

БАЙЄСІВСЬКІ МЕРЕЖІ ЯК ЗАСІБ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Запропонований підхід до побудови моделей якості програмного забезпечення за допомогою апарату байєсівських мереж. Розроблені моделі, призначені для оцінення рівня якості робочих продуктів та прогнозування якості кінцевого програмного продукту з метою ефективного керування програмними проектами. Встановлений метод композиції моделей якості проміжних робочих продуктів для їх використання при прогнозуванні якості кінцевого програмного продукту.

Вступ

Кожний програмний засіб (ПЗ) незалежно від вибраної моделі життєвого циклу (ЖЦ) програмного забезпечення проходить такі стадії розроблення: визначення вимог, проектування, кодування та тестування. При використанні *процесного підходу* до виконання робіт на цих стадіях базовий процес програмної інженерії організації-розробника програмних засобів (БПО) вибудовується згідно стандарту ДСТУ 3918 з множини *основних процесів*, безпосередньо пов'язаних з тих, що розробляються, а також із групи *процесів підтримки та організаційних процесів*.

У мінімальній конфігурації БПО основні процеси інтегруються з процесами верифікації, забезпечення гарантії (контролю) якості, керування конфігурацією та керування проектом. Така інтеграція уможлиблює впровадження відомого циклу PDCA ("Plan – Do – Check – Action", "План – Робота – Контроль – Регулювання") до кожного *результативного робочого продукту* (РП) основних процесів ЖЦ, а саме: *специфікації вимог, опису проекту, вихідного коду та виконуваних на комп'ютері працездатних версій ПЗ*.

Цикл PDCA для послідовно створюваних РП₁,...,РП_n та кінцевого програмного продукту (ПП) показаний на рис. 1.

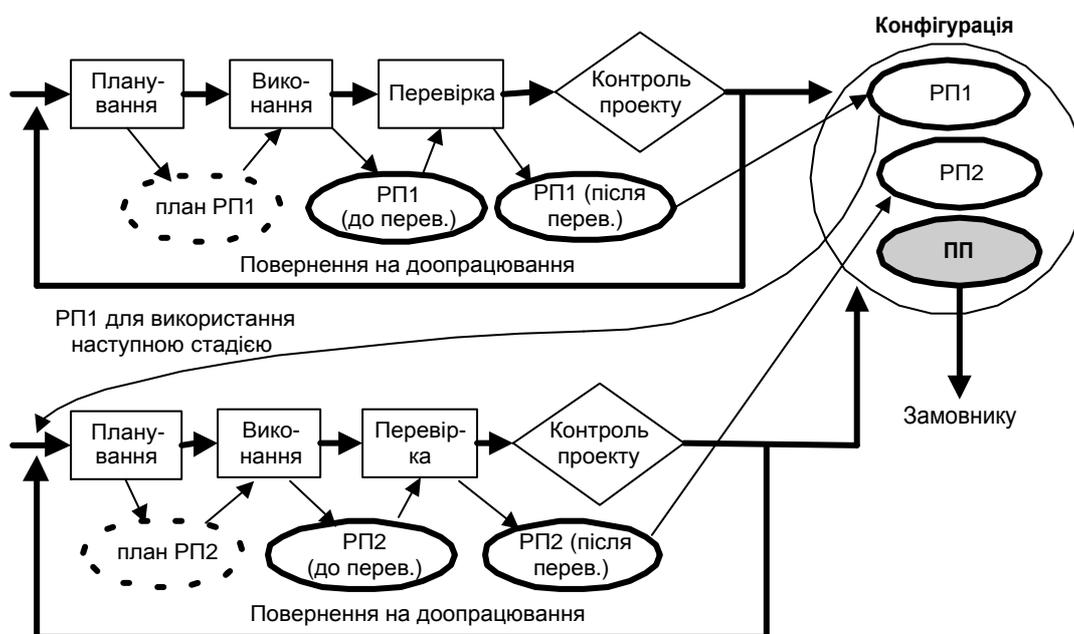


Рис. 1. Життєвий цикл робочих продуктів БПО

Для зручності викладення процесу верифікації та забезпечення гарантії якості умовно поєднані під спільною назвою – *процес перевірки*.

Робочий продукт $РП_i$ на стадії i обумовлюється:

- якістю вихідного РП для стадії - $РП_{i-1}$;
- якістю (зрілістю) основного процесу стадії $ПРО_i$;
- якістю (зрілістю) процесу перевірки $ПРn_i$ та
- адекватністю залучених ресурсів ($РeO_i$ та $РeП_i$, відповідно) до потреб процесів на стадії i .

Зазначені чинники утворюють лише найвищий рівень у розгалуженій ієрархії чинників якості ПЗ, які становлять ризики програмного проекту.

Для того щоб своєчасно виявляти та знищувати загрози якості розроблюваного ПЗ, необхідно не лише оцінювати *якість* кожного РП у контрольних точках проекту з метою визначення його придатності для застосування на наступних стадіях розроблення, але й виконувати *прогнозування якості* кінцевого ПП, спираючись на результати причинно-наслідкового аналізу всієї сукупності чинників якості робочих продуктів, процесів та ресурсів проекту.

В архітектурі БПО організацій, які не подолали бар'єру рівня зрілості СММ 2 (за моделлю СММ), процеси *вимірювання* та *оцінювання* відсутні. За цих обставин збір та (або) упорядкування даних про якість РП ПЗ залишаються не вирішуваною проблемою, а побудова складних прогнозних моделей, базованих на регресійному аналізі зібраних даних, практично неможлива. Існує великий рівень *невизначеності* щодо впливу одних чинників якості на інші та на якість кінцевого ПП.

