

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА
АКАДЕМІЯ»

Факультет інформатики
Кафедра мультимедійних систем

**Експертна оцінка альтернатив на основі попарних порівнянь та методу
аналізу ієрархій**

Керівник курсової роботи
доц., к. н. Олецкий О.В

(підпис)

“ ___ ” _____ 2024 р.

Виконала студентка

Дехтяренко М.С

“ ___ ” _____ 2024 р.

Київ - 2024р

Тема: Експертна оцінка альтернатив на основі попарних порівнянь та методу аналізу ієрархій

Календарний план виконання роботи:

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу	30.07.2023	
2.	Аналіз матеріалів за темою	23.08.2023- 17.10.2023	
3.	Написання текстової частини до курсової роботи	04.04.2024- 10.05.2024	
4.	Розробка застосунку.	04.04.2024- 10.05.2024	
5.	Захист курсової роботи.	20.05.2024	

Студентка :Дехтяренко Мар'яна Сергіївна

Керівник: Олецький Олексій Віталійович

“ _____ ”

Зміст

<i>Перелік скорочень</i>	4
<i>Анотація</i>	5
<i>Вступ</i>	6
<i>РОЗДІЛ 1</i>	9
1.1 Суть та основні положення МАІ.....	9
<i>РОЗДІЛ 2</i>	17
1.1 Переваги	17
1.2 Недоліки	18
1.3 Усунення недоліків у МАІ.....	20
<i>Розділ 3 Опис та реалізація веб-застосунку</i>	25
<i>Висновки</i>	41
<i>Джерела</i>	42

Перелік скорочень

MAI- Метод аналізу ієрархій

PC- Pairwise Comparison

МПП- Матрицях попарних порівнянь

АНР- Analytic hierarchy process

CI- Concordance Index

RI- Random Value Index

Анотація

У курсовій роботі досліджується оцінка альтернатив на основі попарних порівнянь та методу аналізу ієрархій. Також розглядаються основні принципи та етапи МАІ, включаючи побудову матриць попарних порівнянь, визначення вагових коефіцієнтів для кожного критерію та альтернативи, а також розрахунок індексу узгодженості для перевірки консистентності. Основна увага приділяється використанню попарних порівнянь для визначення відносної важливості кожної альтернативи щодо відповідних критеріїв. Робота розглядає теоретичні аспекти МАІ, його застосування в управлінні та прийнятті рішень, а також його практичне застосування в конкретних ситуаціях для оцінки альтернатив. Результати дослідження показують, що МАІ є ефективним інструментом для експертної оцінки альтернатив і може бути успішно використаний для прийняття обґрунтованих рішень у різних галузях.

Вступ

Нерідко в житті ми стикаємося з прийняттям важливих рішень, які вимагають об'єктивного та обґрунтованого підходу. Прийняття таких рішень може бути складним завданням через різноманітні фактори, які звичайна людина не завжди може або хоче враховувати. Чи то в бізнесі, де вибір стратегії може визначити майбутній успіх компанії, або в особистому житті ефективний процес оцінки альтернатив може мати суттєвий вплив на майбутнє.

Стрес, обмежений час та велика кількість варіантів можуть перешкоджати та негативно впливати на процес прийняття об'єктивного рішення. Наприклад, коли ми вибираємо, яку футболку одягнемо сьогодні, навіть якщо наше рішення не буде найкращим з можливих це не сильно вплине на наше життя, але коли мова йде про величезні корпорації, то погане рішення керівництва може призвести до скорочення великої кількості персоналу, що очевидно матиме негативні наслідки не лише на репутації цієї компанії, але й на життя її працівників. У таких ситуаціях нам потрібні інструменти, які допомагатимуть структурувати процес прийняття рішень та об'єктивно порівнювати всі існуючі альтернативи.

Метод аналізу ієрархій і попарних порівнянь, розроблений американським вченим Томасом Л. Сааті в 1970-х роках і заснований на математиці та психології, виявляється надзвичайно корисним у цьому контексті. Сааті навчався в Колумбійському союзному коледжі та Католицькому університеті Америки. Певний час працював під керівництвом математика Джорджа Г. О'Брайєна, де він вивчав вплив на рішення лінійного програмування збурень даних, що і спонукало його до захоплення операційними дослідженнями. Після цього досвіду він почав досліджувати прийняття рішень і створив метод аналізу ієрархій.

Метод базується на трьох основних принципах: розкладання складної проблеми на різні елементи, порівняння цих елементів для визначення їх впливу на ціле, і синтез для формування розуміння зібраної інформації. Вони дозволяють

систематизувати складні рішення, виокремлюючи ключові критерії та важливість кожної альтернативи. Такий підхід дозволяє досягти об'єктивності у процесі прийняття рішень та зменшує ймовірність поганих виборів, особливо коли вибір є важливим, як у випадку багатомільйонних компаній. Надійний і обґрунтований вибір найкращої з альтернатив є складним завданням у контексті постійної зміни умов і обмежень.

Метою нашого дослідження є розробка системного підходу до експертної оцінки альтернатив, що базується на попарних порівняннях та методі аналізу ієрархій. Наша ідея полягає в тому, щоб надати практичні поради щодо прийняття найкращих рішень у різних сферах діяльності.

Результат цього дослідження зосереджується на перевагах та недоліках застосування методу аналізу ієрархій, а також на потенційних проблемах, з якими користувачі можуть зіштовхнутися під час його використання. Для підтвердження та уточнення цих аспектів було проведено відповідні експерименти, а також були запропоновані методи уникнення або вирішення цих проблем.

Робота складається з трьох розділів:

Перший розділ розповість про основи методу аналізу ієрархій, беручи до уваги опис методу, його алгоритм і загальну інформацію про застосування.

Другий розділ буде присвячений аналізу методу з попарним порівнянням альтернатив, де будуть розглянуті переваги та недоліки використання цього методу. Також будуть описані алгоритми та стратегії, які допоможуть збалансувати й усунути недоліки під час прийняття рішень на основі попарних порівнянь.

У третьому розділі буде представлено практичну реалізацію методу аналізу ієрархій з використанням попарних порівнянь, також буде описана програма, спрямована на застосування методу аналізу ієрархій та попарних порівнянь для оцінки альтернатив.

Постановка задачі:

1. Провести дослідження методу аналізу ієрархій з використанням попарних порівнянь для експертної оцінки альтернатив у прийнятті рішень.

2. Визначити переваги та недоліки використання методу аналізу ієрархій з попарним порівнянням альтернатив.

3. Розробити власний застосунок, який робить експертну оцінку альтернатив, яка базується на попарних порівняннях і аналізі ієрархій.

4. Оцінити результати експериментів та зробити висновки щодо ефективності та застосовності методу аналізу ієрархій з попарним порівнянням альтернатив у ситуаціях прийняття рішень.

Метою дослідження є розробка та оцінка системного підходу до прийняття об'єктивних і обґрунтованих рішень, використовуючи метод аналізу ієрархій та попарних порівнянь.

Об'єктом дослідження є спосіб дії та ефективність методу аналізу ієрархій, зокрема його алгоритми та стратегії застосування у процесах прийняття рішень.

Предметом дослідження є метод аналізу ієрархій, його властивості, можливості та обмеження.

Структура та обсяг роботи: робота викладена на 44 сторінках, містить 1 додаток, 27 рисунків, 3 таблиці та 6 формул.

РОЗДІЛ 1.

Теоретичні засади Методу аналізу ієрархій

1.1 Суть та основні положення МАІ

Прийняття рішень є не лише необхідною, але й надзвичайно важливою частиною нашого життя. Кожного дня ми стикаємося з безліччю виборів, які потрібно зробити, починаючи від найпростіших щоденних рішень, таких як вибрати їсти на сніданок, і закінчуючи важливими особистими, професійними чи фінансовими рішеннями. Те які рішення ми приймаємо, і як саме ми це робимо, може впливати на кожен аспект нашого життя - від нашої повсякденної рутини до професійного успіху.

Зазвичай цей процес є досить простим, оскільки ми вибираємо між обмеженою кількістю альтернатив, попарно порівнюючи декілька варіантів, які мають визначені критерії. У більшості випадків це стосується простих рішень, які ми робимо щодня. Наприклад, коли викладач має поставити оцінку за певну роботу учню і використовує шкалу оцінок, яка є визначеною і зрозумілою для всіх у навчальному закладі. Вчителі оцінюють роботу учнів за допомогою шкали від "1" (не задовільно) до "12" (відмінно) або за іншими системами, такими як "F" до "A". Ці системи допомагають визначити рівень успішності учнів та їхні досягнення у навчанні. Хоча ці шкали можуть значно відрізнятися в різних країнах, але ми все одно підсвідомо розуміємо, як вони працюють і поставити оцінку учню або студенту не викликає у нас жодних труднощів.

Однак у деяких ситуаціях процес прийняття рішення буде суттєво ускладнений, особливо коли маємо багато альтернатив і не маємо жодного способу оцінити ці критерії чи варіанти за вже існуючими метриками. Це не

обов'язково має бути якимось складним чи глобальним рішенням, як наприклад розгляд переїзду до іншої країни або вибір професії чи кар'єрного шляху. Це може бути щось легке чи несуттєве, як наприклад вибір який серіал подивитися на ніч чи яку книгу купити в магазині тощо, але ми все одно можемо витратити години нашого часу, щоб вирішити, яка з альтернатив буде найкращою.

Незважаючи на складність прийняття нашого рішення, ми все одно проводимо певні "розрахунки" у своїй голові, робимо невеличкий аналіз відповідно до визначених критеріїв і надаємо певну вагу та пріоритетність кожному з варіантів, які ми розглядаємо. Крім того ми також виступаємо в ролі експертів, щоб оцінити власні критерії та надати перевагу одному з рішень на основі несвідомо визначених нами ваг. Часто наша оцінка не є об'єктивною через наші ж упередження або неправильно визначені пріоритети і похибка може коливатися від незначної до значущої, тому важливо використовувати структуровані методи прийняття рішень, щоб максимально зменшити недоліки, які виникають через вплив людського фактору.

Для цього у сьогоденні часто використовується аналітичний ієрархічний процес, розроблений у 1970 році Томасом Л. Сааті, він є одним із найбільш використовуваних методів у багатокритеріальному аналізі. Використовуючи об'єктивні та суб'єктивні фактори, такі як досвід, знання та інтуїція осіб, що приймають рішення, він дозволяє вирішувати складні питання прийняття рішень.

