

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«НАЦІОНАЛЬНИЙ ГІРНИЧИЙ УНІВЕРСИТЕТ»



Л.В. САРИЧЕВА, К.Л. СЕРГЄЄВА

КОМП'ЮТЕРНА ПІДТРИМКА ПРИЙНЯТТЯ РІШЕНЬ

Навчальний посібник

Дніпро
НГУ
2016

УДК 519.816
ББК 32.813
С20

Рекомендовано вченою радою як навчальний посібник для студентів спеціальності 122 Комп'ютерні науки та інформаційні технології (протокол № 15 від 15.11.2016 р.).

Рецензенти:

О.О. Сдвижкова – д-р техн. наук, проф., завідувач кафедри вищої математики (Державний вищий навчальний заклад «Національний гірничий університет»);

С.Л. Нікулін – д-р геол. наук, доц., професор кафедри геоінформаційних систем (Державний вищий навчальний заклад «Національний гірничий університет»).

ISBN 978–966–350–624–1

Саричева Л.В.

С20 Комп'ютерна підтримка прийняття рішень: навч. посіб. / Л.В. Саричева, К.Л. Сергєєва; М-во освіти і науки України, Нац. гірн. ун-т. – Дніпро: НГУ, 2016. – 99 с.

Зміст видання відповідає освітньо-професійній програмі підготовки магістрів зі спеціальності 122 Комп'ютерні науки та інформаційні технології та програмі дисципліни «Комп'ютерна підтримка прийняття рішень».

Розглянуто методи підтримки прийняття рішень, що дозволяють полегшити методологічні труднощі й знизити психологічний бар'єр експерта або керівника при комп'ютерній підтримці прийняття рішень. Ці методи дозволяють: формалізувати процес породження варіантів рішень на основі наявних даних; використовувати формалізовані процедури узгодження при прийнятті колективних рішень; використовувати формальні процедури прогнозування наслідків прийнятих рішень; вибрати варіант, що дозволяє вирішити проблему.

Поряд з математичним формулюванням задач і їх вирішенням розглядається неформальне, змістовне роз'яснення запропонованих методів, які проілюстровані численними прикладами.

Рекомендації орієнтовано на активізацію виконавчого етапу навчальної діяльності студентів.

УДК 519.816
ББК 32.813

ISBN 978–966–350–624–1

© Л.В. Саричева, К.Л. Сергєєва, 2016
© Державний ВНЗ «НГУ», 2016

ЗМІСТ

ПЕРЕДМОВА.....	5
1 СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ	7
1.1 Завдання комп'ютерних систем підтримки прийняття рішень.....	7
1.2 Генерація можливих рішень (сценаріїв).....	7
1.3 Узгодження рішень (сценаріїв)	9
1.4 Оцінка можливих рішень (сценаріїв).....	10
1.5 Комп'ютерний аналіз динаміки розвитку ситуації.....	11
1.6 Вибір рішення (сценарію)	11
2 ГЕНЕРАЦІЯ РІШЕНЬ. АНАЛІЗ КОГНІТИВНИХ КАРТ.....	12
2.1. Методи генерації рішень	12
2.2. Когнітивні карти	13
2.2.1. Визначення і методи побудови когнітивних карт.....	13
2.2.2. Аналіз когнітивних карт.....	14
2.2.3. Приклади оцінки ситуації і можливості прийняття рішення за допомогою когнітивних карт.....	17
3 ДЕКОМПОЗИЦІЯ СИСТЕМИ. СИЛЬНО ЗВ'ЯЗНІ ОБЛАСТІ.....	33
3.1 Алгоритм знаходження сильно зв'язаних областей у графі.....	33
3.2 Програма знаходження сильно зв'язаних областей у MATLAB.....	37
4 СЦЕНАРІЇ – МОЗАЇКА ОПЕРАЦІЙ	39
4.1 Від когнітивних карт до наборів сценаріїв.....	39
4.2 Формальні граматики	39
4.3 Типи мов за класифікацією Хомського	44
5 РОЗПОДІЛ РЕСУРСІВ ШЛЯХОМ УЗГОДЖЕННЯ ГРУПОВИХ РІШЕНЬ З ВИКОРИСТАННЯМ λ -КОЕФІЦІЄНТІВ.....	54
6 ПРИЗНАЧЕННЯ І СТРУКТУРА КОМП'ЮТЕРНИХ СИСТЕМ ПІДТРИМКИ ПЕРЕГОВОРІВ З ПРИЙНЯТТЯ УПРАВЛІНСЬКИХ РІШЕНЬ.....	57
6.1 Комерційні пакети, використовувані в системах підтримки переговорів .	60
6.2 Оцінка характеру можливого виграшу	60
6.3 Характер вирішення протиріч.....	62
7 ВИБІР МЕТОДІВ ОЦІНКИ ВАРІАНТІВ РІШЕНЬ.....	63
7.1 Огляд методів оцінки ефективності рішень	63
7.2 Математичний апарат СППР для оцінки варіантів рішень	66
8 РОЗРОБКА НЕЧІТКИХ МОДЕЛЕЙ СИСТЕМ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ В СОЦІАЛЬНО-ЕКОНОМІЧНИХ СИСТЕМАХ	68
8.1 Змістовна постановка задачі оцінювання ефективності інновації.....	68
8.2 Описання вхідних і вихідних змінних задач, що розглядається	69

8.3 Фазифікація вхідних і вихідних змінних.....	71
8.4 Формування моделі системи нечіткого виводу.....	71
9 ПРОГРАМНІ ПРОДУКТИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ	76
9.1 СППР "Мислитель"	76
9.2 СППР "Вибір"	79
9.3 СППР Right Solution	83
9.4 СППР FreeMind	86
ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ	89
ЗАВДАННЯ ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ.....	90
ВИСНОВКИ	95
БІБЛІОГРАФІЧНИЙ СПИСОК.....	97

ПЕРЕДМОВА

Збільшення обсягу інформації, що надходить до органів управління і безпосередньо до осіб, що приймають рішення (ОПР), ускладнення вирішуваних задач, необхідність врахування великої кількості взаємопов'язаних факторів і швидко змінюваної ситуації вимагають широко використовувати обчислювальну техніку в процесі прийняття рішень. У зв'язку з цим у системах управління все ширше використовується новий клас обчислювальних систем – системи комп'ютерної підтримки прийняття рішень.

Комп'ютерна підтримка прийняття рішень дозволяє ОПР поряд з об'єктивними оцінками використовувати свої суб'єктивні, властиві тільки йому методи генерації та оцінки можливих варіантів прийнятих рішень, використовуючи всю міць програмного забезпечення для реалізації свого стилю вироблення і прийняття рішень.

Генерацію рішень можна поділити на:

- несподівані, принципово нові, новаторські рішення, які обчислювальна машина породжувати поки не в змозі, але якщо вони згенеровані ОПР, комп'ютерна система підтримки прийняття рішень може їх розвинути, проаналізувати і оцінити;

- рішення, що базуються на типових сценаріях, на аналогіях, на основі комбінацій відомих окремих рішень. Породження таких рішень не тільки доступно обчислювальній системі, але вона може згенерувати всі можливі комбінації таких рішень, оцінити їх і вибрати найкраще.

Узгодження рішень є однією з найбільш важко формалізованих процедур прийняття рішень. Формалізовані методи, що дозволяють у багатьох випадках замінити ранжування конкретних рішень узгодженням оцінок параметрів і алгоритмів, за якими здійснюється ранжування, є досить ефективним інструментом узгодження рішень.

Мова формалізованих оцінок дозволяє полегшити взаєморозуміння всередині групи фахівців, які приймають рішення, сконцентрувати увагу на критичних точках й уникнути непотрібних дискусій з другорядних питань. Надзвичайно важливим є представлення системою підтримки інформації про можливий розвиток ситуації при прийнятті того чи іншого рішення. Така інформація може бути отримана в процесі комп'ютерної гри, що моделює реакцію протидії середовища на чергове рішення.

Сьогодні фахівці, які пов'язані з прийняттям рішень, усвідомлюють необхідність використання комп'ютерних систем підтримки прийняття рішень, які містять формальні методи генерації, оцінки та оптимізації рішень.

Таке розуміння викликано, зокрема, і тим, що:

- сучасні комп'ютерні системи підтримки прийняття рішень дозволяють генерувати не одне, а множину можливих рішень, що задовольняють заданим обмеженням;

- кожне з них необхідно оцінити і вибрати найкраще;

– рішення в більшості випадків має задовольняти декільком суперечливим критеріям і тому необхідно враховувати можливість і повноту досягнення цих показників;

– комп'ютерна система підтримки прийняття рішень дозволяє ОНР ясніше зрозуміти становище, в якому вона знаходиться (області в критеріальному і параметричному просторах, що характеризують поточний стан розв'язуваної проблеми);

– незважаючи на те, що методи комп'ютерної підтримки прийняття рішень ще досить слабкі і вимагають свого розвитку, вони в багатьох випадках забезпечують прийняття правильного і своєчасного рішення.

Цей напрямок стане одним з перспективних напрямків застосування обчислювальної техніки.

Навчальний посібник спрямований на розвиток спеціалізовано-професійних компетенцій студентів, а саме отримання:

- знань закономірностей випадкових явищ і вмінь застосовувати ймовірностно-статистичні методи для вирішення професійних завдань;

- умінь розробляти та застосовувати моделі відображення знань, стратегії логічного виведення, технологій інженерії знань, технологій і інструментальних засобів побудови інтелектуальних систем;

- знань теоретичних особливостей чисельних методів, можливостей їх адаптації до інженерних задач, уміння використовувати чисельні методи під час розв'язання різних прикладних задач;

- знань принципів і правил формалізації соціально-економічних ситуацій, умінь застосовувати математичні методи обґрунтування та прийняття управлінських і технічних рішень у різних ситуаціях;

- знань концепцій оперативної аналітичної обробки та інтелектуального аналізу даних;

- умінь виявляти в даних раніше не відомі знання, необхідні для прийняття рішень в різних сферах професійної діяльності.

1 СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

1.1 Завдання комп'ютерних систем підтримки прийняття рішень

Збільшення обсягу інформації, ускладнення вирішуваних завдань, необхідність врахування великої кількості взаємопов'язаних факторів і швидко мінливих вимог до проєктованого об'єкту вимагають використовувати обчислювальну техніку в процесі прийняття рішень. У зв'язку з цим з'явився новий клас обчислювальних систем – системи підтримки прийняття рішень (СППР).

"Системи підтримки прийняття рішень є людино-машинними об'єктами, які дозволяють особам, що приймають рішення, використовувати дані, знання, об'єктивні і суб'єктивні моделі для аналізу і вирішення слабо структурованих і неструктурованих проблем". У цьому визначенні підкреслюється призначення СППР для вирішення слабо структурованих і неструктурованих задач (які містять як кількісні, так і якісні оцінки проблеми, причому якісні мають тенденцію домінувати). Неструктуровані проблеми мають лише якісний опис.

Людино-машинна процедура прийняття рішень в системах автоматизованого проєктування (САПР) за допомогою СППР являє собою ітеративний процес взаємодії конструктора й комп'ютера.

Системи підтримки прийняття рішень в САПР:

1. Генерують можливі варіанти конструкторських рішень.
2. Здійснюють оцінку цих варіантів і вибирають найкращий.
3. Забезпечують постійний обмін інформацією між конструкторами про прийняті ними рішення і допомагають узгоджувати групові рішення.
4. Моделюють прийняті рішення (у тих випадках, коли це можливо).
5. Оцінюють відповідність виконання прийнятих конструкторських рішень наміченим цілям.

1.2 Генерація можливих рішень (сценаріїв)

На рис. 1.1 представлено схему функціонування системи підтримки прийняття рішень.

Генерацію можливих рішень можна реалізувати за допомогою: програмної реалізації аналітичних моделей, з використанням експертних систем, генерації сценаріїв шляхом комбінації різних операцій, заданих ОПР або взятих з бази даних, і, нарешті, використовуючи підхід, що отримав назву ситуаційного управління.

А. Генерація рішень за допомогою аналітичних моделей.

У наш час розроблено дуже багато методів і алгоритмів, що дозволяють моделювати можливі результати прийняття рішень.

До них відноситься величезна кількість алгоритмів чисельних методів розв'язання систем рівнянь, статичні методи, методи ситуаційного моделювання.

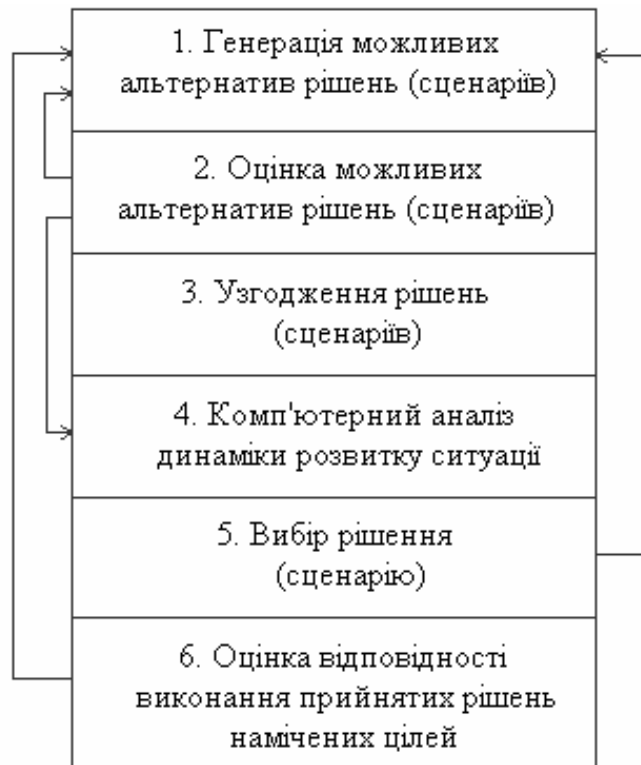


Рис. 1.1 Схема функціонування системи підтримки прийняття рішень

Основна особливість сучасного моделювання – це візуалізація.

Існує широкий клас програмних додатків (соціальні науки, у багатьох випадках економіка, тощо), де ідеальна точність, яку пропонує математика, для особи, що приймає рішення (ОПР) не є потрібною. Їй цілком достатньо грубих, але надійних оцінок.

В. Генерація рішень за допомогою експертних систем.

Експертні системи, що використовують евристичні знання, застосовуються в тих випадках, коли сформулювати рішення задачі в традиційних математичних термінах не вдається. Для всіх найбільш успішних застосувань експертних систем характерна, принаймні, одна спільна риса – вони працюють в одній обмеженій предметній області знань. Спроби розширити предметну область, навіть у межах однієї галузі знань (наприклад, в медицині), в переважній більшості випадків успіху не давали.

Експертні системи підтримки прийняття рішень містять:

- лінгвістичний процесор для спілкування з користувачем;
- базу знань, що містить евристичні знання;
- базу даних;
- інтерпретатор, який на основі вхідних даних, бази знань і бази даних формує рішення задачі.

Головною особливістю експертної системи є можливість робити висновки на основі знань, що зберігаються в базах знань. Знання в експертних системах можуть бути представлені у вигляді семантичних мереж, на основі логічних підходів, у вигляді фреймів і у вигляді системи продукцій. Система продукцій є однією з найбільш популярних форм представлення знань.

Розподілені експертні системи в кожному вузлі системи підтримки прийняття рішень будуються так само, як зосереджені, але крім цього характеризуються тільки їм притаманними особливостями:

- фізичною або логічною розподіленістю залежно від того, чи розташовуються вони на різних вузлах мережі або на одному вузлі;
- розподіленістю даних і баз знань;
- стратегією інформаційного обміну, що враховує достовірність інформації, її важливість, локальність або глобальність розв'язуваної задачі, час очікування завдання в черзі і т.п.

С. Генерація рішень в нестандартних ситуаціях.

При виникненні нестандартної ситуації, характер якої може ідентифікуватися за її приналежністю до даного класу екстремальних ситуацій, СППР пропонує набір можливих дій (операцій). Якщо такий набір не передбачався заздалегідь, він може бути створений екпертом або ОПР. ОПР або експерт повинен вказати можливу послідовність виконання операцій, а також відзначити, які з них можуть виконуватися паралельно (одночасно). Ця інформація може зберігатися в базі даних разом зі списком операцій. На підставі цих даних, а також часу виконання кожної операції, СППР породжує можливі послідовності операцій (варіанти сценаріїв). Система породжує всі можливі сценарії і виникає задача вибору найкращого.

1.3 Узгодження рішень (сценаріїв)

Для того, щоб процедура узгодження рішення реалізовувалася ефективно фахівцем, треба запропонувати якісь правила, за якими здійснювався би пошук компромісу. У загальному випадку ці правила можна розділити на дві категорії: "переговорні" (без використання обчислювальної техніки), і людино-машинні, що спираються на комп'ютерні процедури. Ці комп'ютерні процедури в більшості випадків досить прості у застосовуванні на практиці. Нижче дається короткий перелік деяких процедур.

А. Метод ідеальної точки. Точка **a** називається ідеальною, якщо вона оптимальна відразу за всіма критеріями. Як правило, такої точки, відповідної оцінкам хоча б одного пред'явленого об'єкта, не існує, але правилом пошуку компромісу може бути мінімізація відстані до ідеальної точки, яка може бути легко визначена.

В. Метод поступок. Суть методу – знаходження компромісу, який визначає "плату" за втрату показників за якимсь критерієм (або частиною критеріїв) за рахунок виграшу за іншим критерієм. У цьому методі використовується апарат кусочно-лінійної апроксимації.

С. Метод узгодження рішень за головним критерієм. У деяких випадках задачу з кількома показниками вдається звести до задачі з одним єдиним показником, і прагнути звернути її до екстремуму, а для решти показників ввести деякі обмеження. Тоді проблема узгодження зводиться до виділення головного критерію, погодженню обмежень для всіх інших критеріїв, знаходженню компромісу за головним критерієм.

D. Метод узгодження рішення при лексикографічному упорядкуванні. У тих випадках, коли можуть бути визначені важливості критеріїв, упорядкування можна проводити спочатку за найбільш важливим критерієм, якщо за цим критерієм рівною виявляться кілька станів, то за другим за важливістю критерієм і т.д.

E. Метод узгодження групових рішень з використанням ранжирування за Парето.

F. Метод узгодження за функцією або відношенням переваги (корисності). Формується функція або відношення, що відображає перевагу ОПП (його бачення "що таке добре і що таке погано"). Обчислюються значення функцій для альтернатив рішення, які ранжуються за відношенням корисності.

G. Метод узгодження групових рішень з використанням λ -коефіцієнтів.

Після узгодження рішень відбувається перехід до блоку «комп'ютерний аналіз динаміки розвитку ситуації» або до блоку «вибір рішення (сценарію)».

1.4 Оцінка можливих рішень (сценаріїв)

Оцінка можливих варіантів рішень необхідна для всіх типів завдань і типів систем. Вона передуює остаточному вибору рішення і є його складовим елементом.

A. Традиційні методи оцінки можливих рішень.

У тих випадках, коли фахівець може дати оцінку кожного прийнятого рішення, наприклад, в балах – це, звичайно, найкращий варіант. Однак, дуже часто це зробити не вдається. Тому останнім часом, посилена увага теоретиків і прикладників приділялася розробці нових методів оцінки, які наведені нижче.

B. Використання нечіткої логіки для оцінки можливих рішень.

Необхідність появи нечіткої логіки викликана тим, що при зростанні складності систем знижується наша здатність робити точні і в той же час значущі твердження щодо їхньої поведінки, поки не буде досягнутий поріг, за яким точність і значущість – майже взаємовиключні характеристики. Для роботи з розмитими множинами введені лінгвістичні змінні. Лінгвістичною називається змінна, що задана на деякій кількісній шкалі і яка приймає значення у вигляді слів і словосполучень природної мови. Значення лінгвістичної змінної описуються нечіткими змінними. Лінгвістичні змінні і їх значення служать для якісного словесного опису деякої кількісної величини.

C. Багатокритеріальні оцінки.

Оцінка варіанту рішення за багатьма критеріями означає, що фахівець переслідує більш ніж одну мету. Ці цілі можуть мати різну ступінь важливості, тобто фахівець керується більш ніж одним показником якості прийнятого рішення. При цьому характерна незвідність критеріїв природним чином до одного змістовно значимого показником якості.

Після виконання оцінки варіантів рішень за необхідністю здійснюється перехід до блоку 3 (узгодження рішень) або до блоку 4 (комп'ютерний аналіз динаміки розвитку ситуації), потім до блоку 5 – вибір рішення. Якщо не знайдено жодного задовільного рішення, проводиться перехід до блоку 1.

1.5 Комп'ютерний аналіз динаміки розвитку ситуації

Надзвичайно важливою є можливість подання ОПР інформації про можливий розвиток ситуації при прийнятті того чи іншого рішення. Така інформація виникає в процесі комп'ютерної гри, що моделює реакцію протидії середовища після чергового рішення.

Процес прийняття рішення можна розглядати як гру двох (кількох) супротивників. Одним з них є ОПР, іншим – його супротивник: природа, економіка, соціальна структура і т.д. Методи генерації рішень та їх оцінки можуть бути використані на заключній стадії ухвалення рішення – аналізі можливих наслідків прийнятого рішення як комп'ютерної гри. Така гра може розглядатися як віддалений аналог комп'ютерних ігор в шахи, покер і т.д.

У такій грі можливі три варіанти:

А) Всі можливі ходи визначені наперед. Тобто набір операцій є заздалегідь заданий (додавати нові не можна). Можна міняти послідовність їх виконання, частину не використовувати взагалі, можна змінювати обсяг і тривалість виконання операцій. Такий варіант гри можливий у фінансових, державних відомствах, банках, і т.д. за відсутності екстремальних ситуацій.

В) Частина ходів є визначеною наперед, але можливі ходи непередбачені заздалегідь. Варіант В відрізняється від варіанту А можливістю впровадження нових операцій, досвіду проведення, якого немає або є досвід їх проведення, але в іншій обстановці. Можливі всі зміни, зазначені у варіанті А.

С) Ситуація настільки непередбачена, що реакція на неї не планувалася. Тому процедуру прийняття рішення треба починати з "білого аркуша". Тобто визначати параметри, критерії, список операцій і т.д.

