

УДК 004.4'2

Н.П. Стасюк, А.Й. Савицький

ЗАСТОСУВАННЯ МЕТОДУ АНАЛІЗУ ІЄРАРХІЙ ПРИ ВИБОРІ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анотація: Робота присвячена системам підтримки прийняття рішень, побудованим на основі методу аналізу ієрархій. Визначено приклад декомпозиції проблеми на її складові частини і побудови відповідної ієрархії, а також розглянута процедура визначення рейтингів доступних альтернатив.

Ключові слова: система підтримки прийняття рішень, аналіз ієрархій, альтернативний варіант вибору.

Вступ

Сьогодні існує безліч різних програмних інструментальних засобів, подібних за можливостями. Всі вони мають широкий функціонал, цілий набір вбудованих засобів і інструментів, постійно розвиваються, отримуючи все нові і нові функції, але, як результат, і ціна їх також не стає меншою. Як вибрати з цього різноманіття саме те, що буде найкращим чином вирішувати поставлені завдання [1]?

Яким би не був важким такий вибір, робити його доводиться, а значить повинен бути алгоритм – як прийняти вірне рішення і таким чином повернути інвестиції, вкладені в програмне забезпечення.

Але, на жаль, алгоритму, який дозволив би з великою точністю сказати, що саме для цієї конкретної задачі найбільш оптимальним буде вибір конкретного типу програмного забезпечення, а витрати на нього будуть мінімальними, не існує. Тому прийняття рішень щодо вибору інструментальних засобів розробки здійснюється ситуативно, або у відповідності до знань про ці засоби людиною, яка обирає [2].

Постановка задачі

В силу доступності великої кількості різноманітних програмних інструментальних засобів розробки актуальною є спроба автоматизувати процес вибору таких інструментальних засобів. Для підтримки прийняття рішень щодо вибору інструментальних засобів розробки пропонується використати метод аналізу ієрархій Томаса Сааті.

Метод аналізу ієрархій

Метод аналізу ієрархій (МАІ) [3] полягає в декомпозиції проблеми на все більш прості складові частини і подальшого попарного порівняння складових частин на кожному наступному рівні ієрархії. У результаті може бути виражена відносна ступінь взаємодії

елементів в ієрархії. Ці судження потім виражаються чисельно. Метод включає процедури синтезу множинних суджень, отримання пріоритетності критеріїв і знаходження оцінок альтернативних рішень.

На першому етапі виявляються найбільш важливі елементи проблеми, на другому – найкращий спосіб перевірки спостережень і оцінки елементів; наступним етапом може бути розробка способу застосування рішення і оцінка його якості.

Ієрархія будується від вершини (цілі – з точки зору управління), через проміжні рівні (критерії, від яких залежать наступні рівні) до найнижчого рівня, який зазвичай є переліком альтернатив. Ієрархія вважається повною, якщо кожен елемент заданого рівня функціонує як критерій для всіх елементів рівня, що знаходиться нижче. В іншому випадку ієрархія є неповною.

У МАІ елементи задачі порівнюються попарно по відношенню до їх впливу на загальну для них характеристику.

Етапи МАІ:

1. Постановка завдання.
2. Побудова ієрархії.
3. Побудова набору матриць парних порівнянь для кожного з нижніх рівнів - по одній матриці для кожного елемента, що приймає до верхнього рівня.

Вирішальним перевагою акредитуючого підходу над більшістю існуючих методів оцінювання альтернатив являється його внесок в аналіз структури проблеми і чітке вираження суджень.

Побудова ієрархії критеріїв оцінки інтегрованих середовищ розробки

Нехай задача полягає у виборі між двома альтернативними варіантами інтегрованих середовищ розробки (для прикладу нехай це будуть Microsoft Visual Studio і NetBeans IDE). Обидві альтернативи мають як свої переваги, так і недоліки. При цьому одне середовище розробки являється платним, інше розповсюджується за відкритою ліцензією.

У відповідності до методу аналізу ієрархій на верхньому рівні ієрархії знаходиться ціль, а саме вибір інтегрованого середовища розробки. На наступному рівні знаходяться критерії досягнення цілі. Для спрощення виділимо тільки 5 наступних критеріїв:

- Графічний інтерфейс. Надзвичайно важлива характеристика будь-якої комп'ютерної програми. Саме інтерфейс визначає, наскільки зручним є даний засіб, і наскільки ефективно користувач може здійснювати роботу.
- Вартість покупки і підтримки. Один з визначальних критеріїв, що впливає на вибір інструментальних засобів розробки. Часто виступає головним критерієм. Визначає, скільки коштів буде витрачено на використання даного засобу.