З огляду на визначені умови виконання проектів ПЗ доцільним є залучення сучасних засобів інтелектуального аналізу даних до моделювання причинно-наслідкових зв'язків чинників якості, а саме: генетичних алгоритмів, нейронних або байєсівських мереж.

У даній статті запропонований підхід до побудови моделей якості ПЗ за

допомогою апарату *байєсівських мереж* (БМ). Розроблені моделі призначені для оцінки рівня якості робочих продуктів та прогнозу якості кінцевого ПП з метою ефективного керування проектами ПЗ. Розроблений метод композиції моделей якості РП для їх використання при прогнозуванні якості кінцевих програмних продуктів.

1. Актуальність проблеми оцінювання та прогнозування в сучасній парадигмі якості ПЗ

Сучасна програмна індустрія досягла такого рівня розвитку, на якому вимоги до забезпечення якості та (або) надійності стали обов'язковим пунктом укладених угод (контрактів) на розроблення ПЗ [1]. Це обумовлює необхідність застосування *систематизованих підходів* до прогнозування й оцінки якості розроблюваних ПЗ на всіх стадіях їх ЖЦ.

Процесний підхід до розроблення ПЗ фактично спричинив зміну *парадигми якості* [2]. Ключовими моментами сучасної парадигми є процесна програмна інженерія, інтеграція процесів ЖЦ, а також цілеорієнтована програма вимірювання та оцінювання артефактів проекту як ключова ланка циклу керування проектом і вдосконалення процесів ЖЦ.

З появою в Україні стандартів ДСТУ 3918 [3] та ДСТУ ISO/IEC TR 15504 [4], що регламентують застосування процесів ЖЦ ПЗ і узагальнюють погляди на оцінювання їх зрілості, поступово відбувається осмислення розроблення ПЗ як інженерної *діяльності*. В цих стандартах виділені 40 процесів ЖЦ ПЗ і поданий механізм інтеграції в БПО нових практичних прийомів програмної інженерії.

Прагнення організацій-розробників забезпечити конкуренто-спроможність своєї програмної продукції підштовхує їх до інституціалізації згаданих процесів у програмні проекти, їх оцінення, вдосконалення та отримання сертифікатів зрілості (або сертифікатів якості продукції). А це, в свою чергу, дає підстави говорити про те, що проблеми *інженерії якості* (зокрема, її прогнозу-

вання та оцінення) не просто є актуальними, але й отримують реальне підґрунтя для вирішення [5].

Прогнозування якості кінцевого ПП полягає у побудові проєкції показника якості, визначеного за результатами вимірювання та оцінювання зібраних і оброблених даних про *робочі продукти, процеси та ресурси* розроблення на певній стадії життєвого циклу (яка передує випуску ПЗ в експлуатацію), його значення на будь-якій наступній стадії та наприкінці розроблення ПЗ.

Прогнозування є ітеративним процесом застосування комбінації прогнозних моделей якості ПЗ із послідовним уточненням їх параметрів протягом розроблення програмного продукту.

2. Задачі прогнозування якості

В угодах, стосовних розроблення ПЗ, можуть встановлюватися цільові значення одного *інтегрального* показника якості або окремих характеристик якості [6]. Інтегральний показник якості поєднує значення декількох характеристик якості, наприклад функціональності, надійності та супроводжуваності. Навіть якщо умовами угод цільові значення показників якості не встановлені, надії замовників ПЗ покладаються на те, що розроблений програмний продукт відповідатиме відомим типовим стандартам якості для певного класу ПЗ. Тому першою задачею прогнозування якості є визначення множини (діапазону) можливих значень характеристик якості ПЗ різних класів, які б могли скласти *нормативну базу* для оцінення та прогнозування якості (деякі з них подані в [7]).

Другою задачею є встановлення причинно-наслідкових зв'язків у системі чинників якості в категоріях процесів, продуктів та ресурсів розроблення ПЗ [5,8].

Третя задача полягає у побудові моделей прогнозування якості (або окремих характеристик якості), які б, з одного боку, відповідали існуючому здебільшого невисокому рівню інженерії якості ПЗ (не вище СММ 2), а з іншого – могли

удосконалюватися у міру накопичення історичних даних про вплив чинників якості на значення її характеристик.

Четвертою задачею є побудова БПО з урахуванням обох випадків – відсутності процесів вимірювання, та оцінювання, їх наявності. Очевидно, що у першому випадку прогнозування якості повинне виконуватися в рамках процесу керування проєктом безпосередньо його керівником (менеджером). Частиною вирішення цієї задачі є створення інструментальних технологічних модулів прогнозування якості.

3. Причинно-наслідкові зв'язки в процесах розроблення ПЗ

На даному етапі виконання досліджень за проєктом ПС НАН України 1/02-02 “Розроблення концепції, методів та методичного апарату вдосконалення та стандартизації процесів життєвого циклу систем ПЗ” поставлені задачі прогнозування якості вирішуються стосовно найбільш критичної характеристики – надійності ПЗ [7, 9].

Загально визнаним “мірилом” якості ПЗ є внутрішня метрика надійності – *кількість виявлених дефектів* в робочих продуктах ПЗ та зовнішні метрики – очікувана кількість прередбачених *прихованих дефектів* у ПЗ, реальна кількість *виявлених дефектів* у ПЗ, кількість *усунутих (відкоригованих) дефектів*, а також *щільність дефектів*.

