

МЕТОД ПРОГНОЗУВАННЯ УСПІШНОСТІ ПРОГРАМНИХ ПРОЕКТІВ НА ОСНОВІ АНАЛІЗУ ПРОГНОЗОВАНИХ ЗНАЧЕНЬ ХАРАКТЕРИСТИК ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Красій А.В., аспірант кафедри системного програмування Хмельницького національного університету, Хмельницький, Україна;

Говорущенко Т. О., канд. техн. наук, доцент, старший науковий співробітник, доцент кафедри системного програмування Хмельницького національного університету, Хмельницький, Україна.

Вирішено завдання розроблення методологічного забезпечення для прогнозування успішності програмних проектів. Об'єктом дослідження є процес прогнозування успішності програмних проектів на ранніх етапах життєвого циклу. Предметом дослідження є метод прогнозування успішності програмних проектів на основі аналізу прогнозованих значень характеристик програмного забезпечення. Мета роботи – підвищити достовірність вибору програмних проектів та успішність програмних проектів. Запропоновано інтелектуальний метод прогнозування успішності програмних проектів, який дозволяє обчислювати інтегративні показники успішності різних програмних проектів на основі прогнозованих (штучною нейронною мережею) значень характеристик розроблюваного за проектом програмного забезпечення, а також обчислювати ймовірності успішності цих програмних проектів. Метод відрізняється від відомих тим, що дозволяє прогнозувати успішність програмних проектів, порівнювати програмні проекти комплексно за основними характеристиками проекту і розроблюваного програмного забезпечення та прогнозованим значенням ймовірності успішності (а не тільки за вартістю та тривалістю, як відбувається наразі) та виконувати обґрунтований вибір програмного проекту замовником і розробником для подальшої реалізації. Проведені експерименти з дослідження роботоздатності методу. Результати експериментів дозволяють рекомендувати запропонований метод для використання на практиці.

Ключові слова: програмне забезпечення, програмний проект, специфікація вимог до програмного забезпечення, інтегративний показник проекту, успішність програмного проекту.

МЕТОД ПРОГНОЗИРОВАНИЯ УСПЕШНОСТИ ПРОГРАММНЫХ ПРОЕКТОВ НА ОСНОВЕ АНАЛИЗА ПРОГНОЗИРОВАННЫХ ЗНАЧЕНИЙ ХАРАКТЕРИСТИК ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Красий А.В., аспирант кафедры системного программирования Хмельницкого национального университета, Хмельницкий, Украина;

Говорущенко Т. А., канд. техн. наук, доцент, старший научный сотрудник, доцент кафедры системного программирования Хмельницкого национального университета, Хмельницкий, Украина.

Решена задача разработки методологического обеспечения для прогнозирования успешности программных проектов. Объектом исследования является процесс прогнозирования успешности программных проектов на ранних этапах жизненного цикла. Предметом исследования является метод прогнозирования успешности программных проектов на основе анализа прогнозируемых значений характеристик программного обеспечения. Цель работы – повысить достоверность выбора программных проектов и успешность программных проектов. Предложен интеллектуальный метод прогнозирования успешности программных проектов, который позволяет вычислять интегративные показатели успешности различных программных проектов на основе прогнозируемых (искусственной нейронной сетью) значений характеристик разрабатываемого по проекту программного обеспечения, а также вычислять вероятности успешности этих программных проектов. Метод отличается от известных тем, что позволяет прогнозировать успешность программных проектов, сравнивать программные проекты комплексно по основным характеристикам проекта и разрабатываемого программного обеспечения и по прогнозированным значениям вероятности успешности (а не только по стоимости и продолжительности, как происходит сейчас) и выполнять обоснованный выбор программного проекта заказчиком и разработчиком для дальнейшей реализации. Проведены эксперименты по исследованию работоспособности метода. Результаты экспериментов позволяют использовать предложенный метод на практике.

Ключевые слова: программное обеспечение, программный проект, спецификация требований к программному обеспечению, интегративный показатель проекта, успешность программного проекта.

THE METHOD OF PREDICTION OF SOFTWARE PROJECTS SUCCESS BASED ON THE ANALYSIS OF PREDICTED VALUES OF SOFTWARE CHARACTERISTICS

Krasiy A.V., post-graduate student of system programming department of Khmelnytsky National University, Khmelnytsky, Ukraine;

Hovorushchenko T.O., PhD, Associate Professor, Senior Researcher, Lecturer of system programming department of Khmelnytsky National University, Khmelnytsky, Ukraine.

The task of development of methodological support for predicting of the software projects success is solved. The object of research is the process of predicting of software projects

success at the early lifecycle stages. The subject of the research is a method of prediction of software projects success based on the analysis of predicted values of software characteristics. The purpose of the work is to increase the accuracy of selection of software projects and to increase the success of software projects. The intelligent method of prediction of software projects success is proposed. It provides to calculate the integrative indicators of success of various software projects based on the predicted (by artificial neural network) values of the characteristics of the software developed for the project. It also provides to calculate the probability of success of software projects. The method differs from the known methods, that provides to predict the success of software projects, compare software projects on the main characteristics of the project and the software and on the predicted value of the probability of success (and not just on the cost and duration, as is happening now), and perform the grounded choice of software project by customer and by developer for further realization. Experiments on investigations of the method operability were performed. The experimental results provide to recommend the proposed method for use in practice.

Keywords: software, software project, software requirements specification, integrative indicator of software project, success of software project.

