

ЗАСТОСУВАННЯ ГЕНЕТИЧНОГО АЛГОРИТМУ ДЛЯ ОЦІНКИ РИЗИКІВ ПРОЕКТУ

Мельник А.М.¹⁾, Пик І.Т.²⁾

Тернопільський національний економічний університет

¹⁾ к.т.н., доцент; ²⁾ магістрант

I. Постановка проблеми

Проект, в основному, відбувається в умовах невизначеності і ризику, і це робить його необхідним для виявлення та ідентифікації ризиків, аналізу і їх оцінки, вибору методів управління, розробки і вживання заходів щодо скорочення цих ризиків, здійснення контролю і оцінки результатів цих заходів [1]. Процес оцінки ризиків ПЗ на етапі проектування є важливим і дещо складним та тривалим в часі, результат якого може істотно вплинути на вартість програми.

Тому розробка програмного засобу для моделювання оцінки ризиків із використанням генетичного алгоритму є надзвичайно актуальною задачею.

II. Мета роботи

У даній науковій роботі апробовано можливості використання генетичного алгоритму в задачах моделювання оцінки ризиків програмного забезпечення на етапі проектування.

Для формування числових значень ризиків використано їх символічне представлення для роботи генетичного алгоритму.

III. Алгоритм програми

Для функціонування програми, перш за все, необхідно вказати вхідні дані. У нашому випадку, відповідно до алгоритму функціонування програми (рис. 1), користувачу необхідно задати (етапи 14): інформацію про стан виконання проекту, залежність степені збитку від можливих ризиків, налаштування генетичних операторів, значення ймовірностей виникнення ризиків. На етапі налаштування генетичного алгоритму встановлюємо параметри функціонування генетичного алгоритму, а саме: методи схрещування, імовірність мутації, процент відсічення популяції.

На етапі 5 формуємо символічне представлення інформації про стан виконання проекту та граничних значень 5 типів ризиків. Символьна модель представлення ризиків – це кодування вхідних даних у двійковий код для подальшого формування фенотипу генів хромосоми.

Для формування початкової популяції (етап 6) вибрано стратегію «фокусування», що дає можливість формувати всі можливі комбінації вхідних даних, у випадку, коли кілька груп ризиків не буду змінюватися, тобто є константою. Далі, на цьому ж етапі, оцінюємо здоров'я популяції.

Значення фітнес-функції ранжуємо (сортуюмо) від меншого до більшого для подальшої селекції (етап 7). Селекцію (етап 8) здійснюємо на основі методу відсікання, а саме відсікаємо заданий відсоток хромосом із найбільшим значенням здоров'я.

Для схрещування (етап 9) вибираємо батьків випадковим чином із присутніх в утвореній популяції. Для схрещування передбачено два методи: одноточкове та двоточкове схрещування.

На етапі 10 виконуємо мутацію із заданою імовірністю, тобто якщо хромосомі слід мутувати, то ми її повністю замінюємо на іншу сформовану випадковим чином або ж випадковий ген.

Оцінка здоров'я популяції (етап 11) здійснюється на основі значення фітнес-функції (у нашому випадку – сума степенів збитків присутніх ризиків програмного забезпечення). Для оцінки здоров'я хромосома розбивається на гени, після чого значення генів, через їх символічне представлення, множаться на задані коефіцієнти впливу групи ризиків на степінь збитку. Кращим здоров'ям вважається найменше значення фітнес-функції, що означає мінімальні збитки. За результатами значення здоров'я зображуємо на графіку (етап 12) поточної епохи функціонування генетичного алгоритму мінімальне, максимальне та середнє його значення.

На етапі 13 перевіряємо чи досягнута задана кількість епох роботи генетичного алгоритму або ж у популяції залишилося 2 хромосоми. Якщо ні, то переходимо на етап 7, якщо так, то виводимо результати (етап 14) оцінки можливих ризиків та степені витрат, до якої вони можуть привести. Для завершення роботи ГА вибрано два варіанти тому, що в залежності від заданого користувачем відсотком відсічки, можливі варіанти, коли ГА завершить роботу на більш ранніх стадіях [5].