АНР виявляється особливо корисним у випадках, коли ситуація є дуже напруженою і прийняте рішення матиме величезні наслідки на подальше майбутнє. Цей метод допомагає знайти оптимальне рішення, яке найкраще відповідає потребам та цілям особи або організації, яка його використовує. Він дозволяє уникнути упередженості та забезпечує об'єктивність у процесі прийняття рішень, що робить його популярним інструментом у бізнесі, науці, громадському управлінні та інших сферах.

Даний метод ґрунтується на концепції декомпозиції складних проблем на менш складні компоненти, їх порівнянні та синтезуванні з метою досягнення оптимального рішення. Таким чином, МАІ моделює проблему у вигляді певної ієрархічної структури. Ця структура може включати постановку такого питання: «Яку книжку мені вибрати?». Після цього нам, потрібно визначити критерії та підкритерії, якщо вони існують, за якими буде проводитися вибір між різними альтернативами, а також очевидно надати ці альтернативи. Після того як ми створили нашу ієрархічну структуру, метод визначає відносні пріоритети альтернатив відповідно до бажань експертів або тих, хто приймає рішення. На кожному рівні ієрархії особа, що приймає рішення, повинна проводити порівняння критеріїв та альтернатив.

Таким чином МАІ можна поділити на п'ять етапів:

- Чітке визначення проблеми та її представлення у вигляді певної ієрархії.
- Створення бінарної матриці порівняння та визначення вагових коефіцієнтів
- Обчислення власного вектора
- Обчислення кількісної міри
- Визначення рейтингу альтернатив

Як зазначалося раніше першим кроком в АНР є чітке визначення мети, критеріїв та альтернатив. Далі ми починаємо моделювати рішення, тобто створювати ієрархію. На вершині ієрархії знаходиться особа, яка приймає рішення і має чітку мету, яку потрібно досягти. У нашому випадку метою є вибір найкращої книги. Автор, жанр, рейтинг і кількість сторінок складають другий рівень ієрархії. На останньому рівні у нас наведено шість альтернатив книг: книга1, книга2, книга3, і т. д. . За допомогою такої структури можна легко

проводити порівняння між критеріями та альтернативами, визначеними для досягнення мети.

Сааяті визначав три наступні пункти при створенні ієрархії:

- Проблему слід пояснити чітко і точно.
- Проблема повинна охоплювати всі фактори.
- Слід враховувати всю інформацію, яка вплине на результат. [1]

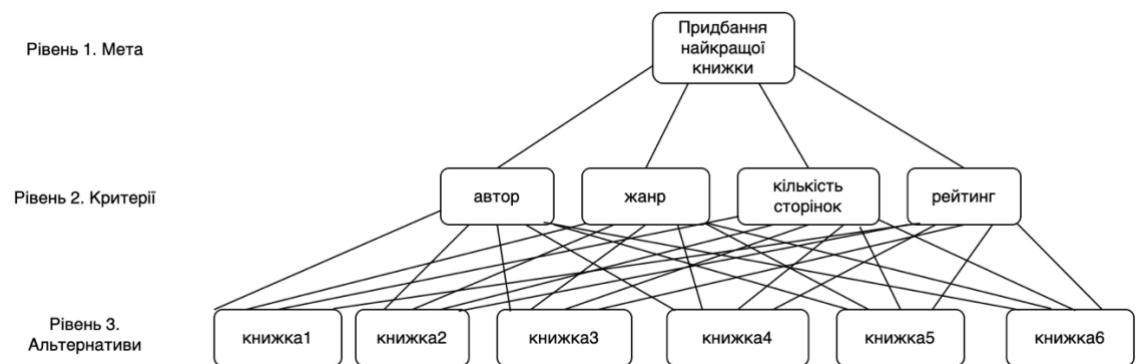


Рисунок 1.1 Приклад створення ієрархії

Після створення ієрархії експерти повинні порівняти кожну пару критеріїв, щоб визначити їхню відносну важливість. У методі аналізу ієрархій після визначення ієрархічної структури критерії порівнюються з точки зору їхнього впливу на досягнення поставленої мети. Експерти з відповідної галузі визначають ступінь важливості кожного критерію шляхом порівняння на кожному рівні ієрархії. У матриці з n елементів проводиться $n(n-1)/2$ порівнянь.

	Автор	Жанр	Кількість сторінок	Рейтинг
Автор	X ₁₁	X ₁₂	X ₁₃	X ₁₄
Жанр	X ₂₁	X ₂₂	X ₂₃	X ₂₄
Кількість сторінок	X ₃₁	X ₃₂	X ₃₃	X ₃₄
Рейтинг	X ₄₁	X ₄₂	X ₄₃	X ₄₄

Таблиця 1.1 Матриця попарних порівнянь

Наприклад, критерії “автор” і “жанр” (табл.1) вимагають оцінки «який з цих двох критеріїв важливіший і наскільки важливий для досягнення цілі». МПП також є обернено-симетричною, тому буде виконуватися формула $x_{ij} = 1/x_{ji}$. Зважаючи на те, що кожен критерій або альтернатива будуть рівноцінними самі собі, то всі діагональні елементи дорівнюватимуть одиниці.

Після встановлення ієрархії необхідно розрахувати, наскільки критерії важливі один в порівнянні з іншим, яка їх відносна важливість. Особа, яка приймає рішення, зазвичай використовує шкалу, щоб визначити ступінь важливості критерію, використовуючи оцінки від 1 до 9. Метод порівняння двох елементів з використанням АНР може бути реалізований різними способами. Проте, шкала відносної важливості, яка була запропонована Сааті, є найбільш розповсюдженою. Ця шкала присвоює значення від 1 до 9, щоб визначити відносну важливість одного елемента в порівнянні з іншим.

Відносна важливість	Визначення	Пояснення
1	Однакова важливість	Внесок альтернатив є однаковим
3	Один трохи важливіший за інший	Досвід та судження сприяє поставити один елемент трохи вище за другий
5	Суттєва перевага	Суттєва перевага одного над другим
7	Значна перевага	Перша альтернатива надзвичайно важлива в порівнянні з іншою
9	Абсолютна перевага одного над іншим	Очевидність переваги альтернативи підтверджується більшістю
2,4,6,8	Проміжні оцінки між сусідніми твердженнями	Компромісне рішення

Таблиця 2.1 Шкала порівнянь Сааті

Тобто експерт має оцінити значимість кожного з критеріїв у порівнянні з іншими критеріями щодо його впливу на основний критерій вищого рівня в ієрархії, для якого складається матриця.

Розрахунок векторів локальних пріоритетів є важливою частиною процесу аналізу ієрархій. Ці вектори визначають відносну важливість кожного критерію або альтернативи на певному рівні ієрархії. Після того, як парні матриці порівнянь були визначені, ми розглядаємо кожну матрицю окремо і знаходимо вектор, який відповідає найбільшому власному значенню матриці. $XW = \lambda W$. У даному випадку λ - це розраховане найбільше власне значення матриці. А W - це власний вектор матриці. У випадку, коли власний вектор нормований, тобто сума його елементів рівна одиниці, то він відповідно є вектором пріоритетів для матриці X . Якщо ж його сума перевищує одиницю, тоді ми маємо внормувати вектор, для цього нам потрібно утворити новий вектор W' елементи якого є відношенням елементів вектора W на суму всіх його елементів.

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ x_{n1} & x_{n2} & \dots & x_{nn} \end{bmatrix} \begin{bmatrix} W_1 \\ \dots \\ W_n \end{bmatrix} = \lambda \begin{bmatrix} W_1 \\ \dots \\ W_n \end{bmatrix}$$

Рисунок 2.1 Власний вектор матриці

Після проведення всіх вищезгаданих операцій перевіряється узгодженість для всіх матриць, адже визначені коефіцієнти та оцінки експертів можуть мати різні похибки від яких матриця стане неузгодженою. Для цього в кожній матриці попарного порівняння розраховується коефіцієнт неузгодженості, який має становити не більше 10%. Якщо коефіцієнт виходить за можливі рамки

неузгодженості, то тоді рекомендується переглянути елементи парного порівняння матриці X , щоб отримати більш узгоджену матрицю.

Для того, щоб визначити рівень узгодженості матриці розраховується коефіцієнт узгодженості. Щоб розрахувати цей коефіцієнт ми спочатку маємо знайти стохастичний коефіцієнт матриці X . Формула за якою обчислюється $RI = 1.98(n-2)/n$ не є точною оскільки визначається емпірично, як середнє значення CI для великої вибірки генерованих випадковим чином матриць порівнянь X . Замість нього ми можемо використати вже розраховані значення індексу рандомізації.

Розмір матриці(n)	3	4	5	6	7	8	9	10
Табличний індекс	0	0	1	1	1	1	1	1
	.52	.89	.11	.25	.35	.40	.45	.49

Таблиця 3.1 Значення індексу рандомізації

Далі ми повинні обчислити значення λ_{\max} . Його можна визначити шляхом обчислення вектора-стовпця XW' і після знайти суму його елементів.

Для того, щоб отримати коефіцієнт узгодженості матриць, ми також маємо дізнатися коефіцієнт узгодженості суджень експертів за цією формулою $CI = \lambda_{\max} - n/n - 1$. Після розрахунку CI , ми обчислюємо коефіцієнт узгодженості CR , який є відношенням CI до RI .

Після перевірки відповідності стандартам коефіцієнта узгодженості потрібно повторити всі ці процедури для кожної запропонованої альтернативи щодо всіх критеріїв. Цей процес дозволяє оцінити альтернативи та провести аналіз, щоб дізнатися, чи відповідають вони поставленій меті. Такий метод допомагає визначити найкращий вибір, враховуючи всі фактори.

РОЗДІЛ 2.

Переваги та недоліки застосування методу аналізу ієрархій з попарним порівнянням альтернатив.

1.1 Переваги

Сьогодні метод аналізу ієрархій є одним з найпопулярніших способів прийняття рішень у багатьох галузях, від менеджменту до повсякденного життя. Він вважається революційним завдяки роботі Томаса Сааті, який розробив та вперше впровадив цей метод. МАІ не просто є популярним методом для прийняття рішень, він широко застосовується по всьому світу, допомагаючи зробити складні вибори не лише в повсякденному житті, а й на державному та комерційному рівнях. Його популярність можна пояснити його системним та структурованим підходом до аналізу складних проблем. МАІ дозволяє розбити складну задачу на менші частини, враховуючи важливість кожної з них та їх взаємозв'язок. Цей метод допомагає уникнути суб'єктивності та недоліків які існують у інших методах прийняття рішень, а також забезпечує об'єктивні та зважені результати.

