УДК 004.415.2.045 (076.5)

МЕТОД ТА ЗАСІБ ДЛЯ ЕМПІРИЧНИХ ДОСЛІДЖЕНЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Сидоров М.О., Дишлевий О.П.

Національний авіаційний університет

nikolay.sidorov@livenau.net

Стаття присвячена розробленню методу та засобу для емпіричних досліджень програмного забезпечення. Розглянуто середовища для емпіричних досліджень, зроблено висновок про потребу в розробленні спеціального середовища для емпіричних досліджень програмного забезпечення. У статті розглянуто експерименти, які визначили особливості методу, пропонується його схема. Подається засіб, який реалізує розроблений метод, з описанням архітектури, бази даних і функціонування.

The article is dedicated to designing of a method and tool for empirical study of software. Existing environments for empirical studies are reviewed, and the conclusion about need for development of special environment for empirical studies of the software is made. Article describes the experiments which defined specifics of a method, and offers its scheme. The tool which enables the developed method, including the description of architecture, database and functionality is presented.

Вступ

В інженерії програмного забезпечення постають питання досліджень, які належать до емпіричної інженерії програмного забезпечення [1]. Основним методом досліджень є вимірювання, а одним із головних інструментів цих досліджень є метрики, за допомогою яких оцінюють властивості складових розроблення програмного забезпечення (продуктів, процесів). Існують прямі та непрямі метрики [2]. Прямі метрики піддаються вимірюванню, але їх недостатньо для оцінювання більшості властивостей. Тоді використовуються непрямі метрики, які формуються на основі прямих. Головне завдання, яке при цьому постає, — визначення виду та ступеня залежності непрямої метрики від прямої [3]. Для цього використовують два підходи — *статистичний аналіз* [4] *та нейронні мережі* [5].

Для розв'язання завдання з визначення залежностей між метриками за допомогою нейромереж потрібно знати не тільки значення метрик (вхідних величин), а має бути проведене навчання нейромережі [5]. Навчання нейромережі проводиться на вже отриманих раніше даних. Так як питаннями визначення залежностей між метриками до цього часу не займалися, сформулювати правила для навчання нейромережі неможливо. Для розв'язання завдання з визначення залежностей між метриками за допомогою статистичного аналізу достатньо знати тільки значення метрик, а залежності будуються статистичними методами [4]. Тому пропонується застосовувати статистичний аналіз.

Постановка завдання

Статистичний аналіз проводиться за допомогою відповідних математичних програмних середовищ, до яких належать *MatLab, MathCad, Maple, Mathematica, MS Excel*. Крім них можна використати статистичні програмні середовища загального призначення *Statistica*, SPSS, SAS, *Systat, Minitab, Statgraphics* чи програмні середовища спеціального призначення SYSTAT, S-plus, STATA, PRISM, STADIA, Олимп, Класс-Мастер, Статистик-Консультант. Математичні програмні середовища та статистичні програмні середовища загального призначення для розв'язання поставленого завдання потребують додаткового програмування з використанням статистичних алгоритмів. Середовища для емпіричних досліджень у програмному забезпеченні немає.

Отже, для визначення залежностей між метриками програмного забезпечення потрібний метод і середовище, яке буде реалізовувати цей метод. У статті пропонується метод обробки даних емпіричних досліджень програмного забезпечення за допомогою статистичного аналізу та засіб — статистичне середовище спеціального призначення, яке реалізує даний метод.

Метод обробки даних емпіричних досліджень програмного забезпечення на основі статистичного аналізу. До методів емпіричних досліджень програмного забезпечення належать: керовані експерименти, дослідження конкретних випадків (*case studies*), дослідження-огляди, етнографії, дослідження дій [3]. Звдання визначення залежностей між метриками програмного забезпечення відноситься до дослідження огляду [3]. У рамках дослідження відбувається вимірю-вання прямих метрик програмного забезпечення, збір даних — результатів вимірювань, обробка даних та визначення залежностей непрямих метрик від прямих. Розроблення методу визначення залежностей між метриками програмного забезпечення здійснюється шляхом аналізу відомого статистичного аналізу з урахуванням особливостей програмного забезпечення.