Важливо підкреслити, що хід – це, як правило, не одна операція, а сценарій. Сучасні програми шахової гри аналізують варіанти на багато ходів вперед, вміють грати за противника, оскільки ходи фігур на дошці строго фіксовані, а правила визначення "ваги" кожної фігури в ситуації, що склалася, строго визначені. При грі з мінливими правилами типу А і В ці умови порушуються. Проте мета залишається та ж: не тільки знайти краще рішення на глибину в один хід, промодельовати його виконання, а переглянути обстановку на кілька ходів вперед, оскільки хід, що здається найбільш виграшний в даний момент, при більш глибокому аналізі (розвитку подій) може виявитися фатальним.

Використовуючи ігровий підхід, аналогічний комп'ютерним іграм, можна промодельовати хід досягнення поставленої задачі, знайти найкращий, виходячи не тільки з оцінки ситуації на даний момент, а й з аналізу можливих варіантів розвитку подій.

1.6 Вибір рішення (сценарію)

Після того, як правила ранжирування рішень узгоджені, їх оцінки в багатьох випадках доводиться перераховувати заново і ОПР вибирає краще з її точки зору рішення з декількох варіантів, які отримали найвищі оцінки. Після процедури вибору рішення СПР переходить до блоку 6 або, якщо вибір виробити не вдалося, до блоків 1 або 2.

2 ГЕНЕРАЦІЯ РІШЕНЬ. АНАЛІЗ КОГНІТИВНИХ КАРТ

2.1. Методи генерації рішень

Процес генерації рішень, що базується на евристичних перевагах ОПР, можна розділити на три послідовних етапи. Кожному етапу в системі підтримки прийняття рішень повинна відповідати своя підсистема генерації рішень. При цьому треба підкреслити, що в СППР можуть бути включені не всі три підсистеми, більше того, кожна з цих підсистем може функціонувати і самостійно.

1. *Формування когнітивної карти.*

На цьому етапі визначаються основні фактори, що впливають на вирішення проблеми і їх причинно-наслідкові зв'язки. Система підтримки прийняття рішень стає інструментом, що полегшує змістовний аналіз цих зв'язків, ступінь впливу різних факторів один на одного і на систему в цілому і допомагає експерту або ОПР неформальними методами виробити рішення на основі проведеного аналізу.

Формування когнітивної карти може стати також і вихідними даними для подальших етапів генерації рішення.

2. *Створення бази знань експертної системи.*

Після того, як на основі когнітивного аналізу вироблені пропозиції або прийняті рішення про виконання певного набору дій (операцій), до бази даних експертної системи (або декількох експертних систем) записуються умови, при яких можуть бути виконані ці дії і деталі їх виконання відповідно до реальних умов. На основі цієї інформації, записаної в базі знань, експертна система відповідно до конкретної обстановки, генерує рішення про порядок виконання операцій (дій).

Експертна система може бути створена і без етапу попереднього когнітивного аналізу, хоча, як правило, в явному або неявному вигляді, він виконується.

3. *Сценарій – мозаїка операцій.*

Розглядаючи набір операцій, отриманий в результаті когнітивного аналізу або сформованих в експертній системі як вихідні дані, система підтримки прийняття рішень формує можливі сценарії – послідовності виконання таких операцій (дій). Сценарії можуть відрізнятися не тільки послідовністю дій, а й складом. Визначення такого сценарію і є рішенням ОПР або рекомендацією експерта.

Сценарій може бути створений і без формального виконання двох раніше зазначених етапів.

Як вже зазначалося вище, при створенні систем підтримки прийняття рішень для різних програмних додатків можуть бути використані або один з етапів генерації рішень, або їх різні комбінації (рис. 2.1.).



Рис. 2.1 Послідовність створення підсистем генерації рішень

2.2. Когнітивні карти

2.2.1. Визначення і методи побудови когнітивних карт

Когнітивна карта (карта пізнання) – це вид математичної або представленої у вигляді графа моделі, яка дозволяє описувати суб'єктивне сприйняття людиною або групою людей будь-якого складного об'єкта, проблеми або функціонування системи. Когнітивна карта призначена для виявлення структури причинних зв'язків між елементами системи, складного об'єкта, складовими проблеми і т.п. і оцінки наслідків, що відбуваються під впливом впливу на ці елементи або зміни характеру зв'язків.

Когнітивна карта – це суб'єктивна модель сприйняття людиною частки світу, за допомогою якої вона хоче усвідомити закономірності цієї частки, не вдаючись у багато подробиць, оскільки вони надзвичайно ускладнюють картину і її сприйняття.

Теорія графів надає засоби відображення структури причинно-наслідкових зв'язків: це шляхи, цикли і компоненти. Вони виявляються корисними для аналізу складних структур взаємозалежностей. Ці три об'єкти можуть вступати один до одного у деякі відношення, і кожне відношення може бути інтерпретоване як додатне чи від'ємне: наприклад, «згоден чи не згоден», «любить чи не любить».

Елементи системи або об'єкта, що вивчається, називаються концептами. Концепти у графі представляються вершинами, причинно-наслідкові зв'язки – спрямованими дугами, що зв'язують концепти. Такий графічний вигляд представлення досліджуваного об'єкта називається когнітивною картою.

Політичні альтернативи, різноманітні економічні причини і ефекти, цілі та необхідні засоби їх досягнення можуть розглядатися як змінні концептів. У цьому випадку вони представляються вершинами когнітивної карти. Їх вплив один на одну позначаються на когнітивній карті спрямованими дугами з навішеними на них знаками. Знаки визначають характер впливу. Такий граф називається знаковим графом.

Популярність когнітивних карт пояснюється відносною легкістю подання причинних зв'язків (відношень) між концептами і загальної структури досліджуваного об'єкта.

Методи побудови когнітивних карт повинні відповідати наступним вимогам:

- вони повинні бути конструктивними і зручними;
- вони не повинні вимагати від упорядника когнітивної карти попередньої специфікації концептів;
- вони повинні бути тісно пов'язані з методами оцінок результатів аналізу так, щоб у процесі прийняття рішень когнітивних карта могла служити порадиником і навіть критиком ОНР;
- вони повинні точно відображати уявлення ОНР про концепти і відношення між ними.

Існує кілька методів побудови когнітивних карт:

1. Когнітивну карту будує сама ОНР на основі своїх знань і уявлень без залучень експертів та довідкових матеріалів. Ефективність цього методу визначається кваліфікацією ОНР: її знаннями і умінням визначати характер відношень між концептами. Побудова когнітивної карти допомагає ОНР ясніше уявити собі проблему, краще зрозуміти роль окремих концептів і характер відношень між ними. Це найшвидший спосіб побудови когнітивної карти.

2. Побудова когнітивних карт на основі вивчення документів. Цей метод має дві переваги: він зручний і дозволяє використовувати дані, які використовує сам ОНР. Проте вивчення документів експертами – процес досить трудомісткий.

3. Побудова когнітивних карт на основі опитувань групи експертів, що мають можливість оцінювати причинні зв'язки. Перевага цього методу полягає в можливості агрегувати індивідуальні думки. Він базується на більшому діапазоні оцінок, ніж можна витягти з досліджуваних документів.

4. Побудова когнітивних карт на підставі відкритих вибіркового опитувань. Цей метод може бути використаний для побудови порівнюваних когнітивних карт. Його перевага полягає в наданні досліднику можливості вести активний діалог з джерелами інформації.

2.2.2. Аналіз когнітивних карт

Як вже зазначалося, когнітивна карта містить елементи двох типів: концепти і причинні зв'язки. Концепти розглядаються як змінні когнітивної карти, а причинні зв'язки – як відношення між кожною парою змінних. Ці відношення причинності можуть мати різні значення. Основні значення – додатне, від'ємне і нуль. При додатному значенні тих двох змінних, які вона пов'язує, зміни відбуваються в одному напрямку (не обов'язково зі знаком плюс). Наприклад, збільшення числа заводів викликає зростання споживання енергії, що також означає скорочення числа заводів і має призвести до зменшення споживання енергії.

При від'ємному значенні відношення причинності збільшення значення однієї із змінних, пов'язаних цим відношенням, викликає зменшення значення

іншої (і навпаки). Тобто їх зміни відбуваються в протилежних напрямках. Так, збільшення ціни на енергію призводить до скорочення її. Споживання і, відповідно, зменшення ціни – до збільшення споживання енергії. Нульове відношення причинності показує, що зв'язків між двома концептами немає (рис. 2.2).

Структуру когнітивного графа найкраще представити у вигляді орієнтованого графа, в якому вершини є концептами (змінними концептів), а дуги висловлюють відношення причинності. Дуги можуть позначатися знаками "+", "-" або 0, що означає відповідно додатне, від'ємне або нульове причинне відношення.

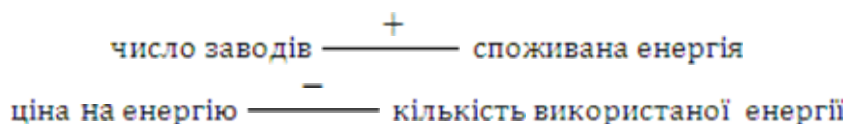


Рис. 2.2 Приклади додатних і від'ємних відношень

Можливі базові значення, які використовуються в когнітивних картах, є логічною комбінацією значень "додатне", "від'ємне" і "нульове". Таких комбінацій дев'ять: + додатне; – від'ємне; 0 нульове; (+) невід'ємне; тобто {0, +}, (-) недодатне, тобто {0, -}; *m*-ненульове, тобто {+, -}; *u*-універсальне, тобто {+, -, 0}; *a* – амбівалентне.

Як вже зазначалося раніше, направлений граф з перерахованими вище мітками, якими позначаються його дуги, називається *знаковим графом* (рис. 2.3).

У багатьох випадках причинний концепт, з якого виходить дуга, є одночасно і концептом результату, тобто в нього входить дуга виходить з іншого причинного концепту.

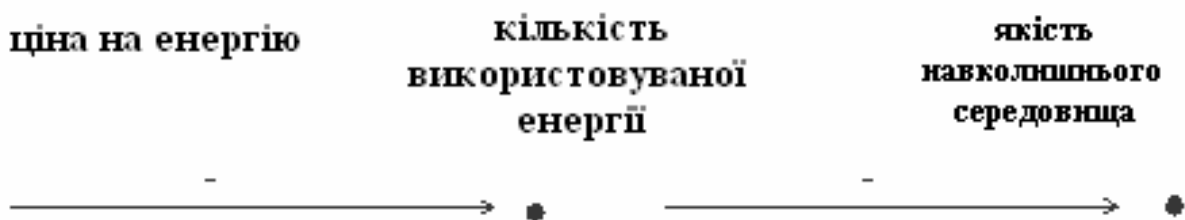


Рис. 2.3 Приклад знакового графа

Тепер розглянемо, як може бути проведений аналіз для комплексної когнітивної карти: визначимо її структурні властивості і висновки, які можна отримати з такої когнітивної карти. Спочатку розглянемо причинні відношення. Так, на графі рис. 2.3 показано шлях з двох дуг від ціни на енергію до якості навколишнього середовища. Показана наступна послідовність:

- 1) підвищення цін на енергію викликає скорочення її споживання, оскільки відношення причинності від'ємне;
- 2) скорочення кількості споживаної енергії веде до поліпшення якості навколишнього середовища, оскільки відношення причинності знову від'ємне.

Таким чином, збільшення значення першої змінної через два від'ємних відношення причинності призводить до збільшення значення останньої змінної графа.

На рис. 2.3 був показаний ациклічний граф. Тепер також на типовому часто використовуваному прикладі перейдемо до розгляду циклічного знакового графа, показаного на рис. 2.4.



Рис. 2.4 Приклад циклічного знакового графа

Шлях в цьому графі починається з вершини "виробництво енергії" і пройшовши через кілька вершин графа знову повертається до вершини "виробництво енергії". Значення початкової змінної при проході по дугах графа тільки зі знаком «+» може бути сильно збільшено.

У прикладі рис.2.4 збільшення виробництва енергії збільшує число заводів, це збільшує число робочих місць, що в свою чергу збільшує чисельність населення, воно збільшує споживання енергії, останнє вимагає збільшення її виробництва. Таким чином утворюється *додатний зворотний зв'язок*, який може значно посилити початкову зміну. Якщо у знаковому графі є кілька таких додатних циклів, це може привести до нестабільності системи, тому незначне початкове зміна може привести до непередбаченого значного стрибка.

Але в графі на рис. 2.4 є і *від'ємний зворотний зв'язок*: підвищення виробництва енергії веде до погіршення якості навколишнього середовища (через ціну на енергію і її споживання), що в свою чергу веде до скорочення населення, скорочення споживання енергії і, в кінцевому рахунку, до скорочення її виробництва. Від'ємні зворотні зв'язки часто призводять до стабілізації системи, проте вони можуть викликати все більші й більші коливання системи, тобто до іншої нестабільної ситуації. Від'ємний або додатний зворотний зв'язок в системі може бути легко визначений: у циклі існує додатний зворотний зв'язок тоді і тільки тоді, якщо в ній парна кількість знаків мінус (або коли знаків мінус немає).

Загальне правило поведінки шляхів і циклів наступне:

1. Зворотній зв'язок у циклі *додатний*, якщо число дуг зі знаком мінус парне (або їх немає), і *від'ємний*, якщо число дуг зі знаком мінус – непарне.

2. *Загальний зворотний зв'язок* вершини А на вершину В (можливо на ту ж вершину А) є сумою зворотних зв'язків в циклах на всьому шляху від А до В (можливо від А до А).

Якщо зворотні зв'язки у всіх циклах додатні, то сума зворотних зв'язків додатною. Якщо сума зворотних зв'язків у всіх циклах від'ємна, то сума є від'ємною. В шляху або циклі з парним числом дуг зі знаком мінус початкове зміна посилюється. В шляху або циклі з непарним числом дуг зі знаком мінус первісним зміни чиниться протидія.

У прийнятті групових рішень проблема полягає в узгодженні рішень. Винятком є варіант, коли лідер приймає рішення, а інші учасники прийняття рішення виступають фактично в ролі порадників без права вирішального голосу. При побудові когнітивних карт таке узгодження зводиться до трьох питань:

1. Погодження списку концептів.
2. Узгодження відношень причинності між ними.
3. Узгодження значень відношень причинності.

Для вирішення всіх цих питань можуть бути використані процедури роботи з експертами, зокрема процедури голосування або формалізовані процедури.

Як б процедура не використовувалася, в результаті має бути отриманий список концептів (вершин знакового графа), список відношень причинності (дуг графа) і список значень відношень причинності кожної дуги. Таким чином, завдання зводиться до побудови когнітивної карти ОІР (колективу ОІР).

2.2.3. Приклади оцінки ситуації і можливості прийняття рішення за допомогою когнітивних карт

Метод когнітивних карт використовується для підтримки прийняття рішень. Однак багато хто вважає його тільки хорошим допоміжним засобом для з'ясування структури повсталі проблеми: визначення факторів, що характеризують або впливають на проблему (концепти), визначення зв'язків між ними (відношень причинності) і з'ясування характеру зв'язків (значень відношень причинності).

Розглянемо когнітивну карту як засіб підтримки прийняття рішення. У цьому випадку виникають три проблеми.

1. *Проблема оцінки рішень.* Якщо в когнітивній карті є кілька змінних (концептів), що визначають різні варіанти рішень, природно постає питання: які змінні (тобто рішення) повинні бути прийняті і які відкинуті?

Природно прийняти ті рішення, які спричиняють кращий додатний ефект і відкинути, ті що спричиняють від'ємний.

2. *Проблема передбачення наслідків прийняття рішення.*

а) Якщо значення деяких змінних буде збільшено, а деяких зменшено, що станеться зі значенням інших змінних. Необхідно перерахувати граф. Якщо всі вхідні в вершину дуги (вірніше, шляхи) додатні, то результат додатний. Можна

навіть говорити про сумарний додатний вплив. Якщо всі вхідні в вершину дуги (шляху) від'ємні, то результат від'ємний. Можна говорити навіть про суму від'ємних впливів. Нарешті, якщо частина дуг, що входять в вершину додатна, а частина від'ємна, то результат оцінити складніше: він проміжний.

б) Які будуть наслідки зміни знака у однієї зі змінних?

Для вирішення питання необхідно здійснити перерахунок когнітивної карти. Оцінки генеруються так само, як у випадку а).

в) Які будуть наслідки, якщо виключити деякі змінні і ввести інші? Для цього необхідно фактично згенерувати нову когнітивну карту.

3. *Пояснення результатів і прийняття рішення.* Застосування когнітивних карт, як правило, вимагає для прийняття рішення додаткового аналізу та інтерпретації.

Таким чином, процес використання когнітивних карт для прийняття рішень можна представити у вигляді схеми, показаної на рис.2.5.

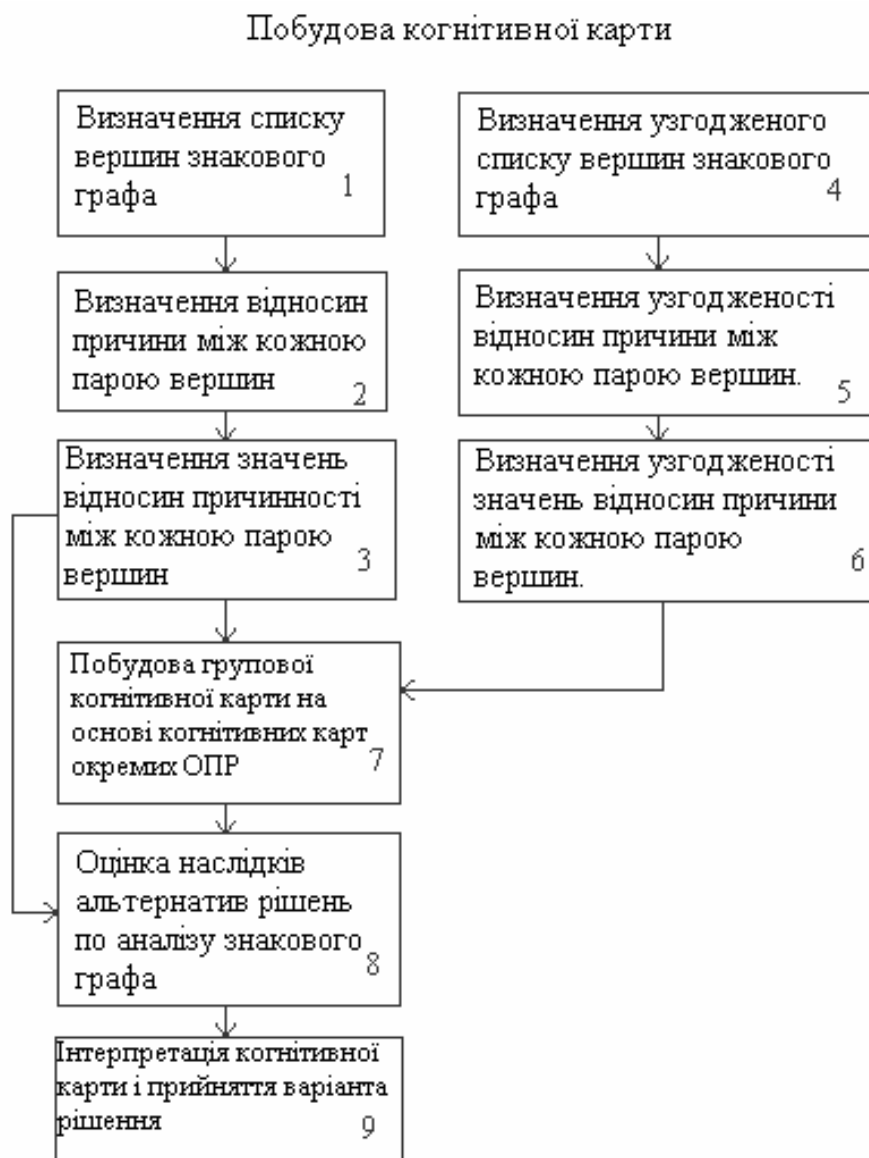


Рис. 2.5 Схема процесу використання когнітивних карт для прийняття рішень

Блоки 1, 2, 3 формують когнітивну карту одного ОПР, блоки 4, 5, 6 – групи ОПР, блок 7 формує групову когнітивну карту за когнітивними картами окремих ОПР (якщо це потрібно, якщо ні – показаний перехід з блоку 3 в блок 8). Блоки 8 і 9 інтерпретації не вимагають.

Одна з переваг методу знакових графів полягає в можливості визначення результатів (якісних і дуже грубих) різних "стратегій", тобто змін, які треба здійснити в досліджуваній системі. Ці зміни можуть відбуватися незалежно від нашої волі, в цьому випадку намагаємося тільки оцінити їх наслідки. У наведеному прикладі можна отримати оцінку стабільності системи споживання енергії, оцінити результати можливих змін, визначити стабільні й нестабільні змінні.

Приклад. Аналіз транспортної системи споживання енергії [20].

У процесі дослідження проведені три опитування групи експертів. Перше опитування пов'язане з визначенням списку змінних (концептів), друге – з обмеженням цього списку розумними розмірами. Третє опитування пов'язане з визначенням дуг (відношень причинності) та їх знаків.

Опитування перше. Визначення списку змінних. Анкета з питаннями була розіслана групі експертів з проханням вказати якомога більше змінних, пов'язаних зі зростанням потреби в енергії (обмеження, впливи, причини, дії і т.п.) в галузі перевезень. Експерти обрані із кола фахівців з широким кругозором, що добре розбираються в проблемі, але не обов'язково представляють яку-небудь спеціальну технічну дисципліну.

На питання по першій анкеті названо більше 500 змінних, що відносяться до використання енергії на транспорті. Необхідно відзначити, що дослідження проводилося без застосування розподіленої системи підтримки прийняття рішень.

Всі змінні були розбиті на вісім груп наступним чином:

1. Описові змінні (характеризують транспортні системи).
2. Конструкторські змінні (описують конструкції різних транспортних засобів – автобусів, вантажівок, вагонів і т.п.).
3. Демографічні змінні (описують стиль життя, тенденції населення і т.д.).
4. Економічні змінні.
5. Змінні, пов'язані з виділенням шкідливих речовин від згоряння пального при перевезенні вантажів.
6. Екологічні змінні, що описують вплив на навколишнє середовище та естетичний стан.
7. Соціальні змінні, що характеризують якість життя.
8. Змінні, що описують енергосистеми.

