заходи в обох випадках однаковими за витратами.

Уявимо заходи в разі першої аномальної ситуації в вигляді $R = \{r_1; r_2; r_3\}$, а заходи для усунення другої аномальної ситуації в вигляді $R = \{r_1; r_4\}$. Тоді число елементів системи складе $N = 2$, а загальне число різних заходів для їх усунення $m = 4$.

Так само представимо описані аномальні ситуації j по відношенню з окремими модулями і блоками ПЗ як послідовні з'єднання (по надійності). Помилки (відмови) і їх усунення не впливають на надійність інших модулів і елементів ПЗ. Час усунення помилки не залежить від того, виявляються чи інші помилки в залишкових модулях. Іншими словами, кожна помилка виявляється і усувається незалежно одна від одної.

Позначимо через $F_j(t)$ функцію розподілу часу експлуатації системи між двома послідовними аномальними ситуаціями типу j , а через $G_j^{(k)}(t)$ – функцію розподілу часу усунення помилки після аномальної ситуації типу j при прийнятті рішення r . Будемо вважати, що $F_j(t)$ і $G_j^{(k)}(t)$ підкоряються експоненціальним законами з функціями інтенсивності відповідно λ_j і $\mu_j^{(k)}$:

$$F_j(t) = 1 - e^{-\lambda_j t}, \quad G_j^{(k)}(t) = 1 - e^{-\mu_j^{(k)} t}, \quad (26)$$

де $\lambda_j = 1/T_{j1}$; $\mu_j^{(r)} = 1/T_{j2}$; T_{j1} – середній час експлуатаційного етапу життєвого циклу розробки ПЗ

між двома аномальними ситуаціями типу j ; T_{j2} – середній час відновлюваного етапу життєвого циклу розробки ПЗ після виявлення помилки типу j при рішенні r . Нехай $d_i^{(k)}$ – нерандомізована стаціонарна стратегія системи в стані i ($i \in S$) при рішенні k (ймовірність прийняття рішення r в стані i),

$$d_i^{(r)} \in \{0,1\}, \sum_{k \in K} d_i^{(k)} \in 1, i \in S. \quad (27)$$

Тоді закон розподілу експлуатації і закон розподілу усунення помилок в цілому запишемо як

$$F_j(t) = 1 - e^{-\lambda t}, G_j^{(r)}(t) = 1 - e^{-\mu t},$$

де
$$\lambda = \sum_{j=1}^N \lambda_j; \mu = \sum_{j=1}^N \sum_{r=1}^N d_j^{(r)} \mu_j^{(r)}.$$

Нехай

$$\begin{aligned} T_{11} = 8 \text{ у.е.}, T_{21} = 8 \text{ у.е.}, T_{12}^{(1)} = 1 \text{ у.е.}, T_{12}^{(2)} = 2 \text{ у.е.}, \\ T_{12}^{(3)} = 1 \text{ у.е.}, T_{22}^{(1)} = 1 \text{ у.е.}, T_{22}^{(4)} = 0,5 \text{ у.е.} \end{aligned} \quad (28)$$

Припустимо, що $i = 0$ фіксує нормальний стан, $i = 1$ аномальний стан (ситуація 1), $i = 2$ аномальний стан (ситуація 2); $S = \{0,1,2\}$, $\tilde{S} = \{1,2\}$. У відповідності з (28) і враховуючи (25)-(27) отримаємо:

$$\begin{aligned} F_0(t) = F(t) = 1 - e^{-0,25t}, F_j(t) = 1 - e^{-0,125t}, \\ j = 1,2; G_1^{(1)}(t) = 1 - e^{-t}, G_1^{(2)}(t) = 1 - e^{-0,5t}, \\ G_1^{(3)}(t) = 1 - e^{-t}, G_2^{(1)}(t) = 1 - e^{-t}, G_2^{(4)}(t) = 1 - e^{-2t}. \end{aligned}$$