Доцільно нагадати ланку причинно-наслідкових зв'язків базових понять інженерії надійності [5, 8] – *вада* (процесу), *помилка* (виконавця), *дефект* (продукту), *відмова* (працюючого на комп'ютері ПЗ).

Дефект завжди є наслідком помилки виконавця процесу на будь-якому з етапів розроблення. Він може бути притаманний специфікації вимог, проєктним документам, тексту коду, експлуатаційній документації тощо. Вихідні результівні РП одних процесів, що містять не усунені при перевірці дефекти, служать вхідними РП для інших процесів, а дефекти в них – джерелом помилок виконавців цих про-

цесів. Крім того, помилки виконавців можуть бути наслідком *вад* у визначенні процесів (невірна послідовність дій, та вибраний інструмент тощо), що сприяє неправильній інтерпретації початкової інформації людиною і прийняттю невірних рішень, а також просто відображає недостатню професійну зрілість кадрів. Помилки виконавців, у свою чергу, призводять до дефекту в РП, і цикл “*помилка - дефект - помилка*” повторюється.

З цих позицій основні процеси ЖЦ ПЗ будемо асоціювати з процесами *внесення дефектів*, а процеси перевірки – з процесами *виявлення та усунення дефектів* (рис.2).

4. Байєсівські мережі та інструментальні засоби їх побудови

В основі теорії байєсівських мереж лежить правило поширення (множення) ймовірностей та теорема Байєса. Зробимо стислий огляд основних рівнянь, пов’язаних з визначенням байєсівської мережі.

Нехай A та B – випадкові події, причому ймовірність $P(B) > 0$. Тоді $\frac{P(AB)}{P(B)}$ – умовна ймовірність події A за умови події B , тобто $P(A|B)$. Таким чином,

$$P(A|B) = \frac{P(AB)}{P(B)} \quad (\text{формула умовної ймовірності}). \quad (1)$$

Помножимо на $P(B)$: $P(AB) = P(B) \cdot P(A|B)$.

За правилом симетрії відносно A та B

$$P(AB) = P(A) \cdot P(B|A) \quad (\text{правило множення ймовірностей}). \quad (2)$$

Тоді

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (\text{правило Байєса}). \quad (3)$$

Потужність правила Байєса полягає у тому, що в багатьох ситуаціях, коли треба обчислити $P(A|B)$, але це важко зробити безпосередньо, можна отримати інформацію про $P(B|A)$, тобто обчислити $P(A|B)$ в термінах $P(B|A)$.

Знаменник $P(B)$ у рівнянні можна визначити за формулою повної ймовірності:

$$P(B) = \sum_{k=1}^n P(BA_k) = \sum_{k=1}^n P(B|A_k) \cdot P(A_k),$$

де A_1, A_2, \dots, A_n – повна система гіпотетичних подій (гіпотез), а B – випадкова подія. Ймовірність події B є сума добутків ймовірностей кожної гіпотези A_k та умовних ймовірностей події B за цими гіпотезами.

Тоді правило Байєса можна переписати у іншому вигляді:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{\sum_{i=1}^n P(B|A_i) \cdot P(A_i)} \quad (4)$$

В узагальненому вигляді нехай H_1, H_2, \dots, H_n утворюють повну систему подій (гіпотез), а A – деяка подія, для якої $P(A) \neq 0$. Тоді для кожної пари (H_i, A) має місце *формула Байєса*:

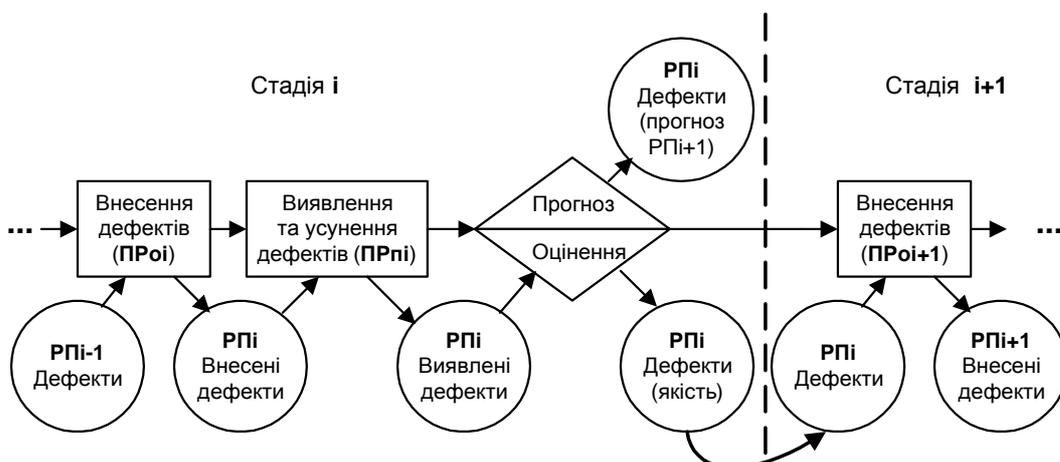


Рис. 2. Причинно-наслідковий зв’язок дефектів у робочих продуктах ПЗ

$$P(H_k | A) = \frac{P(H_k) \cdot P(A | H_k)}{\sum_{i=1}^n P(H_i) \cdot P(A | H_i)}$$

(теорема Байєса). (5)

Подане вище правило множення ймовірностей (2) можна переписати, наприклад, для трьох подій:

$$P(ABC) = P(A | BC) \cdot P(BC) = P(A | BC) \cdot P(B | C) \cdot P(C)$$

або у загальному вигляді (правило мережі):

$$P(A_1, A_2, \dots, A_n) = P(A_1/A_2, \dots, A_n) \cdot P(A_2/A_3, \dots, A_n) \cdot \dots \cdot P(A_{n-1}/A_n) \cdot P(A_n). \quad (6)$$

Ця формула особливо важлива для байєсівських мереж. Вона надає засіб обчислення сумісного розподілення ймовірностей. Оскільки в мережах багато випадкових величин (ВВ), A_i бувають умовно незалежні, тому формулу можна спростити.