Номенклатура

ПЗ	- програмне забезпечення;
ЖЦ ПЗ	- життєвий цикл програмного забезпечення
ПНМ	- штучна нейронна мережа;
C_p	- кросплатформність програмного забезпечення;
C_s	- вартість програмного забезпечення;
C_x	- складність програмного забезпечення;
D_{sp}	- тривалість програмного проекту;
E_{cef}	- економічна ефективність програмного забезпечення;
$I_{rip}Max_{best}$	- максимальне значення інтегративного показника успішності програмного проекту за характеристиками ПЗ, які вимагають максимізації;
$I_{rip}Max_{Sp}$	- інтегративний показник успішності програмного проекту за характеристиками ПЗ, які вимагають максимізації;
$I_{rip}Min_{bad}$	- максимальне значення інтегративного показника успішності програмного проекту за характеристиками ПЗ, які вимагають мінімізації;
$I_{rip}Min_{Sp}$	- інтегративний показник успішності програмного проекту за

	характеристиками ПЗ, які вимагають мінімізації;
P_{MaxSp}	- ймовірність успішності програмного проекту за інтегративним показником $Irip_{MaxSp}$ успішності програмного проекту;
P_{MinSp}	- ймовірність успішності програмного проекту за інтегративним показником $Irip_{MinSp}$ успішності програмного проекту;
P_{Sp}	- ймовірність успішності програмного проекту;
Qs	- якість програмного забезпечення;
$R1$	- множина показників розділу 1 специфікації вимог до ПЗ;
$R2$	- множина показників розділу 2 специфікації вимог до ПЗ;
$R3$	- множина показників розділу 3 специфікації вимог до ПЗ;
$R4$	- множина показників розділу 4 специфікації вимог до ПЗ;
Rs	- надійність програмного забезпечення;
SCH_{MaxSp}	- множина характеристик програмного проекту та розроблюваного ПЗ, збільшення значень яких викликають збільшення успішності програмного проекту;
SCH_{MinSp}	- множина характеристик програмного проекту та розроблюваного ПЗ, збільшення значень яких викликають зменшення успішності програмного проекту;
SCH_{Sp}	- множина основних характеристик програмного проекту;
$Slcm$	- тип моделі життєвого циклу програмного забезпечення;
Sp	- програмний проект;
SRS	- специфікація вимог до ПЗ;
Ub	- зручність використання програмного забезпечення.

Вступ

Програмний проект – це комплекс взаємозв’язаних заходів, спрямованих на досягнення поставлених задач з чітко визначеними цілями протягом заданого періоду часу та при встановленому бюджеті [1]. Наразі великі програмні проекти часто виконуються з відставанням від графіка або з перевищенням кошторису витрат, розроблений продукт часто не має необхідних функціональних можливостей [2].

Сьогодні, коли кількість високобюджетних програмних проектів стрімко зростає, дуже важливим і актуальним є вміння оцінити можливу успішність програмного проекту на ранніх етапах життєвого циклу, а також допомогти замовнику і розробнику обрати ймовірно успішний програмний проект з множини альтернативних програмних проектів.

Статистика успішності програмних проєктів за даними The Standish Group International (Chaos reports) [3] представлена у таблиці 1.

Таблиця 1 – Успішність програмних проєктів

Категорія проєкту	1994	2000	2006	2012
Успішні, %	16	28	35	39
Проблемні, %	53	49	46	43
Провальні, %	31	23	19	18

Аналіз даних таблиці 1 дав можливість побачити, що частка проблемних проєктів є досить сталою величиною і складає мінімум 43%. При цьому лише невелика кількість (максимум 39% за 1994-2012 роки) проєктів має необхідні функціональні можливості при існуючих обмеженнях за вартістю та терміном [3].

Програмні проєкти часто зазнають невдач через помилки на ранніх етапах життєвого циклу ПЗ, а саме [2]: 1) неадекватне формулювання вимог; 2) невдале проектування або неефективне планування; 3) невірне розуміння або недостатній аналіз специфікації; 4) нереалістичні проєктні плани; 5) невірно обрану модель життєвого циклу.

Помилки формулювання вимог та проектування складають 25-55% всіх помилок, причому чим більший обсяг ПЗ, тим більше помилок вноситься саме на ранніх етапах [3]. Слід врахувати й той факт, що вартість виправлення помилки проектування в два-чотири рази вища вартості виправлення помилки конструювання [3].

1. Постановка задачі

Нехай програмний проєкт задано специфікацією вимог до ПЗ [4] у наступному формалізованому вигляді [5]: $SRS = \langle R1, R2, R3, R4 \rangle$, де $R1$ - множина показників розділу 1 специфікації вимог до ПЗ, $R2$ - множина показників розділу 2, $R3$ - множина показників розділу 3, $R4$ - множина показників розділу 4.

Тоді задача оцінювання успішності програмного проєкту полягає у:

1) прогнозуванні (на основі множин показників специфікації $R1, R2, R3, R4$) характеристик програмного проєкту та розроблюваного за ним ПЗ – множини значень $SCH_{Sp} = \{Cs, Dsp, Ecef, Cp, Cx, Ub, Qs, Rs, Slcm\}$;

2) інтерпретації отриманих відносних значень характеристик проєкту та ПЗ - критеріями для такої інтерпретації є інтегративні показники успішності $Irup_{Min_{Sp}}$ (за

характеристиками ПЗ $SCH_{Min_{Sp}} = \{Cs, Dsp, Cx\}$, $SCH_{Min_{Sp}} \subset SCH_{Sp}$) та $Irup_{Max_{Sp}}$ (за характеристиками ПЗ $SCH_{Max_{Sp}} = \{Ecef, Ub, Cp, Qs, Rs\}$, $SCH_{Max_{Sp}} \subset SCH_{Sp}$);

3) оцінюванні ймовірності успішності програмного проекту P_{Sp} - критеріями є ймовірність успішності $P_{Min_{Sp}}$ за інтегративним показником $Irup_{Min_{Sp}}$ та ймовірність успішності $P_{Max_{Sp}}$ за інтегративним показником $Irup_{Max_{Sp}}$.

2. Огляд літератури

Аналіз основних характеристик ПЗ [6-8] зробив очевидним факт, що існуючі математичні інструменти та методи визначення основних характеристик ПЗ не придатні для оцінки їх значень на ранніх етапах життєвого циклу (на етапах формулювання вимог та проектування), оскільки вони орієнтовані на готовий програмний код, а не наявну на етапі проектування специфікацію вимог до програмного забезпечення. Саме аналіз специфікації вимог до ПЗ може надати різнопланову інформацію для подальшого розрахунку тривалості програмного проекту, вартості, кросплатформності, ефективності, складності, якості та надійності розроблюваного за проектом ПЗ, а також для вибору прийнятної моделі життєвого циклу ПЗ.