Нехай витрати від невиконання своїх функцій програмної системи на протязі часу $T = 1ч$ складуть 70000 у.о. Позначимо через $c_{k,j}$ витрати на захід k в разі аномальної ситуації j . Тоді, нехтуючи витратами на тестування в заході $r = 1$, отримуємо $c_{1,1} = c_{1,2} = c_1 = 70000$ у.о. Нехай далі $c_{2,1} = 300, c_{3,1} = 400, c_{4,2} = 600$ у.о. Будемо вважати, що в стані $i = 0$ прийнято єдине рішення ($r = 0$) - продовжити нормальне функціонування, і в цьому стані задано наступний розподіл ймовірностей:

$$p_{00}^{(0)} = 0,7, p_{01}^{(0)} = 0,1, p_{02}^{(0)} = 0,2. \quad (29)$$

Функції (1) і (9) запишемо як:

$$\begin{aligned} Q_{00}^{(0)}(t) = 0,7(1 - e^{-0,25t}), Q_{01}^{(0)}(t) = 0,1(1 - e^{-0,125t}); \\ Q_{02}^{(0)}(t) = 0,2(1 - e^{-0,125t}), Q_{10}^{(1)}(t) = (1 - e^{-t}), \\ Q_{10}^{(2)}(t) = (1 - e^{-0,5t}), Q_{10}^{(3)}(t) = (1 - e^{-t}), \\ Q_{11}^{(k)}(t) = Q_{12}^{(k)}(t) = 0, \quad (k = 1,2,3), \end{aligned}$$

$$\begin{aligned} Q_{20}^{(1)}(t) = (1 - e^{-t}), Q_{20}^{(4)}(t) = (1 - e^{-2t}); \\ H_0^{(0)}(t) = 0,7(1 - e^{-0,25t}) + 0,3(1 - e^{-0,125t}), \\ H_1^{(0)}(t) = (1 - e^{-0,25t}), H_1^{(2)}(t) = (1 - e^{-0,5t}). \\ H_2^{(1)}(t) = (1 - e^{-t}), H_2^{(4)}(t) = (1 - e^{-2t}), H_1^{(3)}(t) = (1 - e^{-t}). \end{aligned}$$

З урахуванням позначень $x_{00} = 1, x_{k,j} = d_j^{(k)}$, $r \in R_j, j \in \tilde{S}$ матриця $q(\alpha, x) = [q_{i,j}(\alpha, x)] (i, j \in S)$ з $q(\alpha, x) = \sum_{r \in R_j} x_{k,i} q_{i,j}^{(r)}(\alpha) (i, j \in S)$ є такою:

$$q(\alpha, x) = \begin{bmatrix} \frac{0,175}{\alpha + 0,25} & \frac{0,0125}{\alpha + 0,125} & \frac{0,025}{\alpha + 0,125} \\ \frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} & 0 & 0 \\ \frac{x_{12}}{\alpha + 1} + \frac{x_{42}}{\alpha + 2} & 0 & 0 \end{bmatrix}$$

Знайдемо визначник матриці $[I - q(\alpha, x)]$:

$$\begin{aligned} D(\alpha, x) = -\frac{0,025}{\alpha + 0,125} \left(\frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right) + 1 - \frac{0,175}{\alpha + 0,25} - \\ - \frac{0,0125}{\alpha + 0,125} \left(\frac{x_{11}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right). \end{aligned}$$

Матриця $[I - q(\alpha, x)]^{-1} = [\mu_{i,j}(\alpha, x)] (i, j \in S)$ має такі елементи:

$$\begin{aligned} \mu_{00}(\alpha, x) = 0, \mu_{01}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,0125}{\alpha + 0,125}, \\ \mu_{02}(\alpha, x) = 1/D(\alpha, x) \cdot (0,025/(\alpha + 0,125)), \\ \mu_{10}(\alpha, x) = \frac{1}{D(\alpha, x)} \left(\frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right), \\ \mu_{11}(\alpha, x) = 1/D(\alpha, x) \times \\ \times \left(1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0125}{\alpha + 0,125} \left(\frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right) \right), \\ \mu_{12}(\alpha, x) = 1/D(\alpha, x) \times \\ \times \frac{0,025}{\alpha + 0,125} \left(\frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right), \\ \mu_{20}(\alpha, x) = 1/D(\alpha, x) \cdot (x_{12}/(\alpha + 1) + 2x_{42}/(\alpha + 2)), \\ \mu_{21}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,0125}{\alpha + 0,125} \left(\frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right), \\ \mu_{22}(\alpha, x) = 1/D(\alpha, x) \times \\ \times \left(1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0125}{\alpha + 0,125} \left(\frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right) \right). \end{aligned}$$