Метод аналізу ієрархій відрізняється здатністю глибоко аналізувати відносну важливість кожного критерію та їх взаємозв'язок у порівнянні з іншими методами прийняття рішень. Наприклад, у сфері бізнесу, при розгляді стратегічних рішень щодо розширення ринків, метод дозволяє не лише врахувати фінансові аспекти, але й оцінити вплив на репутацію компанії та ризики. Подібний глибокий аналіз критеріїв може визначити найбільш оптимальний шлях дії.

Крім того, аналіз ієрархій дозволяє зберігати структурованість та об'єктивність у прийнятті рішень, навіть у випадку, коли група експертів має різні точки зору. Наприклад, коли ми розглядаємо вибір місця для будівництва нового заводу, МАІ може систематично порівняти різні фактори, такі як вартість землі,

доступність робочої сили та екологічні ризики, і забезпечити об'єктивність у виборі оптимального місця.

Навчання та використання методів аналізу ієрархій не вимагає значного попереднього досвіду або знань. Навіть люди, які поверхнево ознайомлені з ним можуть швидко освоїти його основи і застосовувати його для вирішення різноманітних завдань. Таким чином, МАІ не тільки дозволяє проводити ретельний аналіз і зберігати об'єктивність під час прийняття рішень, але він також доступний і простий у використанні навіть для початківців.

Більш того, АНР дозволяє враховувати різні типи критеріїв, що можуть бути як і числовими так і просто словами. Це надає методу суттєву перевагу над іншими способами прийняття рішень, де складно або неможливо виміряти всі параметри числами.

Здатність перевіряти узгодженість є наступною важливою перевагою методу. Це означає, що МАІ може виявити розбіжності в оцінках експертів або помилки в їхніх рішеннях. Також певні експерти можуть свідомо махлювати оцінками надаючи певному критерію суттєву перевагу над іншими, що може негативно вплинути на кінцевий результат.

Отже, МАІ до сих пір залишається потужним інструментом, який не тільки допомагає уникнути суб'єктивності деяких експертів, а й спрощує процес прийняття рішень для широкого кола користувачів, незалежно від їхнього рівня знань.

1.2 Недоліки

Незважаючи на всі переваги, АНР не є ідеальним методом прийняття рішень. Він має недоліки, які варто враховувати, щоб використовувати його ефективно.

МАІ порівнює кожен альтернативу за кожним критерієм, а також критерії між собою. Це означає, що порівняння у випадку, коли у нас є велика кількість критеріїв або альтернатив може бути дуже складним процесом і вимагати багато часу та зосередженості. Наприклад, якщо у вас є десять критеріїв і десять альтернатив, вам потрібно буде провести сто порівнянь, щоб визначити відносну вагомість кожної альтернативи. Крім того, ви повинні порівняти всі критерії один з одним. Це може призвести до значних затримок у прийнятті рішень і вимагати від користувачів значних зусиль. Експерти можуть втомитися та втратити увагу під час тривалого процесу аналізу, що може вплинути на якість порівнянь і об'єктивність результатів. Швидке прийняття рішень або недостатня увага до деталей можуть призвести до недостовірних результатів.

Шкала Сааті, яка використовує лише дев'ятибальну систему, при великій кількості альтернатив може бути дещо обмеженою, оскільки не вистачатиме можливостей для достатньої диференціації окремих елементів. Це може призвести до узагальнень або недостатньо точних порівнянь, що швидше за все не дасть нам узгоджену матрицю порівнянь і гарні результати відповідно будуть майже неможливими.

Іншим недоліком методу аналізу ієрархій є проблема узгодженості, яка є досить поширеною при використанні даного методу. Якщо матриця попарних порівнянь матиме розмір більший ніж 3×3 є висока ймовірність, що вона не буде узгодженою. Постійне узгодження матриць потребує виконання зайвих кроків, щоб перевірити чи коефіцієнт узгодженості не перевищує більше 10 %. Відповідно метод таким чином ускладнюється і це забирає багато часу. Також у випадку, коли CR перевищив задану міру ми повинні переглянути створені матриці попарних порівнянь та у разі необхідності змінити коефіцієнти, що не полегшує процес роботи.

Це також стосується додавання нових альтернатив до моделі або видаленні старих, оскільки може змінитися пріоритетний порядок інших альтернатив. Така

операція може відбутися без зміни значень попарних порівнянь для окремих критеріїв, що в подальшому призведе до неточних результатів або неузгоджених матриць. Хоча на перший погляд може здатися, що введення або вилучення альтернатив з оцінюваної моделі є малоімовірним або безглуздим, на практиці така потреба може виникнути. Наприклад, нові альтернативи можуть виникнути через зміну обставин або додаткову інформацію, що стала доступною з часом. Також може знадобитися видалити деякі альтернативи через зміну умов або недоступність деяких варіантів.

Отже, важливо усвідомлювати, що хоча МАІ і є потужним інструментом для прийняття рішень, він може займати значну кількість часу та вимагати великої концентрації уваги з боку користувачів, особливо при роботі з великими обсягами даних або редагуванням вже існуючих даних.

1.3 Усунення недоліків у МАІ

З наведеного аналізу переваг та недоліків МАІ можна зробити висновок, що метод має значні переваги у системному аналізі складних проблем, але також супроводжується певними недоліками. Один з найбільш важливих недоліків полягає у неузгодженості матриць попарних порівнянь. Тобто у нас є постійна необхідність узгоджувати матриці парних порівнянь, крім випадків, коли використовується нормалізована матриця з однаковими стовпцями. Проте, нормалізована матриця є більше винятком із правил ніж самим правилом і зазвичай спостерігається лише у випадках, коли ми маємо не більше двох критеріїв. Матриці розміру менше 3 на 3 є відносно рідкісними, оскільки навіть звичайні користувачі зазвичай працюють з багатьма критеріями, що вже можна говорити про великі компанії або проекти, які зазвичай і користуються цим методом. І це створює необхідність у постійному узгодженні матриць у реальних

задачах застосування МАІ. Такий процес узгодження вимагає додаткових зусиль та уваги з боку користувача, що може ускладнити процедуру використання методу та знизити його ефективність.

Наукові дослідження в галузі методу аналізу ієрархій свідчать про пошук шляхів для підвищення узгодженості матриць парних порівнянь. Одним із таких підходів є використання розв'язків систем лінійних алгебраїчних рівнянь. Цей метод дозволяє ефективно покращити узгодження матриць та забезпечити більш точні результати в процесі прийняття рішень за методом аналізу ієрархій.

Ми розглянемо цю техніку на прикладі однорівневого МАІ, коли альтернативи порівнюються загально, без занурення в конкретні характеристики або критерії, які впливають на оцінки.

Функція ранжування у МАІ використовується для порівняння альтернатив та отримання їх ранжування на основі матриці попарних порівнянь. Іноді, коли є складність у порівнянні абсолютних значень переваг між альтернативами, застосовується логарифмічне представлення МПП. В таких випадках, замість безпосередніх оцінок переваг, у матриці фігурують їх логарифми за певною основою, що спрощує порівняння ступенів переваг.

Логарифмічне представлення матриць відображається у контексті транзитивних шкал, що дозволяє визначити градацію переваг між альтернативами відносно до базового рівня. Цей метод допомагає оцінити не лише наявність переваги, але й її важливість у порівнянні з іншими альтернативами. Такий підхід є дуже важливим у прийнятті рішень, де потрібно оцінити не лише наявність переваги, а й її роль у контексті інших альтернатив.

Ступінь переваги однієї альтернативи над іншою можна визначити, порівнюючи кількість цілих градацій між ними. Повні МПП, до якої застосовано транзитивну шкалу, демонструється на рисунку 3.2[5]

$$\begin{pmatrix} 1 & \tau & \tau^3 \\ \frac{1}{\tau} & 1 & \tau^2 \\ \tau & \frac{1}{\tau} & 1 \\ \frac{1}{\tau^3} & \frac{1}{\tau^2} & 1 \end{pmatrix}.$$

Рисунок 3.2 Транзитивна шкала

Тобто, якщо s_{ij} визначає перевагу i -го варіанту над j -м у логарифмічній формі, тоді градації переваг можна виразити через логарифмічну шкалу, де кожна наступна градація більша від попередньої в задане число разів. Це дозволяє розробляти матриці парних порівнянь, які є більш узгодженими і відображають тонші відмінності між альтернативами, спираючись на більш об'єктивне та зважене прийняття рішень.

В матрицях попарних порівнянь, представлених у логарифмічному форматі, обов'язково має бути дотримане правило: для будь-якого індексу i , s_{ij} має дорівнювати нулю, що вказує на відсутність переваги та означає рівність між альтернативами, або s_{ij} має дорівнювати $-s_{ji}$ для всіх індексів i та j . Це свідчить про необхідність збереження консистентності в оцінках.

У методі аналізу ієрархій також надзвичайно важливо враховувати узгодженість матриць попарних порівнянь. Невідповідності, які можуть виникати через суб'єктивні судження експертів, потребують детального аналізу та коригування, щоб уникнути некоректних висновків. Оцінка узгодженості можлива за допомогою індексу узгодженості та інших аналітичних методів.

На практиці ж можливо створити ідеально узгоджену матрицю за допомогою Перронового вектору, навіть якщо вхідні дані були випадковими. Він є

вектором, який відповідає максимальному за модулем власному значенню матриці і використовується для обчислення вагових коефіцієнтів альтернатив або критеріїв. Це дає змогу отримати нормалізовані ваги, які є кращими оцінками у порівнянні з суб'єктивними вхідними даними.

$$v = (v_1, \dots, v_n)$$

Рисунок 4.2 Перроновий вектор

Якщо матриця є ординально узгодженою, це підтверджує її транзитивність у відношеннях переваг між альтернативами.

Розглянувши структуру матриць та можливості їх удосконалення через лінійні алгебраїчні рівняння, важливо розуміти, що правильність системи також залежить від точності вхідних даних. Система рівнянь, що містить як експертні оцінки, так і вимоги до узгодженості, може бути перевизначеною, і її розв'язок здійснюється за допомогою псевдооберненої матриці Мура–Пенроуза.