Аналіз специфікації вимог до ПЗ [6, 9, 10] показав, що вже на етапі проектування, коли розробник надав готову специфікацію, аналіз її вимог дозволяє сформулювати множини кількісних та якісних показників, на основі яких замовник і розробник можуть отримати прогнозовані кількісні значення характеристик розроблюваного за програмним проектом ПЗ, які дозволяють отримати прогноз успішності даного програмного проекту.

З проведеного аналізу [6, 11, 12] відомих автоматизованих засобів аналізу специфікацій та оцінювання характеристик програмного забезпечення зрозуміло, що всі вони призначені для роботи з вимогами та специфікаціями, причому деякі з них спрямовані на контроль за реалізацією вимог, але жоден з них не визначає прогнозованих значень характеристик ПЗ за специфікацією. Очевидно тоді, що існуючі засоби не прийнятні для кількісного оцінювання характеристик ПЗ на основі аналізу специфікацій.

3. Матеріали та методи

На характеристики тривалості програмного проекту, типу моделі життєвого циклу, вартості, економічної ефективності, складності, зручності використання, кросплатформності, якості та надійності ПЗ впливають певні показники специфікації [5],

але невідомі функції (формули, залежності), за якими можна обчислити значення тієї чи іншої характеристики ПЗ на основі множини впливових показників специфікації - всі наявні формули та методика оцінювання характеристик ПЗ орієнтовані на готовий програмний код, а не на специфікацію вимог.

Теорема Хехт-Нільсена [13] доводить можливість розв'язку задачі представлення багатовимірної функції довільного вигляду на ШНМ, тому для реалізації невідомих функцій залежності характеристик ПЗ від показників специфікації використовуватимемо саме ШНМ. Важкоформалізованою задачею прогнозування характеристик ПЗ є визначення ваг та взаємовпливу показників специфікації в межах кожної характеристики ПЗ. Ця задача може бути вирішена за допомогою використання навченої ШНМ.

У [14] розроблено ШНМ, яка опрацьовує показники специфікації, здійснює апроксимацію показників та надає прогнозовані кількісні оцінки характеристик ПЗ, на основі яких можна зробити прогноз успішності програмного проекту та здійснити вибір моделі життєвого циклу. Концепцію прогнозування основних характеристик ПЗ на основі аналізу специфікації вимог із використанням ШНМ представлено на рис.1.

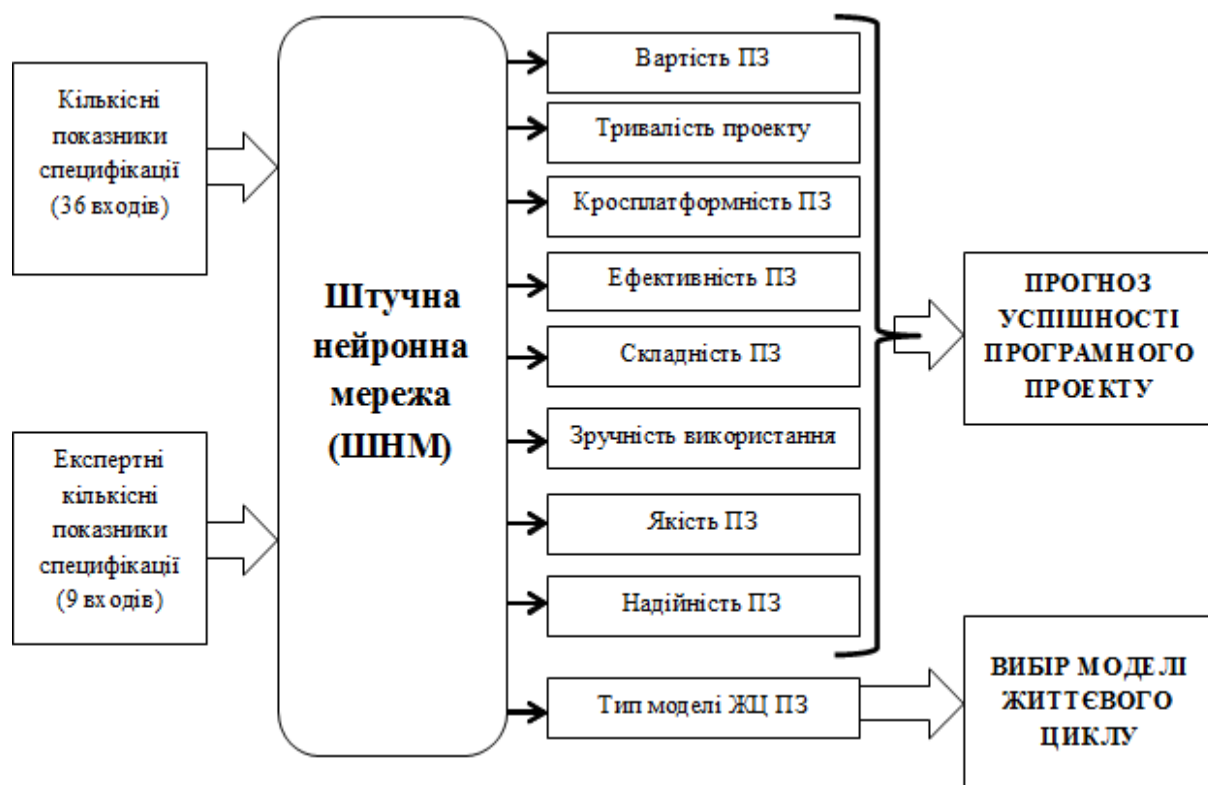


Рисунок 1 - Концепція прогнозування характеристик ПЗ на основі аналізу специфікацій із використанням нейромережних інформаційних технологій.