У загальному вигляді статистичний аналіз, який виконується з метою визначення залежностей, складається з трьох етапів: первинний статистичний аналіз, кореляційний аналіз та регресійний аналіз [4]. Залежності будуються на етапах кореляційного та регресійного аналізу, але без попереднього аналізу даних на першому етапі зробити це неможливо. Це пов'язано з тим, що метою першого етапу є визначення виду розподілу досліджуваної величини, від якого залежать наступні етапи. Існує велика кількість розподілів [8], але для визначення подальших

досліджень суттєву роль грає наявність чи відсутність нормального розподілу [8]. Залежно від наявності нормального розподілу використовуються ті, чи інші алгоритми побудови залежностей. Тому важливим питанням є визначення чи мають нормальний розподіл досліджувані метрики.

Дослідження залежностей у прикладних науках показало, що для кожної із них існують свої закономірності притаманні тільки даним з їх предметної області [6; 7]. Тому був проведений експеримент з визначення закону розподілу метрик програмного забезпечення. Експеримент проводився в рамках дослі-дження повторного використання програмного забезпечення [9].

Метою дослідження було визначення програмного коду, придатного для повторного використання. Для цього завдання раніше використовувалися експерти, які оцінювали програмний код для ряду непрямих метрик. Пропонувалося автоматизувати процес оцінювання шляхом визначення прямих метрик, які отримуються в результаті вимірювання програмного забезпечення, на непрямі метрики, які оцінюють експерти. Усі метрики (прямі та непрямі) були вибрані з розрахунку можливості їх використання для визначення повторно використовуваного програмного коду [2].

Для розв'язання поставленого завдання перш за все потрібно було дослідити вибрані метрики. Завданням експерименту з визначення законів розподілу було розрахувати оптимальні значення для розглянутих метрик та можливий діапазон їхніх відхилень. Для цього були виміряні значення всіх метрик. Для визначення оптимальної кількості значень метрик при побудові законів розподілу вимірювання проводилося в кілька етапів. Спочатку були виміряні п'ятдесят програм. Для кожної метрики були розраховані математичні характеристики та їх відхилення [9]. Далі були виміряні ще десять програм. Для значень метрик, отриманих з усіх програм, знову були проведені аналогічні розрахунки. Таким чином були виміряні сто програм та проведені подібні обчислення. Початкова кількість вимірюваних програм була взята випадковим чином, а крок у десять програм вибраний на основі розрахованих відхилень математичного сподівання [8]. При додаванні наступних десяти програм їх математичне сподівання виходило за межі розрахованих попередньо відхилень.

Загальна кількість у сто програм зумовлена тим, що для останніх двадцяти математичне сподівання входило в межі відхилень. Тому, виходячи із закону великих чисел [9], був зроблений висновок про недоцільність подальшого збільшення програм для вимірювань. Далі був побудований для кожної метрики закон розподілу та обчислені відхилення. Отримані типові закони розподілу метрик подано на рис. 1.