Нехай $\{A_1, A_2, \dots, A_n\}$ – множина випадкових величин у вершинах мережі, а запис $Предки(A_i)$ позначає множину предків вершини A_i . Тоді сумісне розподілення ймовірностей для $\{A_1, A_2, \dots, A_n\}$

$$P(A_1, A_2, \dots, A_n) = \prod_{i=1}^n P(A_i | Предки(A_i)).$$

Байєсівська мережа є орієнтованим ациклічним графом, що покладається в основу математичних моделей систем з невизначеністю, між елементами яких існують причинно-наслідкові зв'язки [10].

З кожною вершиною апіорі пов'язуються параметри, відповідні закону розподілу ВВ. Дуга між будь-якими двома вершинами A та B встановлює причинно-наслідковий зв'язок “ A спричиняє B ”. У мережі справедливе відношення умовної незалежності вершин: вершина-нащадок B не залежить від вершин-предків, що знаходяться вище вершини батька (тобто A).

Якщо БМ містить дискретні ВВ, параметри кожної вершини можуть бути представлені *таблицею ймовірностей вершини* (ТВ).

Нехай, наприклад, БМ має три вершини (рис. 3).

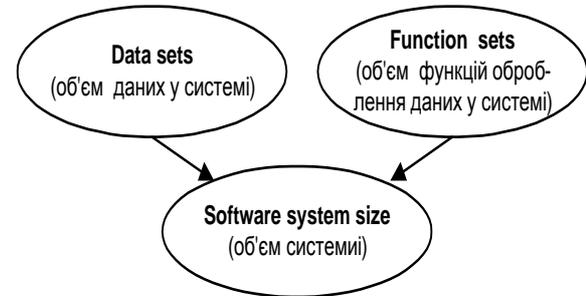


Рис. 3. Байєсівська мережа з трьома вершинами

Дискретні значення ВВ у вершинах визначені за порядковою шкалою: “High” (великий), “Medium” (середній), “Low” (малий) (табл. 1 – для вершин-батьків, табл. 2 – для вершини-сина).

Хоча байєсівські мережі відомі давно, широке практичне застосування вони почали знаходити лише нещодавно, з появою ефективних алгоритмів поширення ймовірності у мережі, а також інструментальних засобів побудови БМ [11].

Аналіз можливостей доступних Internet-версій інструментів *Hugin Lite 6.5* (корпорації HUGIN Expert), *Netica* (корпорації NORSYS Software) та *MSBNx* (від Microsoft Research) показав, що вони забезпечують підтримку оброблення як дискретних, так і неперервних ВВ у вершинах (за умови нормального (Гауса) розподілення для вершин-предків).

Приклад застосування MSBNx для побудови БМ поданий на рис. 4.

Таблиця 1. Ймовірності вершин-батьків

Data sets		Function sets	
Значення	Ймовірність	Значення	Ймовірність
High	0,3	High	0,3
Medium	0,5	Medium	0,3
Low	0,2	Low	0,4

Таблиця 2. Ймовірності вершини-сина

Вершини-батьки		Software system size		
Function sets	Data sets	High	Medium	Low
High	High	0,6	0,3	0,1
	Medium	0,5	0,3	0,2
	Low	0,4	0,4	0,2
Medium	High	0,4	0,4	0,2
	Medium	0,3	0,4	0,3
	Low	0,3	0,3	0,4
Low	High	0,2	0,3	0,5
	Medium	0,2	0,2	0,6
	Low	0,1	0,2	0,7

5. Моделювання взаємозв'язків чинників якості

Оскільки для прогнозування якості суттєвим є визначення впливових чинників на кожній стадії розроблення, байєсівську мережу якості ПЗ у загальному вигляді можна подати як композицію якості для окремих стадій (рис. 5).

В узагальненій БМ всі вершини (крім першої та останньої) і асоційовані з

ними БМ другого та третього рівнів є типовими з точки зору побудови БМ, тобто за причинно-наслідковими зв'язками оцінюваної якості робочих продуктів у відповідних вершинах.

Приклад типової БМ для моделювання якості проміжних робочих продуктів поданий на рис. 6.