Отже, за рисунком 1, ШНМ прогнозування характеристик програмного забезпечення на основі аналізу специфікацій видає множину

$SCH_{Sp} = \{Cs, Dsp, Ecef, Cp, Cx, Ub, Qs, Rs, Slcm\}$ значень основних характеристик програмного проекту Sp та розроблюваного за ним ПЗ. ШНМ прогнозування характеристик ПЗ на основі аналізу специфікацій навчена так, що всі значення характеристик ПЗ (результуючі дані ШНМ) належатимуть інтервалу $(0;1]$. На основі отриманих з ШНМ значень основних характеристик ПЗ замовнику складно оцінити успішність проекту, оскільки складно вірно інтерпретувати одержані значення характеристик. Крім цього, успішність програмного проекту та розроблюваного за ним ПЗ слід оцінювати, приймаючи до уваги всі отримані значення характеристик в комплексі та враховуючи які значення характеристик викликають зменшення успішності програмного проекту, а які значення – її збільшення. Така задача є ще більш важкою та незрозумілою для замовника. Тому варто визначити метод, який допомагатиме замовнику оцінити успішність програмного проекту, інтерпретуючи та інтегруючи одержані значення характеристик ПЗ.

Збільшення значень деяких характеристик $SCH_{Min_{Sp}} = \{Cs, Dsp, Cx\}$ ($SCH_{Min_{Sp}} \subset SCH_{Sp}$) викликають зменшення успішності програмного проекту. Тоді значення характеристик множини $SCH_{Min_{Sp}}$, близькі до 0, будуть вказувати на високу успішність програмного проекту. Збільшення значень інших характеристик $SCH_{Max_{Sp}} = \{Ecef, Ub, Cp, Qs, Rs\}$ ($SCH_{Max_{Sp}} \subset SCH_{Sp}$) викликають збільшення успішності програмного проекту. Тоді значення характеристик множини $SCH_{Max_{Sp}}$, близькі до 1, вказуватимуть на високу успішність програмного проекту.

Визначення 1. Інтегративний показник успішності програмного проекту Sp – це кількісний показник успішності програмного проекту на основі множини значень прогнозованих характеристик $SCH_{Sp} = \{Cs, Dsp, Ecef, Cp, Cx, Ub, Qs, Rs, Slcm\}$.

Оскільки, як показано вище, значення різних характеристик по-різному впливають на успішність програмного проекту, то введемо два інтегративні показники успішності.

Визначення 2. Інтегративний показник успішності $I_{pup_{Min_{Sp}}}$ програмного проекту Sp – це кількісний показник успішності програмного проекту на основі множини значень прогнозованих характеристик $SCH_{Min_{Sp}} = \{Cs, Dsp, Cx\}$.

Визначення 3. Інтегративний показник успішності $Irip_{Max_{Sp}}$ програмного проекту Sp – це кількісний показник успішності програмного проекту на основі множини значень прогнозованих характеристик $SCH_{Max_{Sp}} = \{Ecef, Ub, Cp, Qs, Rs\}$.

Для отримання $Irip_{Min_{Sp}}$ проекту Sp створимо графік в системі координат, яка має три основих вісі (для трьох характеристик – $Cs \in (0;1]$, $Dsp \in (0;1]$, $Cx \in (0;1]$) – рис.2. Для отримання $Irip_{Max_{Sp}}$ проекту Sp створимо графік в системі координат, яка має п'ять основих осей (для п'яти характеристик – $Ecef \in (0;1]$, $Ub \in (0;1]$, $Cp \in (0;1]$, $Qs \in (0;1]$, $Rs \in (0;1]$) – рис.3.

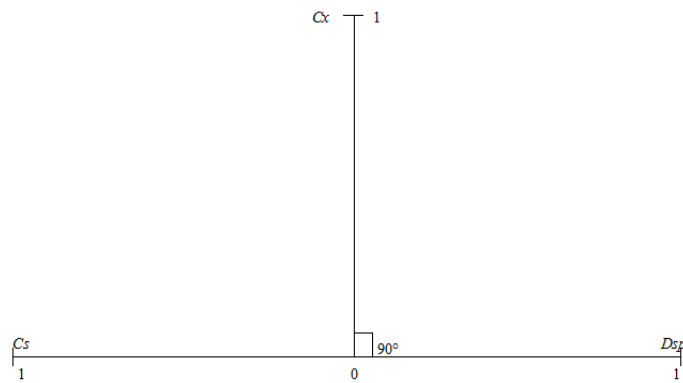


Рисунок 2 - Система координат для $Irip_{Min_{Sp}}$

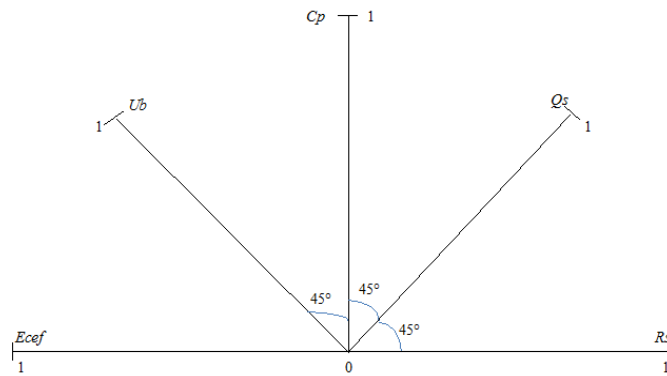


Рисунок 3 - Система координат для $Irip_{Max_{Sp}}$

Нехай ШНМ прогнозування характеристик ПЗ на основі аналізу специфікацій надала нам наступні значення характеристик для Sp : $Cs_{ANN} \in (0;1]$, $Dsp_{ANN} \in (0;1]$, $Cx_{ANN} \in (0;1]$; $Ecef_{ANN} \in (0;1]$, $Ub_{ANN} \in (0;1]$, $Cp_{ANN} \in (0;1]$, $Qs_{ANN} \in (0;1]$, $Rs_{ANN} \in (0;1]$. Відкладемо тоді отримані значення у відповідних системах координат і

отримаємо наступні графічні представлення $I_{rupMinSp}$ (рис.4) та $I_{rupMaxSp}$ (рис.5) проекту Sp .