У виразі (18) $k_i^{(r)}$ приймають такі значення:

$k_0^{(0)} = 70000 \text{ у.е.}$, $k_1^{(1)} = -70000 \text{ у.е.}$, $k_1^{(2)} = -150 \text{ у.е.}$,
 $k_1^{(3)} = -400 \text{ у.е.}$, $k_2^{(1)} = -70000 \text{ у.е.}$, $k_2^{(4)} = -12000 \text{ у.е.}$,
 а величини $\zeta_i^{(r)}$ записуються у вигляді:

$$\zeta_0^{(0)}(\alpha) = \frac{70000}{\alpha} \left(1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0375}{\alpha + 0,125} \right),$$

$$\zeta_1^{(1)}(\alpha) = -\frac{70000}{\alpha} \left(1 - \frac{1}{\alpha + 1} \right), \quad \zeta_1^{(2)}(\alpha) = -\frac{150}{\alpha} \left(1 - \frac{0,5}{\alpha + 0,5} \right),$$

$$\zeta_1^{(3)}(\alpha) = -\frac{400}{\alpha} \left(1 - \frac{1}{\alpha + 1} \right), \quad \zeta_2^{(1)}(\alpha) = -\frac{70000}{\alpha} \left(1 - \frac{1}{\alpha + 1} \right),$$

$$\zeta_2^{(4)}(\alpha) = -\frac{1200}{\alpha} \left(1 - \frac{2}{\alpha + 2} \right).$$

Знайдемо рішення системи \tilde{C} за допомогою алгоритму перегину рішень окремих нерівностей (24), вважаючи, що праві частини b_k цих нерівностей відповідають умовам:

$$c_{11} < b_1 < 2c_1, \quad b_2 > c_{21}, \quad b_3 > c_{31}, \quad b_4 > c_{42}. \quad (30)$$

З урахуванням умов (30) знаходимо наступні рішення окремих нерівностей системи \tilde{C} :

$$\eta_1 = 1:1 - (1,0); \quad 2 - (0,1), \quad 3 - (0,0);$$

$$\eta_1 = 2:1 - (1,0); \quad 2 - (0,0);$$

$$\eta_1 = 3:1 - (1,0); \quad 2 - (0,0); \quad \eta_1 = 4:1 - (0,1); \quad 2 - (0,0).$$

При наявності $Z^{(0)} = \{\{1,2,3\}, \{1,4\}\}$ на останньому кроці алгоритму отримуємо:

$$Z_{1221}^{(4)} = \{\{1\}, \{4\}\}, \quad Z_{1222}^{(4)} = \{\{1\}, \text{пуста множина}\},$$

$$Z_{2121}^{(4)} = \{\{2\}, \text{пуста множина}\}, \quad Z_{2122}^{(4)} = \{\{2\}, \{1\}\},$$

$$Z_{2211}^{(4)} = \{\{3\}, \text{пуста множина}\}, \quad Z_{2212}^{(4)} = \{\{3\}, \{1\}\},$$

$$Z_{3121}^{(4)} = \{\{2\}, \{4\}\}, \quad Z_{3122}^{(4)} = \{\{2\}, \text{пуста множина}\},$$

$$Z_{3211}^{(4)} = \{\{3\}, \{4\}\}, \quad Z_{3212}^{(4)} = \{\{3\}, \text{пуста множина}\}.$$

Таким чином, рішенням системи \tilde{C} є вектори:

$$1 - \{\{1\}, \{4\}\}, \quad 2 - \{\{2\}, \{1\}\}, \quad 3 - \{\{3\}, \{1\}\},$$

$$4 - \{\{2\}, \{4\}\}, \quad 5 - \{\{3\}, \{4\}\}.$$

Їм відповідають такі значення булевих змінних

$$\{x_{k,j}\}: \quad x_{1,1} = 1, \quad x_{4,2} = 1; \quad x_{2,1} = 1, \quad x_{1,2} = 1;$$

$$x_{2,1} = 1, \quad x_{4,2} = 1; \quad x_{3,1} = 1, \quad x_{1,2} = 1; \quad x_{3,1} = 1, \quad x_{4,2} = 1.$$

Не вказані змінні у кожному i -му рішенні дорівнюють нулю.