Нетиповими є перша та остання БМ у ланцюжку (рис. 5). Перша – моделює систему взаємозв'язку понять, стосую-

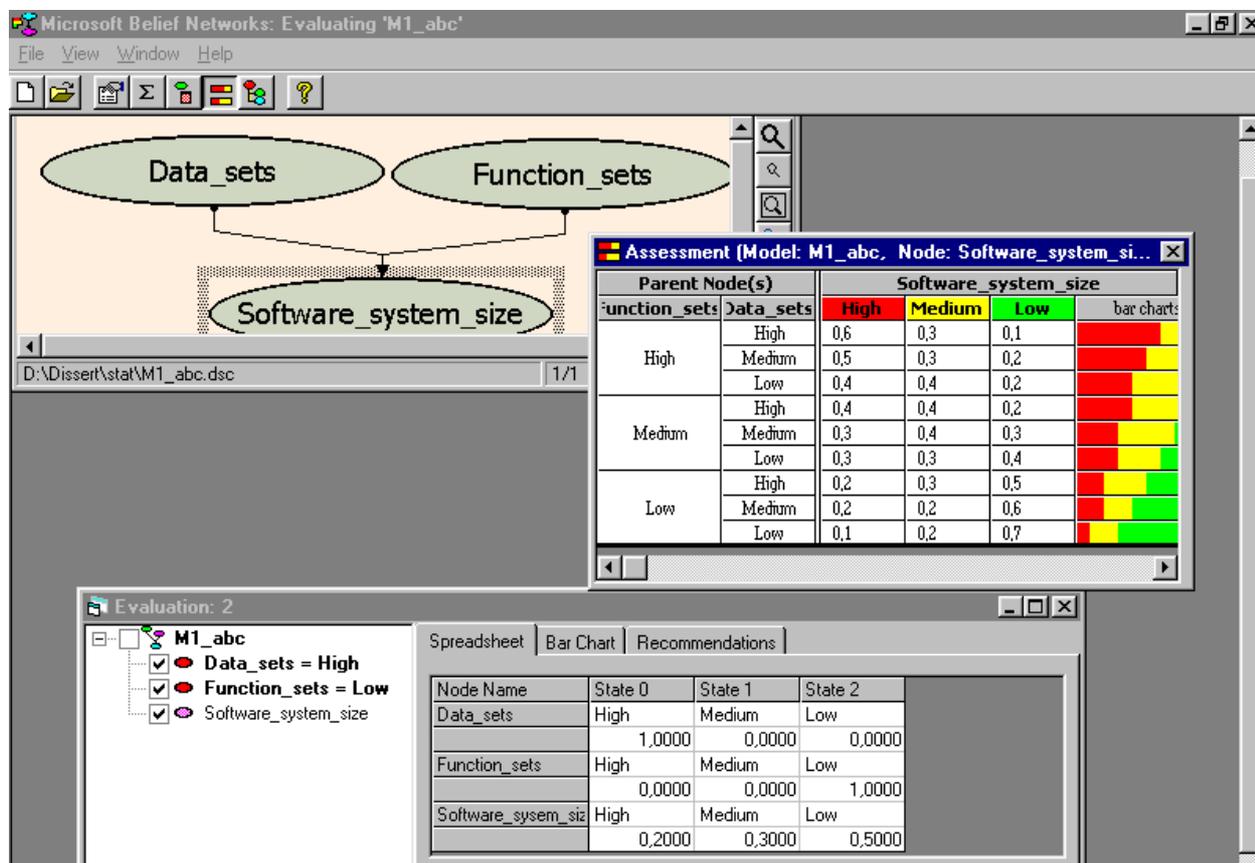


Рис. 4. Приклад БМ, побудованої за допомогою MSBNx

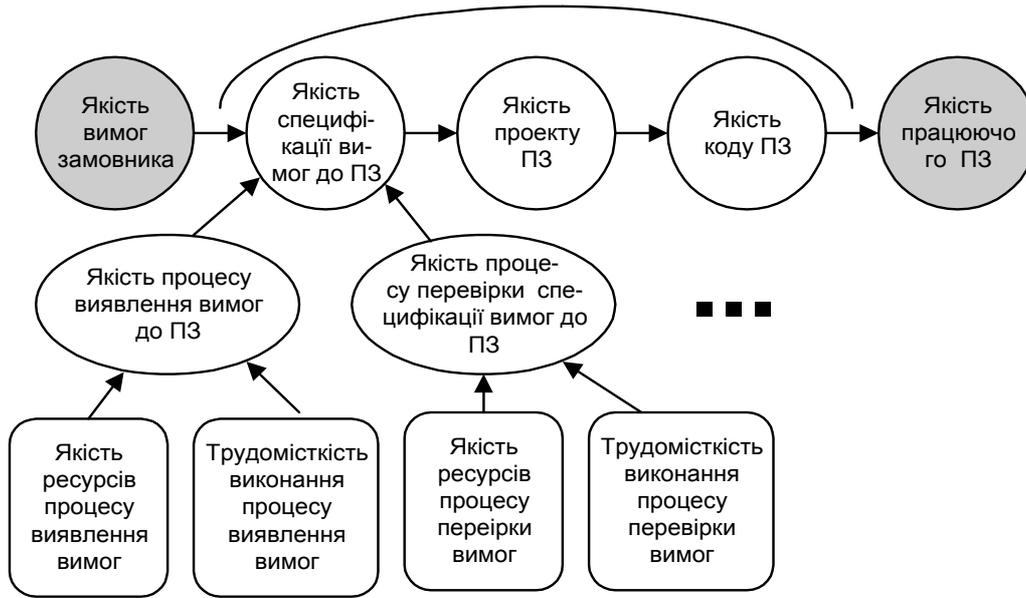


Рис. 5. Узагальнена (примітивна) байєсівська мережа якості ПЗ

вних об'єкта автоматизації і відповідної предметної галузі. Приклад такої моделі поданий на рис. 7.