Псевдообернена матриця корисна для знаходження псевдорозв'язків у випадках, коли система рівнянь є несумісною або перевизначеною. Оскільки псевдообернена матриця забезпечує рішення з мінімальною нормою нев'язки, що робить його найбільш логічним у таких обставинах. Однак, у випадках, коли система має багато можливих розв'язків, розв'язок з мінімальною нормою може бути не оптимальним. Додатково, можна призначати ваги рівнянням, відображаючи міру довіри до відповідних суджень. Це сприяє отриманню більш точних і обґрунтованих рішень, враховуючи значимість кожного рівняння у системі. Після розв'язання системи рівнянь буде створено нову матрицю попарних порівнянь з елементами $m_{ij} = \tau^{x_{ij}}$, доповнену відповідно до вимог оберненої симетричності. Це забезпечить узгодженість і точність отриманих результатів.

Також, у методі аналізу ієрархій важливо розглядати питання узгодженості МПП, яка може бути оцінена за допомогою індексу узгодженості та інших методів. І потрібно не забувати звертати увагу на невідповідності в судженнях експертів, оскільки вони можуть призводити до порушення порядку в оцінках, що може призвести до неточних або поганих рішень.

Розділ 3 Опис та реалізація веб-застосунку

Для цієї курсової роботи було розроблено веб-застосунок- бібліотеку, яка має схожі риси з інтернет-магазином книг.

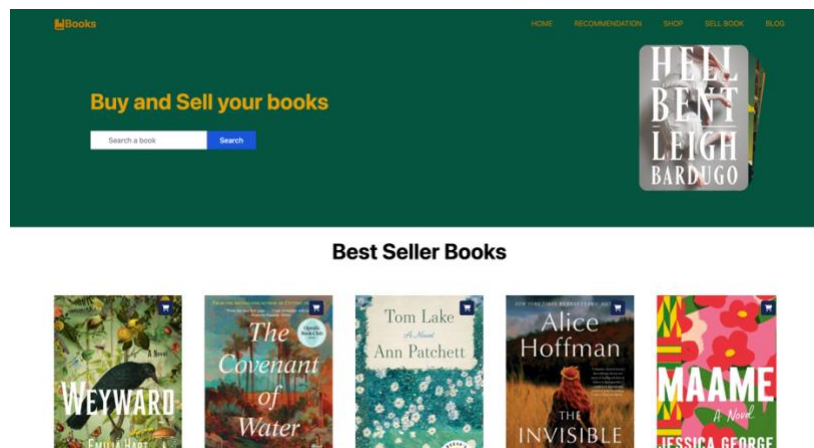


Рисунок 5.3 Головна сторінка сайту

На головній сторінці знаходяться каталог з найпопулярнішими книгами, а також наявний банер з пропозицією "Знайдіть вашу улюблену книгу тут", який при натисненні на кнопку "Explore more" сприяє переходу до каталогу з всіма книжками, що були завантажені на сайт.

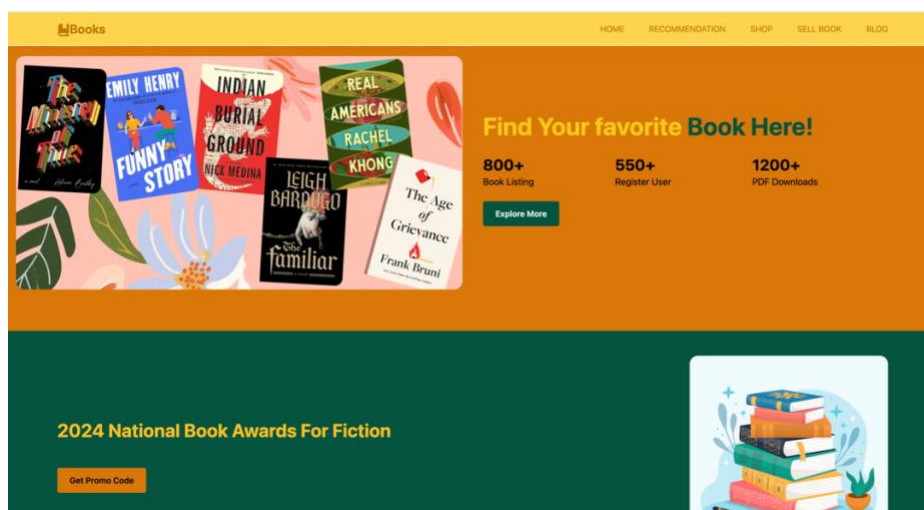


Рисунок 6.3 Банер "Знайдіть вашу улюблену книгу тут"

Звісно, як і в більшості книжкових, у застосунку ви можете побачити різні відгуки від користувачів.

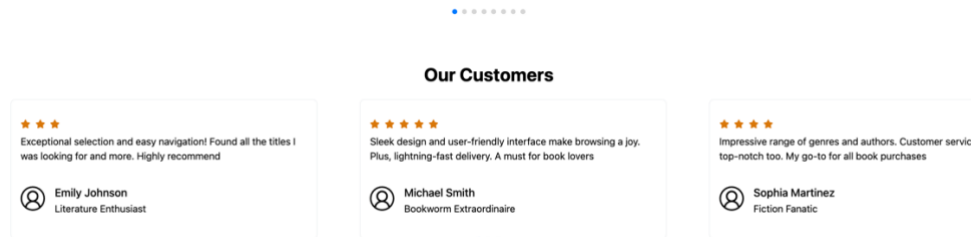


Рисунок 7.3 Відгуки користувачів

Якщо користувачі зареєстровані на сайті, вони мають можливість додавати, редагувати та видаляти книги.

The image shows a web interface for uploading a book. On the left is a sidebar menu with options: Home, Upload Book, Manage Books, Other profile, Log out, Documentation, and Help. The main content area is titled "Upload a Book" and contains the following fields:

- Book Title (text input)
- Image URL (text input)
- Rating (dropdown menu, currently set to "Any")
- Book Description (text area)
- Book PDF (text input)
- Author Name (text input)
- Book Category (dropdown menu, currently set to "Fiction")
- Number Of Pages (dropdown menu, currently set to "Any")

At the bottom of the form is a green "Upload Book" button.

Рисунок 8.3 Редагування, видалення, додавання книг

Можливість реєстрації через обліковий запис Google також доступна, що дозволяє легко та швидко створити акаунт без самостійного та довгого введення інформації про себе.

Sign up

Your email

Password

Sign up

Already have an account? [Login Here](#)


 [Login with Google](#)

Рисунок 9.3 Реєстрація за допомогою Google

Найцікавішою частиною сайту є розділ «Recommendation». Після переходу в цей розділ користувачеві потрібно заповнити невелику форму, за допомогою якої вони отримають індивідуальні рекомендації книг.

Books HOME RECOMMENDATION SHOP SELL BOOK BLOG

Recommendation Form
Please fill the form on your right so we can give you recommendations

Your favorite authors Your favorite genres

Authors Genres

Min Rating Number Of Pages

Any Any

Generate Recommendations

Рисунок 10.3 Розділ "Recommendation"

Якщо жодна з запропонованих книг не підходить, користувач може отримати більш детальні рекомендації, натиснувши кнопку **Click Here!** в кінці сторінки «Recommendation».

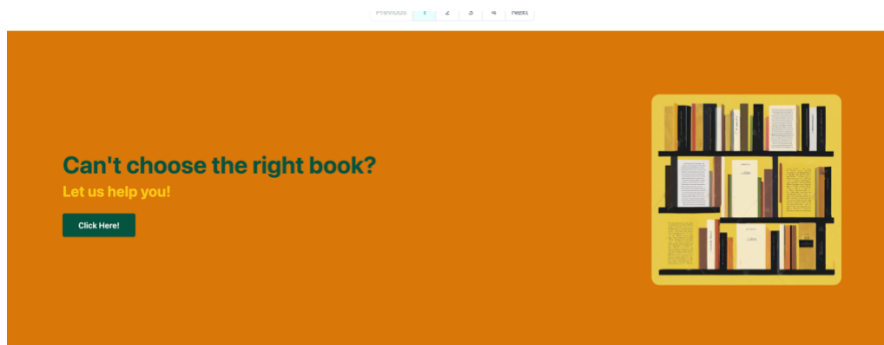


Рисунок 11.3 Банер з кнопкою “Click Here!”

Це перенесе вас на нову сторінку, яка вже використовує методи аналізу ієрархій і попарних порівнянь, щоб надати більш точні поради щодо вибору книги. На сторінці користувачам пояснюється, що рекомендації будуть надані за допомогою АНР. Також на сторінці наведено п’ять таблиць, які потрібно заповнити користувачу, щоб відповідно отримати рекомендації.

Першою є таблиця з критеріями, такими як: автор, жанр, кількість сторінок, рейтинг.

Table 1
Criteria Comparison

	AUTHOR	GENRE	NUMBER OF PAGES	RATING
Author	1	1.000	0.500	0.250
Genre	1 <input type="text"/>	1	0.333	0.200
Number Of Pages	2 <input type="text"/>	3 <input type="text"/>	1	0.500
Rating	4 <input type="text"/>	5 <input type="text"/>	2 <input type="text"/>	1

Рисунок 12.3 Таблиця з експертними оцінками щодо критеріїв

CR цієї матриці 0.6%, тобто він не перевищує норму і відповідно матриця є узгодженою. За розрахунками попарної матриці порівнянь найважливішим у нас

вийшов критерій “жанр”- 41%, потім “автор”- 35%, далі вже “кількість сторінок”- 15.8% та “рейтинг”- 8%

Наступним кроком у нас є таблиця 2, де ми порівнюємо кожну альтернативу за критерієм “автор”. Альтернативи запропоновані нам: “Weyward”, “The Covenant of Water”, “Tom Lake”, “The Invisible Hour”, “Maame”. Матриця сформована на основі інформації виглядатиме так:

Table 2
Comparison Of Alternatives By Category Author

	WEYWARD	THE COVENANT OF WATER	TOM LAKE	THE INVISIBLE HOUR	MAAME
Weyward	1	0.250	0.167	0.333	0.167
The Covenant of Water	4	1	1.000	0.333	0.250
Tom Lake	6	1	1	0.500	0.333
The Invisible Hour	3	3	2	1	0.333
Maame	6	4	3	3	1

Рисунок 13.3 Порівняння альтернатив за критерієм "автор"

Отримані результати: CR матриці-7%, “Weyward”- 51%, “The Covenant of Water”-18%, “Tom Lake”-14%, “The Invisible Hour”-10%, “Maame”- 2%.