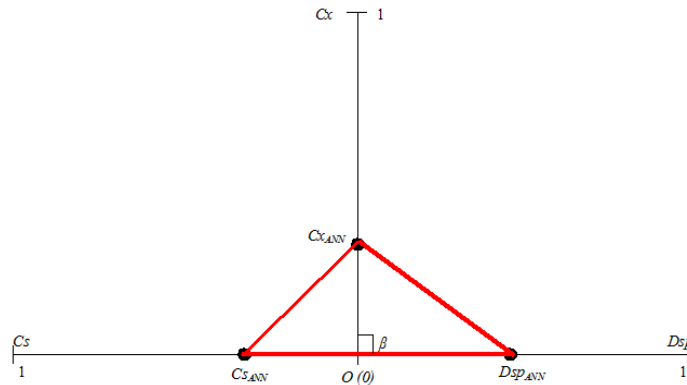


Рисунок 4 - Графічне представлення $I_{rupMinSp}$ програмного проекту Sp

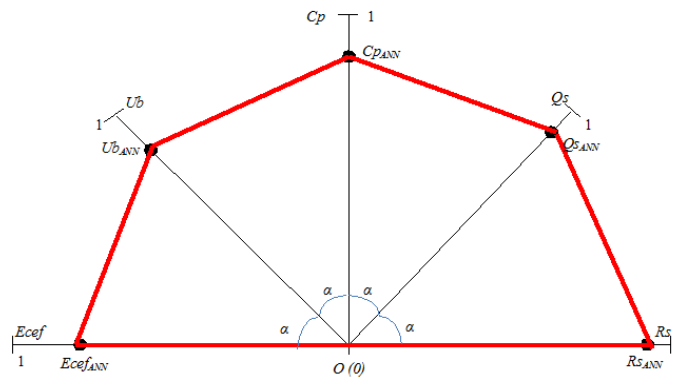


Рисунок 5 - Графічне представлення $I_{rupMaxSp}$ програмного проекту Sp

Інтегративним показником успішності $I_{rupMinSp}$ програмного проекту Sp є площа виділеного суцільною жирною лінією на рис.4 трикутника $Cs_{ANN}Dsp_{ANN}Cx_{ANN}$, а інтегративним показником успішності $I_{rupMaxSp}$ проекту Sp є площа виділеного суцільною жирною лінією на рис.5 п'ятикутника $Ecef_{ANN}Ub_{ANN}Cp_{ANN}Qs_{ANN}Rs_{ANN}$.

Тоді для визначення $I_{rupMinSp}$ проекту Sp (визначення площі трикутника $Cs_{ANN}Dsp_{ANN}Cx_{ANN}$) застосуємо формулу площі трикутника за відомою стороною та висотою:

$$I_{rupMinSp} = \frac{1}{2} \cdot (Cs_{ANN} + Dsp_{ANN}) \cdot Cx_{ANN}. \quad (1)$$

Визначальною характеристикою щодо прогнозування успішності програмного проекту (серед характеристик тривалості проекту, вартості та складності ПЗ) є характеристика складності, оскільки постійне зростання складності функцій ПЗ неминуче

призводить до збільшення трудомісткості створення програмних проектів, кількості помилок у програмному кодї, а також до зменшення успішності програмних проектів [3]. Тому система координат для $I_{rupMin_{Sp}}$ (рис.2) побудована таким чином, що при визначенні $I_{rupMin_{Sp}}$ як площі трикутника $Cs_{ANN}Dsp_{ANN}Cx_{ANN}$ саме характеристика складності є визначальною і не підлягає компенсації іншими характеристиками.

Для визначення $I_{rupMax_{Sp}}$ проекту Sp (площі п'ятикутника $Ecef_{ANN}Ub_{ANN}Cp_{ANN}Qs_{ANN}Rs_{ANN}$) розіб'ємо п'ятикутник на чотири трикутники - $Ecef_{ANN}OUb_{ANN}$, $Ub_{ANN}OCp_{ANN}$, $Cp_{ANN}OQs_{ANN}$, $Qs_{ANN}ORs_{ANN}$ і знайдемо площу для кожного з трикутників, застосовуючи формулу площі трикутника за відомими двома сторонами та кутом між ними:

$$\begin{aligned}
 I_{rupMax_{Sp}} &= S_{Ecef_{ANN}OUb_{ANN}} + S_{Ub_{ANN}OCp_{ANN}} + \\
 &+ S_{Cp_{ANN}OQs_{ANN}} + S_{Qs_{ANN}ORs_{ANN}} = \\
 &= \frac{1}{2} \cdot Ecef_{ANN} \cdot Ub_{ANN} \cdot \sin 45^\circ + \frac{1}{2} \cdot Ub_{ANN} \cdot Cp_{ANN} \cdot \sin 45^\circ + \\
 &+ \frac{1}{2} \cdot Cp_{ANN} \cdot Qs_{ANN} \cdot \sin 45^\circ + \frac{1}{2} \cdot Qs_{ANN} \cdot Rs_{ANN} \cdot \sin 45^\circ = \\
 &= 0.3536 \cdot (Ecef_{ANN} \cdot Ub_{ANN} + Ub_{ANN} \cdot Cp_{ANN} + \\
 &+ Cp_{ANN} \cdot Qs_{ANN} + Qs_{ANN} \cdot Rs_{ANN})
 \end{aligned} \tag{2}$$

Інтегративні показники успішності програмного проекту дають змогу визначити ймовірність його успішності.

Для визначення ймовірності успішності програмного проекту за інтегративним показником успішності $I_{rupMin_{Sp}}$ слід знайти максимальне значення інтегративного показника успішності $I_{rupMin_{bad}}$ (площу максимально можливого трикутника $CsDspCx$, окресленого пунктирною лінією на рис.6).