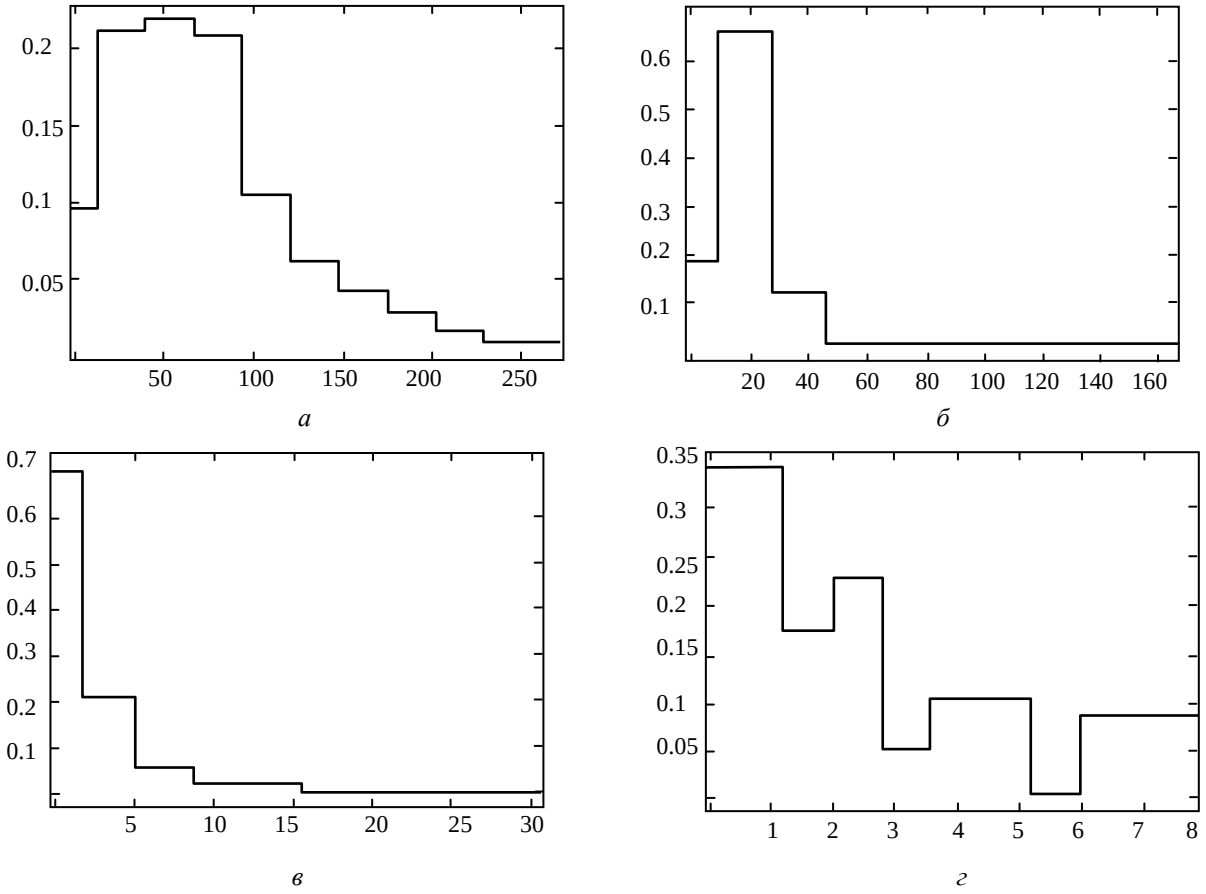

Рис. 1. Гістограми метрик:
а — максимальна кількість непустих рядків у модулі; *б* — максимальна кількість викликів інших функцій,

обчислена в модулі; *в* — середня кількість викликів вводу—виводу, використаних у кожному модулі; *г* — середня кількість аргументів (параметрів), використаних у кожному модулі

Аналіз усіх гістограм показує, що їх вигляд можна звести до наступних чотирьох типових (рис. 1, *а* — 1, *г*): рис 1, *а*, 1, *б* — гістограми з лівою асиметрією; рис 1, *в* — унімодальна гістограма з сильною лівою асиметрією; рис 1, *г* — багатомодальна гістограма.

Експеримент показав дві особливості програмного забезпечення:

1. доступність програм як об'єктів дослідження за кількістю та різноманітністю (це є проблемою в інших науках через складність отримання даних), що дає можливість використовувати точні, а не наближені методи розрахунків;

2. відсутність нормального закону розподілу метрик.

Із результатів експерименту можна зробити висновок: оскільки у метрик з досліджених наборів немає нормального закону розподілу, то визначати закон розподілу в розроблюваному методі недоцільно, але необхідно обов'язково перевіряти обсяг вибірки.

Для визначення наявності залежності непрямої метрики від прямої проводиться *кореляційний аналіз*. Загалом кореляційний аналіз можна проводити двома шляхами [10]: простий розрахунок коефіцієнтів парної кореляції та розрахунок парної рангової кореляції. Перший використовується тоді, коли досліджувані величини мають нормальний розподіл, другий — коли нормального закону розподілу немає. Для емпірич-них досліджень програмного забезпечення з визначення залежності непрямої метрики від прямої потрібно використати розрахунок парної рангової кореляції, що пов'язано з розподілом, відмінним від нормального.

Суть парної рангової кореляції полягає в порівнянні не самих значень величин чи їх статистичних характеристик, а рангів, тобто номерів досліджуваних величин у відповідних матрицях (наборах метрик). Визначається парна рангова кореляція методом обчислення коефіцієнта Спірмена чи Кендала [10]. Залежність для значень коефіцієнтів, відмінних від –1, 0 та 1, підтверджується розрахунком значущості. При проведенні розрахунків не потрібно перевіряти точність отриманих значень, оскільки вимірювання метрик програмного забезпечення не має похибок, пов'язаних з людським фактором чи засобом вимірювань. Отже, кореляційний аналіз у розроблюваному методі характеризується:

- проведенням розрахунків парної рангової кореляції;
- відсутністю перевірок на точність отриманих даних.