- Форматування тексту програми. Доведено, що при якісному форматуванні тексту програми, а також підсвічуванні найбільш важливих компонентів програмного коду, ефективність роботи програміста підвищується. Тому це є важливим фактором роботи інтегрованого середовища розробки.
- Гарячі комбінації клавіш. Дозволяють пришвидшити доступ до найбільш використовуваних функцій програми, а отже пришвидшують розробку взагалі.
- Засоби відстежування залежностей. Важливий компонент інтегрованого середовища розробки, коли проект містить велику кількість залежностей між своїми компонентами. Оптимізує процес розробки через спрощення доступу до перегляду таких залежностей.

Для наглядності приведеного прикладу на наступному рівні ієрархії знаходяться можливі альтернативні варіанти вибору (рис. 1), а саме: Microsoft Visual Studio і NetBeans IDE.

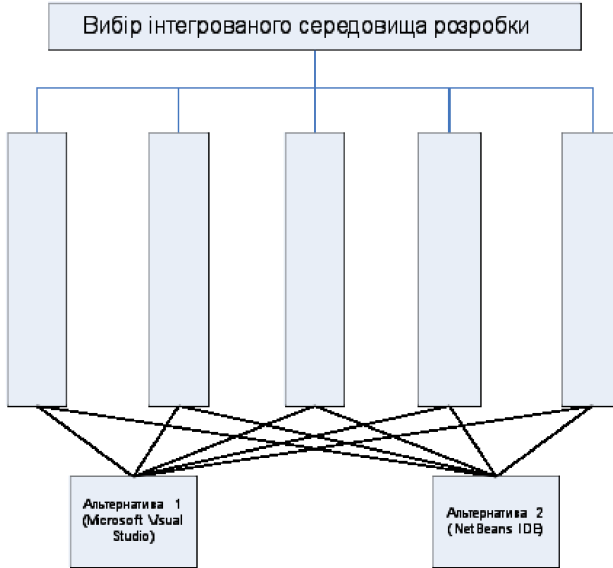


Рис. 1 – Схема методу аналізу ієрархій для задачі вибору інтегрованого середовища розробки програмного забезпечення

Закон ієрархічної безперервності вимагає, щоб елементи верхнього рівня ієрархії були порівнянні попарно по відношенню до елементів наступного рівня і т.д. аж до вершини ієрархії [3]. Наприклад, треба отримати змістовні відповіді на питання типу: “Наскільки альтернатива один краще альтернативи 2 за критерієм

графічного інтерфейсу?” Або “Наскільки по відношенню до основної мети критерій вартості покупки і підтримки важливіший за критерій графічного інтерфейсу?”.

Для проведення суб'єктивних парних порівнянь використовується шкала відносної важливості елементів по відношенню до загальної мети (табл. 1)

Таблиця 1

Шкала відносної важливості

Відносна важливість	Визначення
1	Рівна важливість
3	Помірна перевага одного над іншим
5	Суттєва або сильна перевага
7	Значна перевага
9	Дуже сильна перевага
2,4,6,8	Проміжне судження між двома сусідніми
Зворотні величини	Якщо при порівняння А і Б отримано одне з вищевказаних чисел x , то при порівнянні Б і А отримується зворотна величина $1/x$

Необхідно побудувати $n + 1$ матрицю попарних порівнянь.

Матриця попарних порівнянь для другого рівня задачі вибору інтегрованого середовища розробки (табл. 2) має розмірність 5 (по кількості критеріїв).

У відповідності до наведених вище варіантів альтернатив (Microsoft Visual Studio і NetBeans IDE) попарні порівняння для третього рівня задачі вибору інтегрованого середовища розробки наведено у таблиці 3.

Для групи матриць парних порівнянь формуються набори локальних пріоритетів, які виражають їх відносний вплив на елементи більш високого рівня.

Локальні пріоритети критеріїв L_i визначаються як:

$$L_i = \sqrt[n]{\prod_{j=1}^n x_{ij}} \quad (1)$$

Так, локальний пріоритет графічного інтерфейсу:

$$L_1 = \sqrt[5]{1 \cdot 1/7 \cdot 2 \cdot 1/3 \cdot 1/3} = 0.499 \quad (2)$$

Аналогічно $L_2 = 4.258$, $L_3 = 0.401$, $L_4 = 1.125$, $L_5 = 1.095$.