У третій таблиці ми порівнюємо кожну альтернативу з критерієм “жанр”

Table 3
Comparison Of Alternatives By Category Genre

	WEYWARD	THE COVENANT OF WATER	TOM LAKE	THE INVISIBLE HOUR	MAAME
Weyward	1	0.333	0.111	0.111	0.167
The Covenant of Water	3	1	0.333	0.200	0.333
Tom Lake	9	3	1	0.500	1.000
The Invisible Hour	9	5	2	1	1.000
Maame	6	3	1	1	1

Рисунок 14.3 Порівняння альтернатив за критерієм "жанр"

Отримані результати: CR -2%. "Weyward"- 56%, "The Covenant of Water"- 22%, "Tom Lake"-8%, "The Invisible Hour"-5%, "Maame"- 7%

В наступній таблиці ми порівнюємо альтернативи за критерієм "кількість сторінок".

Table 4
Comparison Of Alternatives By Category Number Of Pages

	WEYWARD	THE COVENANT OF WATER	TOM LAKE	THE INVISIBLE HOUR	MAAME
Weyward	1	0.333	0.200	0.250	0.111
The Covenant of Water	3	1	0.333	0.200	0.111
Tom Lake	5	3	1	1.000	0.333
The Invisible Hour	4	5	1	1	0.200
Maame	9	9	3	5	1

Рисунок 15.3 Порівняння альтернатив за критерієм "кількість сторінок"

Отримані результати: CR -5%, "Weyward"- 48%, "The Covenant of Water"- 29%, "Tom Lake"-9%, "The Invisible Hour"-10%, "Maame"- 3%

Останньою є п'ята таблиця, де порівнюються альтернативи відповідно до критерія “рейтинг”

Table 5
Comparison Of Alternatives By Category Rating


	WEYWARD	THE COVENANT OF WATER	TOM LAKE	THE INVISIBLE HOUR	MAAME
Weyward	1	0.111	1.000	0.200	0.143
The Covenant of Water	9 ▾	1	1.000	0.333	0.500
Tom Lake	1 ▾	1 ▾	1	0.200	0.333
The Invisible Hour	5 ▾	3 ▾	5 ▾	1	1.000
Maame	7 ▾	2 ▾	3 ▾	1 ▾	1

Рисунок 16.3 Порівняння альтернатив за критерієм "рейтинг"

Отримані результати: CR -8%, “Weyward”- 49%, “The Covenant of Water”- 13%, “Tom Lake”-24%, “The Invisible Hour”-5%, “Maame”- 6%

Після заповнення даних у всіх таблицях ми натискаємо кнопку “next” і отримуємо результати. Weyward”- 0.5193, “The Covenant of Water”-0,2071, “Tom Lake”-0,1145, “The Invisible Hour”-0,0745, “Maame”- 0,045. Отже, найкращою альтернативою серед запропонованих є книжка “Weyward”.

Our best recommendation for you:



Weyward
Emilia Hart

★★★★★

One June day when Mia Jacob can no longer see a way to survive, the power of words saves her. The Scarlet Letter was written almost two hundred years earlier, but it seems to tell the story of Mia's mother, Ivy, and their life inside the Community—an oppressive cult in western Massachusetts where contact with the outside world is forbidden. But how could this be? How could Nathaniel Hawthorne have so perfectly captured the pain and loss that Mia carries inside her?

Fiction
488 pages
AHP: 0.5193

Рисунок 17.3 Найкраща рекомендація за МАІ

Данне завдання було реалізоване на js та React з використанням додаткових бібліотек, таких як flowbite-react, щоб полегшити написання певних елементів на сайті.

Клас ANRTables містить в собі чотири таблиці, у яких експерти можуть порівнювати альтернативи за кількома категоріями, такими як автор, жанр, кількість сторінок та рейтинг. Для кожної категорії створюється окрема таблиця, де користувач може вказати свої переваги для кожної пари альтернатив. Значення в кожній таблиці визначаються за допомогою випадуючого списку, де користувач обирає свою оцінку для порівняння пар альтернатив. Користувач або експерт має вибрати значення лише у лівій частині таблиці, оскільки матриці є обернено симетричними. По діагоналі скрізь стоять одиниці, тому що ми не можемо зазначити, що однакові значення будуть мати різний пріоритет. Таким чином ми полегшуємо завдання для експертів і їм не потрібно самостійно вводити кожен елемент таблиці та запам'ятовувати, що вона є обернено симетричною


```

<div className='bg-green-800 px-11 p-11'>
  <div>
    <h3 className='text-4xl font-bold text-yellow-300 pb-3 text-center'>Table 1</h3>
    <p className='text-xl font-semibold text-yellow-300 pb-11 text-center'>Criteria Comparison </p>
  </div>
  <Table>
    <Table.Head>
      <Table.HeadCell> </Table.HeadCell>
      <Table.HeadCell>Author</Table.HeadCell>
      <Table.HeadCell>Genre</Table.HeadCell>
      <Table.HeadCell>Number Of Pages</Table.HeadCell>
      <Table.HeadCell>Rating</Table.HeadCell>
    </Table.Head>
    <Table.Body className='divide-y'>
      {[1, 2, 3, 4].map((rowItem, rowIndex) => (
        <Table.Row key={rowIndex} className='bg-white dark:bg-gray-800 font-bold'>
          <Table.Cell>{rowIndex === 0 ? 'Author' : rowIndex === 1 ? 'Genre' : rowIndex === 2 ? 'Number Of Pages' : 'Rating'}</Table.Cell>
          {[1, 2, 3, 4].map((colItem, colIndex) => (
            <Table.Cell key={colIndex}>
              {rowIndex > colIndex ? (
                <Select
                  value={selectedValuesCategoriesTable[rowIndex][colIndex]}
                  onChange={(e) => handleSelectChange(e.target.value, rowIndex, colIndex, 5)}
                >
                  {[1, 2, 3, 4, 5, 6, 7, 8, 9].map(option => (
                    <option key={option} value={option}>{option}</option>
                  ))}
                </Select>
              ) : (rowIndex === colIndex ? '1' : (1/(selectedValuesCategoriesTable[colIndex][rowIndex])).toFixed(3) || '')}
            </Table.Cell>
          ))}
        </Table.Row>
      ))}
    </Table.Body>
  </Table>
</div>

```

Рисунок 18.3 Таблиця критеріїв

Далі ми використовуємо функцію `createMatrixFromTable`, яка перетворює дані, введені експертами, у матрицю. Оскільки ми плануємо виконувати різні дії з матрицями, виконувати їх безпосередньо з вихідними таблицями неможливо, тому ця операція необхідна.



```
function createMatrixFromTable(table, rowCount ) {
  const matrix = [];
  for (let i = 0; i < rowCount; i++) {
    const row = [];
    for (let j = 0; j < rowCount; j++) {

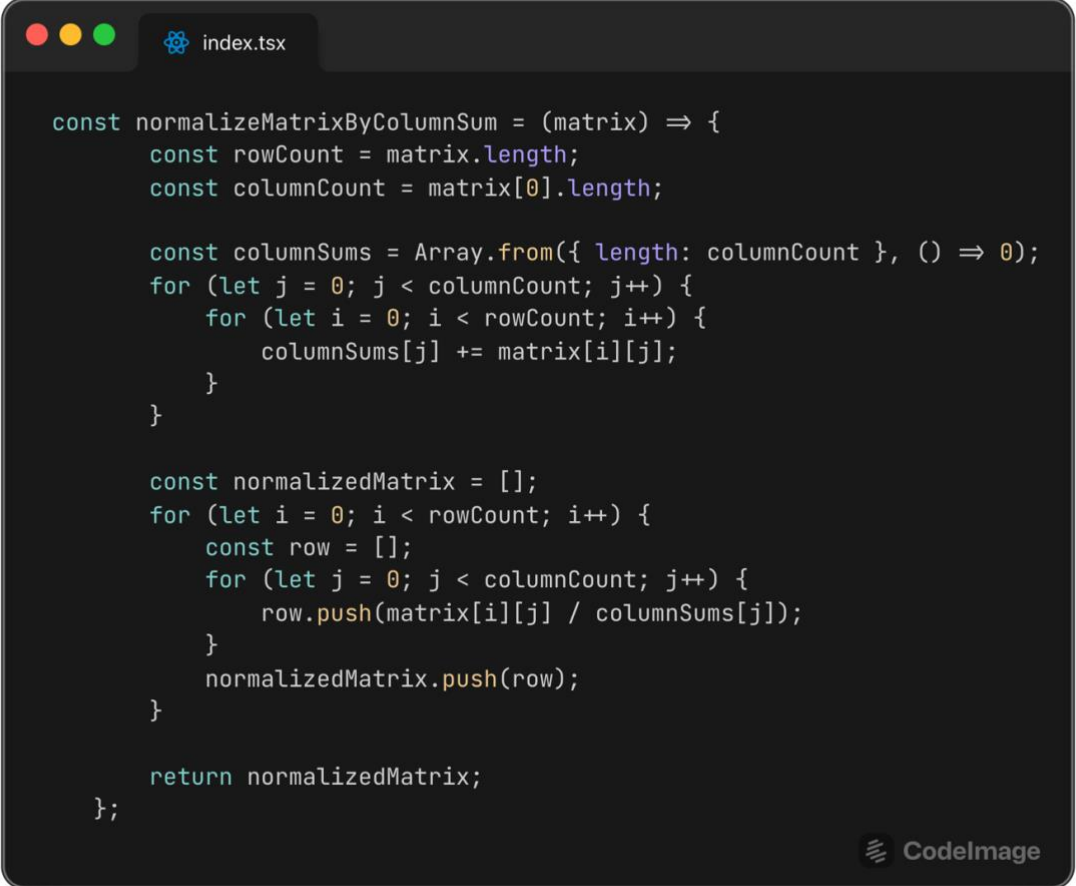
      if (i === j) {
        row.push('1');
      } else {
        if (table[j] && table[j][i]) {
          row.push((1 / table[j][i]).toFixed(3));
        } else {
          row.push(table[i][j] || null);
        }
      }
    }
    matrix.push(row);
  }

  return matrix;
}
```

Рисунок 19.3 Функція `createMatrixFromTable`

Ще однією важливою функцією у класі `ANPImplementation` є `normalizeMatrixByColumnSum`. Вона нормалізує матрицю шляхом обчислення суми значень у кожному стовпчику і поділу кожного елемента матриці на відповідну суму стовпчика. Нормалізація допомагає зробити значення усіх елементів матриці порівнюваними та однаково важливими шляхом їх перетворення у відносні пропорції. У методах аналізу ієрархій, де вага кожного елемента визначається за допомогою порівняння з іншими елементами, це