Остання БМ (рис. 5) моделює систему взаємозв'язку понять, стосовних безпосереднього оцінювання зовнішньої характеристики якості. Наприклад, якщо оцінюваною характеристикою є надійність ПЗ, то для цього застосовуються моделі зростання надійності [12]. У БМ повинні бути присутні вершини, пов'язані з параметрами відповідної моделі, а також з умовами експлуатації ПЗ,

які відбиваються операційним профілем ПЗ [13].

Очевидно, що "глибина" та розгалуженість побудованих БМ залежить, по-перше, від системи знань людини-будівника про предмет дослідження, по-друге, від доцільності поглибленої деталізації БМ для дослідження предмету і, по-третє, від наявності об'єктивних (фактичних) даних про причинно-наслідкові зв'язки понять у вершинах БМ.

Перевагою моделей, побудованих за допомогою байєсівських мереж, є те,

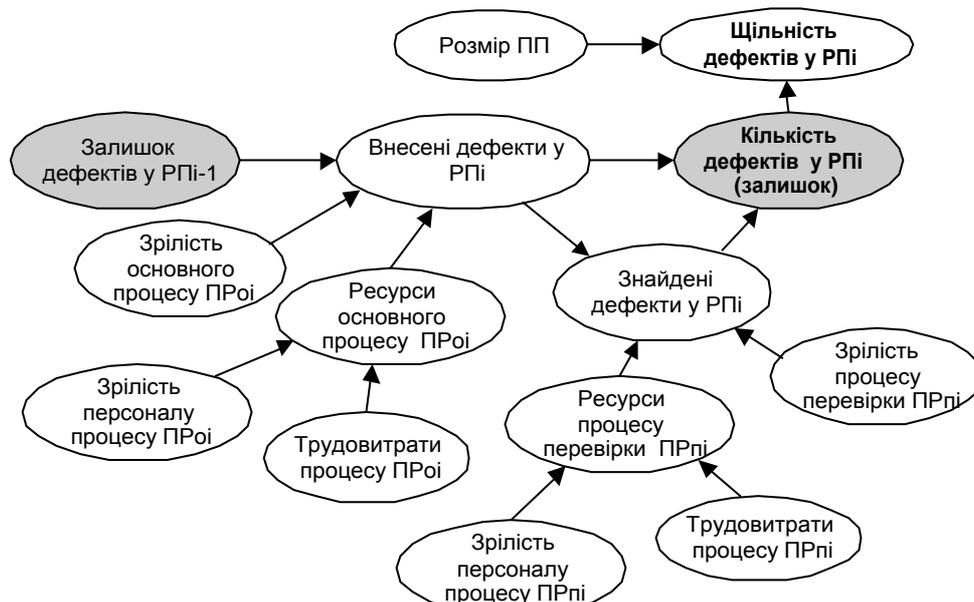


Рис. 6. Типова БМ для моделювання якості проміжних РП

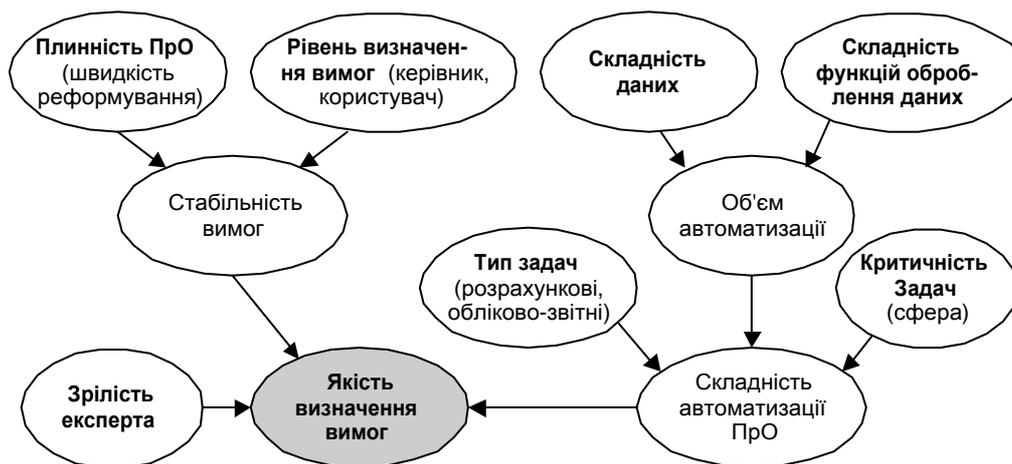


Рис. 7. БМ для моделювання якості вимог до ПЗ

що від шкали значень випадкових величин (ВВ) у вершинах може бути змінений після її побудови, тобто у ході застосування мережі (за наявності автоматизованої підтримки використання БМ).

За відсутності в архітектурі БПО процесів вимірювання та оцінювання (або у примітивних БМ, як на рис. 5) шкала значень може бути якісною – *номінальною* або *порядковою* (наприклад, “високий”, “середній”, “низький” рівень значень). Саме так може бути оцінена якість вихідних вимог до ПЗ, які з’ясовує аналітик проекту з представниками організації замовника (рис. 7). Якість оцінок за цією моделлю залежить від рівня знань замовника та аналітика про предметну галузь та потреби її автоматизації.

Якщо є більш суттєві знання про властивості ВВ (що ґрунтуються на історичних даних вимірювань у проектах), доцільно використовувати кількісні шкали: *інтервальну, відносну*.