Опитування друге. Обмеження числа змінних відповідно до рейтингу їх важливості. Важливість змінних визначилася шляхом опитування експертів. Кожну змінну їм пропонувалося оцінити за двома параметрами: "абсолютна важливість" і "відносна важливість".

Абсолютна важливість визначалася за 7-ми бальною шкалою. Бал 1 позначав "не має значення", бал 7 – "виняткова важливість". Відносна важливість визначалася за 100 бальною шкалою. Порівнювалися змінні однієї і тієї ж категорії. Вибиралася найважливіша, з точки зору експерта змінна даної категорії, і їй присвоювався ранг 100.

Решта змінних порівнювалася зі змінною, визначеною як найважливіша. Так, змінна, що отримала рейтинг 50, вважається "двічі менш важливою". Відповіді експертів були оброблені. Приклад результатів обробки для підкатегорії "якість життя в процесі пересування" показаний в табл. 2.1. Зауважимо, що жодна змінна не має бала 7 та бала 100. Це означає, що експерти не зійшлися в думці, які змінні є найважливішими.

Таблиця 2.1

Результати обробки показників важливості для підкатегорії
"якість життя в процесі пересування"

№	Змінні	Медіана абсолютної важливості (від 1 до 7)	Середнє геометричне відносної важливості (від 1 до 100)
1	Комфорт пересування	4	68,7
2	Число зручностей (їдальня, бар, туалет і т.д.)	3	36,2
3	Імовірність затримки	5	82,3
4	Рівень шуму (всередині транспортного засобу)	4	50,5
5	Якість їзди	2	28,1
6	Рівень уваги, необхідної при їзді	1	13,2
7	Терпимість до переповнення транспортного засобу	4	58,3
8	Норма завантаження (число пасажирів на м ²)	4	68,9
9	Число пасажирів у транспортному засобі	3	42,7
10	Число місць у ряді крісел	3	59,3
11	Естетичне враження від маршруту	3	27,0
12	Зручність розкладу (частота рейсів)	6	85,6
13	Число рейсів на день	5	80,4

У табл. 2.2 змінні впорядковані за відносною важливістю. Упорядкування за відносною важливістю майже повністю збігається з впорядкуванням за абсолютною важливістю. Табл. 2.2 є ілюстрацією того факту, що метод оцінки за відносними або абсолютними значеннями, середнім або ще яким-небудь способом у більшості випадків не є вирішальним. Вирішальним є кваліфікація та оцінка експерта. Існує багато методів вибору змінних в якості концептів знакового графа, після того як деякий їх набір був визначений експертами після другого опитування.

Таблиця 2.2

Результати упорядкування змінних за відносною важливістю

№	Змінні	Місце за медіаною абсолютної важливості (від 1 до 7)	Місце за середнім геометричним відносною важливістю (від 1 до 100)
1	Зручність розкладу (частота рейсів)	1 (6)	1 (85,6)
2	Імовірність затримки	2 (5)	2 (82,3)
3	Число рейсів на день	2 (5)	3 (80,4)
4	Норма завантаження (число пасажирів на м ²)	3 (4)	4 (68,9)
5	Комфорт пересування	3 (4)	5 (68,7)
6	Кількість місць у ряду крісел	4 (3)	6 (59,3)
7	Терпимість до переповнення транспортного засобу	3 (4)	7 (58,3)
8	Рівень шуму всередині транспортного засобу	3 (4)	8 (50,5)
9	Число пасажирів у транспортному засобі	4 (3)	9 (42,7)
10	Число зручностей (їдальня, бар, туалет і т.д.)	4 (3)	10 (36,2)
11	Якість їзди.	5 (2)	11 (28,1)
12	Естетичне враження від маршруту	4 (3)	12 (27,0)
13	Рівень уваги, необхідного при їзді	6 (1)	13 (13,2)

Наприклад:

1. Вибрати одну або дві змінних з кожної підкатегорії, що мають найбільший абсолютний або відносний бал.

2. Вибрати змінні з кожної підкатегорії, що має абсолютну важливість, наприклад, не менше 6 балів і / або відносну важливість не менше 80 балів.

3. Вибрати всі змінні, незалежно від їх приналежності до тієї чи іншої підкатегорії, які мають високий бал, наприклад, не менше 5 абсолютної значущості або не менше 70 відносної. Правда, в цьому випадку не всі підкатегорії будуть представлені в списку обраних концептів.

4. Вибрати одну або дві змінні, отримані за кращою сумою балів абсолютної і відносної значущості.

Після отримання попереднього списку його склад може уточнюватися різними способами.

Зауважимо, що дуже важко оцінити, який із запропонованих методів краще або в яких випадках який метод застосовувати. Одна з причин такої невизначеності полягає в тому, що не визначається важливість підкатегорії, всі підкатегорії змінних вважаються однаково важливими, хоча насправді це не так. Більш того, в дослідженнях, що мають різні цілі, в залежності від призначення досліджувані підкатегорії можуть мати різну значимість (важливість).

У наведеному прикладі вибір здійснювався за максимальним значенням медіани абсолютної важливості змінної в кожній групі, а в тих випадках, коли максимальним значенням оцінювалося кілька змінних, вибиралася та, у якій більше середнє геометричне відносної важливості. Після додаткової обробки були відібрані змінні, показані в табл. 2.3.

Таблиця 2.3

Відбір змінних

№	Змінні	Скорочена назва	Категорія змінної
1	Число пасажиро-кілометрів (середнє всіх видах транспорту)	пасажиро-кілометри	описова
2	Економія пального (середнє за всіма видами транспорту)	економія пального	конструкторська
3	Чисельність населення	чисельність населення	демографічна
4	Вартість транспортного засобу	вартість транспортного	економічна
5	Вартість проїзного квитка (середнє за всіма видами транспорту)	вартість проїзного квитка	економічна
6	Кількість шкідливих продуктів згорання на пасажиро-кілометр (середнє за всіма видами транспорту)	згорання	забруднення

7	Аварії	аварії	впливу на навколишнє середовище і естетику
8	Ймовірність затримки	імовірність	якість життя
9	Загальна витрата пального (за всіма видами транспорту)	витрата пального	енергосистеми

Опитування третє. Вибір дуг, знаків, і побудова знакового графа. Після того, як обрано 9 змінних, експертам була відіслана анкета з проханням вказати значення відношень причинності для кожної пари змінних x, y з можливими базовими значеннями відношень причинності, розглянутими вище (+, -, 0, (+) і т.д.). Кожному експерту був виданий пакет з 72 карт, (на кожну пару змінних x, y – одна карта). Карти були перемішані, щоб мінімізувати ефект впливу порядку карт. Кожен експерт отримав пакет, в яких карти були покладені в різних порядках. Результати роботи експертів зведені в табл. 2.4.

Таблиця 2.4

Результати визначення експертами відношень причинності

Змінні x, y	Експерти							Результат 0 + - ?	Змінні x, y	Експерти							Результат 0 + - ?
	1	2	3	4	5	6	7			1	2	3	4	5	6	7	
1,2	+	0	0	0	0	+	+	430	5,6	0	-	0	+	0	+	+	331
1,3	0	0	0	0	0	0	0	700	5,7	0	-	0	+	0	+	0	421
1,4	0	0	-	0	+	-	0	412	5,8	0	-	0	+	0	-	-	313
1,5	-	-	+	+	+	+	-	034	5,9	0	0	0	+	-	+	+	331
1,6	+	+	0	+	+	+	0	250	6,1	+	0	0	0	-	-	-	313
1,7	+	+	+	+	+	+	+	070	6,2	-	+	0	0	+	-	-	223
1,8	0	+	+	+	+	+	+	160	6,3	0	0	0	0	-	-	-	403
1,9	+	+	+	+	+	0	+	160	6,4	-	+	0	0	-	-	0	313
2,1	-	0	0	+	+	0	+	331	6,5	0	-	0	0	0	+	0	511
2,3	0	0	0	0	0	0	0	700	6,7	+	+	+	0	0	+	+	250
2,4	-	(-)	-	0	0	-	0	403	6,8	0	0	0	0	0	+	0	610
2,5	-	-	0	-	-	-	-	106	6,9	+	-	0	0	+	-	-	223
2,6	-	-	-	-	-	-	-	007	7,1	0	0	0	-	+	-	-	313
2,7	0	0	0	0	-	+	0	511	7,2	0	(-)	0	-	0	0	0	601
1,8	0	0	0	0	-	-	0	601	7,3	-	(-)	0	-	0	-	-	304
2,9	-	-	-	-	-	-	+	016	7,4	0	+	0	-	0	0	0	511
3,1	+	+	+	+	+	+	+	070	7,5	0	+	? 0	0	+	0	+	4201

3,2	0 0 0 0 0 0 0	700	7,6	0 (+) 0 + 0 - 0	511
3,4	0 - 0 - 0 0 0	502	7,8	+ + + + + + +	070
3,5	0 (-) 0 - + 0 0	511	7,9	0 0 0 + 0 0 0	610
3,6	+ + + + 0 + +	160	8,1	0 - - - - -	106
3,7	+ + 0 + + + 0	250	8,2	0 - 0 0 - - -	304
3,8	+ + + + + + +	070	8,3	- 0 0 ? 0 0 0	5011
3,9	+ + + + + + +	070	8,4	0 0 0 0 + 0 0	610
4,1	0 0 0 0 - 0 0	60 1	8,5	0 + 0 0 ? + 0	4201
4,2	- (+) 0 0 - - +	313	8,6	0 + + 0 + + +	250
4,3	0 0 0 0 0 0 0	700	8,7	- 0 0 0 + + -	322
4,5	+ - 0 - + + 0	232	8,9	- + + 0 0 - +	232
4,6	+ - 0 0 0 - 0	412	9,1	+ 0 0 0 0 - 0	511
4,7	- (-) 0 0 0 - -	403	9,2	- - - - 0 - 0	205
4,8	0 0 0 0 0 0 0	700	9,3	0 0 0 0 0 0 0	700
4,9	+ + 0 0 + - 0	331	9,4	+ 0 0 0 0 + 0	520
5,1	+ - - - - - -	016	9,5	+ + 0 - 0 - 0	322
5,2	0 + - 0 0 0 0	511	9,6	+ + + + + - +	061
5,3	0 0 0 0 0 0 0	700	9,7	+ 0 0 0 0 0 0	610
5,4	+ 0 0 - + + 0	331	9,8	0 0 0 0 + - 0	511

Примітка: знаки (+) і (-) оцінюються як 0.

Формування табл. 2.4 мала б виконати система підтримки прийняття рішень після введення в неї оцінок, даних експертами. Фактично, експерти повинні були відповісти на два питання:

- чи повинна бути дуга (відношення причинності) між вершинами графа x і y ,
- який знак цієї дуги.

При обробці табл. 2.4 була прийнята наступна методика:

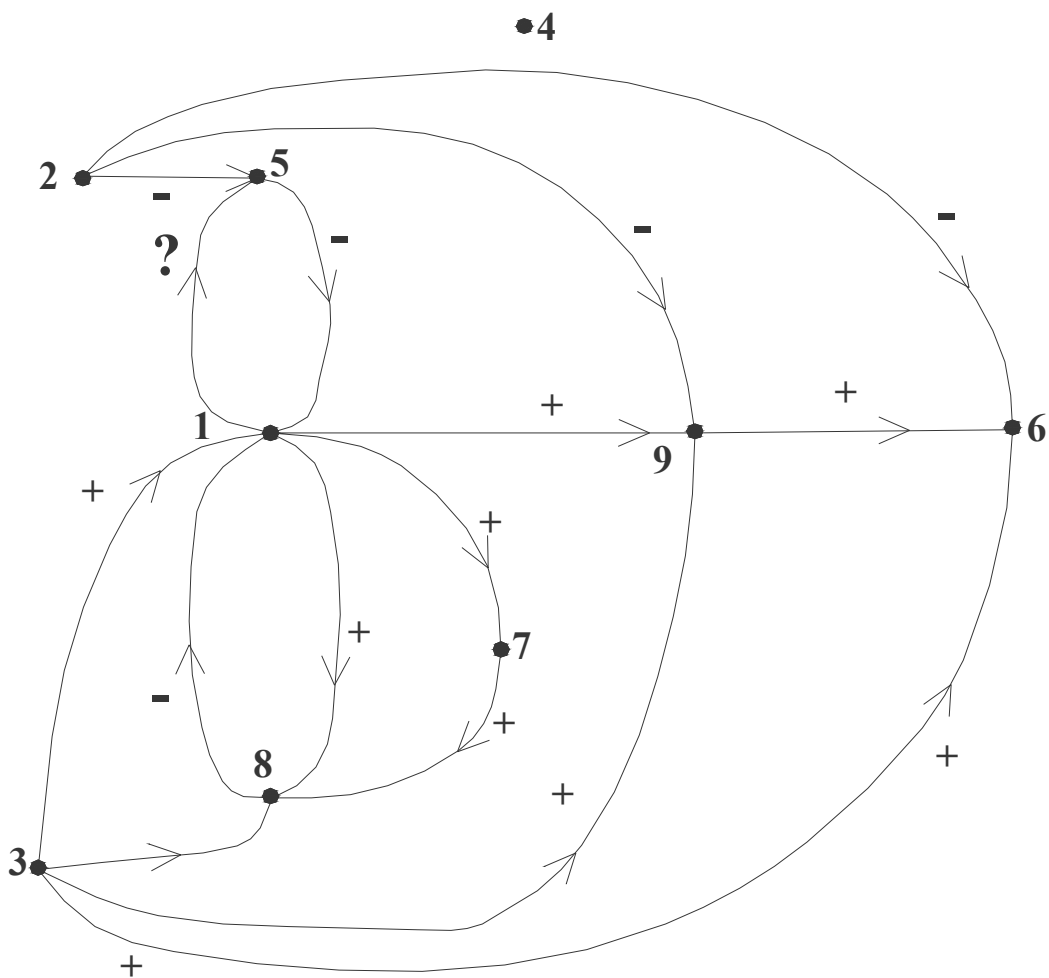
- якщо принаймні 6 з 7 експертів вважають, що вершина x суттєво впливає на вершину y , то проводиться дуга (x, y). Якщо таку оцінку давало менше число експертів, то вважалось, що відношення причинності між цими вершинами (концептами) немає;

- знак дуги (значення відношення причинності) проставлявся тільки у тому випадку, коли думка 60% експертів збігається. У нашому прикладі (див. табл. 2.4) істотна розбіжність між експертами відбулася тільки по дузі (1, 5). Всі сім експертів вважали, що вершина 1 робить істотний вплив на вершину 5 (пасажиро-кілометри на вартість проїзду). Але три експерта вважали, що вплив додатний, а чотири – від’ємний. У результаті аналізу даних експертів був отриманий граф, показаний на рис. 2.6. У зв’язку з розбіжностями експертів по знаку дуги (1, 5) на ній проставлений «?».

Знаковий граф може бути використаний для якісної оцінки впливу окремих вершин знакового графа на стійкість системи. Тобто, як зміниться стійкість системи, якщо деякі зовнішні сили нададуть досить сильний вплив на один або декілька концептів (вершин знакового графа) або змінять знак дуги (дуг) – значення відношення причинності.

Назвемо систему, що описується знаковим графом, *стійкою*, якщо значення всіх тих змінних у вершинах графа, на які не чинять вплив зовнішні фактори, не змінюються під впливом будь-яких змін значень однієї або декількох змінних, викликаних впливом зовнішніх факторів. Назвемо змінну x *стабільною початковою змінною*, якщо різке збільшення її значення не призведе до збільшення значень інших змінних системи.

Пояснимо сказане на прикладі знакового графа рис. 2.6.



1 – пасажиро-кілометри; 2-економія пального; 3 – чисельність населення; 4 – вартість транспортного засобу; 5 – вартість проїзного квитка; 6 – згоряння; 7 – аварії; 8-ймовірність затримки; 9 – витрата пального

Рис. 2.6 Знаковий граф

У табл. 2.5 показано зміну стійкості знакового графа за умови, що дуга (1, 5) має значення «+» залежно від зміни знаків його ребер. Стійкість графа визначається характером його зворотнього зв'язку. Стабільними початковими змінними в табл. 2.5 виявилися ті змінні, які або не впливають на інші змінні (вершини 4 і 6) або безпосередньо впливають на ті, які не впливають на інші (вершина 9).

У табл. 2.6 показано зміну стійкості знакового графа за умови, що дуга (1,5) має значення «-». У цьому випадку всі вершини виявляються стабільними початковими змінними, тому що збільшення значення одних змінних компенсується зменшенням інших, те ж саме відбудеться, якщо поміняти знаки всіх дуг. Також легко проінтерпретувати і проміжні зміни, показані в табл. 2.6.

Таблиця 2.5

Приклад оцінки стійкості знакового графа за умови, що дуга (1, 5) має значення «+»

Зміна дуг знакового графа	Оцінка стійкості знакового графа	Стабільні початкові змінні в зміненому графі	Нестабільні початкові змінні в зміненому графі
не змінюються	нестійкий	(4,6,9)	(1,2,3,5,7,8)
Змінено знаки дуг (1, 8), (5,1)	стійкий	всі	жодного
Змінено знаки всіх інших дуг	нестійкий	(4, 6, 9)	(1,2,3,5,7,8)

Таблиця 2.6

Приклад оцінки стійкості знакового графа за умови, що дуга (1,5) має значення «-»

Зміна дуг знакового графа	Оцінка стійкості знакового графа	Стабільні початкові змінні в зміненому графі	Нестабільні початкові змінні в зміненому графі
не змінюються	стійкий	все	жодної
змінені знаки дуг (1,7), (7, 8)	нестійкий	(3, 4, 6, 9)	(1,2,5,7,8)
змінені знаки дуг (1,8), (5,1), (8,1)	нестійкий	(4, 6, 9)	(1,2,3,5,7,8)
змінені знаки всіх інших дуг	стійкий	все	жодної

У табл. 2.7 показано зміну стійкості знакового графа за умови, що дуга (1,5) має значення 0.

Таблиця 2.8 показує стабільність вершини графа в залежності від умов табл. 2.5-2.7.

Аналіз табл. 2.5-2.8 показує:

1. Якщо ціни на проїзні квитки є функцією загальної відстані (кілометражу) поїздки пасажирів і ціна поїзду одного кілометра зменшується зі збільшенням відстані, тоді система споживання пального (для міжміських перевезень) буде стійкою.

2. Якщо ціни на пасажирські квитки не залежать істотно від відстані, яку проїхав пасажир, або ціна за кілометр збільшується зі збільшенням відстані – система стає нестійкою.

3. Якщо дуга (1, 5) має знак мінус, тоді різке збільшення населення веде до нестабільності тільки у випадку, якщо дуги (1, 8) і (5, 1) або (8, 1) змінюють знак. Аналогічно, якщо дуга (1,5) має знак плюс – будь-яке збільшення ціни квитків веде до нестабільності.

4. Зауважимо, що в нашому прикладі ціна на квиток розглядається як постійна величина. Однак у період годин пік ціни на квитки можуть бути підняті, а в години, менш зручні для поїздок, знижені. У результаті ретельного урахування інтересу і потреб пасажирів можна таким чином підвищити ціну за поїзд одного кілометра пасажиром, не знижуючи числа пасажирів. На графі рис. 2.6 така можливість не відображена. Відображення на знаковому графі більш складних залежностей, ніж це показано на рис. 2.6, сильно ускладнило б граф і ускладнило б аналіз залежностей між змінними в його вершинах.

5. Аналогічно можна було б розглянути можливість зміни знака дуги (1,8) (пасажиро-кілометри, ймовірність затримки) з плюса на мінус. Якщо розклад руху скласти так, щоб в години пік потреба у транспортних засобах скоротилася, тоді незважаючи на зростання пасажиро-кілометрів ймовірність затримок могла б скоротитися.

6. На знаковому графі рис. 2.6 вплив вартості транспортного засобу на систему споживання пального є незначною, оскільки вершина 4 не пов'язана дугами з іншими вершинами. Введення нової змінної обміну відносно зношених транспортних засобів на нові з зовнішнім власником нових транспортних засобів може виявитися серйозним фактором зниження вартості перевезень. Таким чином, правильний і досить повний вибір вершин є важливим фактором успішного проведення аналізу. Зауважимо, що і в цьому випадку виникає проблема великого числа вершин і складності графа.

З пунктів 4, 5, 6 видно, що виникає серйозне протиріччя між ступенем детальності знакового графа і можливостями аналізу, що часто впливає на достовірність, цінність і змістовність аналізу викладеними вище методами.

7. Цикл 1-7-8-1 від'ємний, цикл 1-8-1 теж від'ємний, а знак циклу 1-5-1 не визначений. Цікаво зауважити, що знак (-) дуги (1, 5), що стабілізує обстановку, робить цикл додатним, а не від'ємним. Це ілюстрація того, що додатні цикли іноді стабілізують ситуацію. У нашому прикладі стабілізуючу роль відіграє протидія від'ємного циклу 1-8-1. Дійсно, якщо невизначений знак дуги (1, 5) вважати плюсом, тоді цикл 1-5-1 стає від'ємним і ситуація виявиться нестійкою доки знак дуги (5, 1) не зміниться так, щоб цикл 1-5-1 став знову додатним чи знак дуги (1,8) зміниться так, щоб інший цикл, а саме 1-8-1, став додатним.

Пункт 7 в деякій мірі ілюструє складність аналізу за допомогою когнітивних карт.

Таблиця 2.7

Приклад оцінки стійкості знакового графа за умови, що дуга (1.5) має значення «0»

Зміна дуг знакового графа	Оцінка стійкості знакового графа	Стабільні початкові змінні в зміненому графі	Нестабільні початкові змінні в зміненому графі
не змінюються	нестійкий	(4, 6, 9)	(1,2,3,5,7,8)
змінити знаки всіх інших дуг	нестійкий	(4, 6, 9)	(1,2,3,5,7,8)

Таблиця 2.8

Стабільність вершини графа

Вершини	Стабільність поведінки
1, 6, 7, 8, 9	Необмежені зміни залежно від значень імпульсів в будь-якому з нестабільних початкових змінних, що впливають на них
2, 3, 4	Обмежені зміни незалежно від початкових змінних
5	Необмежені зміни залежно від імпульсів нестабільних початкових змінних за умови, що існує дуга (1,5)

Ще більшу складність викликає бідність образотворчих засобів: слабкий, сильний і дуже сильний додатний вплив позначається на знаковому графі одним символом – плюс (або мінус). Більше того: в одному діапазоні значень відношення між двома змінними може бути додатним, а в іншому-від’ємним. Для подолання цієї труднощі часто використовують лінгвістичні змінні замість знаків дуги, що хоча і робить когнітивну карту більш змістовною, але ускладнює її аналіз.