особливо важливо. Нормалізація допомагає зберегти структуру порівнянь, а результати аналізу стають більш точними та об'єктивними.



```
const normalizeMatrixByColumnSum = (matrix) => {
  const rowCount = matrix.length;
  const columnCount = matrix[0].length;

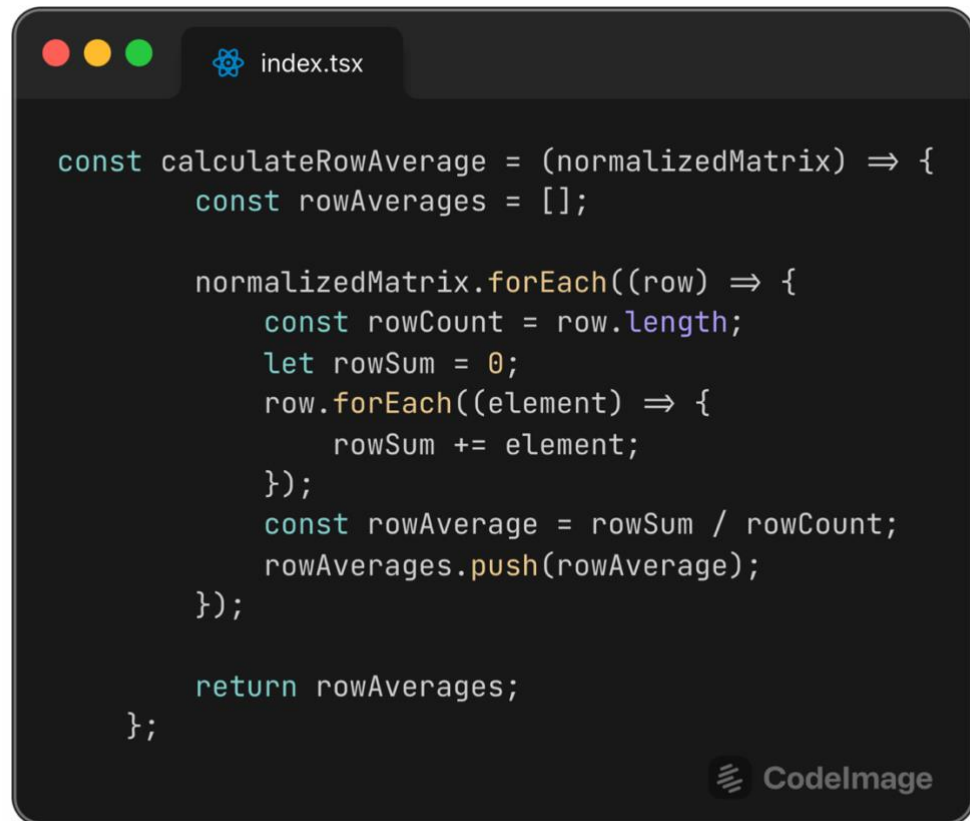
  const columnSums = Array.from({ length: columnCount }, () => 0);
  for (let j = 0; j < columnCount; j++) {
    for (let i = 0; i < rowCount; i++) {
      columnSums[j] += matrix[i][j];
    }
  }

  const normalizedMatrix = [];
  for (let i = 0; i < rowCount; i++) {
    const row = [];
    for (let j = 0; j < columnCount; j++) {
      row.push(matrix[i][j] / columnSums[j]);
    }
    normalizedMatrix.push(row);
  }

  return normalizedMatrix;
};
```

Рисунок 20.3 Функція *normalizeMatrixByColumnSum*

Наступною є функція *calculateRowAverage*, яка має на меті визначити ваговий коефіцієнт W кожного рядка нормалізованої матриці. Ми можемо оцінити вплив кожного елементу на загальний результат аналізу, обчисливши середнє значення для кожного рядка. Це допомагає визначити, які критерії є більш важливими для прийняття рішень.



```
const calculateRowAverage = (normalizedMatrix) => {
  const rowAverages = [];

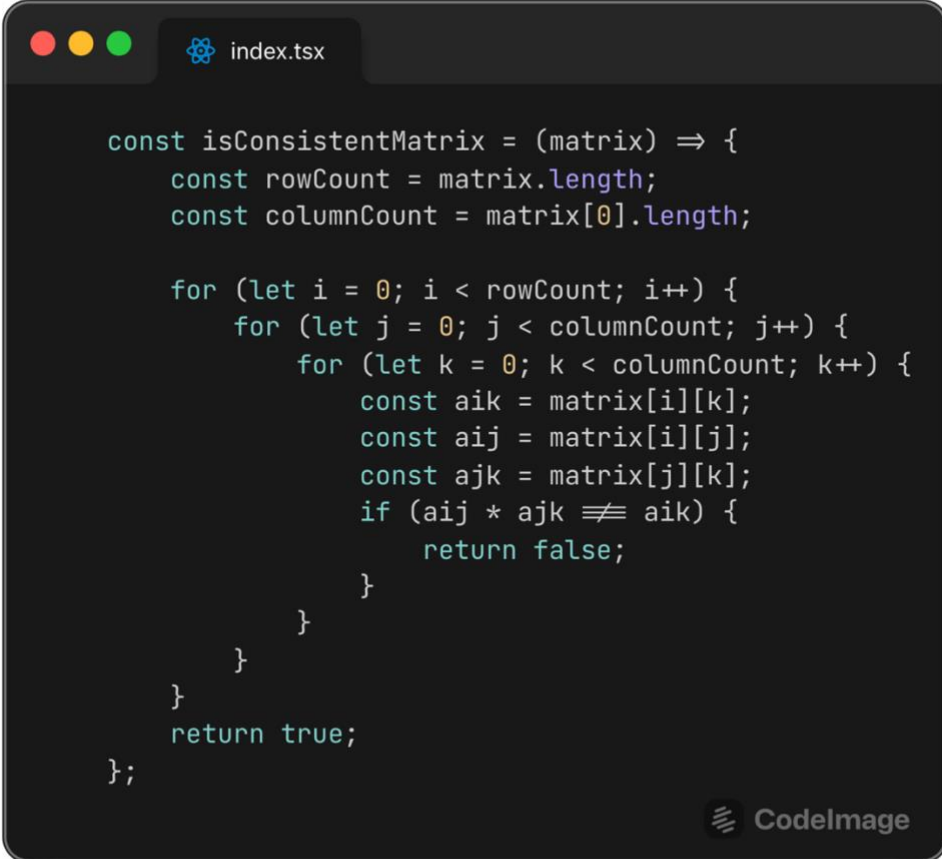
  normalizedMatrix.forEach((row) => {
    const rowCount = row.length;
    let rowSum = 0;
    row.forEach((element) => {
      rowSum += element;
    });
    const rowAverage = rowSum / rowCount;
    rowAverages.push(rowAverage);
  });

  return rowAverages;
};
```

CodelImage

Рисунок 21.3 Функція calculateRowAverage

Функція `isConsistentMatrix` перевіряє, чи відповідає матриця умові консистентності. Консистентність матриці є важливою для отримання правильних результатів під час аналізу методу аналізу ієрархій. Конфлікт матриці може свідчити про неправильне або непослідовне визначення відносної важливості критеріїв або альтернатив. Використовуючи відповідність парних порівнянь, функція перебирає всі елементи матриці та порівнює їх. Якщо умова консистентності для одного елементу порушується, функція повертає `false`, що означає, що матриця не є консистентною. У випадку, якщо всі елементи відповідають умові консистентності, функція повертає `true`.



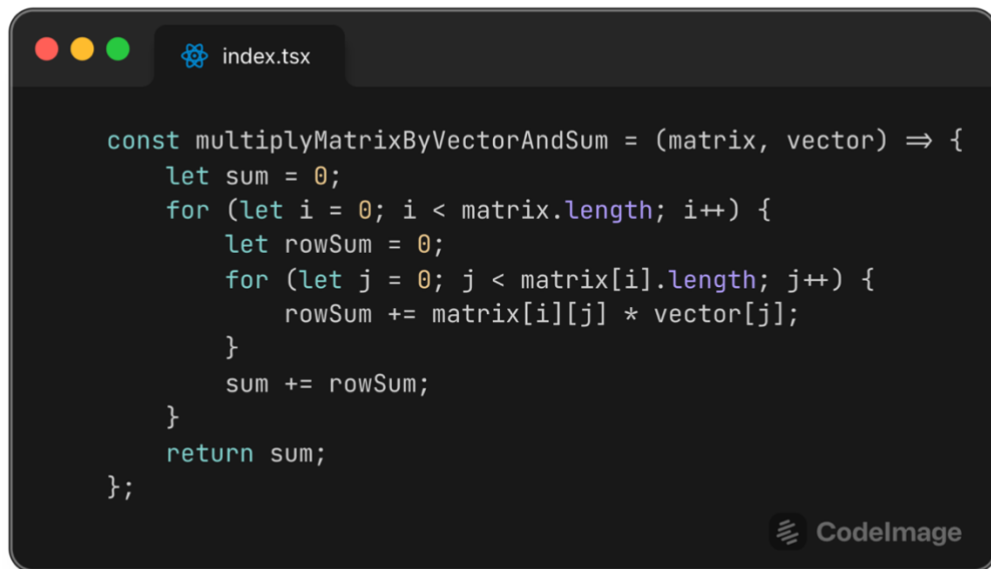
```
const isConsistentMatrix = (matrix) => {
  const rowCount = matrix.length;
  const columnCount = matrix[0].length;

  for (let i = 0; i < rowCount; i++) {
    for (let j = 0; j < columnCount; j++) {
      for (let k = 0; k < columnCount; k++) {
        const aik = matrix[i][k];
        const aij = matrix[i][j];
        const ajk = matrix[j][k];
        if (aij * ajk !== aik) {
          return false;
        }
      }
    }
  }
  return true;
};
```

CodemImage

Рисунок 22.3 Функція *isConsistentMatrix*

Ще однією важливою функцією є `multiplyMatrixByVectorAndSum`. Ця функція виконує множення матриці на вектор та сумує результати. У контексті методу АНР, вона дозволяє нам знайти значення λ_{\max} , яке буде використано у подальшому обчисленні коефіцієнту узгодженості. Кожен елемент рядка матриці множиться на відповідний елемент вектора, а результати додаються разом для кожного рядка матриці. Функція повертає загальну суму цих обчислень.