Для визначення виду залежності непрямої метрики від прямої застосовується *регресійний аналіз*. Він полягає в побудові та розрахунках коефіцієнтів функції регресії, яка відображає залежність непрямої метрики від прямої. Виділяють два види регресій — *лінійну* та *нелінійну* [10]. Лінійна регресія будується у разі, коли при кореляційному аналізі був зроблений висновок про наявність лінійної залежності, в іншому випадку будується нелінійна регресія. Лінія регресії будується на основі кореляційного поля. Якщо побудовані точки кореляційного поля потрапляють в коло, то робиться висновок про відсутність залежності. Якщо ж кореляційне поле не вписується у коло, а має інший вигляд, то робиться висновок про нелінійну залежність у лінії регресії.

Оскілки дані досліджень програмного забезпечення не мають нормального закону розподілу, то будується нелінійна регресія. Обов'язковою передумовою побудови будь-якої регресії є нормальний закон розподілу залежної метрики або обох метрик, який відсутній. У зв'язку з великою вибіркою дана передумова ігнорується. Єдиної теорії побудови нелінійної регресії немає [10]. Тому при регресійному аналізі залежно від даних використовується той чи інший метод нелінійної регресії. Для реалізації регресійного аналізу в розроблюваному методі нелінійної регресії були використані результати експерименту за вибором методів побудови регресії на даних досліджень програмного забезпечення. Будувалися наближені лінії регресії методом найменших квадратів, поліномів Чебишева та лінеаризації (побудова найпростіших наближених функцій).

Суть експерименту полягає в побудові ліній регресії різними способами для кожного з кореляційних полів з подальшим визначенням найточнішої лінії. У зв'язку з великим обсягом проведення розрахунків було прийняте рішення про побудову лінії регресії спочатку для однієї прямої метрики та непрямої метрики «простота сприйняття», а далі для контролю отриманих даних побудувати лінії регресії для кількох метрик. Кількість метрик збільшувалась доти, доки не була підтверджена закономірність. Спочатку для побудови лінії регресії була взята метрика «середнє значення непустих рядків у модулі». Її вибір пов'язаний з великим значенням значущості, що говорить про її суттєвий вплив на непряму метрику «простота сприйняття» [9]. Для неї були побудовані наближені функції регресії трьома способами. Виявилося, що максимальний ступінь функції — 3, що говорить про простоту функції регресії. Її слід вибирати серед невеликого переліку простих функцій [10]. Після розрахунків коефіцієнтів наближених функцій та перевірки відхилення функцій виявилося, що найоптимальнішою функцією є експоненційна функція (рис. 2).

Далі були побудовані лінії регресії ще для трьох метрик. Результати побудови та розрахунків коефіцієнтів лінії регресії для даних метрик підтвердили відсутність складних функцій залежності з великими степенями. Результати порів-няння методів нелінійної регресії показали, що перші два методи недоцільно вико-

ристовувати для побудови регресії, так як максимальний ступінь найближчої наближеної лінії регресії — 3. Тому для реалізації запропонованого методу було обрано метод лінеаризації.

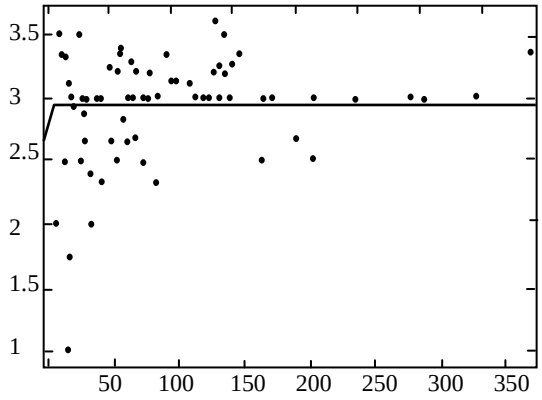

Рис. 2. Лінія регресії прямої метрики «середнє значення непустих рядків у модулі» та непрямої метрики «простота сприйняття»

Таким чином, на основі проведених досліджень пропонується метод обробки даних емпіричних досліджень програмного забезпечення на основі статистичного аналізу. Суть методу полягає в тому, що для визначення виду залежності непрямої метрики від прямої використовується розрахунок тільки парної рангової кореляції метрик без великої кількості перевірок та будуються тільки нелінійні функції регресії у зв'язку з відсутністю нормального розподілу метрик без проведення первинного статистичного аналізу. В інших галузях досліджувані величини, як правило, мають нормальний розподіл, і тому для визначення їх залежностей використовується простий розрахунок парної кореляції, а також наявна лінійна регресія між величинами [6; 7]. Запропонований метод дає змогу визначати залежність непрямої метрики від прямої без попереднього визначення їх законів розподілу та визначити лінію регресії без попереднього аналізу досліджуваних величин.