У знаковому графі відсутнє поняття часу. Нарешті необхідно розрізняти послідовність і паралельність впливу або виконання врахованих чинників, чого не можна зробити в знаковому графі.

Проте, незважаючи на зазначені недоліки, когнітивні карти дозволяють оцінити можливі наслідки деяких рішень і визначити, чи можуть ці рішення дестабілізувати систему і які якісні зміни відбудуться у вузлах системи після того, як ці рішення будуть виконані.

Розглянемо ще два приклади, на яких надалі будуть показані деякі методи прийняття рішень, у тому числі й ті, які долають перераховані вище недоліки методу когнітивних карт.

Приклад. Ліквідація наслідків аварійного розливу нафти [20].

Для ліквідації наслідків розливу нафти в гавані можливе виконання наступного набору операцій (дій).

- A. Активізація командного центру.
 - B. Збір та приведення в готовність місцевого обладнання.
 - C. Закриття гавані.
 - D. Доставка додаткового обладнання повітряним шляхом.
 - E. Створення механічних загороджень.
 - F. Розпилення хімічних речовин.
 - G. Використання механічних засобів для зняття плівки.
 - H. Використання біологічних речовин.
 - I. Використання горючих речовин.
 - J. Очищення узбережжя.
- Побудуємо табл. 2.9 відношення причинності.

Таблиця 2.9

Таблиця відношень причинності

	A	B	C	D	E	F	G	H	I	J
A		(+)	(+)							
B				(+)	(+)	(+)	(+)	(+)	(+)	
C										
D					+	+	+	+	+	
E							-		-	
F								-		-
G								-		-
H										-
I							-			
J										

У даному випадку відношення причинності добре відомі. Вони можуть зберігатися в базі даних, і система підтримки прийняття рішень видає їх на дисплеї для аналізу ОПР. Якщо ОПР не вносить змін, то система підтримки прийняття рішень видає на дисплей знаковий граф, показаний на рис. 2.7.

Може бути і зворотний порядок. ОПР малює схему типу показаної на рис. 2.7, а СППР видає таблицю типу 2.9. Зауважимо, що граф рис. 2.7 немає сенсу аналізувати на стійкість. Якщо прийняти деяку дозволу на графі послідовність дій, наприклад. А, В, С, F, G, H, J, то гавань буде очищена від нафти, що розлилася.

Цей же результат може бути отриманий і при іншій дозволений послідовності дій. Тут постає інше питання, на яке метод когнітивних карт не дає відповіді: яка послідовність дешевше, при якій послідовності дій очистка гавані буде виконана за найкоротший термін, яка послідовність дій безпечніше і т.д.

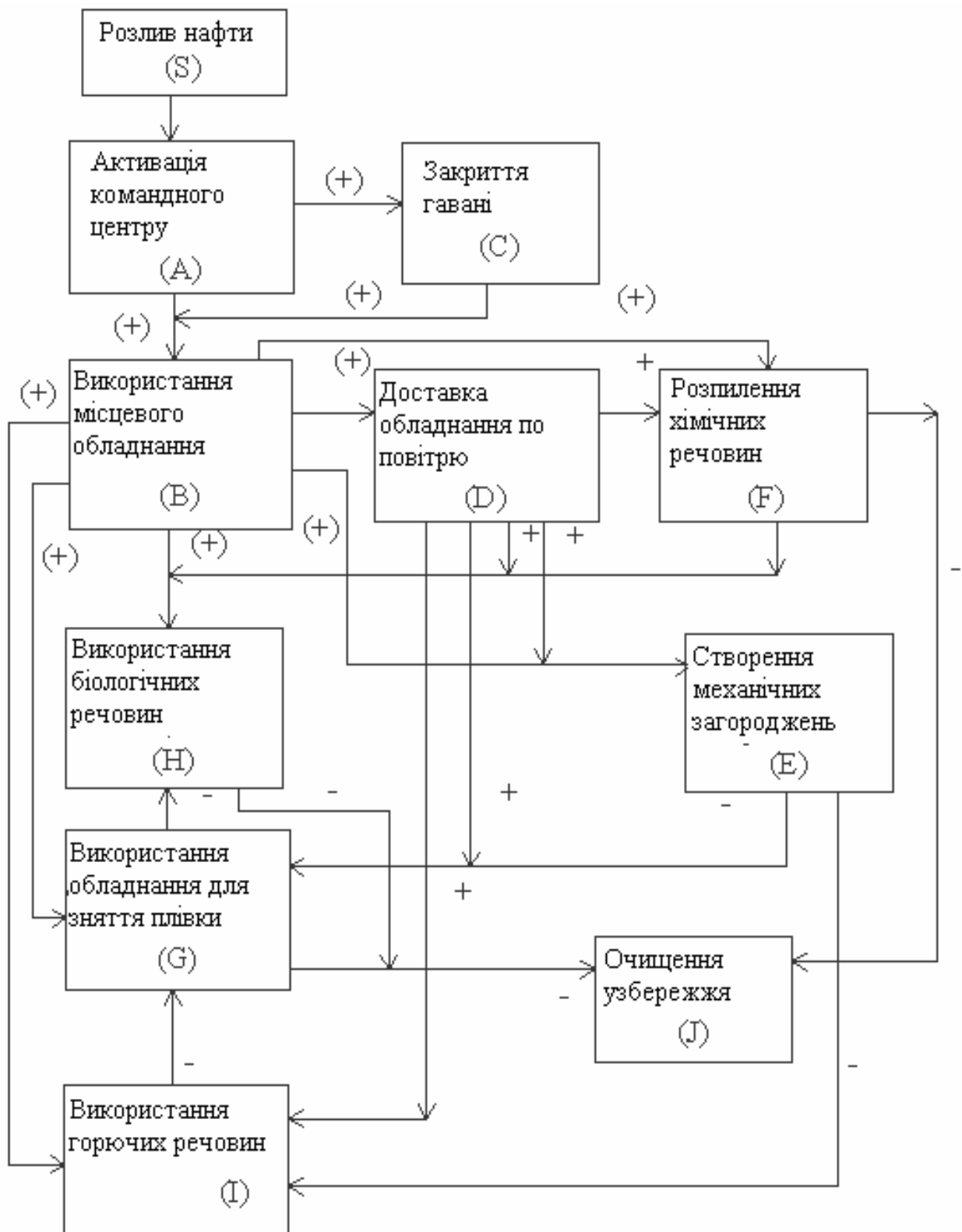


Рис. 2.7 Знаковий граф системи ліквідації наслідків аварійного розливу нафти

Приклад С. Боротьба з сепаратизмом.

Відзначимо відразу, що наведені в прикладі варіанти рішень та їх оцінки не є відображенням думки будь-яких експертів.

На деякій території виникли сепаратистські сили, що борються за відділення її від держави. Центральний уряд оцінює обстановку і розглядає можливі варіанти рішень боротьби з сепаратизмом.

Розглядаються наступні варіанти дій:

1. Укладення договору між центральним урядом і адміністрацією території. Розмежування повноважень з наданням адміністрації широких прав.
2. Зміна адміністрації території, яка прагне до відокремлення від держави.
3. Перекриття енергоканалів.
4. Блокування території.
5. Придушення сепаратистів військовою силою (військові дії).

Бездіяльність (стан на момент прийняття рішення).

Побудуємо табл. 2.10 відношень причинності, аналогічну табл. 2.4, вважаючи, що обробка результатів опитування експертів вже проведена і в табл. 2.10 представлені результати обробки.

Зауважимо, що можливі варіанти дій цими шістьма варіантами не вичерпуються. Підкреслимо, що значення відношень причинності в табл. 2.10 не безперечні.

Таблиця 2.10

Таблиця відношень причинності

	1	2	3	4	5	6
1						
2	+				+	
3	+	+		0	+	
4	+	+	0		+	
5	+	+	0	0		
6	-	-	0	0	-	

Таблиця 2.10 може бути побудована ОПР безпосередньо на дисплеї або отримана системою підтримки прийняття рішень в результаті обробки даних експертів. За табл. 2.10 СППР будує знаковий граф, показаний на рис. 2.8.

Цей знаковий граф дозволяє провести аналіз стійкості системи так само, як граф рис. 2.5. Але це робитиметься не буде не тільки в силу досить складних відношень причинності, дуже ускладнюють такий аналіз, а за набагато серйозної причини.

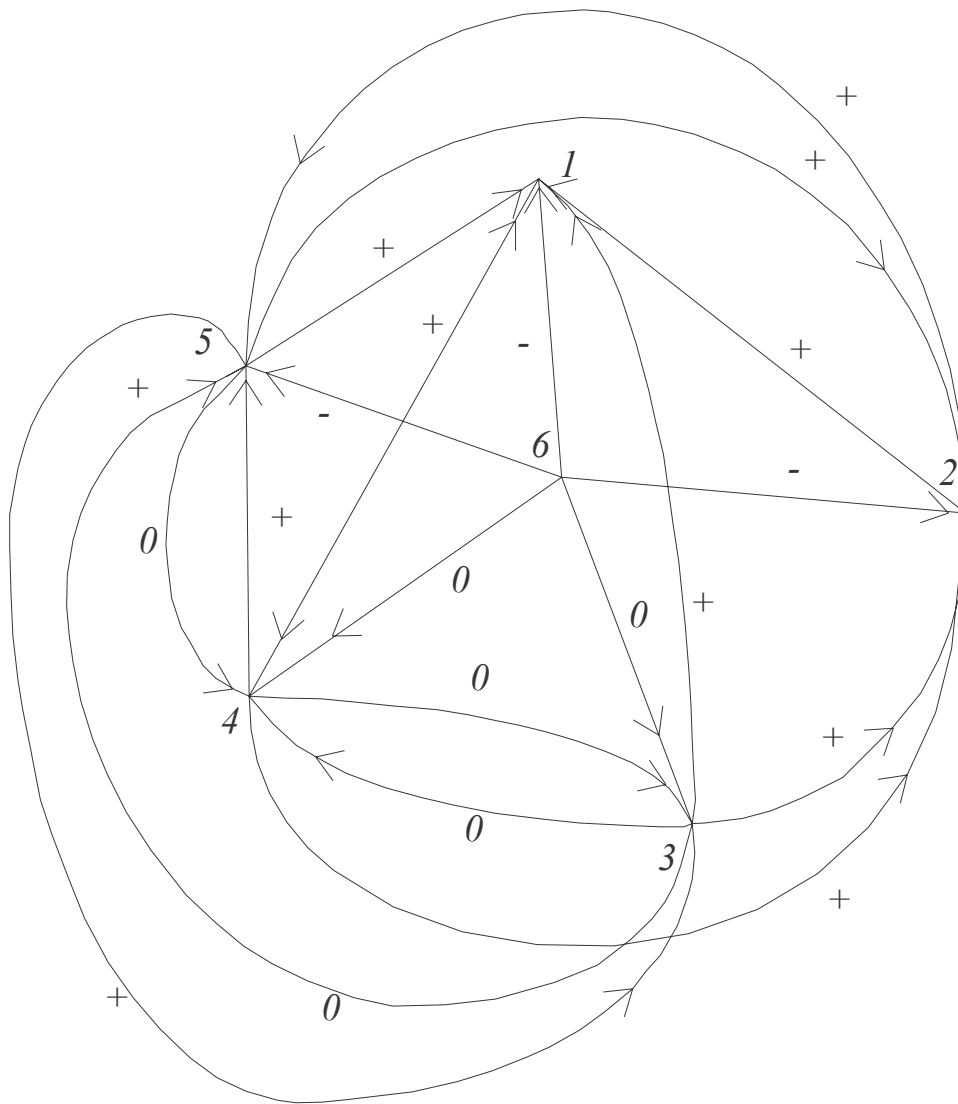


Рис. 2.8 Знаковий граф системи боротьби з сепаратизмом

Знаковий граф рис. 2.7 не відображає реакції противника, тобто сепаратистів на дії центрального уряду. І навіть якщо створити систему, зовсім стійку за критеріями методу когнітивних карт, активні дії навколишнього середовища, в даному випадку сепаратистів, можуть зруйнувати самі ретельно розроблені плани.

3 ДЕКОМПОЗИЦІЯ СИСТЕМИ. СИЛЬНО ЗВ'ЯЗНІ ОБЛАСТІ

Аналізувати граф, що містить багато концептів, вкрай складно. Можна зробити декомпозицію графа, розбивши його на деяке число областей, а потім провести аналіз як кожної частини, так і узагальненого графа. В узагальненому графі кожна поділена область є вершиною графа. Одним з можливих методів декомпозиції графа є розбиття його на тісно пов'язані області.

У багатьох випадках краще йти не "згори вниз", тобто не створювати спочатку дуже складний граф, а потім проводити його декомпозицію, а відразу виділити деякі області і з них будувати узагальнений граф.

Нехай G – спрямований граф, а B – підмножина його вершин.

Будемо називати U *сильно зв'язаною областю* в G , якщо:

- в B існує така єдина вершина a , що знайдеться шлях з початкової вершини графа G в a , що не проходить ні через який інший ділянку з B ;
- існує шлях (ненульовий довжини), що лежить цілком всередині B і веде з будь-якої вершини B в будь-яку іншу вершину в B .

В даний час розроблено досить багато алгоритмів виділення тісно зв'язаних областей, вони досить складні. Для пояснення ідей виділення тісно зв'язаних областей, наведемо один з найпростіших.

3.1 Алгоритм знаходження сильно зв'язаних областей у графі

Будемо розглядати сильно зв'язані області R_j , що характеризуються такими властивостями:

1. R_j є сильно зв'язною областю, що складається з множини вузлів, що представляють підсистеми, в тому числі й елементарні підсистеми, кожен з яких передуює сам собі (а значить і слід сам за собою) всередині цієї множини.

2. $R_i \neq R_j$.

3. Для кожного $i < j$ або $R_i \cap R_j = \emptyset$, або $R_i \cap R_j = R_i$, тобто $R_j \in R_i$.

Приклад. Граф на рис. 3.1 утворює наступний список сильно зв'язаних областей **5, 5-6, 2-7-3-4, 3-4**.

Вузли **2-7** і **2-3-4** також утворюють сильно зв'язані області, але якщо в список сильно зв'язаних областей включити як область **2-7**, так і область **2-3-4**, то порушується умова 3, оскільки $2-7 \cap 2-3-4 \neq 2-3-4$.

Тісно зв'язані області можна визначати неформальним шляхом.

Приклад. Граф рис. 3.1 може бути розбитий на наступні області так, як це показано в табл. 3.1 і на рис. 3.2-а.

Зауважимо, що один і той же вузол може входити в декілька тісно зв'язаних областей, але при цьому обов'язково менша область залишається підмножиною більшої.

На рис. 3.2-б показані підсистеми другого рівня, що входять до граф рис. 3.2-а.

Вершини 3 і 4 утворюють область третього рівня.

Кожна тісно зв'язана область може розглядатися як деяка підсистема, що "живе своїм життям", але пов'язана з усією системою.

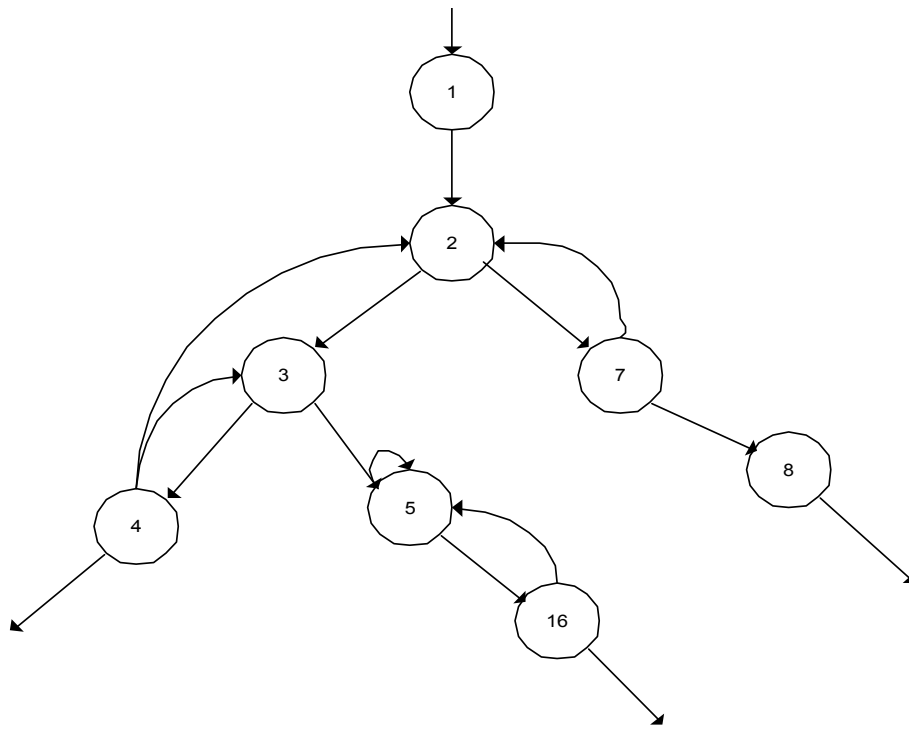


Рис. 3.1 Приклад графа, що складається з сильно зв'язаних областей

Таблиця 3.1

Перелік тісно зв'язаних областей графа (рис. 3.1)

Номер підсистеми	Вершини, що входять в систему
1	1
2	5
3	8
4	3,4
5	5,6
6	2,7
7	2,3,4,7

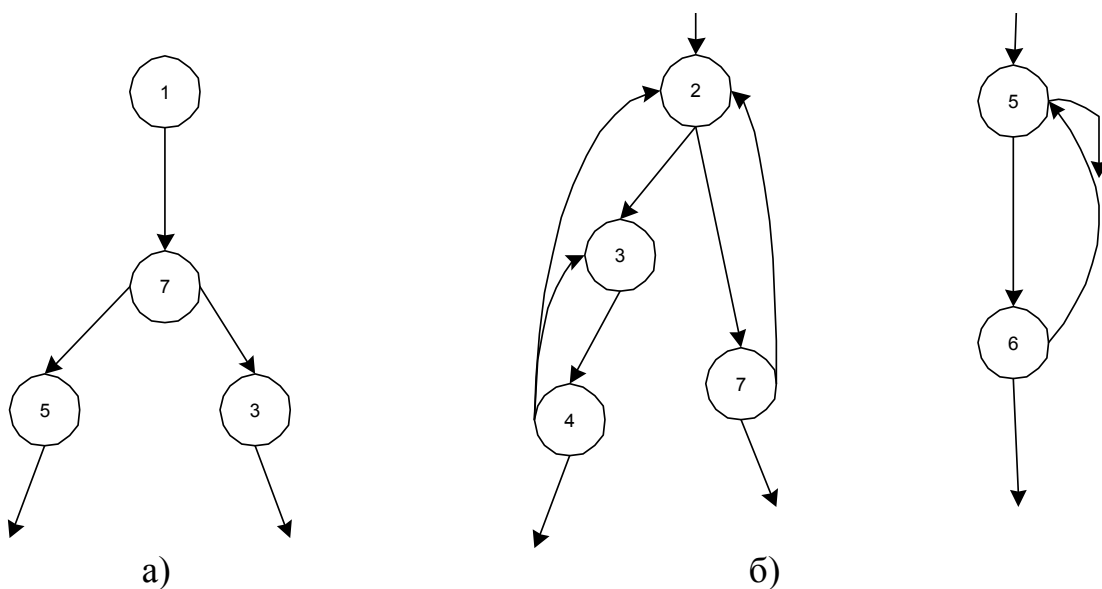


Рис.3.2 Сильно зв'язані області графа (рис. 3.1)

Граф системи рис.3.1 може бути представлений графом рис. 3.2-а, значно простішим і легшим для аналізу. Підграфи рис. 3.2-б можуть аналізуватися окремо.

Тепер розглянемо алгоритм знаходження сильно зв'язкових областей, що характеризуються властивостями 1-3 цього параграфа.

Алгоритм розглянемо на прикладі графа, показаного на рис. 3.1.

1. Формується таблиця із зазначенням номерів підсистем, на які передається управління при виході з даної підсистеми. Такою таблицею для графа на рис. 3.1 є табл. 3.2.

Таблиця 3.2

Підсистеми графа (рис. 3.1)

Номер вершини	Номер вершини, на яку передається керування
1	2
2	3,7
3	4,5
4	2,3, ВИХІД
5	5,6
6	5, ВИХІД
7	2,8
8	ВИХІД

2. Формується булева матриця зв'язку C розмірності $n \times n$, де n – число вершин у графі. Елемент матриці $C_{ij} = 1$, якщо вершина j безпосередньо слідує за вершиною i . В іншому випадку $C_{ij} = 0$.

Матриця зв'язку для графа (рис. 3.1) показана в табл. 3.3.

Таблиця 3.3

Матриця зв'язку для графа (рис. 3.1)

	1	2	3	4	5	6	7	8
1	0	1	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0
3	0	0	0	1	1	0	0	0
4	0	1	1	0	0	0	0	0
5	0	0	0	0	1	1	0	0
6	0	0	0	0	1	0	0	0
7	0	1	0	0	0	0	0	1
8	0	0	0	0	0	0	0	0

Вводяться булеві вектори \mathbf{R}_k , що визначають вузли графа на деякому замкнутому шляху k . Елемент вектора $\mathbf{R}_{kj} = 1$, якщо через вузол j проходить замкнутий шлях k .

Для знаходження векторів \mathbf{R}_k :

а) знаходяться вершини, що безпосередньо самі за собою, і такі, для яких $C_{ii} = 1$. У матриці табл. 3.3 такою вершиною буде п'ята: $C_{55} = 1$, звідси $\mathbf{P}_1 = 00001000$, тобто сильно пов'язана область, яку позначимо номером 1, складається з одного вузла 5;

б) вводиться матриця C^1 ($1 = 1, 2, \dots, n$) довжини шляхів (не замкнутих) з вершини i до вершини j . Елемент матриці $C^1_{ij} = 1$, якщо з вершини i в вершину j існує шлях довжиною в 1. Зауважимо, що якщо $C^1_{ii} = 1$, то $C^2_{ii} = 1$, $C^3_{ii} = 1$ і т.д.