З урахуванням виразів (28) і (29) $f(\alpha, x)$ при $\alpha = 0,1$, що відповідає інфляції, що дорівнює 10%, і початковому розподіленні $y = (1,0,\dots,0)$ приймає значення: $f_1 = -51903$; $f_2 = -5329040$; $f_3 = -17161$; $f_4 = -26939$; $f_5 = -3325$.

Таким чином, при $\alpha = 0,1$ і $y = (1,0,\dots,0)$ оптимальною нерандомізованою марковською стаціонарною стратегією буде стратегія $x_{1,1} = 0, x_{2,1} = 0, x_{3,1} = 1, x_{1,2} = 0, x_{4,2} = 1$, що відповідає рішенню $\{\{3\}, \{4\}\}$ системи \tilde{C} .

Висновки

У розділі удосконалено метод управління ризиками розробки ПЗ. В основу даного методу було покладено напівмарківську модель прийняття рішень для керованого марківського процесу в безперервному часі. Відмінною особливістю запропонованого методу є використання псевдобулевих методів бівалентного програмування з нелінійною цільовою функцією і лінійними обмеженнями для визначення оптимальної стратегії усунення експлуатаційних помилок.

Проведені дослідження показали, що використані в даному розділі теоретичні положення в достатньому обсязі відображають стандарти та можливості сучасних методологій тестування ПЗ.

Слід зауважити, що представлений в розділі метод доцільно використовувати не тільки при управлінні ризиками безпеки ПЗ, але і при функціональному, навантажувальному, стресовому та інших видах тестування для запобігання можливих витрат.

Подальші дослідження направлені на розробку комплексу математичних моделей технології тестування WEB-застосунків, на розробку математичних моделей технології тестування DOM XSS уразливості і технології тестування уразливості до SQL ін'єкцій на основі підходу GERT-мережевого синтезу.

СПИСОК ЛІТЕРАТУРИ

1. Krishnan M. Soumya Software Development Risk Aspects and Success Frequency on Spiral and Agile Model. International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007). 2015. № 3 (1). P.301-310.
2. Zeng Y. Risk Management For Enterprise Resource Planning System Implementations in Project-Based Firms : dis. for the degree of PHD. Maryland, 2010. 210 p.
3. Бриткин А. И. Риски, связанные с внедрением технологий, в проектах разработки программного обеспечения. Социально-экономические и технические системы. 2007. № 8 (42). С. 156-168
4. Будников С.А. Полумарковская модель сложного конфликта радиоэлектронных систем. Методы и средства управления технологическими процессами: работы межд. Кон (Саранск, 19 – 21 ноября 2009 г.). Саранск, 2009. С. 201-223.
5. Коваленко А.В., Смирнов А.А., Коваленко А.С. Задачи распознавания ситуаций в ERP системах. Системы обработки информации. 2014. №4(120). С. 161-164
6. Коваленко А.В., Смирнов А.А. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения. Системы обработки информации. 2016. №5(142). С. 153-157
7. Смірнов О.А., Коваленко О.В., Мелешко Є.В. Інженерія програмного забезпечення. К.: ПВЛ КНТУ, 2013. 409 с.
8. Jacques J., Raimondo M. Semi-Markov risk models for finance, insurance and reliability Boston, Ma : Springer Science + Business Media LLC. 2007.

REFERENCES

1. Krishnan, M. Soumya (2015), "Software Development Risk Aspects and Success Frequency on Spiral and Agile Model", *International Journal of Innovative Research in Computer and Communication Engineering*, No. 3 (1), pp.301-310.
2. Zeng, Y. (2010), *Risk Management For Enterprise Resource Planning System Implementations in Project-Based Firms*, dis. for the degree of PHD, Maryland, 210 p.
3. Britkin, A.I. (2007), "Risks associated with the introduction of technologies in software development projects", *Social'no-jeconomicheskie i tehniczeskie sistemy*, No. 8 (42), pp. 156-168.
4. Budnikov, S.A. (2009), "Semi-Markov model of complex conflict of radio electronic systems", *Metody i sredstva upravlenija tehnologicheskimi processami*, Saransk, pp. 201-223.
5. Kovalenko, O.V., Smirnov, O.A. and Kovalenko, A.S. (2014), "Problems of recognition of situations in ERP systems", *Sistemi obrobki informacii*, No. 4(120), pp. 161-164/
6. Kovalenko, O.V. and Smirnov A.A. (2016), "Methods for qualitative analysis and quantification of software development risks", *Sistemi obrobki informacii*, No. 5(142), pp. 153-157.
7. Smirnov, O.A., Kovalenko, O.V. and Meleshko E.V. (2013), *Software Engineering: Training Guide*, KNTU, Kyiv, 409 p.
8. Jacques, J. and Raimondo M. (2007), "Semi-Markov risk models for finance, insurance and reliability", Springer Science + Business Media LLC, Boston.