Отже, запропонований метод статистичної обробки даних досліджень програмного забезпечення характеризується так (рис. 3): відсутність визначення закону розподілу метрики; обов'яз-кове велике значення вибірки; використання розрахунку тільки парної рангової кореляції; від-сутність перевірки точності коефіцієнтів кореляції; відсутність перевірки спільного закону розподілу метрик; побудова регресії методом лінеаризації.

Програмне забезпечення для методу обробки даних емпіричних досліджень на основі статистичного аналізу. Середовище, яке реалізує запропонований метод, складається з блоків — введення даних, попередніх обчислень, кореляційного аналізу, регресійного аналізу, побудови графіків, блоку виведення та бази даних метрик.

Архітектура середовища. Архітектура має такий вигляд — рис. 4. Блок введення даних — призначений для відображення даних з бази даних та для ручного введення даних. Також він призначений для вибору метрик для досліджень. Блок попередніх обчислень призначений для обчислення математичного сподівання, коефіцієнтів асиметрії та ексцесу.

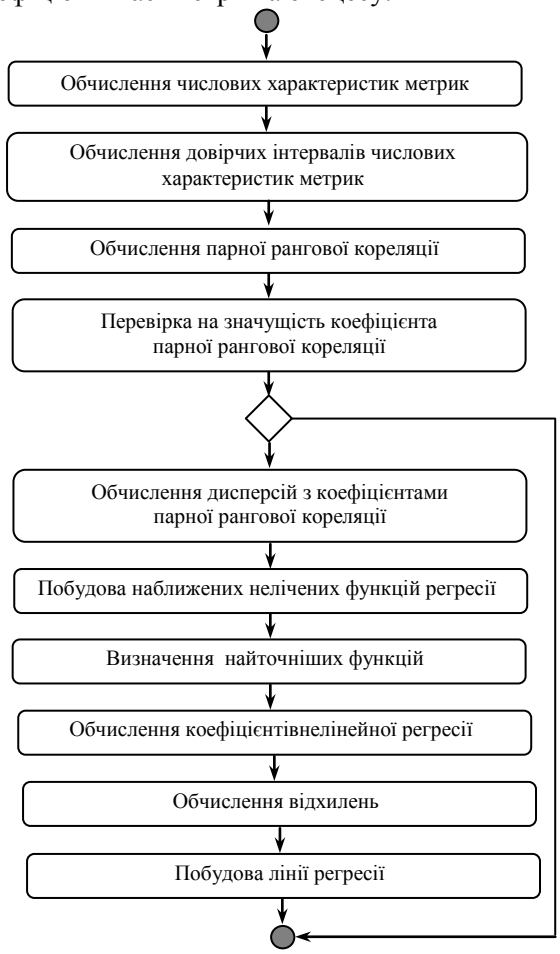

Рис. 3. Метод визначення залежностей між метриками програмного забезпечення за допомогою статистичного аналізу

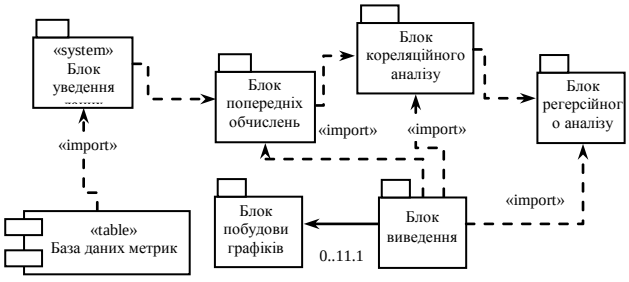

Рис. 4. Архітектура статистичного пакету

Блок кореляційного аналізу призначений для розрахунків коефіцієн-тів кореляції, значущості залежності та визначення наявності чи відсутності залежності. Блок регресійного аналізу призначений для визначення виду функції залежності однієї метрики від іншої. Блок побудови графіків призначений для побудови та виведення розподілів та графіків функцій залежності. Блок виведення призначений для ви-

ведення обчислених результатів та висновків за отриманими закономірностями.