Якщо вершини i і j знаходяться на одному замкнутому шляху довжиною 1 і $C^1_{ii} = 1$, то і $C^1_{jj} = 1$.

Для знаходження матриці C^{1+1} за матрицею C^1 позначимо через C^1_i рядок i матриці C^1 . Попередньо будемо вважати, що всі елементи матриці C^{1+1} дорівнюють нулю ($C^{1+1}_{ij} = 0$, $i, j = 1, 2, \dots, n$). Тоді якщо вершина j безпосередньо слідує за вершиною i , то $C^{1+1}_i = C^1_j \vee C^{1+1}_i$.

Пояснимо цю формулу на прикладі графа рис.4.1. За вершиною 2 безпосередньо слідує вершини 3 і 7.

$$C_3' = 00011000, C_7' = 01000001.$$

За нашою умовою на початку обчислення $C_2^2 = 00000000$ (матриця C^2 попередньо обнулена за формулою $C_2^2 = C_3^1 \vee C_7^2 = 00011000$).

Це перший крок в обчисленні значення $C_2^2 = C_7^1 \vee C_2^2 = 01000001 \vee 00011000 = 01011001$. Це остаточне значення C_2^2 .

Ясно, що якщо за третій вершиною графа безпосередньо слідувала тільки одна вершина, то для обчислення C_2^2 потрібно було б зробити тільки один крок, якби за неї слід було n вершин – n кроків.

Отриманий результат говорить, що шлях довжини 2 пов'язує вершину 2 з вершинами 4, 5 і 8, а також із самою вершиною 2, що легко перевірити безпосередньо.

Таким чином знаходяться всі рядки C^1 ($1 = 2, 3, \dots, n$). Зауважимо при цьому, що для отримання матриці C^{1+1} потрібно пам'ятати не всі попередні матриці, а тільки C^1 , тому процес легко організовується рекурсивно;

в) для знаходження найкоротшого шляху від вершини i до вершини j кожній матриці C^1 ставиться у відповідність матриця \tilde{D}^1 , елементами якої є цілі числа, що визначають довжину найкоротшого шляху між відповідними вершинами.

$\tilde{1}$ вказує на відповідність матриці \tilde{D}^1 матриці C^1 , позначає довжину шляху не більше 1.

Елемент матриці \tilde{D}^1

$$\tilde{d}_{ij}^{l-1}, \text{ якщо } \tilde{d}_{ij}^{l-1} \neq 0;$$

$$\tilde{d}_{ij}^1 = 1, \text{ якщо } C_{ij}^1 = 1 \wedge \tilde{d}_{ij}^{l-1} = 0 \ (\tilde{1}, i, j = 1, 2, \dots, n);$$

$$0, \text{ якщо } C_{ij}^1 = 0 \wedge \tilde{d}_{ij}^{l-1} = 0.$$

Для графа рис. 4.1 и табл. 4.3

$$\tilde{d}_{22}^2 = 2; \tilde{d}_{28}^2 = 2; \tilde{d}_{24}^2 = 2; \tilde{d}_{25}^2 = 2; \tilde{d}_{23}^2 = 1; \tilde{d}_{27}^2 = 1, \text{ вектор } \tilde{d}_2^2 = 02122012$$

Таким чином, довжина шляху від вершини 2 до вершини 3 і 7 дорівнює 1, до вершин 2, 4, 5, і 8 дорівнює 2 і до вершин 1 та 6 шляху, довжиною не більше 2, немає. Це перевіряється безпосередньо;

г) замкнутий шлях і вектор R_k (що його визначає) знаходяться наступним чином:

1. Для матриці C' з найменшим значенням 1, у якій хоча б один з елементів $C'_{ij}=1$, виписуються вектори R_k ($k=1,2,\dots$), такі, що $R_{kj}=1$. Справді, якщо $C'_{ij}=1$, значить вузол j є своїм власним, безпосереднім приймачем і шлях замкнутий.

Для графа рис. 3.1 такою матрицею буде матриця Z табл. 3.3 з елементом $C_{55}=1$.

2. Для всіх $i \neq j$ ($1 \leq i \leq n$) матриці D^* , якщо її елементи $d_{ij}^1 \neq 0$ і $\bar{d}_{ji}^1 \neq 0$, а $\bar{d}_{ij}^1 + \bar{d}_{ji}^1 \leq 1$, $R_{ki}=1$, переглядаючи всі i , закінчуємо формування вектора R_k .

Якщо $\bar{d}_{ij}^1 \neq 0$, значить існує шлях довжини не більш 1 від вершини i до вершини j , якщо $\bar{d}_{ji}^1 \neq 0$, то існує зворотний шлях від вершини j до вершини i довжиною теж не більше 1. Нарешті, якщо $\bar{d}_{ij}^1 + \bar{d}_{ji}^1 \leq 1$, значить існує замкнутий шлях від вершини i до вершини j завдовжки не більше 1. Остання умова введена для того, щоб розглядати тільки ті шляхи, довжина яких не більше 1.

Необхідно відзначити, що поділ системи на підсистеми можна провести різними методами, в багатьох випадках з урахуванням ваг вершин і дуг. Але чим складніше вимога до розбиття, тим складніше алгоритм.

3.2 Програма знаходження сильно зв'язаних областей у MATLAB

Наведемо текст програми, яка призначена для знаходження сильно зв'язаних областей за матрицею суміжності.

Знаходимо матрицю суміжності C . За нею обчислюємо матрицю досяжності за формулою:

$$A = \mathbf{B}[(E + C)^{n-1}],$$

де E – одинична матриця, n – число вершин графа.

Обчислюємо матрицю $R = A^2$ та матрицю D – поелементного добутку матриці A . Елемент головної діагоналі матриці R показує, скільки вершин входить в сильно зв'язану область разом з вершиною, номер якої співпадає з номером цього елемента. Рядки матриці D дають інформацію про склад зв'язаних областей (в одному рядку елементи, що дорівнюють 1 – з однієї області).

Для графу (рис. 3.1) маємо у MATLAB:

```
C=[0    1    0    0    0    0    0    0;
    0    0    1    0    0    0    1    0;
    0    0    0    1    1    0    0    0;
    0    1    1    0    0    0    0    0;
    0    0    0    0    1    1    0    0;
    0    0    0    0    1    0    0    0;
    0    1    0    0    0    0    0    1;
    0    0    0    0    0    0    0    0]
```

```

n=8; % кількість вершин графа
E=eye(n,n); % одинична матриця
EC=(E+C)
n1=n-1
ECn1=EC^n1; % (E+C) у ступені (n-1)
for i=1:n % цикл для булевого перетворення матриці
досяжності
    for j=1:n
        if ECn1(i,j)==0
            A(i,j)=0;
        else
            A(i,j)=1;
        end
    end
end
A1=A'; %транспонування A
for i=1:n
    for j=1:n
        D(i,j)=A(i,j)*A1(i,j); % поелементний добуток
    end
end
R=A*A

```

Отримуємо результат:

<p>D =</p> <table style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table>	1	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	<p>R =</p> <table style="border-collapse: collapse; width: 100%;"> <tr><td>1</td><td>5</td><td>5</td><td>5</td><td>7</td><td>7</td><td>5</td><td>6</td></tr> <tr><td>0</td><td>4</td><td>4</td><td>4</td><td>6</td><td>6</td><td>4</td><td>5</td></tr> <tr><td>0</td><td>4</td><td>4</td><td>4</td><td>6</td><td>6</td><td>4</td><td>5</td></tr> <tr><td>0</td><td>4</td><td>4</td><td>4</td><td>6</td><td>6</td><td>4</td><td>5</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>2</td><td>2</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>2</td><td>2</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>4</td><td>4</td><td>4</td><td>6</td><td>6</td><td>4</td><td>5</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table>	1	5	5	5	7	7	5	6	0	4	4	4	6	6	4	5	0	4	4	4	6	6	4	5	0	4	4	4	6	6	4	5	0	0	0	0	2	2	0	0	0	0	0	0	2	2	0	0	0	4	4	4	6	6	4	5	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0																																																																																																																										
0	1	1	1	0	0	1	0																																																																																																																										
0	1	1	1	0	0	1	0																																																																																																																										
0	1	1	1	0	0	1	0																																																																																																																										
0	0	0	0	1	1	0	0																																																																																																																										
0	0	0	0	1	1	0	0																																																																																																																										
0	1	1	1	0	0	1	0																																																																																																																										
0	0	0	0	0	0	0	1																																																																																																																										
1	5	5	5	7	7	5	6																																																																																																																										
0	4	4	4	6	6	4	5																																																																																																																										
0	4	4	4	6	6	4	5																																																																																																																										
0	4	4	4	6	6	4	5																																																																																																																										
0	0	0	0	2	2	0	0																																																																																																																										
0	0	0	0	2	2	0	0																																																																																																																										
0	4	4	4	6	6	4	5																																																																																																																										
0	0	0	0	0	0	0	1																																																																																																																										

Елементи $r_{11} = 1$, $r_{88} = 1$ матриці R показують, що вершини 1 та 8 утворюють одноелементні області, а елементи $r_{22} = r_{33} = r_{44} = r_{77} = 4$ показують, що вершини 2, 3, 4 та 7 входять в одну сильно зв'язану область, що складається з чотирьох елементів; $r_{55} = r_{66} = 2$ показують, що вершини 5 та 6 входять в одну сильно зв'язану область, що складається з двох елементів (рис. 3.2-б).

4 СЦЕНАРІЇ – МОЗАЙКА ОПЕРАЦІЙ

4.1 Від когнітивних карт до наборів сценаріїв

Метод когнітивних карт дозволяє представити проблему у вигляді знакового графа, у вершинах якого позначено множину факторів, що характеризують проблему, а поміченими дугами визначається зв'язок між цими факторами і характер їх взаємного впливу.

Аналіз стійкості знакового графа в багатьох випадках є дуже корисним, проте абсолютно недостатнім для прийняття рішення щодо виниклої проблеми. Тому в даний час зроблені зусилля по доповненню методу когнітивних карт іншими методами, що дозволяють здійснити прийняття рішення.

Один з таких методів дозволяє генерувати можливі варіанти сценаріїв (варіанти рішень) із заданого набору операцій. Під сценарієм будемо розуміти послідовність дій, яка повинна привести до вирішення виниклої проблеми. Формування можливого сценарію і є варіант рішення.

Завдання генерації сценаріїв виникає досить часто в тих випадках, коли сценарії формується з деякого набору можливих дій (операцій). Цей набір може бути заданий когнітивної картою, експертної системою або безпосередньо.

Задача генерації всіх можливих сценаріїв може бути вирішена різними методами. Всі вони, так чи інакше, зводяться до перебору дуг графа, що описує проблему. Тому вирішення задачі можна надати і в термінах теорії графів. Однак з точки зору перспектив вирішення подібних задач для графів із значно складнішою структурою (наприклад, коли на елементи графа накладаються різні умови) більш перспективним є використання апарату формальних граматики.

Твірні граматики можна описувати як спрямованими графами, так і множиною граматичних правил, причому обидва ці описи ідентичні і взаємно однозначно відображаються один на одного. Тому, якщо є спрямований граф, що описує можливі технологічні послідовності виконання операцій, то з нього можна автоматично отримати твірну граматику, що описує цей процес. За її допомогою можна згенерувати всі можливі (допустимі) сценарії вирішення проблеми і таким чином перейти від когнітивної карти до набору сценаріїв.

4.2 Формальні граматики

Будь-яке висловлювання людини може бути виражене тільки на деякій мові. Мова – це множина речень або послідовностей слів. Кожне речення формується зі словника мови відповідно до заданих правил формування. Правила формування називаються *граматикою мови* і визначають її структуру. Однак такого інтуїтивного визначення мови та її граматики для формального аналізу і генерації зовсім недостатньо. Для опису мови також необхідно використовувати будь-яку мову. Мова, що використовується для опису граматики будь-якої мови, називається *метамовою*. Якщо деяка мова L_1 використовується для того, щоб описати деяку іншу мову L_2 , то мова L_1 називається метамовою над L_2 .

Характерна властивість природних мов (української, англійської і т.д.) полягає в тому, що вони можуть бути використані в якості своїх власних

метамов. І дійсно, граматики української мови може бути написана українською мовою. Природна мова допускає неоднозначність, тому описані на ній правила важко зробити однозначними і вже зовсім не можна зробити їх строго формалізованими.

В описі кожної мови, тобто в її граматиці, зазвичай розрізняють синтаксис і семантику.

Синтаксис – це множина формальних правил виводу правильно побудованих рядків або речень мови.

Семантика – смислове, змістовне значення кожного такого правила.

Для формального опису синтаксису розроблено декілька метамов, найбільшого поширення набула *метамова Бекуса*.

Метамова Бекуса – це список синтаксичних правил, що мають вигляд

$$U \rightarrow y,$$

де y – рядок символів, кожен з яких належить або до формальної мови, або до метамови; U – символ, що належить до метамови. Символ \rightarrow позначає слова "є", "це", "визначає". U називається лівою частиною правила, y – правою.

Наприклад:

$$\langle \text{цифра} \rangle \rightarrow 0$$

$$\langle \text{цифра} \rangle \rightarrow 1$$

$$\langle \text{цифра} \rangle \rightarrow 9$$

означають відповідно $\langle \text{цифра} \rangle$ це 0, $\langle \text{цифра} \rangle$ це 1, ..., $\langle \text{цифра} \rangle$ це 9.

Тут поняття $\langle \text{цифра} \rangle$ задано простим перерахуванням.

Таке перерахування можна задати також наступним чином:

$$\langle \text{цифра} \rangle \rightarrow |1|2|3|4|5|6|7|8|9$$

У складніших випадках поняття вводяться рекурсивно.

Наприклад:

$$\langle \text{ідентифікатор} \rangle \rightarrow \langle \text{буква} \rangle$$

$$\langle \text{ідентифікатор} \rangle \rightarrow \langle \text{ідентифікатор} \rangle \langle \text{буква} \rangle$$

$$\langle \text{ідентифікатор} \rangle \rightarrow \langle \text{ідентифікатор} \rangle \langle \text{цифра} \rangle$$

означає: ідентифікатор – це послідовність букв і цифр, що починаються з літери. Поняття $\langle \text{ідентифікатор} \rangle$ зустрічається у правій частині синтаксичного правила, вже рекурсивно визначено в першому випадку як буква, у другому – як послідовність букв, в третьому – як послідовність букв і цифр, що починається з літери.

Для опису семантики загальноновизнаних формалізованих методів поки немає. Їх відсутність викликає серйозні труднощі в усуненні неоднозначності формальних мов.

Наприклад, в програмі:

```
begin array A[4], integer I, k;
```

```
  I:=2; k:=3;
```

```
  for A[I]:=k step 1 until 4 do begin
```

```
    I:=I+1; k:=k+1;
```

```
  end;
```

```
end;
```


можливо принаймні два тлумачення:

а) перший крок

$A[2]:=3; I:=2+l; k:=3+l; A[2]<4$ процес триває;

другий крок

$A[2]:=4; I:=3+l; k:=4+l; A[2]:=4$ процес закінчено;

б) перший крок

$A[2]:=3; I:=2+l; k:=3+l; A[2]<4$ процес триває;

другий крок

$A[3]:=4; I:=3+l; k:=4+l; A[2]<4$ процес триває нескінченно.

Фактично неоднозначність у формальній мові, в тому числі і в мові програмування, не допускається. Але поки це досягається не тільки методами формального опису мови, але засобами трансляції або генерації.

Формальні мови мають наступні особливості:

- незалежність синтаксичних правил від сенсу,
- неможливість появи парадоксів,
- строгість синтаксичних правил та відсутність винятків,
- кінцеве число синтаксичних правил.

Для того, щоб описати формальну мову, як і будь-яку мову, в тому числі і природну, необхідно задати її алфавіт, словник і правила побудови речень, тобто граматику мови.

Алфавіт – це не порожня кінцева множина елементів. Назвемо елементи мови символами.

Надалі не будемо робити відмінностей між словом мови або метамови і знаком алфавіту, об'єднавши ці два поняття в поняття "*символ*".

Символом будемо називати мінімальну за кількістю знаків конструкцію, яка використовується в даній формальній мові або метамові і має самостійне значення. Це може бути одна буква, наприклад A, B, L ; знак операції, наприклад $+, >, \wedge$, роздільник, наприклад: $[,]$, одне слово або група слів, наприклад *begin, go to* і т.д.

Необхідно окремо розглянути випадок групи слів, оскільки у багатьох формальних мовах мають місце випадки, коли сенс мають лише певні комбінації слів, а кожне окреме слово в цій комбінації сенсу не має. Символи будемо позначати: словами (під словом слід розуміти і окрему букву), написаними великими латинськими літерами; словами, написаними прописними українськими літерами, взятими в кутові дужки, знаками операцій і роздільників. Наприклад: $A, begin, <идентифікатор>, (, +, B, C, *, <блок>$ є символами мови.

Множину символів мови позначимо через V і розділимо на дві підмножини.

1. Підмножина символів, що належать власне формальній мові. Цю підмножину назвемо *підмножиною кінцевих або термінальних символів* і позначимо його через V_T . Символи цієї підмножини називаються *кінцевими (термінальними)*, тому що в дереві тексту (див. нижче) ці символи завжди виявляються в кінцевих вузлах. Так, для багатьох формальних мов, в тому числі і мов програмування, кінцевими символами є, наприклад: $\pm, =, go\ to, end, if$ і т.д.

2. Підмножина символів, що відносяться тільки до метамови. Символи цієї підмножини будемо називати *нетермінальними* і позначимо цю підмножину V_N . Символи називаються нетермінальними тому, що в дереві аналізу вони не можуть відповідати його кінцевим вузлам. До цієї підмножини відносяться символи <ідентифікатор>, <блок>, <терм>, <програма> і т.д.

У природних мовах із слів формуються речення. У формалізованій теорії мови введемо поняття рядка, аналогічне поняттю речення у природних мовах.

Рядком назвемо послідовність термінальних і нетермінальних символів даної мови, утворену згідно з правилами граматики аналізованої мови. Рядок будемо позначати малими латинськими буквами. Наприклад: a, b, x, y, z . Для порожнього рядка введемо спеціальний символ s .

Множину всіх рядків, утворених із символів множини V , позначимо через V^* , через z позначимо будь-який рядок, що належить V^* , через x – деякий заданий рядок (можливо, порожній, що теж належить V^*), через U – будь-який символ, що належить V . Множину всіх рядків можна визначити наступним чином:

$$V = \{z / z = \varepsilon, z = xU\}, z, x \in V^*, U \in V,$$

тобто множина всіх рядків складається з рядків, які або можуть бути порожніми, або можуть складатися з рядків, генерованих приєднанням довільного символу до деяких вихідних рядків (у тому числі і порожніх).

Наприклад, якщо $V = \{C, D\}$, то $V^* = \{\varepsilon, C, D, CDD, CCD \dots\}$. Зауважимо, що V^* – нескінченна множина.

Множину всіх синтаксичних правил конкретної формальної мови, записаних деякою метамовою (наприклад, мовою Бекуса), позначимо літерою P .

Будемо говорити, що рядок x безпосередньо породжує рядок y відносно P ($x \rightarrow y$) тоді і тільки тоді, коли існують рядки u, w (можливо, порожні) такі, що $x = uVw$ і $V \rightarrow z \in P$.

Наприклад, нехай правила P_1 задані наступним набором правил:

$$p_1: A \rightarrow BCD; p_2: B \rightarrow BC; p_3: D \rightarrow CD; p_4: B \rightarrow C; p_5: D \rightarrow C.$$

Тоді щодо цього ряду правил можливо безпосереднє породження

$$BCD \rightarrow BCCD.$$

У нашому прикладі можна припустити $x = BCD, y = BCCD, V = B, z = BC, w = CD, B \rightarrow BC \in P$, але можна також інтерпретувати це безпосереднє породження таким чином: $x = BCD, y = BCCD, V = D, z = CD, u = BC, z = \varepsilon, D \rightarrow C \in P$.

Таким чином, символ \rightarrow означає, що один символ ліворуч від \rightarrow відповідно до правил граматики замінюється ланцюжком, що знаходиться праворуч від \rightarrow .

Будемо говорити, що рядок x породжує рядок y відносно P ($x \rightarrow y$) тоді і тільки тоді, коли існує послідовність рядків $x_0, x_1, x_2, \dots, x_n$, така, що $x = x_0, y = x_n, x_{i-1} \rightarrow x_i$ ($i = 1, 2, \dots, n; V \rightarrow z \in P$). Використовуючи синтаксис P_1 , можна записати породження

$$BCD \rightarrow BCCCD.$$

$$x = x_0 = BCD, y = BCCCD;$$

$$x_0 \rightarrow x_1 = BCCD, V = B, z = BC, u = \varepsilon, w = CD;$$

$$x1 \xrightarrow{\bullet} x2 = BCCCD, V=B, z=BC, u=\varepsilon, w=CCD;$$

$$x2 \xrightarrow{\bullet} x3 = BCCCD, V=B, Z=BC, u=\varepsilon, w=CCCD.$$

Таким чином, використовуючи породження рядків щодо синтаксису даної мови, можна генерувати рядки, що формально належать до цієї мови.

Введемо поняття *початкового нетермінального символу*, що породжує будь-який рядок мови. Це виділений нетермінальний символ, що позначає сукупність (клас) всіх тих мовних об'єктів, для опису яких призначається дана граматики.

Формально початковий нетермінальний символ S визначається наступним чином:

$$S \xrightarrow{\bullet} z, S \in V_N, z \in V^*.$$

Формальною граматику назвемо впорядковану четвірку

$$G = (V_N, V_T, P, S),$$

де V_N і V_T – непересічні множини нетермінальних і термінальних (відповідно) символів даної граматики, P – множина синтаксичних правил граматики, S – початковий нетермінальний символ.

Формальна граматики, яка для будь-якого пред'явленого ланцюжка вміє вирішувати, є цей ланцюжок правильним чи ні, називається *граматику*, що *розпізнає*.

Формальна граматики, яка вміє будувати будь-який правильний ланцюжок, даючи вказівки про його будову, не генеруючи при цьому ні одного неправильного ланцюжка, називається *твірною*.