Надійшла (received) 07.06.2018

Прийнята до друку (accepted for publication) 22.08.2018

**Усовершенствованный метод управления рисками программного обеспечения
на основе полумарковской модели принятия решений**

О. В. Коваленко

В работе проведены исследования, которые показали, что управление риском разработки программного обеспечения включает систему мероприятий, осуществляемых как до проявления негативного события, так и после его реализации. Однако превентивный анализ и учет большинства возможных эксплуатационных ошибок позволит снизить финансовые и другие затраты в жизненном цикле разработки программного обеспечения. Рассматриваемая задача управления рисками разработки программного обеспечения при определенных ограничениях на мероприятия по тестированию качества и безопасности, сформулирована в виде полумарковской модели принятия решений для управляемого марковского процесса в непрерывном времени и дисконтированными доходами или расходами. При этом данный вид эксплуатационных рисков отождествляются с последовательно соединенными независимыми элементами, восстанавливаемыми за конечное время. Оптимальная нерандомизированная стационарная стратегия управления определена с помощью псевдобулевых методов бивалентного программирования, находя все решения системы ограничений. Усовершенствован метод управления рисками разработки программного обеспечения на основе полумарковской модели принятия решений для управляемого марковского процесса в непрерывном времени. Отличительной особенностью предложенного метода является использование псевдобулевых методов бивалентного программирования с нелинейной целевой функцией и линейными ограничениями для определения оптимальной стратегии устранения эксплуатационных ошибок. Проведенные исследования в работе показали, что используемые теоретические положения в достаточном объеме отражают стандарты и возможности современных методологий тестирования программного обеспечения. В качестве примера рассмотрены ситуации возникновения ошибок безопасности ПО, и определена оптимальная стратегия управления для устранения указанной аномальной ситуации. Представленный в работе метод целесообразно использовать не только при управлении рисками безопасности программного обеспечения, но и при функциональном, нагрузочном, стрессовом и других видах тестирования для предотвращения возможных потерь.

Ключевые слова: управление рисками; полумарковская модель; принятия решений; псевдобулевы методы бивалентного программирования.

**Improved method of software risk management
based on the semi-Markov decision-making model**

O. Kovalenko

The studies carried out in this work showed that risk management of software development includes a system of activities carried out both before the appearance of a negative event, and after its occurrence. However, preventive analysis and consideration of the majority of possible operational errors will reduce financial and other costs in the life cycle of software development. The considered task of risk management of software development under certain restrictions on quality and safety testing activities is formulated as a semi-Markov decision-making model for a managed Markov process in continuous time and discounted income or expenses. In this case, this type of operational risks are identified with sequentially connected independent elements that are restored in a finite time. The optimal non-randomized stationary control management strategy is defined using pseudo-Boolean methods of bivalent programming, finding all solutions of the constraint system. Objective: to improve the method of risk management of software development on the basis of a semi-Markov decision-making model for a managed Markov process in continuous time. A distinctive feature of the proposed method is the use of pseudo-Boolean methods of bivalent programming with a nonlinear objective function and linear constraints to determine the optimal strategy for eliminating operational errors. The research conducted in the work showed that the applied theoretical positions sufficiently reflect the standards and capabilities of modern software testing methodologies. As an example, the situation of the occurrence of software security errors is considered, and an optimal control strategy for eliminating the indicated anomalous situation is determined. The method presented in the work can be used not only for managing software security risks, but also for functional, load, stress, and other types of testing to prevent possible losses.

Keywords: risk management; semi-Markov model; decision-making; pseudo-Boolean methods of bivalent programming.