База даних призначена для зберігання наборів метрик та результатів вимірювань вихідних кодів програм. База даних відображає такі сутності (рис. 5): *metrica* — містить код, назву, описання метрик, які застосовуються для досліджень програмного забезпечення, а також номер та назву експерименту, коли була виміряна; *unitmetr* — містить значення метрик по кожному виміряному модулю; *unit* — містить вихідні модулі програмного забезпечення, які вимірюються, мову, якою написаний модуль та коментар; *experiment* — містить дані про проведені експерименти з вимірювання програмного забезпечення.

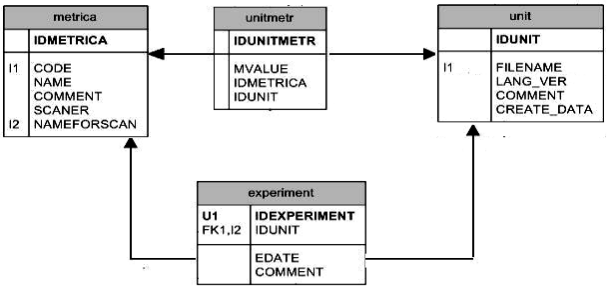

Рис. 5. Схема бази даних

Подана база даних містить невеликий набір сутностей і прості зв'язки між ними. Дані, які являють ці сутності, можуть бути отримані з будь-якого вимірювача програмного забезпечення. Дослідникові потрібно власноруч імпортувати отримані результати вимірювань метрик з вимірювача в базу даних для аналізу.

Застосування статистичного середовища. Статистичне середовище може функціонувати у двох режимах: вибір даних та аналіз.

У режимі «вибір даних» дослідник повинен вибрати метрики, які він досліджуватиме. За потребою він може їх відредагувати (додавши, видаливши чи змінивши метрику та дані до неї). Додати для аналізу дослідник може як одну, так і кілька метрик (рис. 6).

У режимі «аналіз даних» (рис. 7) дослідник може побачити основні статистичні характеристики, коефіцієнти кореляції метрик. Тут він може запустити графічний режим, у якому може побачити функції розподілу метрик та лінії регресії. За потреби дослідник може вивести коефіцієнти функції регресії та результати перевірок.

Висновки. Особливості методу для емпіричної інженерії програмного забезпечення значно спрощують кореляційно-регресійний метод статистичних досліджень, а також допомагають досліднику програмного забезпечення зрозуміти суть та особливості цього дослідження.