За орієнтири при визначенні причинно-наслідкових зв’язків чинників якості пропонується взяти сукупність елементів таксономії ризику проекту [14]. Для вершин, пов’язаних з оцінками працевитрат, доцільно використати атрибути що містяться у вартісних моделях, наприклад СОСОМО [15]. Оцінки обсягу (розміру) розроблюваного ПЗ можуть ґрунтуватися на чинниках моделей, побудованих за методологією FPA [16], а оцінки зрілості БПО організації-розробника – на

чинниках, визначених у моделях SPICE або СММ [17].

6. Програмна підтримка процесів оцінювання та прогнозування якості

Розроблення програмного комплексу (ПК) підтримки оцінювання та прогнозування якості ПЗ виконується із застосуванням бібліотеки функцій (класів) HUGIN API ActiveX Server 6.3 for Visual Basic (далі НАPI). Ця бібліотека надає функції для вбудовування у ПК засобів роботи із сукупністю розроблених БМ.

VB-інтерфейс забезпечує взаємодію байєсівської мережі у ПК з об’єктами, створеними у середовищі MS Office (зокрема Excel, Access). Ця важлива особливість НАPI дозволяє використовувати середовище Excel для підготовки даних для таблиць ймовірностей у вершинах БМ, а середовище Access для накопичення та використання історичних даних щодо артефактів проекту. А це, у свою чергу, дозволяє позбутися обмежень HUGIN стосовно оброблення неперервних значень ВВ за законом нормального розподілення. Розрахунки, виконані на значеннях ВВ, розподілених за іншими законами (наприклад, біноміальним), можуть бути виконані в Excel та експортовані у байєсівську мережу ПК.

Відповідно до описаного підходу до моделювання причинно-наслідкових зв’язків чинників якості розроблені байє-

сівські мережі для прийняття рішень менеджером проекту на кожній стадії ЖЦ. Оскільки для прогнозування надійності важливим є врахування попереднього досвіду оцінення (на стадіях ЖЦ, які передують поточній стадії), пропонується механізм композиції мереж, який полягає у введенні в БМ додаткової вершини, що є “спільною” для двох БМ у композиції.

Наприклад, на рис. 8 подані дві мережі “якість вимог до ПЗ” та “якість РП на стадії”. На першій стадії розроблення оцінюваним РП є “специфікація вимог”. Вважаємо, що достовірна специфікація вимог міститься у документі “постановка задач” (за ГОСТ 34), оскільки відомо, що такі документи, як “технічне завдання” або “пояснювальна записка до проекту” насправді такої специфікації не містять. Дві мережі об’єднуються за допомогою вершини “ЯкістьВихВимог”.

Висновки

Апарат байєсівських мереж використовується при вирішенні задач,

пов’язаних з прийняттям рішень в умовах невизначеності. В програмній інженерії, зокрема, в інженерії якості та надійності програмного забезпечення, вони ще не знайшли свого широкого застосування, хоча проблема прийняття обґрунтованих рішень в галузі керування проектами існує. БМ можуть використовуватися для аналізу ефективності архітектури ПЗ, аналізу причин і наслідків відмов у ПЗ, керування ризиком проекту тощо. Процесний підхід до створення ПЗ та впровадження прийомів вимірювання та оцінювання в процес розроблення складають підґрунтя для накопичування та використання історичних даних, стосовних проектів програмного забезпечення. Для аналізу даних пропонується використовувати апарат байєсівських мереж. Розроблені БМ складають базис моделювання якості ПЗ на стадіях життєвого циклу розроблення ПЗ.