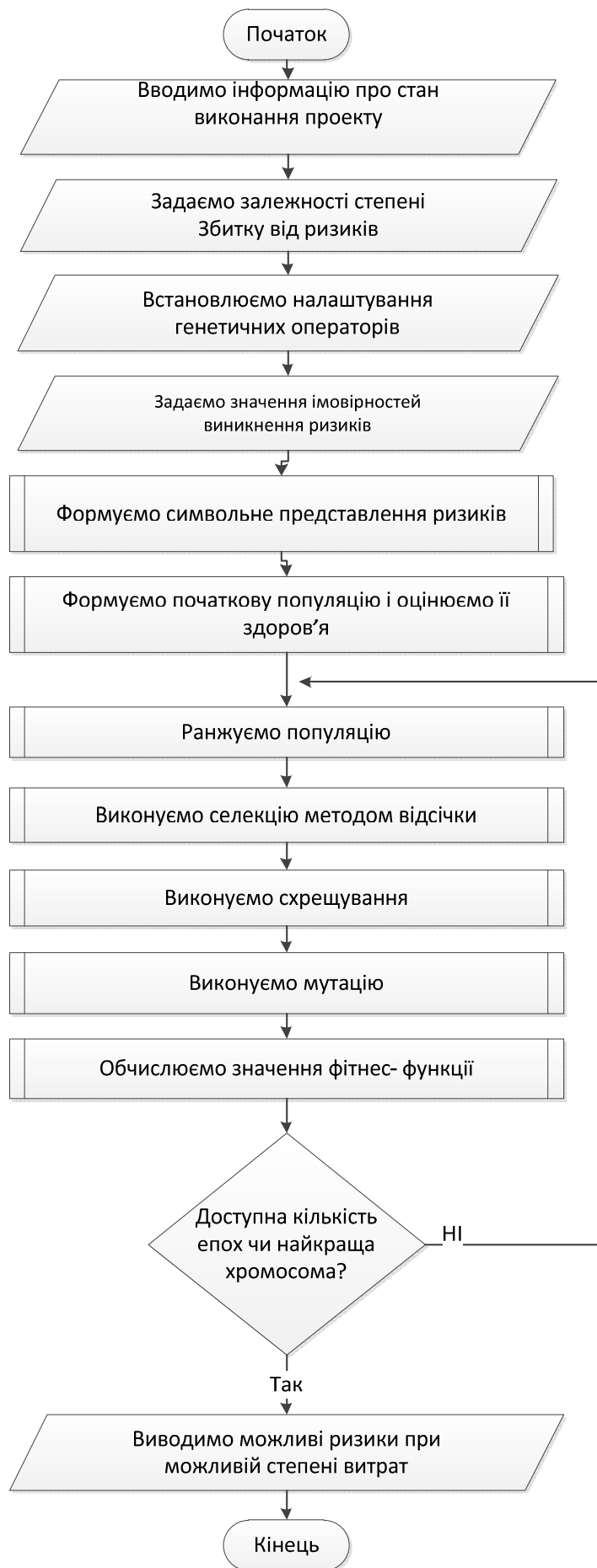


Рисунок 1 - Алгоритм функціонування програми

В роботі використовується символічне представлення, і враховано той факт, що хромосома складається із п'яти частин, тому операції схрещування та мутації працюють паралельно з кожною частиною окремо. В операторі мутації випадковим чином здійснюється заміна частини хромосоми (гена) на довільну іншу хромосому, згенеровану випадковим чином.

Висновок

Запропоновані генетичним алгоритмом оцінки є оптимальними при заданих вхідних значеннях, які супроводжуються експертними оцінками в залежності від ступеню витрат для конкретного типу ризиків.

У результаті моделювання відшуковується найкраща хромосома, за якою визначається ступінь збитку - 3,5%.

Розроблена програма, в залежності від налаштування роботи генетичного алгоритму, за короткий час відшуковує мінімальну ступінь збитків при визначених в процесі моделювання ризиках і може використовуватися як при розробці методичних рекомендацій, так при оцінці ризиків програмного забезпечення на його фірмах-виробниках.

Список використаних джерел

1. Вороновский Г.К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Вороновский Г.К. – Харьков: Основа, 1997. –
2. Гилл Ф. Практическая оптимизация / Гилл Ф., Мюррей У., Райт М. – М.: Мир, 2007. – 509 с.
3. Джеймс Мак-Кэффри. Анализ уязвимости и рисков проекта с использованием PERIL: [Электронный ресурс] / Джеймс Мак-Кэффри. – Режим доступа: <http://msdn.microsoft.com/ru-ru/magazine/dd315417.aspx>.
4. Иванов Д.Е. Ускорение работы генетических алгоритмов при построении тестов // Искусств. Интеллект / Иванов Д.Е., Скобцов Ю.А. 2001. – №1. – С. 52–60.
5. Турчин В.Ф. Феномен науки. Кибернетический подход к эволюции / Турчин В.Ф. – М.: ЭТС, 2000. – 368 с
6. Умрюхин Е.А. Механизмы мозга: информационная модель и оптимизация обучения / Умрюхин Е.А.. – М.: Мир, 1999. – 96 с.

УДК 004.55

ПОБУДОВА АДАПТИВНОЇ МОДЕЛІ КОРИСТУВАЧА В ІНТЕРФЕЙСАХ WEB-ОРІЄНТОВАНИХ СИСТЕМ

Мельник А.М.¹⁾, Співак І.Я.²⁾, Сирник О.Й.³⁾, Дробот І.М.⁴⁾

Тернопільський національний економічний університет

¹⁾ к.т.н., доцент; ²⁾ к.т.н., доцент; ³⁾ викладач; ⁴⁾ магістрант

І. Постановка проблеми

WEB-орієнтовані системи сьогодні стають великими та складними, що для їх розробки необхідна участь злагоджених команд розробників різних спеціальностей та кваліфікацій. Вкладені в їх розробку та витрачені засоби повинні окупатися, тому такі системи повинні існувати та застосовуватися протягом тривалого періоду, розвиваючись від версії до версії, переносячи на своєму життєвому шляху багато змін, покращуючи відомі та додаючи нові функції, коригуючи і усуваючи дефекти і помилки. Тривалий життєвий цикл припускає здатність WEB-орієнтованих систем адаптуватися не лише до зміни умов роботи в новому середовищі, але і до вимог користувача, які постійно змінюються.

II. Мета роботи

Метою роботи є підвищення якості інтерфейсу користувача WEB-орієнтованих систем на основі динамічної класифікації користувачів.

Для досягнення мети необхідно вирішити наступні завдання:

- проаналізувати відомі методи побудови та забезпечення якості інтерфейсів користувачів, розглянути особливості адаптивних інтерфейсів WEB-орієнтованих систем;
- виконати аналіз вимог до моделі користувача в адаптивних інтерфейсах програмного забезпечення;
- розробити модель користувача WEB-системи.