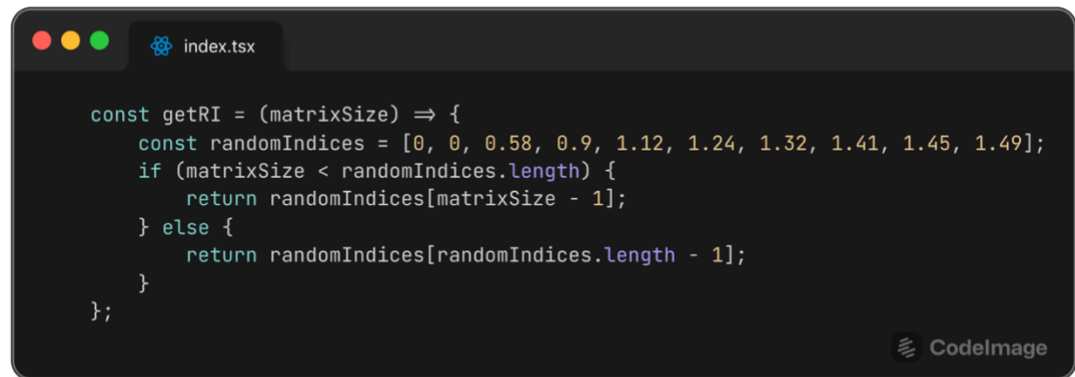


```
const multiplyMatrixByVectorAndSum = (matrix, vector) => {
  let sum = 0;
  for (let i = 0; i < matrix.length; i++) {
    let rowSum = 0;
    for (let j = 0; j < matrix[i].length; j++) {
      rowSum += matrix[i][j] * vector[j];
    }
    sum += rowSum;
  }
  return sum;
};
```

Рисунок 23.3 multiplyMatrixByVectorAndSum

Для CR нам також потрібно отримати значення RI для цього у нас існує функція getRI. Відповідно до розміру матриці порівнянь функція getRI повертає відповідне значення Індексу рандомізації (RI). Після цього це значення використовується для оцінки даних щодо консистентності матриці.

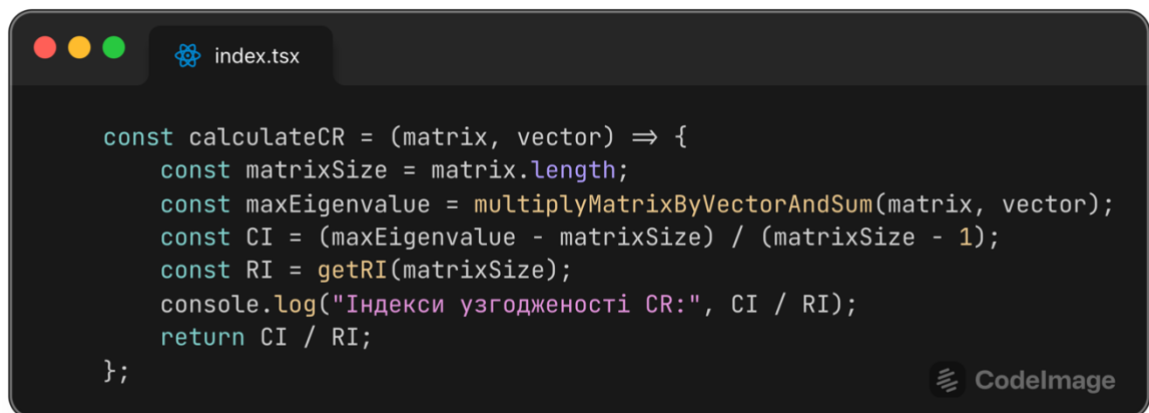
Згідно з емпіричними даними, ці значення RI використовуються для оцінки того, наскільки великі значення випадкових індексів можуть бути прийняті в залежності від розміру матриці.

A screenshot of a code editor window titled 'index.tsx'. The code defines a function 'getRI' that takes 'matrixSize' as an argument. It uses a 'randomIndices' array with values [0, 0, 0.58, 0.9, 1.12, 1.24, 1.32, 1.41, 1.45, 1.49]. The function returns the value at the index 'matrixSize - 1' if it is within the array's length, otherwise it returns the value at the last index. The editor has a dark theme and a 'CodeImage' logo in the bottom right corner.

```
const getRI = (matrixSize) => {
  const randomIndices = [0, 0, 0.58, 0.9, 1.12, 1.24, 1.32, 1.41, 1.45, 1.49];
  if (matrixSize < randomIndices.length) {
    return randomIndices[matrixSize - 1];
  } else {
    return randomIndices[randomIndices.length - 1];
  }
};
```

Рисунок 24.3 Функція getRI

Останнім методом для розрахунку CR у нашій програмі є calculateCR. Застосовуючи функції описані вище вона знаходить коефіцієнт узгодженості матриці.

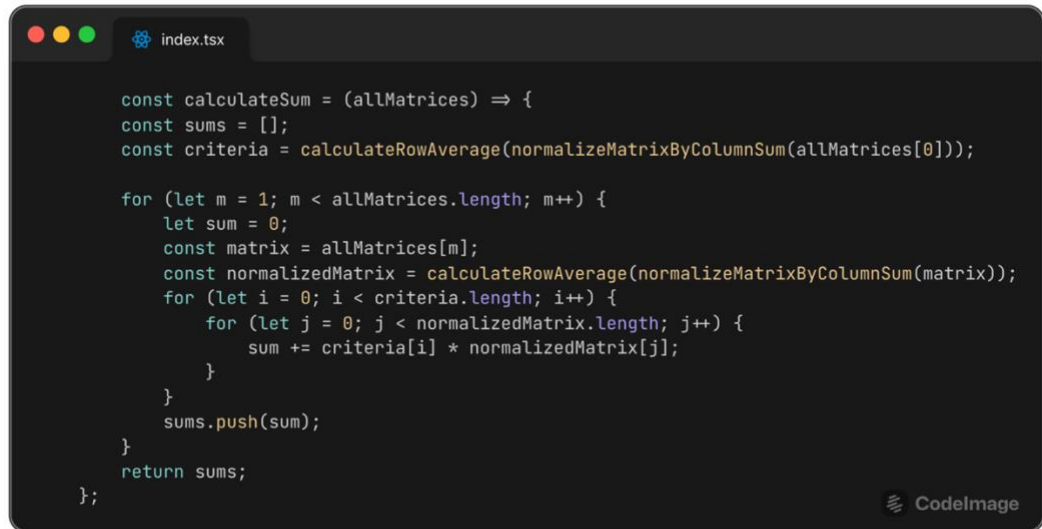
A screenshot of a code editor window titled 'index.tsx'. The code defines a function 'calculateCR' that takes 'matrix' and 'vector' as arguments. It calculates 'matrixSize' from the matrix length, then 'maxEigenvalue' by multiplying the matrix by the vector and summing the results. It then calculates 'CI' as (maxEigenvalue - matrixSize) / (matrixSize - 1) and 'RI' by calling 'getRI(matrixSize)'. Finally, it logs the result and returns CI / RI. The editor has a dark theme and a 'CodeImage' logo in the bottom right corner.

```
const calculateCR = (matrix, vector) => {
  const matrixSize = matrix.length;
  const maxEigenvalue = multiplyMatrixByVectorAndSum(matrix, vector);
  const CI = (maxEigenvalue - matrixSize) / (matrixSize - 1);
  const RI = getRI(matrixSize);
  console.log("Індекси узгодженості CR:", CI / RI);
  return CI / RI;
};
```

Рисунок 25.3 Функція calculateCR

Останнім етапом методу аналізу ієрархій (MAI) є функція «calculateSum», яка обчислює комбіновані вагові коефіцієнти для кожної альтернативи. Ця

функція обчислює суму, кожен коефіцієнт з матриць множиться на вектор критеріїв кожної альтернативи.



```
index.tsx

const calculateSum = (allMatrices) => {
  const sums = [];
  const criteria = calculateRowAverage(normalizeMatrixByColumnSum(allMatrices[0]));

  for (let m = 1; m < allMatrices.length; m++) {
    let sum = 0;
    const matrix = allMatrices[m];
    const normalizedMatrix = calculateRowAverage(normalizeMatrixByColumnSum(matrix));
    for (let i = 0; i < criteria.length; i++) {
      for (let j = 0; j < normalizedMatrix.length; j++) {
        sum += criteria[i] * normalizedMatrix[j];
      }
    }
    sums.push(sum);
  }
  return sums;
};
```

Рисунок 26.3 Функція calculateSum

З функції calculateSum, ми отримуємо масив значень роботи МАІ. Для всіх альтернатив використовуються вищезазначені функції для їх оцінки. Після цього ми знаходимо індекс найбільшого значення серед всіх отриманих результатів (maxIndex) і вибираємо книжку, що відповідає цьому індексу. Таким чином, ця книжка буде найкращою рекомендацією серед усіх альтернатив.

```

index.tsx

<div>
  <div className='pt-28'>
    <h2 className='text-5xl font-bold text-center'>Our best recommendation for you: </h2>
  </div>
  <div className=' pt-8 w-full'>
    {books.find((book, index) => index === maxIndex) && (
      <Card className='w-full h-fit'>
        <div className='w-fit'>
          <div className='flex bg-yellow-400 p-8 rounded-lg shadow-lg'>
            <div className='flex-none w-1/5'>
              <img
                src={books[maxIndex].imageUrl}
                alt=''
                className='h-auto object-contain rounded-lg'
              />
            </div>
            <div className='ml-8'>
              <h2 className='text-4xl font-bold'>{books[maxIndex].bookTitle}</h2><br/>
              <p className='font-medium'><span></span> {books[maxIndex].authorName}</p><br/>
              <div className='flex items-center'>
                <p className='mr-2'>{renderRatingStars(books[maxIndex].rating)}</p>
              </div>
              <p className='mt-4'>{books[3].bookDescription}</p><br/>
              <p className='font-medium'><span></span> {books[maxIndex].category}</p><br/>
              <p className='font-medium'><span></span>{books[maxIndex].numOfPages} pages</p>
              <p className='font-medium'><span></span> AHP: {result[maxIndex]}</p><br/>
            </div>
          </div>
        </div>
      </Card>
    )}
  </div>
</div>

```

Рисунок 27.3 Отримання книги, яка є найкращою альтернативою

Висновки

Метод аналізу ієрархій є потужним і універсальним інструментом для прийняття рішень завдяки системному підходу до аналізу складних проблем. Його популярність пояснюється можливістю розбивати складну задачу на менші частини, враховувати важливість кожного критерію і їх взаємозв'язок. Завдяки цьому, МАІ дозволяє уникнути суб'єктивності й отримувати об'єктивні та зважені результати. Метод також забезпечує можливість роботи з різними типами критеріїв, допомагаючи систематично порівнювати різні альтернативи та забезпечувати глибокий аналіз для обґрунтованих рішень.