Приклад. Формальну граматику $G = (V_N, V_T, P, S)$, що описує спрощений арифметичний вираз, можна задати такими правилами:

$$p_1: \langle \text{арифметичний вираз } (+, *) \rangle \rightarrow E$$

$$p_2: E \rightarrow T$$

$$p_3: T \rightarrow T * P$$

$$p_4: E \rightarrow T + E$$

$$p_5: T \rightarrow P$$

$$p_6: E \rightarrow c$$

$$p_7: T \rightarrow a$$

$$p_8: P \rightarrow b.$$

Тоді $V_N = \{ \langle \text{арифметичний вираз } (+, *) \rangle, E, T, P \}$;

$V_T = \{ a, b, c, +, \perp \}$;

$S = \langle \text{програма} \rangle$

Породження арифметичного виразу в цій граматиці можна задати наступним чином:

$$\langle \text{арифметичний вираз } (+, *) \rangle \rightarrow E \rightarrow T \rightarrow T * P \xrightarrow{\bullet} a * P \xrightarrow{\bullet} a * b, \text{ тобто}$$

$$\langle \text{арифметичний вираз } (+, *) \rangle \xrightarrow{\bullet} a * b.$$

Підкреслимо, що формальна граматики не є алгоритмом. Множина синтаксичних правил P – це не послідовність приписів, а сукупність дозволів виконувати задані підстановки.

Це означає, що правило $A \rightarrow l$ в граматиці, що розпізнає розуміється як "*l* можна замінити на *A*", а в твірній як "*A* можна замінити на *l*", причому ця заміна може бути проведена тільки в певних умовах.

Тепер можна ввести формальне визначення мови, що породжується граматиною. Мова L_G , породжувана граматиною $G = (V_N, V_T, P, S)$, – це сукупність всіх термінальних ланцюжків, що виводяться з початкового символу S у граматиці G , тобто мова визначається наступною множиною:

$$L_G = \{x/x \in V^*T, S \xrightarrow{\bullet} x\}. \quad (4.1)$$

Наприклад, якщо

$$\begin{aligned} V_N &= \{S, A\}, V_T = \{B, C\}, \\ P &= \{S \rightarrow A; A \rightarrow B, A \rightarrow CA\}, \\ L_G &= \{B, CB, CCB, CCCB, \dots\}. \end{aligned}$$

Генерування рядків мови може бути представлено деревом (рис. 4.1).

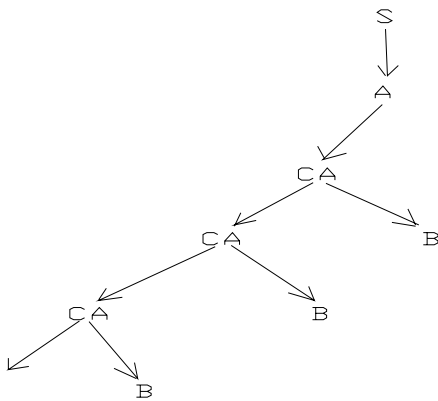


Рис. 4.1 Дерево рядків мови

Зауважимо, що в графі, показаному на рис. 4.1, термінальні символи C і B не породжують нових символів, їх породжують тільки нетермінальні символи A і S .

Таким чином, використовуючи функції породження рядків щодо синтаксису цієї мови, можна генерувати рядки, які формально належать цій мові, тобто відповідають синтаксичним правилам мови.

Ці рядки можуть виявитися семантично абсолютно безглуздими. Відомим прикладом речення, правильного синтаксично, але абсолютно невірною семантично, є фраза "глокляя суздра болданула макленка".

Різні граматики можуть породжувати одну і ту ж мову. Граматики G_1 і G_2 називаються еквівалентними, якщо породжувані ними мови L_{G_1} і L_{G_2} рівні. Еквівалентні перетворення граматик можуть бути викликані вимогою зручного вибору підпрограм, що реалізують окремі конструкції мови, прискорення процесу трансляції та генерації тексту.

4.3 Типи мов за класифікацією Хомського

Формальне визначення мови дозволило ввести їх класифікацію у відповідності з тими обмеженнями, які накладені на множині P граматики мови. Найбільшого поширення набула класифікація Хомського, який розділив мови відповідно до граматик, що їх описують, на чотири типи.

До типу 0 він відніс мови, на синтаксичні правила яких не накладено ніяких обмежень.

До типу 1 належать мови, синтаксичні правила яких можуть мати тільки наступний вигляд:

$$xAy \rightarrow xay \quad (x, y \text{ можуть бути порожніми}).$$

Цей тип мови називається *контекстно-залежною*, оскільки синтаксичне правило справедливе тільки в контексті рядків, що оточують A , тобто x і y .

До *типу 2* відносяться мови, синтаксичні правила яких можуть бути представлені тільки у вигляді

$$A \rightarrow a.$$

Мова типу 2 називається *контекстно-вільною* або *безконтекстною*, оскільки значення A не залежить від оточуючого його контексту.

У мовах *типу 3* синтаксичні правила строго обмежені однією з форм $A \rightarrow B$ або $A \rightarrow BC$, де $A, C \in V_N, B \in V_T$. Мови цього типу зазвичай називають *мовами кінцевих автоматів*, тому що їх граматики можуть бути представлені і проаналізовані апаратом кінцевих автоматів (рис. 4.2).

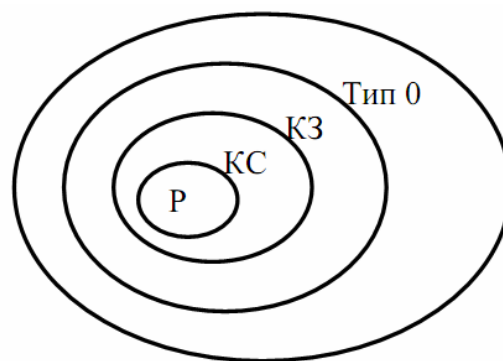


Рис. 4.2. Співвідношення типів формальних мов і граматики: P – регулярна (тип 3), KC – контекстно-вільна (тип 2), KЗ – контекстно-залежна (тип 1)

Приклад регулярної граматики:

$$G = \{V_N, V_T, P, S\}.$$

$$V_N = \{S\};$$

$$V_T = \{a, b\}$$

$$P: S \rightarrow a; S \rightarrow b; S \rightarrow aS; S \rightarrow bS.$$

За допомогою цієї граматики породжуються рядки символів a та b .

Послідовність породження рядків можна пояснити наступною схемою:

Правило виведення	Зміст рядка S
$S \rightarrow aS$	aS
$S \rightarrow aS$	aaS
$S \rightarrow b$	aab

Результатом виведення є рядок aab.

Приклад безконтекстної граматики:

$$G = \{V_N, V_T, P, S\}.$$

$$V_N = \{S\};$$

$$V_T = \{a, b\}$$

$$P: S \rightarrow aSb; S \rightarrow ab.$$

За допомогою граматики породжуються рядки $a^n b^n$.

Приклад виведення рядка $a^3 b^3$.

Правило виведення	Зміст рядка S
$S \rightarrow aSb$	aSb
$S \rightarrow aSb$	aaSbb
$S \rightarrow ab$	aaabbb

Приклад безконтекстної граматики більш складного виду:

$$G = \{V_N, V_T, P, S\}.$$

$$V_N = \{S\};$$

$$V_T = \{IF, THEN, ELSE, U, B\}$$

$$P: S \rightarrow B;$$

$$S \rightarrow IF U THEN S;$$

$$S \rightarrow IF U THEN S ELSE S.$$

Ця граMATика дозволяє формувати різні умовні оператори мови типу Паскаль. Наприклад, оператор $IF \cup THEN \quad IF \cup THEN B \quad ELSE B$ може бути сформований наступним чином.

Правило виведення	Зміст рядка S
$S \rightarrow IF \cup THEN S$	IF \cup U THEN S
$S \rightarrow IF \cup THEN S \quad ELSE S$	IF \cup THEN IF \cup THEN S ELSE S
$S \rightarrow B$	IF \cup THEN IF \cup THEN B ELSE S
$S \rightarrow B$	IF \cup THEN IF \cup THEN B ELSE B.

Можливий другий варіант:

Правило виведення	Зміст рядка S
$S \rightarrow IF \cup U THEN S \quad ELSE S$	IF \cup THEN S ELSE S
$S \rightarrow IF \cup U THEN S$	IF \cup THEN IF \cup THEN S ELSE S
$S \rightarrow B$	IF \cup THEN IF \cup THEN B ELSE S
$S \rightarrow B$	IF \cup THEN IF \cup THEN B ELSE B.

Зауважимо, що одна і та ж мова в багатьох випадках може бути описана граматиками різних типів. У цьому сенсі граматики різних типів можуть виявитися еквівалентними.

Найбільший інтерес представляють граматики типу 1 і 2.

Для граматик, описуваних спрямованою мережею без циклів (граф рис. 4.1 і табл. 4.1), алгоритм автоматичної побудови контекстно-вільної граматики системою підтримки прийняття рішень досить простий.

1. В якості початкового символу S вибирається символ, що позначає вершину, в яку не входить жодна дуга (початкову вершину). Якщо таких вершин кілька – створюється фіктивна початкова вершина.

2. В якості термінальних символів V_T вибираються символи, які означають вершини заключних дій. З них не виходить жодна дуга.

3. В якості нетермінальних символів V_N використовуємо символи, що позначають всі інші вершини графа.

4. Множина граматичних правил P визначається дугами, що з'єднують пари вершин. Вихідна вершина визначає ліву частину граматичного правила, а вершина, до якої входить дуга-праву. На цьому побудову граматики закінчено.

Повернемося до прикладу «Ліквідація наслідків аварійного розливу нафти».

Початковим символом граматики є вершина S , кінцевим – вершина J (у загальному випадку термінальних вершин може бути декілька), решта вершини описуються нетермінальними символами, тобто $V_N = \{A, B, C, D, E, F, G, H, I\}$, $V_T = \{J\}$. Множина граматичних правил P може бути задана граматиною типу 1, тобто контекстно-залежною граматиною з порожнім рядком:

$$\begin{aligned} S \rightarrow A; B \rightarrow BD; B \rightarrow BH; D \rightarrow DG, E \rightarrow EI; G \rightarrow GJ; \\ A \rightarrow AB; B \rightarrow BE; B \rightarrow BI; D \rightarrow DH; F \rightarrow FH; H \rightarrow HJ; \\ A \rightarrow AC; B \rightarrow BF; D \rightarrow DE; D \rightarrow DI; F \rightarrow FJ; I \rightarrow IG; \\ C \rightarrow CB; B \rightarrow BG; D \rightarrow DF; E \rightarrow EG; G \rightarrow GH. \end{aligned} \quad (4.2)$$

Безпосереднє породження рядків системою підтримки прийняття рішень реалізується відповідно до описаного вище алгоритму. Наприклад:

$$S \rightarrow A \rightarrow AB \rightarrow \dots \rightarrow ABD \rightarrow \dots \rightarrow ABDE \rightarrow \dots \rightarrow ABDEG \rightarrow \dots \rightarrow ABDEGJ.$$

Таким чином, символ S породжує рядок

$$S \rightarrow \dots \rightarrow ABCDEGJ$$

Рядок $ABCDEGJ$ – це один зі сценаріїв, тобто можлива послідовність операцій з ліквідації наслідків розливу нафти. Зауважимо, що рядок складається як з термінальних, так і з нетермінальних символів, але закінчується термінальним символом. Нас цікавить не тільки кінцева операція (J – очищення берега), але і всі проміжні операції, що їй передують. Цей рядок показує один з можливих шляхів у графі рис. 4.3.

Потрібно, щоб мова L_G , що породжується граматиною G , була сукупністю всіх правильних ланцюжків, що складаються з термінальних і нетермінальних символів, які виводяться з початкового символу S , тобто

$$L_G = \{x/x \in V^*, S \rightarrow \dots \rightarrow x\}, \quad V = V_N \cup V_T \quad (4.3)$$

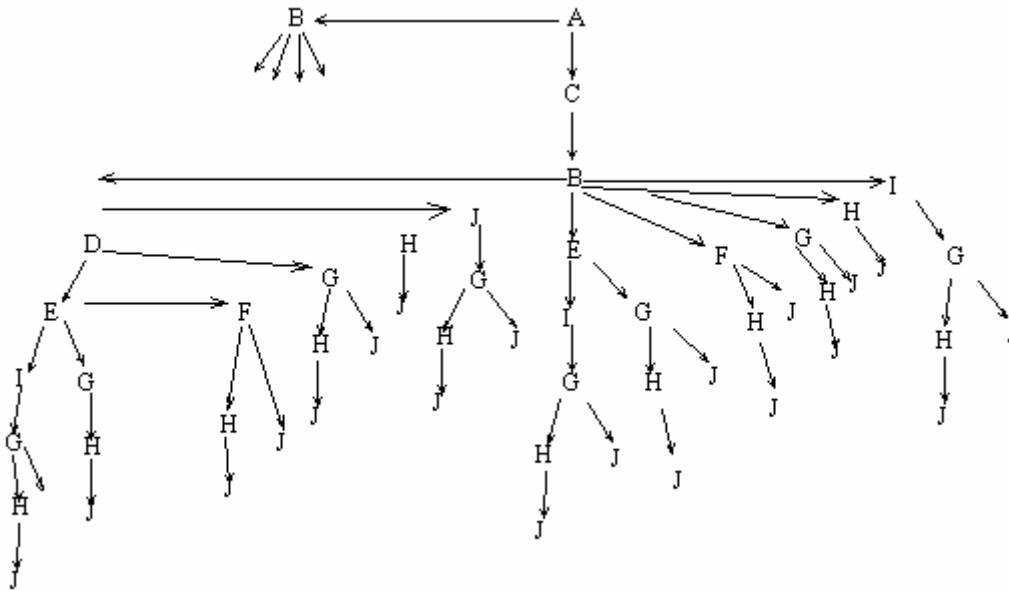


Рис. 4.3. Граф формальної граматики для прикладу «Ліквідація наслідків аварійного розливу нафти»

Відміна символів, що належать множині V_T , від символів, що належать множині V_N полягає в тому, що символи V_T не можуть стояти в лівій частині граматичного правила (породжувати інші символи).

Одна і та ж мова може бути породжена граматиками різних типів. Граматика типу 2 (безконтекстна або контекстно-вільна граматика) простіша, ніж контекстно-залежна граматика типу 1 і працювати з нею в багатьох випадках зручніше. Якщо перейти від граматики типу 1 до граматики типу 2, то для породження рядків мови L_G необхідно по новому сформулювати правило генерації рядків:

Символ $- \bullet ->$ означає, що до символу зліва від знака $- \bullet ->$ відповідно до правила граматики приписується символ, що стоїть в правій частині граматичного правила.

Тепер за наведеним вище алгоритмом можна побудувати контекстно-вільну граматику, яка породжує ту ж саму мову, що і розглянута вище контекстно-залежна граматика. Ця контекстно-вільна граматика буде мати наступну множину правил:

$$\begin{aligned}
 S &\rightarrow A; B \rightarrow D; B \rightarrow H; D \rightarrow G; E \rightarrow I; G \rightarrow J; \\
 A &\rightarrow B; B \rightarrow E; B \rightarrow I; D \rightarrow H; F \rightarrow H; H \rightarrow J; \\
 A &\rightarrow C; B \rightarrow F; D \rightarrow E; D \rightarrow I; F \rightarrow J; I \rightarrow G; \\
 C &\rightarrow B; B \rightarrow G; D \rightarrow F; E \rightarrow G; G \rightarrow H.
 \end{aligned}
 \tag{4.4}$$

Всі можливі рядки, породжені граматиною (4.4) у відповідності з визначенням мови (4.3), показані на рис. 4.3. Кожен символ на рис. 4.3 означає операцію, а гілка графа-послідовність операцій, тобто сценарій.

Надалі будемо працювати з контекстно-вільними граматиками.

Часи виконання операцій можуть бути взяті з бази даних або вказані ОПР. При оцінці часових характеристик t_x виконання операцій X необхідно врахувати вплив операції X_i на час виконання операції X_j . Часи виконання

операцій можуть бути показані в таблицях, аналогічних табл. 4.1. У стовпці t_x табл. 4.1 вказані часи автономного виконання операцій, коли виконання цієї операції не поєднується з іншою, яка доповнює її. Наприклад, розпорошення хімічних речовин не доповнюється використанням біологічних речовин.

Таблиця 4.1

Час виконання операцій

Операції X	t_x	F	G	H	I
A	2	-	-	-	-
B	5	-	-	-	-
C	8	-	-	-	-
D	20	-	-	-	-
E	10	-	-	-	-
F	4	-	1/4	1/2	1/2
G	12	3/4	-	1/2	1/2
H	8	1/2	3/4	-	-
I	5	-	-	-	-
J	3	-	-	-	-

Дріб в клітці табл. 4.1 показує, як скорочується час виконання операції X_i , зазначеної у рядку, якщо вона виконується спільно з операцією X_j , зазначеної у стовпці. Риска в таблиці показує, що час виконання операції X_j не залежить від виконання операції X_i . Часи виконання операції залежать від обсягу робіт, наприклад, розпорошення хімічних речовин, використання біологічних і горючих речовин і т.п., які вказані для одиниці обсягу роботи: 1 км² забрудненої території, 1 тонна пролитої речовини і т.д. на одиницю обладнання. Такі характеристики як час активізації командного пункту, збір і приведення в готовність обладнання і т.п. вказані в абсолютних цифрах.

Треба відзначити чотири особливості, що зазвичай не розглядаються в традиційних твірних граматиках. Необхідно:

- 1) породити всі правильні рядки мови, описувані граматикою, тобто всі можливі сценарії,
- 2) показати паралельно виконувані операції,
- 3) показати операції, що виконуються циклічно,
- 4) вказувати час виконання сценарію, тобто у рядка з'являється ще один параметр – час.

Для виконання перших двох особливостей можна модифікувати відомий алгоритм *backtracking*. Ідея алгоритму проста: якщо можна, то йдемо вперед, породжуючи ланцюжок символів, якщо ланцюжок породжений – повертаємося назад до повного перебору всіх граматичних правил.

1. $i = l$.

2. Якщо i -м елементом є символ A й існує правило $A \rightarrow B (tB)$, де $B \in V_N$, правило позначається і виробляється конкатенація символу B з породжуванним ланцюжком. Символ B стає $i+l$ символом. За ним в круглих дужках вказується час виконання позначеної ним операції.

Пункт 2 повторюється до виконання умови $B \in V_T$.

Якщо $B \in V_T$ - перехід до пункту 3.

3. Якщо i -м елементом є символ A й існує правило $A \rightarrow B(tB)$, де $B \in V_T$, то правило позначається і здійснюється конкатенація символу B з породженим ланцюжком. За ним в круглих дужках вказується час виконання позначеної ним операції. Породження послідовного ланцюжка закінчується. Перехід до пункту 4.

4. Якщо в точці A є ще непомічені правила, то відбувається перехід до пункту 2. Якщо ні, то проводиться перевірка, чи породжені символом A рядки, які повинні бути об'єднані в паралельний рядок.

Якщо так, то проводиться їх об'єднання. Якщо ні, то перехід до пункту 5.

5. Здійснюється перевірка умови $A = S$. Якщо $A \langle \rangle S$, перехід в точці $i-l$ до пункту 4. Якщо $A = S$, то кінець алгоритму.

Часові характеристики беруться з табл. 4.1 в базі даних або вказуються ОНР. Паралельні рядки, породжені паралельно-часовими граматиками з розглянутого алгоритму, будемо брати у квадратні дужки і з'єднувати стрілками, рядок всередині дужки – послідовний, в круглих дужках вказані часові характеристики. Породження рядка може бути описано наступним чином:

$$\begin{aligned} (tA) \text{---}\bullet\text{---}\rangle A'(tA)B(tB) \text{---}\bullet\text{---}\rangle A'(tA)B(tB)F(tF) \text{---}\bullet\text{---}\rangle A'(tA)B(tB)C(tc)F(tF)J(tj) \text{---}\bullet\text{---}\rangle \\ \text{---}\bullet\text{---}\rangle A'(tA)D(tD) \text{---}\bullet\text{---}\rangle A'(tA)D(tD)E(tE) \text{---}\bullet\text{---}\rangle A'(tA)D(tD)E(tE)F(tF) \text{---}\bullet\text{---}\rangle \\ \text{---}\bullet\text{---}\rangle A'(tA)D(tD)E(tE)F(tF)H(tH) \text{---}\bullet\text{---}\rangle A'(tA)D(tD)E(tE)F(tF)H(tH)J(tJ) \end{aligned}$$

Паралельні рядки можуть зливатися в один рядок. У цьому випадку символ, в якому відбувається точка злиття, входить в обидва рядки, при цьому часові характеристики точки злиття (у нашому випадку tF) враховуються один раз. У прикладі показано два "паралельні" рядки:

$$A'(tA)B(tB)F(tF)J(tJ) \text{ і } A'(tA)D(tD)E(tE)F(tF)H(tH)J(tJ).$$

Лівий рядок визначає таку послідовність дій:

- А. Активізація командного центру.
- В. Збір та приведення в готовність місцевого обладнання.
- Ф. Розпилення хімічних речовин.
- Ж. Очищення узбережжя.

Правий рядок формує іншу послідовність:

- А. Активізація командного центру.
- Д. Доставка додаткового обладнання повітрям.
- Є. Створення механічних загороджень.
- Ф. Розпилення хімічних речовин.
- Н. Використання біологічних речовин.
- Ж. Очищення узбережжя.

У дужках після кожного символу вказано час виконання операції, позначеної символом. Чи належить точка $F(tF)$ обом рядкам або тільки одному – питання угоди. Значення мають часові характеристики всього піддерева. Графічно паралельно-часові рядки можуть бути показані так, як це зроблено на рис. 4.3.

$$A'(2.) \xrightarrow{\bullet} [A'(2.)B(5.)\{I(5.)G(12.)\}(4)J(30.)]; [A'(2.)C(8.)]$$

Реалізацію четвертої особливості легко здійснити, підрахувавши сумарний час виконання операцій, записаних в кожному послідовному рядку з урахуванням циклів. Для дерева, що складається з декількох послідовних рядків, тобто для паралельного рядка, час її виконання визначається рядком з максимальним часом виконання. Представити графічно все дерево можливих сценаріїв складно, тим більше важко експерту або ОНР в складніших випадках вибрати кращий з представленою довгого списку та / або дерева можливих сценаріїв. Тому бажано, щоб система підтримки прийняття рішень сама вибрала кілька кращих варіантів сценаріїв і представила їх ОНР. При цьому необхідно, щоб ці варіанти були кращими саме з точки зору ОНР.