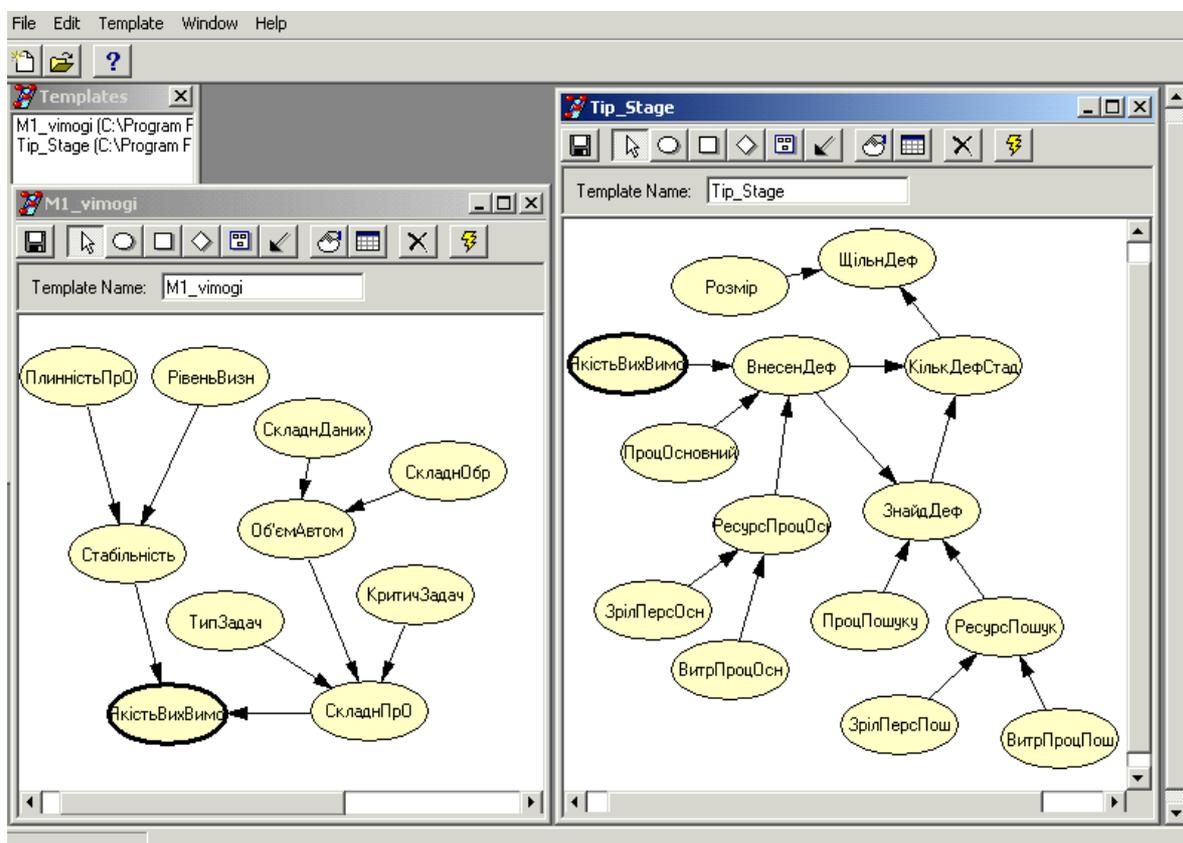


Рис. 8. Дві байєсівські мережі якості, побудовані в HUGIN Lite 6.5

1. *Планирование* обеспечения надежности информационных систем / Г.И. Коваль, Т.М. Коротун, Т.Л. Яблокова, Л.И. Кузаченко // Проблемы программирования. – 2001. – №3-4. – С. 40 - 47.
2. *Парадигма* качества программного обеспечения / Ф.И. Андон, В.Ю. Суслов, Т.М. Коротун, Г.И. Коваль // Там же. – 1999. – №2. – С. 51 - 62.
3. *ДСТУ 3918-99*. Інформаційні технології. Процеси життєвого циклу програмного забезпечення. – Київ: Держстандарт України, 2000. – 49 с.
4. *ДСТУ ISO/IEC TR 15504 (Частини 1–9)*. Інформаційні технології. Оцінювання процесів програмування. – Київ: Держспоживстандарт України, 2002.
5. *Основы* инженерии качества программных систем / Ф. И. Андон, Г. И. Коваль, Т.М. Коротун, В. Ю. Суслов – К.: Академперіодика, 2002. – 504 с.
6. *ДСТУ 2850-94*. Програмні засоби ЕОМ. Показники і методи оцінювання якості. – Держстандарт України. – 1994. – 20 с.
7. *Коваль Г.И.* Подход к прогнозированию надежности ПО при управлении проектом // Проблемы программирования. – 2002. – № 1 – 2. – С. 282 – 290.
8. *Коваль Г.И., Коротун Т.М.* Проблемы анализа причинно-следственных связей отказов и ошибок в ПО // Сб. Программная инженерия. – Киев: Ин-т кибернетики им. В.М. Глушкова НАНУ, 1994. – С. 83 - 93.
9. *Коваль Г.И.* Прогнозирование надежности ПО компьютерных систем // Сб. материалов конференции. УкрПРОГ'98, 2-4 сентября 1998 г., Киев.- 1998. – С. 358 – 361.
10. *Мищенко Н.М.* Байєсові мережі як засіб моделювання причинно-наслідкових зв'язків у системах з невизначеністю // // Проблемы программирования. – 2002. – №3-4. – С. 125 - 131.
11. *Murphy K.P.* Brief Introduction to Graphical Models and Bayesian Networks – <http://HTTP.CS.Berkeley.EDU/~murphyk/Bayes/bayes.html>.
12. *Мороз Г.Б., Лаврищева Е.М.* Модели роста надежности ПО // Киев, 1992. – 25 с. – (Препр. / АН Украины. Ин-т кибернетики им. В.М. Глушкова; 92 - 38).
13. *Мороз Г.Б., Коваль Г.И., Коротун Т.М.* Концепция профилей в инженерии надежности программных систем // Математичні машини і системи. – 2004. – №1. – С. 166 - 184.
14. *Управление* риском проектов ПО / Ф.И. Андон, В.Ю. Суслов, Т.М. Коротун, Г.И. Коваль, О.А. Слабоспицкая // Проблемы программирования. – 1999. – №1. – С. 53 - 62.
15. *Визначення* витрат на створення ПЗ автоматизованих систем / Ф.И. Андон, В.Ю. Суслов, Т.М. Коротун, Г.И. Коваль, О.А. Слабоспицкая // Там же. – 1998. – №3. – С. 23 – 34.

ззованих систем / Ф.И. Андон, В.Ю. Суслов, Т.М. Коротун, Г.И. Коваль, О.А. Слабоспицкая // Там же. – 1998. – №3. – С. 23 – 34.

16. *Коваль Г.И.* Методы определения размера ПО // Там же. – 1999. – №1. – С. 63 - 71.

17. *Модель* оценки технологической зрелости организаций-разработчиков ПО / Ф.И. Андон, В.Ю. Суслов, Т.М. Коротун, Г.И. Коваль, О.А. Слабоспицкая // Там же. – 1998. – №4. – С. 46 -57.

Отримано 28.03.05.

Про автора

Коваль Галина Іванівна,
ст.наук.співр

Місце роботи автора:

Інститут програмних систем НАН
України,
просп. Академіка Глушкова, 40,
Київ-187, 03680, Україна
Тел. 526 4579,
e-mail: galia@isofts.kiev.ua