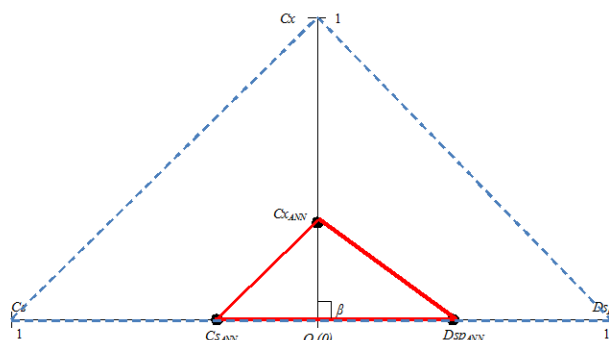


Рисунок 6 - Графічне представлення інтегративного показника успішності $I_{rupMin_{Sp}}$ та максимального значення інтегративного показника успішності $I_{rupMin_{bad}}$

Оскільки ШНМ прогнозування характеристик ПЗ на основі аналізу специфікацій навчена так, що максимальні значення характеристик вартості, тривалості та складності, отримані з неї, становлять 1, то:

$$I_{rupMin_{bad}} = \frac{1}{2} \cdot (Cs + Dsp) \cdot Cx = \frac{1}{2} \cdot (1 + 1) \cdot 1 = 1. \quad (3)$$

Для визначення ймовірності успішності програмного проекту за інтегративним показником успішності $I_{rupMax_{Sp}}$ слід знайти максимальне значення інтегративного показника успішності $I_{rupMax_{best}}$ (площу максимально можливого п'ятикутника $EcefUbCpQsRs$, окресленого пунктирною лінією на рис.7).

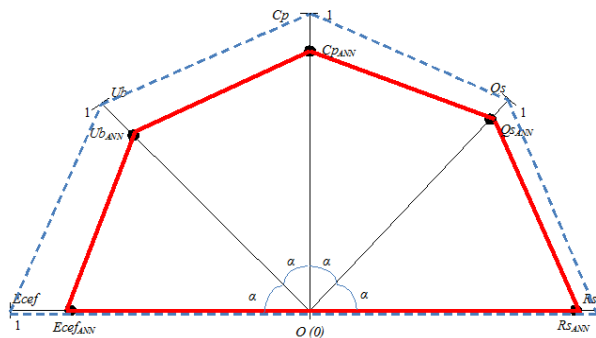


Рисунок 7 - Графічне представлення інтегративного показника успішності $I_{rupMax_{Sp}}$ та максимального значення інтегративного показника успішності $I_{rupMax_{best}}$

Оскільки ШНМ прогнозування характеристик ПЗ на основі аналізу специфікацій навчена так, що максимальні значення характеристик економічної ефективності, зручності використання, кросплатформності, якості та надійності, отримані з неї, становлять 1, то:

$$I_{rupMax_{best}} = 0.3536 \cdot (Ecef_{ANN} \cdot Ub_{ANN} + Ub_{ANN} \cdot Cp_{ANN} + Cp_{ANN} \cdot Qs_{ANN} + Qs_{ANN} \cdot Rs_{ANN}) = 0.3536 \cdot (1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1) = 1.4144 \quad (4)$$

Оскільки характеристики вартості, тривалості та складності вимагають мінімізації, то максимальне значення $I_{rupMin_{Sp}}$, яке складає $I_{rupMin_{bad}} = 1$ (згідно формули (3)), - це найгірше значення $I_{rupMin_{Sp}}$ програмного проекту.

Тоді за $I_{rupMin_{Sp}}$ ймовірність успішності $P_{Min_{Sp}}$ програмного проекту Sp складає:

$$P_{Min_{Sp}} = 1 - \frac{I_{rupMin_{Sp}}}{I_{rupMin_{bad}}} = 1 - \frac{I_{rupMin_{Sp}}}{1} = 1 - I_{rupMin_{Sp}}. \quad (5)$$

Характеристики економічної ефективності, зручності використання, кросплатформності, якості та надійності навпаки вимагають максимізації, тому максимальне значення $Irip_{Max_{Sp}}$, яке складає $Irip_{Max_{best}} = 1.4144$ (згідно формули (4)), - це найкраще значення $Irip_{Max_{Sp}}$ програмного проекту.

Тоді за $Irip_{Max_{Sp}}$ ймовірність успішності $P_{Max_{Sp}}$ програмного проекту Sp складає:

$$P_{Max_{Sp}} = \frac{Irip_{Max_{Sp}}}{Irip_{Max_{best}}} = \frac{Irip_{Max_{Sp}}}{1.4144} = 0.7070 \cdot Irip_{Max_{Sp}} \quad (6)$$

Оскільки $Irip_{Min_{Sp}}$ проекту Sp , на основі якого обчислюється $P_{Min_{Sp}}$ проекту, залежить від трьох основних характеристик ПЗ, а $Irip_{Max_{Sp}}$ проекту Sp , на основі якого обчислюється $P_{Max_{Sp}}$ проекту, залежить від п'яти основних характеристик ПЗ, тоді усереднене значення ймовірності успішності P_{Sp} програмного проекту Sp на основі двох отриманих значень $P_{Min_{Sp}}$ та $P_{Max_{Sp}}$ обчислюється за формулою:

$$P_{Sp} = \frac{3 \cdot P_{Min_{Sp}} + 5 \cdot P_{Max_{Sp}}}{8} \quad (7)$$

Формалізація даного методу прогнозування успішності програмного забезпечення на етапі проектування матиме наступний вигляд:

$$SCH_{Sp} = \{ SCH_{Min_{Sp}}, SCH_{Max_{Sp}} \} \Rightarrow \begin{cases} Irip_{Min_{Sp}} = f_1(SCH_{Min_{Sp}}) \\ Irip_{Max_{Sp}} = f_2(SCH_{Max_{Sp}}) \end{cases} \Rightarrow \begin{cases} P_{Min_{Sp}} = f_3(Irip_{Min_{Sp}}, Irip_{Min_{bad}}) \\ P_{Max_{Sp}} = f_4(Irip_{Max_{Sp}}, Irip_{Max_{best}}) \end{cases} \Rightarrow P_{Sp} = f_5(P_{Min_{Sp}}, P_{Max_{Sp}}) \quad (8)$$

де функція $f_1()$ обчислюється за формулою (1), функція $f_2()$ – за формулою (2), функція $f_3()$ – за формулою (5), функція $f_4()$ – за формулою (6), функція $f_5()$ – за формулою (7); показники $Irip_{Min_{bad}}$ та $Irip_{Max_{best}}$ обчислюються за формулами (3) та (4) відповідно.