Приклад. Боротьба з сепаратизмом. Нагадаємо, що цифри позначають:

- 1 – укладання договору між центральним урядом і адміністрацією території;
- 2 – зміна адміністрації території, яка прагне до відокремлення від держави;
- 3 – перекриття енергоканалів;
- 4 – блокування території;
- 5 – придушення сепаратистів військовою силою;
- 6 – бездіяльність (стан на момент прийняття рішення).

Цифра 2 на перетині 2 рядка і 5-го стовпця вказує, що послідовність операцій 2,5 буде виконана двічі. Це повинен вказати ОНР або експерт.

Виходячи з матриці табл. 4.2 система підтримки прийняття рішень будує твірну граматику, а потім генерує допустимі рядки, тобто формує можливі сценарії. Твірна граMATика P_c :

$$\begin{aligned} &6' \rightarrow 1(t_1); 5 \rightarrow 1(t_1); 4 \rightarrow 3(t_3); 2 \rightarrow 1(t_1); \\ &6' \rightarrow 2(t_2); 5 \rightarrow 2(t_2); 4 \rightarrow 5(t_5); 2(t_2) \rightarrow 5(t_5); \\ &6' \rightarrow 3(t_3); 5 \rightarrow 3(t_3); 3 \rightarrow 1(t_1); \\ &6' \rightarrow 4(t_4); 5 \rightarrow 4(t_4); 3 \rightarrow 2(t_2); \\ &6' \rightarrow 5(t_5); 4 \rightarrow 1(t_1); 3 \rightarrow 4(t_4); 4 \rightarrow 2(t_2); 3 \rightarrow 5(t_5). \end{aligned}$$

Нижче наводяться приклади можливих сценаріїв. Початковим символом у граматиці P_c є символ 6 – не вживати ніяких дій (стан на момент прийняття рішення).

Найкращий для центрального уряду сценарій – укладення договору між центральним урядом і адміністрацією території:

$$6'(t_6) \xrightarrow{\bullet} 6'(t_6)1(t_1)$$

Більш складніший для центру сценарій – зміна адміністрації території і укладення договору:

$$6'(t_6) \xrightarrow{\bullet} 6'(t_6)2(t_2) \xrightarrow{\bullet} 6'(t_6)2(t_2)1(t_1)$$

Далі можна перебирати всі двоходові варіанти. Розглянемо більш складний випадок:

$$\begin{aligned} &6'(t_6) \xrightarrow{\bullet} 6'(t_6)5(t_5) \xrightarrow{\bullet} 6'(t_6)5(t_5)1(t_1) \xrightarrow{\bullet} 6'(t_6)4(t_4) \xrightarrow{\bullet} 6'(t_6)4(t_4)1(t_1) \\ &\quad \xrightarrow{\bullet} 6'(t_6)3(t_3) \xrightarrow{\bullet} 6'(t_6)3(t_3)1(t_1) \end{aligned}$$

Таблиця 4.2

Таблиця послідовності операцій підтримки прийняття рішень

	1	2	3	4	5	6
1						
2	/				2	
3	/	/		/	/	
4	/	/	/		/	
5	/	/	/	/		
6	/	/	*	*	*	

Це сценарій, за яким проводиться придушення сепаратистів військовою силою, блокування території та перекриття енергоканалів, що повинно привести до укладання договору між центральним урядом і адміністрацією.

Запишемо ще кілька сценаріїв без їх виведення.

$$6'(t_6) \text{---}\bullet\text{---} \rightarrow [6'(t_6)5(t_5)4(t_4)3(t_3)2(t_2)l(t_1)]$$

$$6'(t_6) \text{---}\bullet\text{---} \rightarrow [6'(t_6)\{5(t_5)2(t_2)\}l(t_1)]$$

За сценарієм першого рядка послідовно виконуються всі операції: 5, 4, 3, 2, 1. За сценарієм другого рядка повторюються спроби за допомогою військової сили придушити сепаратистів і провести зміну адміністрації території (дві спроби). Часи t_i не представлені. При плануванні всякої операції час виконання кожної її складової частини має бути обмежений. Якщо за цей час її виконати не вдається, то план проведення операції повинен бути переглянутий.

Додаткова складність може виникнути, коли один і той самий символ змістовно може бути термінальним і нетермінальним (символ 2 – зміна адміністрації території, яка прагне до відокремлення від держави). Може така зміна вирішити всі проблеми. З іншого боку, не можна повністю відмовлятися від термінального символу l – укладання договору між центральним урядом і адміністрацією території. У такій ситуації доцільно ввести фіктивний термінальний символ (позначимо його 2) і граматичне правило $2(t_2) \rightarrow 2(t_2')$. Табл. 4.2 прийме вигляд табл. 4.3.

Таблиця 4.3

Таблиця послідовності операцій підтримки прийняття рішень

	1	2	2	3	4	5	6
1							
2	/	/				2	
3	/		/		/	/	
4	/		/	/		/	
5	/		/	/		/	
6	/		/	*	*	*	

Тепер можлива побудова, наприклад, такого рядка:

$$6'(t_6) \text{---}\bullet\text{---} \rightarrow [6'(t_6)4(t_4)3(t_3)2(t_2)2(t_2')].$$

До сценарію цього рядка входить блокування території, перекриття енергоканалів, а зміна адміністрації території є кінцевою метою, досягнутої в результаті перших двох дій. Таким чином, за допомогою твірної граматики, маючи множину можливих операцій та таблицю допустимих послідовностей їх виконання, можна генерувати всі можливі сценарії.

5 РОЗПОДІЛ РЕСУРСІВ ШЛЯХОМ УЗГОДЖЕННЯ ГРУПОВИХ РІШЕНЬ З ВИКОРИСТАННЯМ λ -КОЕФІЦІЄНТІВ

Задача прийняття рішень часто зводиться до розподілу ресурсів. Ресурси можуть бути найрізноманітнішими: гроші між кредиторами при банкрутстві фірми, дороги при переміщенні великих мас транспортних засобів, артилерійські снаряди між цілями, які повинні бути придушені і т.д. Задачу розподілу ресурсів можна розглядати як пару (E, C) , де E – загальна кількість ресурсів, а $C = (c_1, c_2, \dots, c_n)$ – вектор його запитів. Правило розподілу ресурсів – це функція, що визначає кількість ресурсів, які виділяються кожному "споживачеві".

Найпростіше правило, але не завжди прийнятне – це розподіл, пропорційний запитам:

$$\begin{aligned} P(E, C) &= \lambda C, \\ \lambda C &= E. \end{aligned}$$

У цьому випадку проблем з визначенням величини λ не виникає, вона визначається за відношенням:

$$\lambda = \frac{E}{\sum_i^n c_i}.$$

Це правило можна застосовувати тільки в тому випадку, якщо всі "споживачі" ресурсів згодні отримати ресурси пропорційно запитам. Розглянемо більш складний випадок, коли виробляється узгоджене рішення про виділення i "споживачеві" його частки. Цю частку виділяють йому всі інші "споживачі". Вона дорівнює:

$$\lambda_i = E - \sum_{i \neq j} c_j,$$

тобто це залишок від того, що інші споживачі взяли собі. Введемо величини:

$$\lambda_i = \max \left\{ 0, E - \sum_{i \neq j} c_j \right\}, \text{ и } c'_i = \min \{ c_i, E \} \quad (5.1)$$

Тут λ визначається автоматично відповідно до запитів. Цей метод стимулює збільшення запитів, але не дозволяє погоджувати рішення. У цьому випадку задачу розподілу ресурсів можна записати у вигляді

$$P(X, C) = X, \quad X = (x_1, x_2, \dots, x_n),$$

де

$$x_i = \begin{cases} \lambda_i + (c'_i - \lambda_i) \left(\sum_{j \in N} (c'_j - \lambda_j) \right)^{-1} (E - \sum_{j \in N} \lambda_j), & \text{якщо } C > E > 0, \\ c_i, & \text{якщо } E = C, \\ 0, & \text{якщо } E = 0. \end{cases}$$

У тих випадках, коли для знаходження величин λ_i можна спиратися на деякий базовий критерій, наприклад, вартість, гроші або співвідношення типу

(5.1) і всі інші критерії оцінювати, спираючись на нього – безумовно доцільно використовувати такий підхід, проводячи узгодження одним зі способів, описаних вище. Якщо такий базовий критерій вибрати важко, виробляються попарні оцінки [9] значущості (пріоритетності) одного "споживача" стосовно до іншого, як показано в табл. 5.1.

Таблиця 5.1

Визначення попарних оцінок значущості (пріоритетності)

Ступінь важливості	Визначення
1	Однакова значимість
3	Деяка перевага значущості одного порівняно з іншим
5	Істотна або сильна значимість
7	Дуже сильна або очевидна значимість
9	Абсолютна значимість
2, 4, 6, 8	Проміжні величини між сусідніми значеннями шкал
Зворотні величини наведених вище чисел	Якщо значенню i у порівнянні зі значенням j приписується одне з наведених вище чисел, то значенню j у порівнянні зі значенням i приписується зворотна величина

Нехай C_1, C_2, \dots, C_n – сукупність об'єктів або можливих дій. Кількісні порівняння в парах об'єктів або дій (C_i, C_j) представимо матрицею розміру $n \times n$

$$A = (a_{ij}) \quad (i, j = 1, 2, \dots, n).$$

Елементи a_{ij} визначимо таким чином: якщо $a_{ij} = \beta$, то $a_{ji} = 1/\beta$, $\beta \neq 0$, $a_{ii} = 1$ для всіх i . Таким чином, матриця A має вигляд:

$$A = \begin{vmatrix} 1 & a_{12} & \dots & a_{1n} \\ 1/a_{12} & 1 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ 1/a_{1n} & 1/a_{2n} & \dots & 1 \end{vmatrix}$$

Розглянемо ілюстративний приклад. Нехай є чотири споживача ресурсу: C_1, C_2, C_3 і C_4 . За попарною домовленістю один з одним вони прийшли до наступних співвідношень по частці одержуваного ресурсу: $C_1 = 5C_2$, $C_1 = 6C_3$, $C_1 = 7C_4$, $C_2 = 4C_3$, $C_2 = 6C_4$, $C_3 = 4C_4$. Представляючи ці дані у вигляді матриці, показаної у табл. 5.2, легко бачити, що матриця є неузгодженою (що типово для оцінок при попарному порівнянні), якщо за основу не взято базовий параметр.

Тепер за матрицею табл. 5.2 треба знайти пріоритети (або ваги) споживачів, які стали б базою для подальшого узгодження. Існує декілька способів узгодження матриці, що відрізняються за складністю і точністю. Найбільш точним є знаходження головного власного вектора матриці, який після нормалізації стає вектором пріоритетів. Для даного ілюстративного прикладу скористаємося спрощеним методом.

Таблиця 5.2

Матриця співвідношень по частці одержуваного ресурсу

	C_1	C_2	C_3	C_4
C_1	1	5	6	7
C_2	1/5	1	4	6
C_3	1/6	1/4	1	4
C_4	1/7	1/6	1/4	1

Підсумовуємо елементи кожного рядка і нормалізуємо діленням кожної суми на суму елементів. Сума отриманих результатів буде дорівнює одиниці. Перший елемент результуючого вектора буде пріоритетом першого об'єкта, другий – другого і т.д. У даному прикладі – вектор сум рядків матриці показаний у табл. 5.3

Сума елементів вектора дорівнює 37, 18. Вектор пріоритетів – в останньому стовпці табл. 5.3.

Таблиця 5.3

Вектор сум рядків матриці співвідношень

	Сума	Пріоритети
C_1	19,00	0,51
C_2	11,20	0,30
C_3	5,42	0,15
C_4	1,56	0,04

У наведеному прикладі ці значення можуть бути величинами λ_i або базою для узгодження цих значень.

6 ПРИЗНАЧЕННЯ І СТРУКТУРА КОМП'ЮТЕРНИХ СИСТЕМ ПІДТРИМКИ ПЕРЕГОВОРІВ З ПРИЙНЯТТЯ УПРАВЛІНСЬКИХ РІШЕНЬ

Можна розглядати три класи переговорів з використанням СПП.

1. "Переговори" між системами прийняття рішень без участі людини. Такі "переговори" мають місце в технічних і військових системах.

2. Переговори між системою прийняття рішень, з одного боку, і людиною, з іншого. Такі системи мають місце при прийнятті стандартних рішень в процесі купівлі-продажу, надання кредиту і т.д.

3. Переговори між експертами та / або керівниками за допомогою систем підтримки переговорів.

Будемо розглядати 3-й клас переговорів. Підкреслимо, що СППР в цьому класі переговорів не замінює людини. Її вміння, знання та переваги все одно залишаються вирішальними, але посилюються комп'ютерними системами, дозволяючи здійснювати генерацію варіантів (і отримуючи, можливо, несподівані для людини рішення), їх оцінку і прогноз результатів прийняття рішень.

Якщо під мистецтвом ведення переговорів розуміти ймовірність прийняття групою правильних рішень, то воно може бути оцінено кількісно.

Нехай P – вектор ймовірностей прийняття індивідуальних правильних рішень P_i , $i = 1, N$. Тоді ймовірність прийняття правильних рішень (оцінка мистецтва прийняття рішень групою) може бути представлена у вигляді:

$$M_V(P) = \sum_S \prod_{i \in S} P_i \prod_{j \notin S} (1 - P_j),$$

де S – члени групи, що приймають правильне рішення.

Структура системи підтримки переговорів і схема її функціонування представлені на рис. 6.1 [21].

СППР здійснює підтримку на всіх етапах, починаючи від збору інформації і закінчуючи оформленням документів по прийнятому рішенню. СППР можна розглядати як функціонально розподілену систему, якщо вона реалізована тільки на одному комп'ютері в одного з «парламентерів» або як функціонально і просторово розподілену, якщо вона використовується кількома «переговорниками», віддаленими один від одного.

У більшості випадків перерви між переговорами використовуються для прийняття індивідуальних рішень, в ході яких переосмислюється ситуація, здійснюється пошук нових варіантів, оцінка або переоцінка учасників наради тощо. У цьому випадку процес узгодження рішень здійснюється шляхом послідовного чергування процедур, показаних в блоках B , C і D .

Якщо переговори, в яких для досягнення угоди висувуються менш переважні для кожного учасника альтернативи, здаються обіцяючими для знаходження компромісу, то процедури генерації та оцінки таких альтернатив повинні розглядати ці варіанти. Одним зі способів досягнення справедливості правил у процесі переговорів є надання права всім учасникам переговорів висувати альтернативні пропозиції і здійснювати свої оцінки кожної пропозиції (своїй або партнера по переговорах).

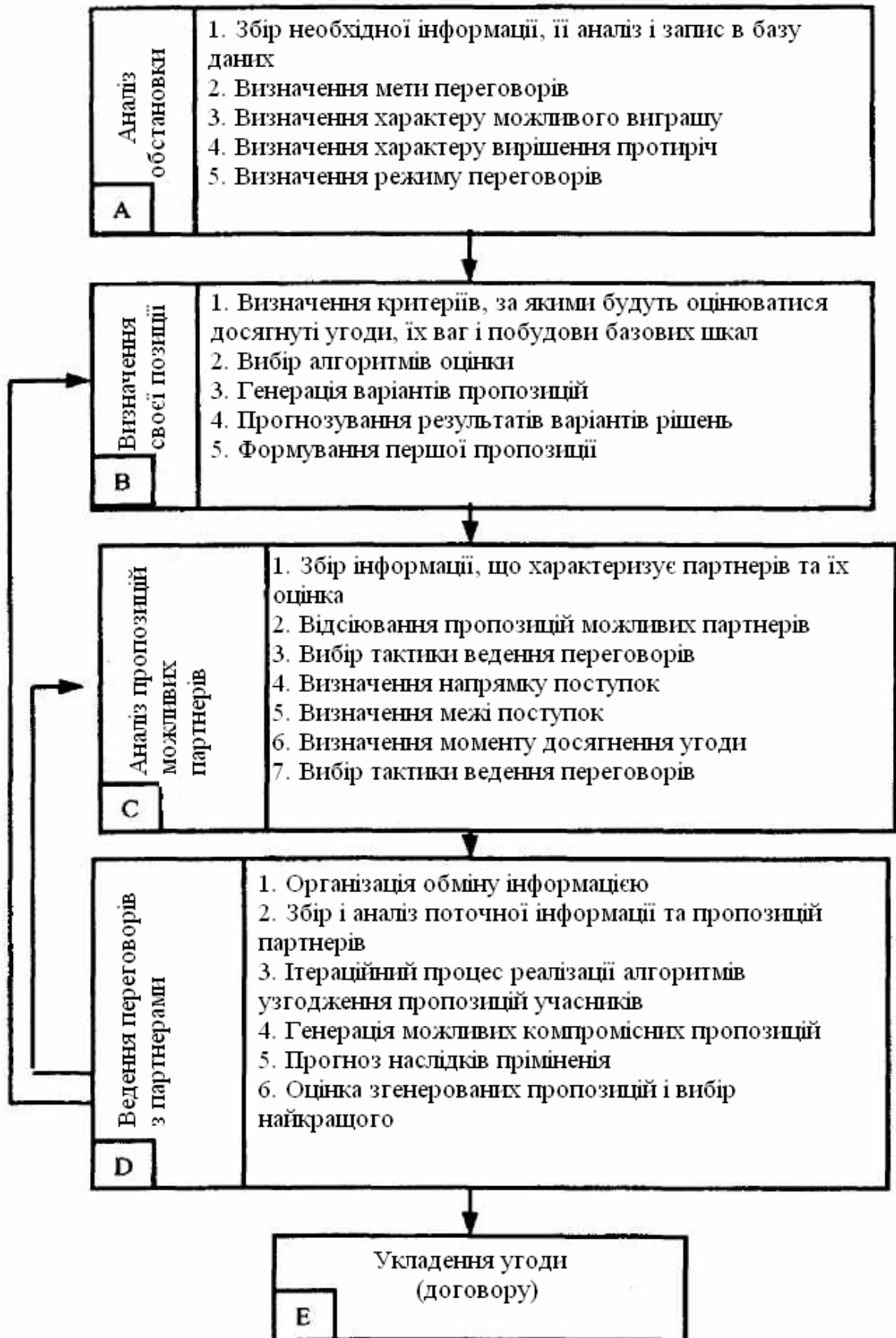


Рис. 6.1 Структура системи підтримки переговорів і схема її функціонування

Розглянемо шлях знаходження таких менш бажаних рішень, які, однак, можуть привести до узгодження у фазі переговорів. Нехай у переговорах беруть участь представники N організацій та запропоновано K варіантів рішень. Кожен учасник має чітко ранжувати переваги по всіх K альтернативам, які представлені матрицею $N \times K$, кожен елемент якої $z_{n,k}$ визначає k -ий рівень переваги варіанта рішення для n -го учасника переговорів. Будемо вважати, що чим більше значення k , тим нижче ранг. Учасники переговорів можуть не знати всю матрицю Z . Алгоритм проведення переговорів наступний:

1. Розглядається найкраща альтернатива рішення кожного учасника. Якщо переваги збігаються – вибір зроблено. Якщо ні, то $n=2$ – перехід до кроку n .

n . Розглядаються наступні послідовні альтернативи. Тепер кожен учасник виставляє n можливих найбільш бажаних рішень. Якщо є їх перетин – вибір зроблено, якщо ні, то $n = n + 1$. Якщо $n \leq N$ – крок n повторюється, якщо $n > N$ – рішення не знайдено.

Приклад. У матриці Z елементи кожного рядка розташовані за спаданням рівня уподобань. У першому рядку, що показує переваги першого учасника, найкращий варіант a , потім b і т.д. У другому рядку, що показує переваги другого учасника, найкращий варіант b , потім d і т.д.

$$Z = \begin{pmatrix} a & b & c & d \\ b & d & a & c \end{pmatrix}$$

шаг 1

$$Z' = \begin{pmatrix} a & b & c & d \\ b & d & a & c \end{pmatrix} \text{ перетинів немає,}$$

шаг 2

$$Z^2 = \begin{pmatrix} a & b & c & d \\ b & d & a & c \end{pmatrix} \quad \text{перетин є – } b. \text{ Варіант } b \text{ може}$$

розглядатися як компромісний. Компромісних варіантів, як показано в наступному прикладі, може виявитися декілька.

$$Z = \begin{pmatrix} a & b & c & d \\ b & a & c & d \end{pmatrix}$$

Крок знову 1 не дає перетинів, крок 2 дає перетин двох елементів.

$$Z^2 = \begin{pmatrix} a & b & c & d \\ b & a & c & d \end{pmatrix}$$

Зауважимо, що всі елементи, що належать отриманій за алгоритмом компромісній множині – Парето-оптимальні. Це спрощений алгоритм пошуку співпадаючих рішень учасників переговорів, а не їх модифікації, але ідею компромісу він ілюструє добре. СПП є найважливішою складовою системи підтримки групових рішень.

6.1 Комерційні пакети, використовувані в системах підтримки переговорів

У табл. 6.1. Підсумовані функції, що виконуються найбільш відомими пакетами програм.

Таблиця 6.1

Функції, що виконуються найбільш відомими пакетами програм					
Функція \ Найменування системи	Oracle Express OLAP	Data Vision	Пакет фірми Cognos	Система управління мед. ресурсами	Система iRenaissance
1. Аналіз ситуації	+	+	+	+	+
2. Генерація можливих рішень	-	-	-	-	-
3. Оцінка генерованих рішень	-	-	-	+	+
4. Обмін інформацією та узгодження рішень	-	-	-	-	+
5. Моделювання та	+	+	+	-	+
6. Вибір варіанту	-	-	-	+	-

З таблиці 6.1 видно, що всі системи так чи інакше аналізують обстановку, деякі здійснюють моделювання розвитку ситуації головним чином за методом "що ..., якщо". У системі *iRenaissance* забезпечується обмін інформацією між співробітниками різних відділів, а в системі *управління медичними ресурсами* видаються відповіді на конкретні питання, які можна з великою натяжкою вважати вибором варіанта.

Жодна система не здійснює генерацію рішень і їх узгодження, жодна система не враховує персональні інтереси, оцінки і переваги керівника.