Попри ці переваги, МАІ не є ідеальним, оскільки має певні недоліки. Найпомітнішими з них є складність порівняння альтернатив при великій їх кількості, що ускладнює процес та може спричинити втрату точності через вплив людського фактора. Крім того, обмежена шкала Сааті не завжди дозволяє точно диференціювати переваги критеріїв. Проблема узгодженості матриці попарних порівнянь також залишається значним викликом для ефективного використання цього методу.

Для усунення цих недоліків пропонується вдосконалювати процес перевірки узгодженості матриць за допомогою розв'язання систем лінійних алгебраїчних рівнянь. Логарифмічне представлення попарних порівнянь може допомогти поліпшити консистентність матриць, дозволяючи виявляти градації переваг між альтернативами. Використання Перронового вектора та інших методів для аналізу узгодженості матриць також сприятиме підвищенню точності й надійності прийняття рішень за допомогою МАІ.

Таким чином, метод аналізу ієрархій залишається потужним інструментом для прийняття рішень, але потребує вдосконалення в аспектах узгодженості й порівняння альтернатив, щоб підвищити свою ефективність у різних галузях.

Джерела

1. Fundamentals of the Analytic Hierarchy Process Thomas L. Saaty University of Pittsburgh, Pittsburgh PA US; 1994; - Електронний ресурс:
<https://mrdongtp.com/wp-content/uploads/2019/02/Saaty-2001.pdf>
2. Kazancoglu Y. Performing Supplier Selection and Performance Evaluation with Operations Research Techniques in the Logistics Management Process: AHP and DEA [Thesis]. İzmir: Ege University Institute of Social Sciences; 2008
3. Dagdeviren M, Akay D, Kurt M. Analytical hierarchy process for job evaluation and application. Journal of the Faculty of Engineering and Architecture of Gazi University. 2004;19(2):131-138
4. Tas C, Bedir N, Eren T, Alagas HM, Cetin S. Polyclinic evaluation with integrating AHP-TOPSIS methods: An Application in Ankara. Journal of Health Management. 2018;2(1):1-17
5. Kayhan G. Generation a Hybrid System Using Fuzzy Ahp/Fuzzy Topsis in Human Resource Performance Evaluation and an Application [Thesis]. Turkiye: Erciyes University, Graduate School of Natural and Applied Sciences; 2010
6. Олецкий О.В. Підвищення узгодженості та якості попарних порівнянь на основі системи лінійних алгебраїчних рівнянь. //Наукові записки НаУКМА. Комп'ютерні науки. Т.5 (2022) – К., 2022. – С.85-91.
DOI: <https://doi.org/10.18523/2617-3808.2022.5.85-91>.
7. Alessio Ishizaka and Ashraf Labib. Review of the main developments in the Analytic Hierarchy Process, 2011.
8. Płatek P. Analytic Hierarchy Process/ Paweł Płatek. – 2017.
9. Emrouznejad A. Fuzzy Analytic Hierarchy Process / A. Emrouznejad, W. Ho. – Boca Raton: CRC Press, 2017.
10. A heuristic method to rank the alternatives in the AHP synthesis Changsheng Lin a , Gang Kou- Режим доступу до ресурсу:
<https://www.sciencedirect.com/science/article/pii/S1568494620308541>

11. On using pairwise comparison in the analytic hierarchy process: Validity is goal while consistency is means. Ting Kuo 2023. Электронный ресурс:

<https://www.sciencedirect.com/science/article/abs/pii/S002002552301215X>

12. Jaroslav Ramik. Ranking Alternatives by Pairwise Comparisons Matrix and Priority Vector. 2017; Электронный ресурс:

https://www.researchgate.net/publication/322669316_Ranking_Alternatives_by_Pairwise_Comparisons_Matrix_and_Priority_Vector

Додаток 1

```

const AHPImplementation = () => {
  const [books, setBooks] = useState([]);
  const location = useLocation();
  const matrices = location.state;
  const renderRatingStars = (rating) => {
    const stars = [];
    const ratingValue = parseInt(rating);
    for (let i = 0; i < ratingValue; i++) {
      stars.push(<FaStar className='text-yellow-300 inline-flex' key={i} />);
    }
    return stars;
  };
  useEffect(() => {
    fetch("http://localhost:4040/allBooks")
      .then(res => res.json())
      .then(data => {
        setBooks(data);
      });
  }, []);
  const mergeMatrices = (matrixArrays) => {
    return matrixArrays.map(matrixArray => {
      return matrixArray.map(row => row.map(cell => parseFloat(cell)));
    });
  };
  const normalizeMatrixByColumnSum = (matrix) => {
    const rowCount = matrix.length;

```

```

const columnCount = matrix[0].length;
const columnSums = Array.from({ length: columnCount }, () => 0);
for (let j = 0; j < columnCount; j++) {
  for (let i = 0; i < rowCount; i++) {
    columnSums[j] += matrix[i][j];
  }
}
const normalizedMatrix = [];
for (let i = 0; i < rowCount; i++) {
  const row = [];
  for (let j = 0; j < columnCount; j++) {
    row.push(matrix[i][j] / columnSums[j]);
  }
  normalizedMatrix.push(row);
}
return normalizedMatrix;
};

const calculateRowAverage = (normalizedMatrix) => {
  const rowAverages = [];
  normalizedMatrix.forEach((row) => {
    const rowCount = row.length;
    let rowSum = 0;
    row.forEach((element) => {
      rowSum += element;
    });
    const rowAverage = rowSum / rowCount;
    rowAverages.push(rowAverage);
  });
};

```

```

    return rowAverages;
};

const isConsistentMatrix = (matrix) => {
    const rowCount = matrix.length;
    const columnCount = matrix[0].length;
    for (let i = 0; i < rowCount; i++) {
        for (let j = 0; j < columnCount; j++) {
            for (let k = 0; k < columnCount; k++) {
                const aik = matrix[i][k];
                const aij = matrix[i][j];
                const ajk = matrix[j][k];
                if (aij * ajk !== aik) {
                    return false;
                }
            }
        }
    }
    return true;
};

const getRI = (matrixSize) => {
    const randomIndices = [0, 0, 0.58, 0.9, 1.12, 1.24, 1.32, 1.41, 1.45, 1.49];
    if (matrixSize < randomIndices.length) {
        return randomIndices[matrixSize - 1];
    } else {
        return randomIndices[randomIndices.length - 1];
    }
};

const multiplyMatrixByVectorAndSum = (matrix, vector) => {

```

```

let sum = 0;
for (let i = 0; i < matrix.length; i++) {
  let rowSum = 0;
  for (let j = 0; j < matrix[i].length; j++) {
    rowSum += matrix[i][j] * vector[j];
  }
  sum += rowSum;
}
return sum;
};

const calculateCR = (matrix, vector) => {
  const matrixSize = matrix.length;
  const maxEigenvalue = multiplyMatrixByVectorAndSum(matrix, vector);
  const CI = (maxEigenvalue - matrixSize) / (matrixSize - 1);
  const RI = getRI(matrixSize);
  return CI / RI;
};

const calculateSum = (allMatrices) => {
  const sums = [];
  const criteria =
calculateRowAverage(normalizeMatrixByColumnSum(allMatrices[0]));
  for (let m = 1; m < allMatrices.length; m++) {
    let sum = 0;
    const matrix = allMatrices[m];
    const normalizedMatrix =
calculateRowAverage(normalizeMatrixByColumnSum(matrix));
    for (let i = 0; i < criteria.length; i++) {
      for (let j = 0; j < normalizedMatrix.length; j++) {

```

```

        sum += criteria[i] * normalizedMatrix[j];
    }
}
sums.push(sum);
}
return sums;
};

const allMatrices = mergeMatrices(matrices);
// Виклик функції з вхідним параметром allMatrices
const result = calculateSum(allMatrices);
const CR= allMatrices.map((matrix, index) => {
    const vector =
calculateRowAverage(normalizeMatrixByColumnSum(matrix));
    return calculateCR(matrix, vector);
});
const maxSum = Math.max(...result);
const maxIndex = result.indexOf(maxSum);
return (
    <div>
        <div className='pt-28'>
            <h2 className='text-5xl font-bold text-center'>Our best
recommendation for you: </h2>
        </div>
        <div className=' pt-8 w-full'>
            {books.find((book, index) => index === maxIndex) && (
                <Card className='w-full h-fit'>
                    <div className='w-fit'>

```

```

<div className='flex bg-yellow-400 p-8 rounded-lg shadow-
lg'>
  <div className='flex-none w-1/5'>
    <img
      src={books[3].imageUrl}
      alt=""
      className='h-auto object-contain rounded-lg'
    />
  </div>
  <div className='ml-8'>
    <h2 className='text-4xl font-
bold'>{books[3].bookTitle}</h2><br/>
    <p className='font-medium'><span></span>
{books[3].authorName}</p><br/>
    <div className='flex items-center'>
      <p className='mr-
2'>{renderRatingStars(books[3].rating)}</p>
    </div>
    <p className='mt-
4'>{books[3].bookDescription}</p><br/>
    <p className='font-medium'><span></span>
{books[3].category}</p><br/>
    <p className='font-
medium'><span></span>{books[3].numOfPages} pages</p>
    <p className='font-medium mt-4'><span></span> CR:
{CR[3].toFixed(4)}%</p><br/>
    <p className='font-medium'><span></span> AHP:
{result[3]}</p><br/>

```

```

        </div>
    </div>
</div>
</Card>
    )}
</div>
<div className='mt-28 px-4 lg:px24'>
    <h2 className='text-5xl font-bold text-center'>All Books</h2>
    <div className='grid gap-8 my-12 lg:grid-cols-4 sm:grid-cols-2 md:grid-
cols-3 grid-cols-1'>
        {books.slice(0, 8).map((book, index) => (
            <Card key={index}>
                <div className="h-full flex flex-col justify-start">
                    <img
                        src={book.imageURL}
                        alt="
                            className='h-auto max-h-160 object-contain'
                    />
                    <div className="p-4">
                        <h5 className="text-2xl font-bold tracking-tight text-gray-
900 dark:text-white">
                            <p>{book.bookTitle}</p>
                        </h5>
                        <p className="font-normal text-gray-700 dark:text-gray-
400">
                            {book.bookDescription}
                        </p>
                    </div>
                </div>
            </div>
        )}
    </div>

```

```
        </div>
      </Card>
    )}
  </div>
</div>
</div>
);
};
export default AHPImplementation;
```