Після того як компоненти власного вектора отримані для всіх п'яти строк, стає можливим їх використання для подальших обчислень. Нормалізуємо їх для оцінки вектора пріоритетів по формулі:

Таблиця 2

Матриця попарних порівнянь другого рівня задачі

	Графічний інтерфейс	Вартість покупки і підтримки	Форматування тексту програми	Гарячі комбінації клавіш	Засоби відстежування залежностей
Графічний інтерфейс	1	7	1/2	3	3
Вартість покупки і підтримки	1/7	1	1/8	1/5	1/5
Форматування тексту програми	2	8	1	3	2
Гарячі комбінації клавіш	1/3	5	1/3	1	1
Засоби відстежування залежностей	1/3	5	1/2	1	1

Таблиця 3

Матриці попарних порівнянь третього рівня задачі вибору інтегрованого середовища розробки

Графічний інтерфейс	1	2	Вартість покупки і підтримки	1	2
Альтернатива 1	1	1/3	Альтернатива 1	1	5
Альтернатива 2	3	1	Альтернатива 2	1/5	1
Форматування тексту програми	1	2	Гарячі комбінації клавіш	1	2
Альтернатива 1	1	1/2	Альтернатива 1	1	1
Альтернатива 2	2	1	Альтернатива 2	1	1
Засоби відстежування залежностей	1	2			
Альтернатива 1	1	1/3			
Альтернатива 2	3	1			

$$\|L_i\| = \frac{L_i}{\sum_{j=1}^n L_j} \quad (3)$$

$$\sum_{j=1}^n L_j = 7.378 \quad (4)$$

Відповідно знаходимо оцінки для всіх п'яти параметрів оцінки інтегрованих середовищ розробки:

$$\|L_1\| = 0.068, \|L_2\| = 0.577, \|L_3\| = 0.054, \|L_4\| = 0.152, \|L_5\| = 0.148.$$

Після цього аналогічні дії виконуємо для п'яти матриць парних порівнянь третього рівня, кожна з яких відповідає одному з параметрів (табл. 4).

Таблиця 4

Нормовані оцінки альтернатив

	Оцінка Альтернативи 1	Оцінка Альтернативи 2
Графічний інтерфейс	$\ L_{11}\ = 0.75$	$\ L_{12}\ = 0.25$
Вартість покупки і підтримки	$\ L_{21}\ = 0.167$	$\ L_{22}\ = 0.883$
Форматування тексту програми	$\ L_{31}\ = 0.667$	$\ L_{32}\ = 0.333$
Гарячі комбінації клавіш	$\ L_{41}\ = 0.5$	$\ L_{42}\ = 0.5$
Засоби відстежування залежностей	$\ L_{51}\ = 0.75$	$\ L_{52}\ = 0.25$

Тепер можна розрахувати глобальні пріоритети для вирішуваної задачі відповідно до наступної формули:

$$G_1 = \sum_{i=1}^n \|L_i\| \cdot \|L_{i1}\|, G_2 = \sum_{i=1}^n \|L_i\| \cdot \|L_{i2}\|, \text{ де } n = 5 \quad (5)$$

Таблиця 5

Глобальні пріоритети альтернатив 1 і 2

Параметр оцінки альтернативи	$\ L_i\ $	$\ L_{i1}\ $	$\ L_{i2}\ $
Графічний інтерфейс	0.068	0.75	0.25
Вартість покупки і підтримки	0.577	0.167	0.883
Форматування тексту програми	0.054	0.667	0.333
Гарячі комбінації клавіш	0.152	0.5	0.5
Параметр оцінки альтернативи	$\ L_i\ $	$\ L_{i1}\ $	$\ L_{i2}\ $
Засоби відстежування залежностей	0.148	0.75	0.25
Узагальнений пріоритет G		0.37	0.63

Таким чином, отримуємо, що використання NetBeans IDE значно прийнятніше за Microsoft Visual Studio, а найбільш важливим критерієм для прийняття такого рішення являється вартість покупки і підтримки.

Аналогічні розрахунки можна легко здійснити для будь-якого іншого набору критеріїв порівняння інтегрованих середовищ розробки і для довільної множини порівнюваних об'єктів.

Висновки

Запропонований варіант застосування методу аналізу ієрархій до процесу вибору інтегрованих середовищ розробки дозволяє оцінити відносні пріоритети вибору альтернатив у відповідності до встановлених пріоритетів критеріїв вибору. Подібний метод розрахунків може бути використаним, наприклад, викладачем при виборі інтегрованих середовищ розробки програмного забезпечення для виконання комп'ютерного практикуму з дисципліни “програмування”.

Список використаних джерел

1. Інформаційні технології та системи. Підручник. / Плещак В.Л., Рогушина Ю.В., Кустова Н.П. // К.: Книга, 2004. – 520 с.
2. Теория и методы принятия решений / Ларичев О.И. – М.: Логос, 2000. – 296 с.
3. Аналитическое планирование. Организация систем. / Саати Т., Керне К. – М.: Радио и связь, 1991. – 224 с.

Отримано 02.09.2014 р.