4. Експерименти

Виконаємо експериментальне дослідження методу прогнозування успішності проектів. Для цього розглянемо чотири альтернативні програмні проекти, розроблені різними софтверними компаніями для вирішення однієї задачі. Нехай для розглянутих чотирьох альтернативних проектів було отримано наступні прогнозовані значення характеристик ПЗ – таблиця 2.

Таблиця 2 - Прогнозовані значення характеристик ПЗ для чотирьох альтернативних програмних проектів

Характеристика	Проект1	Проект2	Проект3	Проект4
1	2	3	4	5
C_s	0,2	0,78	0,61	0,41
D_{sp}	0,1	0,81	0,59	0,43
C_x	0,15	0,69	0,63	0,38
E_{cef}	0,8	0,13	0,52	0,52
U_b	0,85	0,15	0,5	0,56
1	2	3	4	5
C_p	0,87	0,21	0,47	0,57
Q_s	0,89	0,17	0,49	0,61
R_s	0,91	0,24	0,41	0,49

5. Результати

Згідно методу прогнозування успішності програмних проектів (формула (8)), обчислимо $I_{rupMinSp}$ і $I_{rupMaxSp}$ для чотирьох альтернативних програмних проектів, а також P_{MinSp} , P_{MaxSp} , P_{Sp} цих проектів. Результати обчислень представлені у таблиці 3.

Таблиця 3 - Інтегративні показники та ймовірності успішності чотирьох альтернативних програмних проектів

Величина успішності	Проект1	Проект2	Проект3	Проект4
$I_{rupMinSp}$	0,0225	0,5486	0,3780	0,1596
$I_{rupMaxSp}$	1,0621	0,0451	0,3275	0,4445
P_{MinSp}	0,9775 (97,75%)	0,4514 (45,14%)	0,6220 (62,20%)	0,8404 (84,04%)
P_{MaxSp}	0,7509 (75,09%)	0,0319 (3,19%)	0,2316 (23,16%)	0,3143 (31,43%)
P_{Sp}	0,8359 (83,59%)	0,1892 (18,92%)	0,3780 (37,80%)	0,5116 (51,16%)

6. Обговорення

Так, за результатами, наведеними у таблиці 3, проект1 має найбільшу ймовірність успішності (майже 84%), а проект2 – найменшу ймовірність успішності (близько 19%). Враховуючи отримані результати, замовнику для вирішення поставленої задачі

пропонується обрати проект1, оскільки саме він має найкращі характеристики та найвищу ймовірну успішність.

Наразі вибір програмного проекту як замовником, так і розробником здійснюється на основі лише прогнозованих у специфікації характеристик вартості, тривалості та власної інтуїції. Але не завжди розробнику вдається вірно спрогнозувати орієнтовану вартість та тривалість при розробленні специфікації вимог до ПЗ. Як було доведено у [6], вартість і тривалість слід оцінювати комплексно з врахуванням всіх впливових вимог (показників) специфікації.

Прогнозовані в специфікації значення вартості та тривалості чотирьох розглядуваних програмних проектів наведено у таблиці 4.

Таблиця 4 - Прогнозовані в специфікації значення вартості та тривалості для чотирьох альтернативних програмних проектів

Характеристика	Проект1	Проект2	Проект3	Проект4
Прогнозована вартість, вказана у специфікації	87000 грн	89000 грн	85000 грн	86500 грн
Прогнозована тривалість, вказана у специфікації	200 роб.днів	210 роб.днів	198 роб.днів	203 роб.днів

Характеристики програмних проектів з таблиці 4 свідчать, що для всіх чотирьох програмних проектів у специфікаціях зазначено приблизно однакові прогнозовані вартість та тривалість розроблення, але в таблиці 2 показано, що вони мають суттєво різні відносні оцінки всіх характеристик ПЗ, в тому числі вартості та тривалості, обчислені комплексно, з врахуванням всіх значущих показників специфікації. Так, відносна вартість коливається в межах від 0,2 (для проекту1) до 0,78 (для проекту2); відносна тривалість – в межах від 0,1 (для проекту1) до 0,81 (для проекту2). Отже, якщо оцінювати вартість та тривалість, враховуючи всі значущі показники специфікації, то очевидно, що їх значення не будуть однаковими для чотирьох розглядуваних програмних проектів. Тому на основі лише прогнозованих у специфікації вартості та часу розроблення як замовник, так і розробник може прийняти хибний висновок щодо вибору програмного проекту.

Крім цього, успішність програмних проектів залежить не лише від вартості та тривалості, але й від решти основних характеристик ПЗ – складності, ефективності, зручності використання, кросплатформності, якості та надійності, які в явному кількісному вигляді взагалі не зазначені у специфікації вимог до ПЗ. Тому саме оцінки всіх основних характеристик, надані ШНМ, та опрацьовані згідно запропонованого методу, допоможуть зробити вірний вибір і реалізувати проект, який має кращі

значення характеристик та найвищу ймовірність успішності (серед розглянутих чотирьох проектів – це проект1).

Отже, результати роботи запропонованого методу дають можливість замовнику обрати програмний проект з точки зору його основних характеристик та прогнозованої успішності.

Висновки

Вперше запропоновано метод прогнозування успішності програмних проектів на основі аналізу прогнозованих значень характеристик ПЗ, який базується на опрацюванні вихідних функціоналів нейромережної моделі процесу прогнозування характеристик ПЗ, які відповідають кількісним значенням основних характеристик ПЗ і дають можливість оцінити сумарний вплив показників специфікації на характеристики розроблюваного за специфікацією ПЗ. Опрацювання вихідних функціоналів полягає у обчисленні двох інтегративних показників успішності (за характеристиками ПЗ, які вимагають мінімізації, та за характеристиками ПЗ, які вимагають максимізації) для будь-якого описаного специфікацією програмного проекту, на основі яких обчислюються значення ймовірностей успішності програмного проекту за обома інтегративними показниками та усередненої ймовірності успішності програмного проекту.