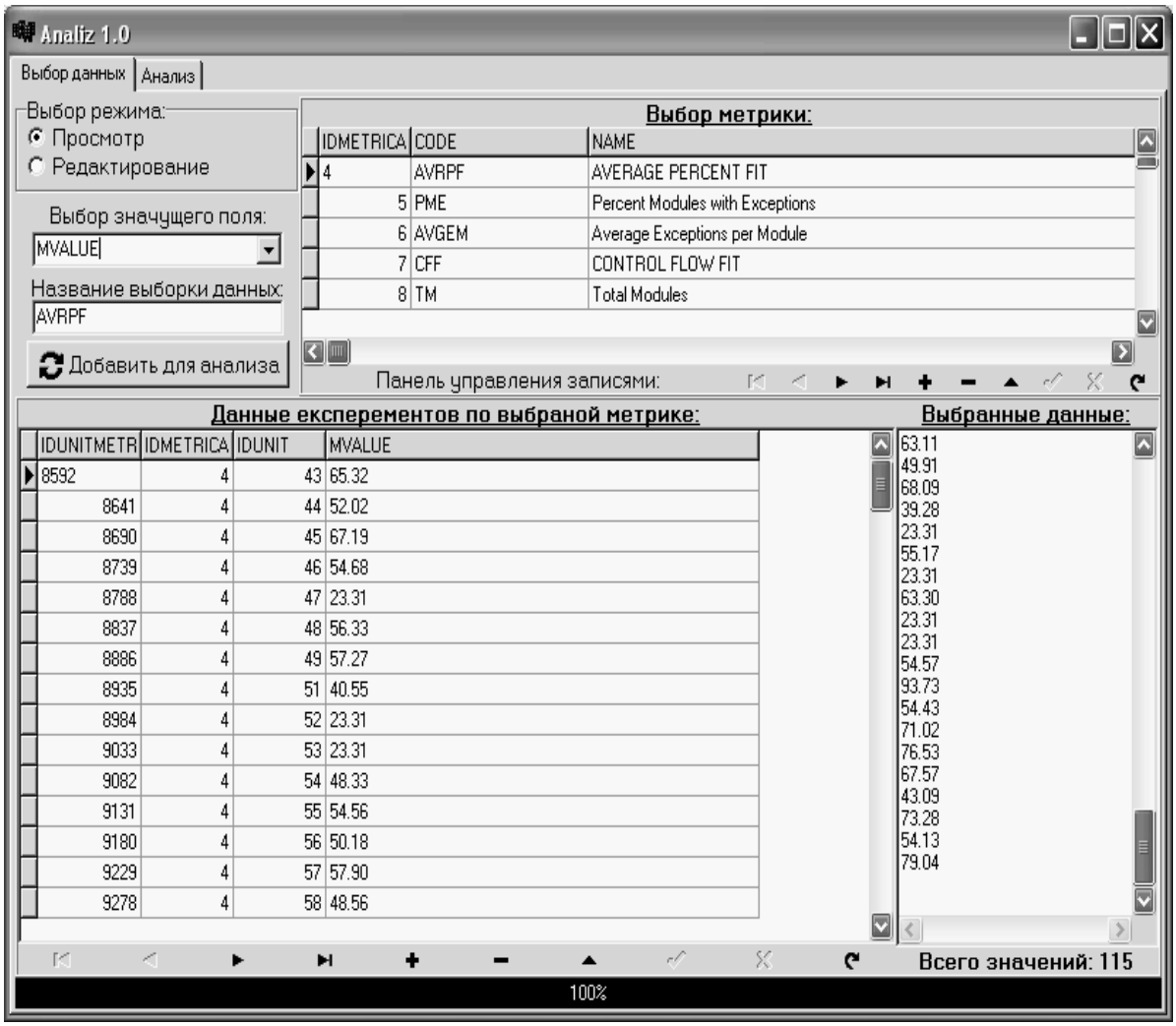

Рис. 6. Режим роботи «вибір даних»