Метод відрізняється від відомих тим, що дозволяє прогнозувати успішність програмних проектів та порівнювати програмні проекти комплексно - за основними характеристиками проекту і розроблюваного ПЗ та прогнозованими значеннями ймовірності успішності (а не тільки за вартістю та тривалістю, як відбувається наразі).

Практичне значення розробленого методу полягає у його використанні для виконання обґрунтованого вибору програмного проекту замовником і розробником для подальшої реалізації.

Перспективою для подальших досліджень авторів є підвищення достовірності роботи даного методу за рахунок підвищення достовірності роботи ШНМ прогнозування основних характеристик ПЗ на основі аналізу специфікацій, для чого необхідним є збір великої кількості показників та значень характеристик ПЗ зі специфікацій вже розроблених проектів для побудови навчальної вибірки ШНМ (для достовірної на 100% роботи ШНМ необхідний об'єм навчальної вибірки становить 20250 векторів).

Список літератури

1. Futrell R.T. Quality software project management / R.T.Futrell, D.F.Shafer, L.I.Shafer - New York: Prentice Hall PTR, 2003. – 1136 p.

2. Pomorova O. Intelligent assessment and prediction of software characteristics at the design stage / O.Pomorova, T.Hovorushchenko // American journal of software engineering and applications (AJSEA). – 2013. – Vol. 2. – P. 25-31.

3. CHAOS Manifesto: think big, act small, 2013 [Electronic resource] - Access mode: <http://www.versionone.com/assets/img/files/CHAOSManifesto2013.pdf>

4. IEEE 830-1998. Recommended practice for software requirements specifications - New York: IEEE, 1998 – 31 p.

5. Говорущенко Т.О. Математичне моделювання специфікації вимог та характеристик програмного забезпечення / Т.О.Говорущенко, А.В.Красій // Радіоелектронні і комп'ютерні системи. – 2014. – № 5. – С. 34-39.

6. Говорущенко Т.О. Визначення характеристик та вибір моделі життєвого циклу програмного забезпечення на основі аналізу специфікацій / Т.О.Говорущенко, А.В.Красій // Вісник Хмельницького національного університету. – 2013. – № 6. – С. 201-208.

7. A.Maedche. Software for people: fundamentals, trends and best practices (Management for professionals) / A.Maedche, A.Botzenhardt, L.Neer. – Berlin: Springer-Verlag Berlin Heidelberg, 2012. – 293 p.

8. C.Jones. The economics of software quality / C.Jones, O.Bonsignour. – Boston: Pearson Education, 2012. – 588 p.

9. K.Wiegers. Software requirements: 3-rd edition / K.Wiegers, J.Beatty. - Washington: MS Press, 2013. – 640 p.

10. K.Wiegers. More about software requirements: Thorny issues and practical advice / K.Wiegers. – Washington: MS Press, 2006. – 203 p.

11. A.Chen. Visual models for software requirements / A.Chen, J.Beatty. - Washington: MS Press, 2012. – 444 p.

12. Open Source Requirements Management Tool [Electronic resource] – Access mode: <http://sourceforge.net/projects/osrmt/>

13. Callan R. The essence of neural networks / R.Callan. – London: Prentice Hall Europe, 2003. – 288 p.

14. Красій А.В. Моделювання процесу прогнозування характеристик програмного забезпечення на основі аналізу специфікацій / А.В.Красій // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2014. – № 2. – С. 66-76.

Т р а н с л і т е р о в а н и й с п и с о к л і т е р а т у р и

1. Futrell R.T. Quality software project management / R.T.Futrell, D.F.Shafer, L.I.Shafer - New York: Prentice Hall PTR, 2003. – 1136 p.

2. Pomorova O. Intelligent assessment and prediction of software characteristics at the design stage / O.Pomorova, T.Hovorushchenko // American journal of software engineering and applications (AJSEA). – 2013. – Vol. 2. – P. 25-31.
3. CHAOS Manifesto: think big, act small, 2013 [Electronic resource] - Access mode: <http://www.versionone.com/assets/img/files/CHAOSManifesto2013.pdf>
4. IEEE 830-1998. Recommended practice for software requirements specifications - New York: IEEE, 1998 – 31 p.
5. Hovorushchenko T.O. Matematichne modeluvannya spetsifikacii vimog ta charakteristik programnogo zabezpechennya / T.O.Hovorushchenko, A.V.Krasiy // Radioelektronni i komputerni sistemi. – 2014. – № 5. – S. 34-39.
6. Hovorushchenko T.O. Vznachennya charakteristik ta vibir modeli jitteвого tsiklu programnogo zabezpechennya na osnovi analizu spetsifikatsiy / T.O.Hovorushchenko, A.V.Krasiy // Visnik Khmel'nitskogo natsionalnogo universitetu. – 2013. – № 6. – S. 201-208.
7. A.Maedche. Software for people: fundamentals, trends and best practices (Management for professionals) / A.Maedche, A.Botzenhardt, L.Neer. – Berlin: Springer-Verlag Berlin Heidelberg, 2012. – 293 p.
8. C.Jones. The economics of software quality / C.Jones, O.Bonsignour. – Boston: Pearson Education, 2012. – 588 p.
9. K.Wiegers. Software requirements: 3-rd edition / K.Wiegers, J.Beatty. - Washington: MS Press, 2013. – 640 p.
10. K.Wiegers. More about software requirements: Thorny issues and practical advice / K.Wiegers. – Washington: MS Press, 2006. – 203 p.
11. A.Chen. Visual models for software requirements / A.Chen, J.Beatty. - Washington: MS Press, 2012. – 444 p.
12. Open Source Requirements Management Tool [Electronic resource] – Access mode: <http://sourceforge.net/projects/osrmt/>
13. Callan R. The essence of neural networks / R.Callan. – London: Prentice Hall Europe, 2003. – 288 p.
14. A.V.Krasiy. Modeluvannya protsesu prognozuvannya charakteristik programnogo zabezpechennya na osnovi analizu spetsifikacij / A.V.Krasiy // Komputerno-integrovani tekhnologii: osvita, nauka, virobnictvo. – 2014. – №2. – S. 66-76.