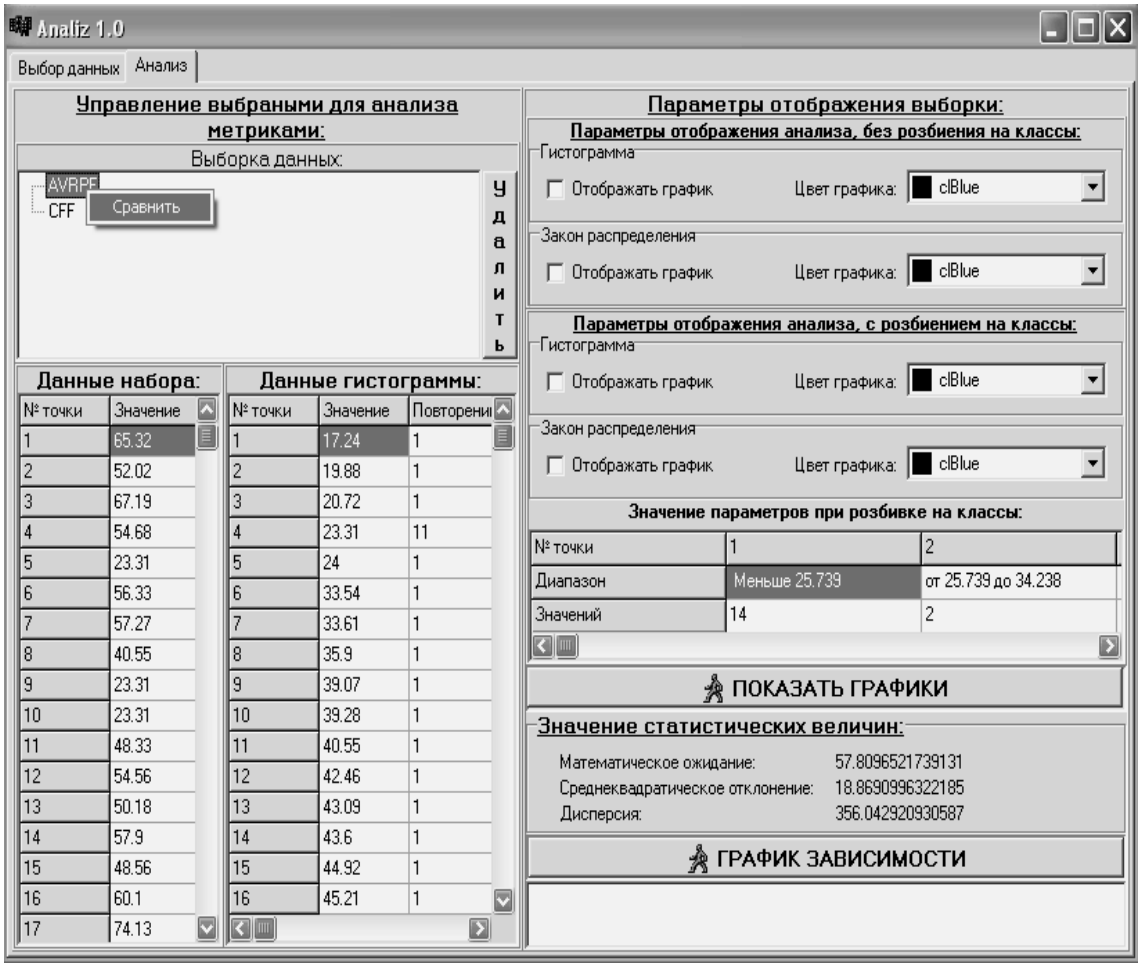

Рис. 7. Режим роботи «аналіз даних»

Запропонований метод являє собою чітку послідовність дій, які повинен виконати дослідник програмного забезпечення при визначенні залежності непрямої метрики від прямої. При цьому досліджуючи при цьому всі методи аналізу, йому не потрібно вибирати серед існуючих статистичних методів той, який йому буде прийнятний.

На основі методу реалізовується засіб для визначення залежностей між метриками, який набагато спрощує роботу дослідника. Тепер не потрібно додатково у статистичних середовищах писати програмний код для проведення необхідних розрахунків. Дослідникові достатньо мати тільки набір метрик, які він хоче дослідити. Після занесення метрик до бази даних йому тепер достатньо натиснути кілька кнопок, щоб отримати результат як у числовому вигляді, так і у графічному.

ЛІТЕРАТУРА

1. *Koji Torii, Kenichi Matsumoto, Kumiyo Nakakoji, Yoshiro Takada, Kaduyuki Shima,* Ginger 2: An Enviroment for Computer-Aided Empirical Software Engineering // IEEE Transactions on Software Engineering. — 1999. — Vol 25. —No 4, July/August. — P. 475 — 486.

2. *Norman E. F.* Shari Lawrence Pfleeger Software. Metrics: A Rigorous and Practical Approach. — Cambridge University Press, 1996. — 638 p.

3. *Forrest Shull, Janice Singer, Dag I.K. Sjoberg.* Guide to Advanced Empirical Software Engineering. — Springer-Verlag London Limited 2008. — 394 p.

4. *Вентцель Е.С.* Теория вероятностей : учеб. для вузов. — 7-е изд. стер. / Е. С. Вентцель. — М. : Высш. шк., 2001. — 575 с.

5. *Уоссермен Ф.* Нейрокомпьютерная техника : Теория и практика: пер. с англ. / Ф. Уоссермен. — М. : Мир, 1992. — 118 с.

6. *Рокицкий П.Ф.* Биологическая статистика. Изд. 3-е, испр. / П.Ф. Рокицкий. — Минск : Высшэйш. шк., 1973. — 320 с.

7. *Дружинин Н.К.* Математическая статистика в экономике / Н.К. Дружинин. — М. : Статистика, 1971. — 262 с.

8. *Кендалл М.* Теория распределений : пер. с англ. / М. Кендалл, А. Стюарт. — М. : Глав. ред. физ.-мат. лит-ы изд-ва «Наука», 1966. — 588 с.

9. *Дишлевий О.П.* Перевірка адекватності метричних моделей властивостей програмного забезпечення // Інженерія програмного забезпечення 2006: Матеріали Всеукр. конф. аспірантів та студентів. — К.: НАУ, 2007. — С. 77—84.

10. *Айвазян С.А. и др.* Прикладная статистика: Исследование зависимостей : справ. изд. / С.А. Айвазян, И.С. Енюков, Л.Д. Мешалкин : под. ред. С.А. Айвазяна. — М.: Финансы и статистика, 1985. — 487 с.

Стаття надійшла до редакції 06.03.09