Треба відзначити, що в проекті системи *Spatial Decision Support System for rural Land Use Planning (SDSS / LUP)* вже сформульована задача вироблення рішень з: вибору ділянок для відновлення зіпсованих ґрунтів; створення інфраструктури для збереження земель (створення дамб, кам'яних стін, тощо); методів рекультивациі ділянок зіпсованих ґрунтів. Таким чином, вже робляться спроби створення комерційних пакетів, які дійсно генерують рішення.

6.2 Оцінка характеру можливого виграшу

Надзвичайно важливим для учасників переговорів є характер можливого виграшу від прийнятого рішення. За характером можливого виграшу переговорні ситуації можуть бути розділені на наступні класи.

– *Розподільчий* (його також називають змагальний, грою з нульовою сумою і т.п.). У переговорах цього класу одна сторона "виграє", а інша – "програє". Це відбувається тоді, коли існує фіксований ресурс, розподіл якого і складає предмет переговорів.

– *Інтеграційний* (його називають дружнім, типу вигреш – вигреш і т.п.). У переговорах цього класу у зв'язку з тим, що величина ресурсу мінлива, не фіксована, обидві сторони можуть "виграти". Домінантою таких переговорів є максимізація сумарного вигрешу. Вона досягається обміном інформації і спільним вирішенням проблем. Це може статися, наприклад, при злитті двох фірм.

– *Інтеграційно-розподільний*, що включає елементи обох перерахованих вище класів. Прикладом переговорів цього класу є відношення постійних постачальників і покупців товару. Кожен з них хоче отримати максимум прибутку за рахунок іншого, але кожна сторона хоче також, щоб інша сторона була задоволена.

Практично оцінки можливого вигрешу роблять, виходячи з функцій корисності, заданих обмежень і того принципу оптимальності, якому слідує фахівець [24]. Покажемо це на ілюстративному прикладі.

Нехай є комплекс робіт з n незалежних операцій. Для кожної операції задана залежність витрат $S_i(\tau_i)$ від тривалості її виконання τ_i , а також обмеження на сумарні витрати:

$$\sum_i S_i(\tau_i) \leq S$$

Потрібно визначити тривалість всіх операцій так, щоб мінімізувати втрати C_i за одиницю часу.

Будемо вважати, що допускається невиконання ряду операцій i , якщо i -а операція не виконується, то втрати від її невиконання дорівнюють d_i . Потрібно знайти таку множину операцій Q , яка виконуватиметься, мінімізуючи можливі втрати, а також тривалість цих операцій при обмеженні:

$$\sum_{i \in Q} S_i(\tau_i) \leq S$$

В такому разі втрати

$$\Phi(\tau_i, Q) = \sum_{i \in Q} C_i \tau_i + \sum_{i \notin Q} d_i$$

Оскільки

$$\sum_{i \notin Q} d_i = \sum_{i=1}^n d_i - \sum_{i \in Q} d_i$$

$$\Phi(\tau_i, Q) = \sum_{i \in Q} (C_i \tau_i - d_i) + \sum_i d_i$$

У разі дискретної залежності приходимо до задачі вибору оптимальної множини реалізованих операцій Q такої, що

$$\sum_{i \in Q} (C_i \tau_i - d_i) \rightarrow \min \text{ при обмеженні } \sum_{i \in Q} S_i \leq S.$$

Це класична задача дискретної оптимізації, званої задачею про ранець. Її можна було б віднести до типу утилітарних, оскільки в ній максимізується сумарна корисність (мінімізуються сумарні втрати).

6.3 Характер вирішення протиріч

Характер вирішення протиріч можна класифікувати у термінах відношення до задоволення вимог своїх і опонента: егоїстичний; компромісний; байдужий; поступливий (наприклад, безумовна капітуляція).

Позначимо функцію корисності результатів переговорів для однієї зі сторін – $\pi(z)$, а для іншої – $\psi(z)$. Кожна сторона знає свою функцію, але, як правило, не знає функції іншої, z – вектори аргументів цих двох функцій. Область визначення аргументів може бути спільною для обох функцій або своя для кожної функції, але вони обов'язково повинні перетинатися. Оскільки нас цікавить тільки область перетину, то вона і позначена вектором z .

Точки екстремумів функцій позначимо $\pi(z^x_0)$ і $\psi(z^y_0)$. У загальному випадку точки z^x_0 і z^y_0 не збігаються. Кожна зі сторін хоче, щоб моментом закінчення переговорів стала точка екстремуму, але точка моменту закінчення в значній мірі залежить від тактики проведення переговорів. Тому функції, що описують хід переговорів, можна розглядати як композиції двох функцій. Для сторони X це композиція функції $\pi(z)$ і $\alpha(z)$, а для сторони Y – композиція функцій $\psi(z)$ і $\beta(z)$, де $\alpha(z)$ і $\beta(z)$ – функції, що характеризують тактику переговорів сторін X і Y відповідно. Ці композиції запишемо у вигляді функцій

$$f_x(\pi(z), \alpha(z)) \quad \text{і} \quad f_y(\psi(z), \beta(z))$$

Якщо позначити точку досягнення угоди через Z_c , то обидві сторони будуть прагнути до того, щоб:

$$\begin{aligned} |\pi(z^x_0) - f^x(\pi(z^x_c), \alpha(z^x_c))| &\rightarrow \min \\ |\psi(z^y_0) - f^y(\psi(z^y_c), \beta(z^y_c))| &\rightarrow \min \end{aligned}$$

При *егоїстичній тактиці* партнера X

$$f^x(\pi(z^x_c), \alpha(z^x_c)) = \pi(z^x_0)$$

незалежно від тактики партнера Y .

При *компромісній тактиці* партнерів X і Y

$$A |\pi(z^x_0) - f^x(\pi(z^x_c), \alpha(z^x_c))| = B |\psi(z^y_0) - f^y(\psi(z^y_c), \beta(z^y_c))|,$$

де A і B – вектори, на яких досягається компроміс, тому що в загальному випадку

$$|\pi(z^x_0) - f^x(\pi(z^x_c), \alpha(z^x_c))| \neq |\psi(z^y_0) - f^y(\psi(z^y_c), \beta(z^y_c))|$$

При *байдужій тактиці* партнерів X і Y ні характер функцій $\alpha(z)$, $\beta(z)$, ні значення функцій $\pi(z^x_c)$ і $\psi(z^y_c)$ ролі не відіграють.

При *поступливій тактиці* партнера X

$$f^x(\pi(z^x_c), \alpha(z^x_c)) = \psi(z^y_0)$$

незалежно від тактики партнера Y .

7 ВИБІР МЕТОДІВ ОЦІНКИ ВАРІАНТІВ РІШЕНЬ

7.1 Огляд методів оцінки ефективності рішень

У процесі прийняття нового рішення людина в змозі розглянути кілька варіантів і в багатьох випадках не помічає кращий або небезпечний (приклад – "пози́хи" в шахах). СППР здатна генерувати, якщо не всі, то дуже велику кількість можливих рішень. Але генерація великого числа рішень має сенс тільки в тому випадку, якщо сама СППР зможе їх оцінити і ранжувати з урахуванням переваг керівника (показувати велике число, наприклад 200 варіантів, керівнику безглуздо, він не в змозі проаналізувати всі варіанти).

Для представлення кращих варіантів рішень, згенерованих за допомогою СППР, їх необхідно оцінити, ранжувати і вибрати кращі. Оцінка варіантів рішень проводиться за допомогою математичних моделей і обов'язково з урахуванням переваг керівника.

Питання про застосування математичних моделей для аналізу проблем прийняття рішень в економіці, екології, управлінні виробництвом та інших областях, закони функціонування яких ще погано формалізовані і вивчені, не може розглядатися також як, наприклад, у фізиці, в якій математичні моделі є результатом багатовікових досить успішних досліджень. У цих областях математичні моделі досить грубі, іноді дають навіть якісні невірні прогнози. Це пов'язано, зокрема, як з величезною складністю цих проблем, так і з їх залежністю від виникаючих в цих завданнях невизначеностей і чисто суб'єктивних факторів. Тому суб'єктивні оцінки параметрів моделей, що визначаються керівниками, мають велике значення.

Вирішення цієї проблеми може бути знайдено, якщо використовувати математичні моделі і методи для генерування та оцінки можливих сценаріїв (рішень), які сприймаються як рекомендації для подальшого обмірковування і, можливо, неформального аналізу [19].

Як приклад корисності таких моделей можна навести одне з перших, якщо не перше, дослідження моделі світової динаміки, здійснене в кінці 60-х років Дж. Форрестером. Він пов'язав основні економічні та демографічні характеристики за допомогою простих співвідношень для того, щоб потім вивчити в динаміці взаємний вплив цих характеристик і отримати якийсь варіант розвитку світової економіки. Незважаючи на довільність багатьох припущень, дослідження змогло передбачити проблеми, що виникли в наступні роки: подорожчання деяких видів ресурсів, наростаюче забруднення навколишнього середовища, брак сільськогосподарської продукції і т.д.

При визначенні набору критеріїв, їх «ваг» і базових шкал керівник або експерт в зрозумілих для нього термінах може висловити свої суб'єктивні оцінки успішності (або невдачі) передбачуваного спільного рішення. При виборі алгоритму оцінки такої апріорної ясності немає. При використанні різних алгоритмів будуть виникати різні оцінки варіантів узгодженого рішення. Звичайно, в деяких випадках вони можуть збігатися, але це виняток, а не правило. Це виключення виникає звичайно в тих випадках, коли один з об'єктів

множини, що ранжується, є гіршим або перевершує інші за всіма або найбільш значимим параметрами.

Методи оцінки рішень надзвичайно різноманітні [5, 7, 14].

Варіант можна оцінювати за співвідношенням:

$$F(X^t, Y^t, z_t) = \max_{x_t \in X^t} \min_{y_t \in Y^t} \sum_t (x_t - y_t) z_t,$$

де z_t – "вага" критерію t ;

$X^t \in X$ – варіант значення критерію t після досягнення угоди;

$y_t \in Y_t$ – варіант поточного значення критерію t ;

X^t – множина можливих значень критерію t після укладення угоди;

Y^t – множина можливих оцінок поточного значення критерію t .

Для оцінки фінансової рентабельності проекту застосовуються кілька більш складних співвідношень. Часто використовується метод, що отримав назву "*чиста поточна вартість*" (від англійського *net present value*):

$$V = \sum_{t=1}^T \frac{B^t - Z^t}{(1 + E)^t},$$

де B^t – дохід за рік t , Z^t – витрати за рік t ; E – ставка (норма дисконту).

Якщо вважати, що витрати визначаються тільки капітальними вкладеннями і витратами на експлуатацію, то залежно від ситуації величина доходів може змінюватися. У цьому випадку на прогноз доходів сильний вплив чинить суб'єктивна оцінка керівником значущості параметрів, яка залежить також від стратегії поведінки керівника. Тоді функцію V можна записати у вигляді:

$$V_{ij} = \sum_{t=1}^T \frac{B_{ij}^t - Z_{ij}^t}{(1 + E)^t},$$

де i – одна з подій, які можуть знизити (підвищити) дохідність і/або збільшити (знизити) витрати, а j – одна із стратегій поведінки, яку керівник збирається реалізувати.

На основі використання функцій типу V_{ij} запропоновано досить велике число оцінок ефективності.

Одним з них є максимальне значення мінімального виграшу. Формально ця оцінка виходить з принципу "гірше не буде". Вона має вигляд:

$$W = \max_i \min_j V_{ij}$$

Ця оцінка відображає позицію крайнього песимізму. Оцінка мінімального ризику ідейно близька до максимальної, але орієнтується не на виграш, а на ризик:

$$S = \max_i \min_j a_{ij}$$

де a_{ij} – величина ризику для варіанту i, j .

Треба сказати, що завдання визначення ризику досить складна сама по собі, і ми її розглядати не будемо. При використанні цієї оцінки намагаються уникнути великого ризику при прийнятті рішень. Нарешті, певний проміжний варіант між вкрай песимістичними варіантами і вкрай оптимістичними:

$$H = \max_i \left[\beta \min_j V_{ij} + (1 - \beta) \max_j V_{ij} \right]$$

де $0 \leq \beta \leq 1$ – "коефіцієнт песимізму" (або "коефіцієнт оптимізму"). При $\beta = 1$ оцінка перетворюється на максимальну, а при $\beta = 0$ вона максимально оптимістична. Необхідно підкреслити, що визначення значення – це прерогатива керівника, і з цієї точки зору, оцінка надзвичайно суб'єктивна.

Функції V , S і H – це тільки кілька прикладів з дуже великого числа функцій, алгоритмів і методів оцінок ефективності капіталовкладень. Нижче наведено кілька типових прикладів таких моделей.

Для так званих виробничих задач характерною є канонічна задача лінійного програмування:

$$\begin{aligned} CX &\rightarrow \max, \\ AX &= B, \\ x_{ij} &\geq 0, \end{aligned}$$

де C – вектор коефіцієнтів, A – матриця параметрів, X – вектор змінних, B – вектор обмежень.

При вирішенні транспортних задач може бути використаний метод пошуку оптимального потоку (поток мінімальної вартості), який доставляє рішення задачі:

$$\sum_{(i,j) \in U} c_{ij} x_{ij} \rightarrow \min, \quad \sum_{j \in I_i^+} x_{ij} - \sum_{j \in I_i^-} x_{ij} = a_{ij} \quad i \in I, x_{ij} > 0, (i, j) \in U$$

де $(i, j) \in U$ – множина дуг, c_{ij} – вартість одиничного дугового потоку, (I_i^+) , (I_i^-) – множина вузлів, які з'єднані з вузлом i дугами з U , що починаються в i (або закінчуються в i), $\sum_{(i,j) \in U} c_{ij} x_{ij}$ – вартість потоку.

Для вирішення широко відомої задачі вибору з сукупності n об'єктів деякого їх кількості, що характеризуються мінімальною вагою (об'єм, величини шкідливих викидів і т.д.) при заданій цінності (продуктивності, токсичності, силі вибуху тощо) може бути застосована задача булевого програмування, що отримала назву задачі про ранець:

$$f(x) = \sum_{i=1}^n P_i x_i \rightarrow \min, \quad \sum_{i=1}^n c_i x_i \geq c, \quad x_i = \{1, 0\}, \quad i = 1, n,$$

де c_i – цінність i -го об'єкта, P_i – вага i -го об'єкта, c – задана цінність обраних об'єктів.

У процесі прийняття групового рішення часто виникає необхідність відмовитися від виконання якоїсь частини проекту або один варіант пропозицій замінити іншим, в якому не передбачається проведення будь-яких робіт. Одним з критеріїв в таких випадках може бути мінімізація упущеної вигоди. Розглянемо постановку такої задачі.

Нехай заданий комплекс з n робіт, незалежних у тому сенсі, що виконання їх не синхронізовано.

Обсяг операції можна визначити співвідношенням:

$$W = \sum_{k=1}^K f[u(t_k)]$$

де t_1 – час початку операції, t_K – час її кінця; $u(t_k)$ – вектор ресурсів в операції в момент k . Набір ресурсів опишемо у вигляді:

$$U_i = \beta_j V_i, \quad j=1, \dots, m,$$

де m – кількість ресурсів, V_i – інтенсивність набору, β_j - кількість ресурсу j -го виду на одиницю інтенсивності набору.

Обмеження на ресурси пов'язані з обмеженням фінансів:

$$\sum_{i=1}^n \sum_{j=1}^m q_j \beta_{ij} V_i(t) \leq S(t)$$

де q_j – вартість одиниці ресурсів j -го виду в одиницю часу, а $S(t)$ – обсяг фінансування в момент t .

Кількість ресурсів на i -ту операцію в момент t визначимо із співвідношення:

$$U_i(t) = \sum_{j=1}^m q_j \beta_{ij} V_i(t)$$

Потрібно визначити план робіт, що задовольняє обмеженням на ресурси і забезпечує мінімум упущеної вигоди:

$$\Phi = \sum_{i=1}^n c_i t_i \rightarrow \min$$

7.2 Математичний апарат СППР для оцінки варіантів рішень

Для опису математичних моделей використовується різний математичний апарат: методи суб'єктивної ймовірності, нечіткі множини, нейронні мережі, кусочно-лінійна апроксимація, марковські випадкові процеси, методи математичного програмування та ін. [17, 18]. Деякі методи (далеко не всі), умови та області застосування показані в табл. 7.1.

Яка математична модель, і який математичний апарат є кращим при комп'ютерній підтримці прийняття рішень? Про це йдуть дискусії між фахівцями, що "сповідують" ті чи інші математичні моделі та методи. Проте використання різних методів і алгоритмів для вирішення одного класу задач в математиці є давно і добре відомим явищем. Конкретний метод вибирається залежно від характеру даних і особливості задачі, знання і досвіду експерта в застосуванні методу. Реалізація математичних моделей широко використовується в комерційних пакетах СППР.

Таблиця 7.1

Методи, умови та області застосування математичних моделей

Найменування моделі	Умови застосування	Області застосування
Суб'єктивні ймовірності (Байєсівський аналіз)	Достатній обсяг надійної інформації, яка може бути оброблена статистичними методами. Досліджуваний процес має бути стаціонарним і описуватися формулою Байєса.	Клінічна медицина, оцінка потреби в різних продуктах харчової промисловості, управління водними ресурсами, оцінка надійності роботи реакторів
Нечіткі множини	Алгоритми управління нескладні і можуть бути описані простими правилами, точне визначення параметрів не потрібно або неможливо. Аналітичний опис системи не потрібний, достатньо опису того, як процесом керує досвідчений оператор.	Системи керування автомобілем, потоками транспортних засобів, медична діагностика, управління холодною прокаткою, оцінка надійності споруд та ін..
Багато-критеріальні функції переваги	Володіння досвідом та знаннями дозволяє здійснити критеріальний аналіз ситуації, прогнозувати динаміку подій, оцінити важливість використовуваних критеріїв, дати критеріальну оцінку значень фізичних параметрів і побудувати функцію переваги.	Надзвичайні ситуації, автоматизація проектування, медична діагностика, управління технологічними процесами, прийняття рішень в політиці, бізнесі та ін..
Нейронні мережі	Уміння побудувати загальну функцію, що описує процес управління або розпізнавання, представити її множиною більш простих функцій і розташувати ці прості функції в ієрархічній мережі нейронів.	Розпізнавання образів, геологорозвідка, економічний аналіз, управління технологічними об'єктами (електричними печами, хімічним виробництвом) і т.д.
Генетичні алгоритми	Сформулювати задачу у вигляді функції (алгоритму) не вдається, або знаходження рішення математично сформульованої задачі вимагає неприйнятно великого перебору. Різновид методу випадкового пошуку.	Банківська діяльність, вирішення економічних задач, вибір маршрутів транспортних засобів, аналіз якості програмного забезпечення, вибір варіантів рішення і сценаріїв дій та ін..
Традиційні методи моделювання, оптимізації	Уміння і можливість сформулювати завдання в строгій математичній постановці.	У традиційних областях застосування математичних моделей

8 РОЗРОБКА НЕЧІТКИХ МОДЕЛЕЙ СИСТЕМ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ В СОЦІАЛЬНО-ЕКОНОМІЧНИХ СИСТЕМАХ

Розглянемо методику проектування і побудови СППР на основі моделі нечіткого виводу та бази знань нечітких продукцій на прикладі оцінювання ефективності фінансування інноваційного проекту.

Основою прийняття рішення є бізнес-план проекту, в якому зазначені соціальне значення нововведення, ступінь ризику та оцінка чистого приведенного доходу як міра фінансової ефективності.

8.1 Змістовна постановка задачі оцінювання ефективності інновації

Під *інновацією (нововведенням)* розуміють кінцевий результат інноваційної діяльності, що одержав реалізацію у виді нового або удосконаленого продукту, реалізованого на ринку, нового або удосконаленого технологічного процесу, використовуваного в практичній діяльності. Інновація являє собою результат, отриманий від вкладення капіталу в нову техніку або технологію, у нові форми організації виробництва праці, обслуговування, керування і т.п. Процес створення, освоєння і поширення інновацій називається **інноваційною діяльністю** або інноваційним процесом.

У залежності від технологічних параметрів інновації підрозділяються на продуктові (застосування нових матеріалів, нових напівфабрикатів і комплектуючих; одержання принципово нових продуктів) і процесові (нові методи організації виробництва). Інноваційний процес являє собою послідовність дій по розробці нових продуктів і операцій, по їхній реалізації на ринку.

Інноваційний процес складається з наступних основних кроків:

- зародження ідеї інновації;
- обґрунтування необхідності інноваційної зміни;
- оцінка ефективності інновації;
- розробка та технічна реалізація інноваційної ідеї;
- реалізація інновації в поведінці системи;
- просування інновації на ринку вимог.

Модель інноваційної діяльності в економіці можна представити у вигляді системи управління зі зворотнім зв'язком, як показано на рис 8.1.

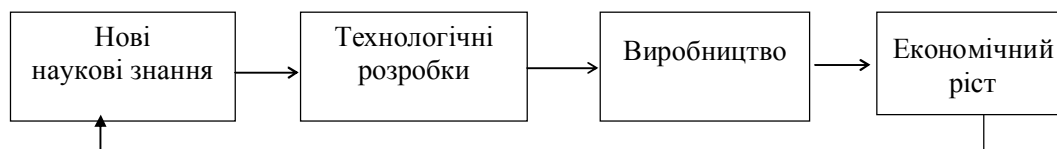


Рис. 8.1. Функціональна послідовність інноваційного процесу в економічній системі

Згідно цієї концепції, нові знання, одержані в результаті фундаментальних, а потім і прикладних досліджень, народжують автоматично нові технології і продукти. Тому нові знання повинні автоматично вести до економічного росту. Кожна стадія потребує фінансових та інших витрат на своє здійснення.

Суб'єкта, який витрачає свої кошти і зусилля на проведення інноваційної діяльності, будемо називати інвестором. Звичайно, інвестор зацікавлений у тому, щоб інноваційна діяльність була ефективною.

Ефективність інноваційної зміни виражається через різноманітні показники. В умовах ринкових відносин не може бути уніфікованої системи показників. Кожен інвестор самостійно визначає цю систему виходячи з особливостей інноваційного проекту, професіоналізму фахівців.

Впровадження нововведень може дати наступні види ефекту: науковий (приріст інформації); продуктовий (покращення якості товару); технологічний (ріст продуктивності праці, покращення умов праці); функціональний (ріст ефективності управління організацією); соціальний (покращення умов життя); екологічний (покращення стану довкілля).

Економічний ефект розробки, впровадження (перетворення в інновацію) або продажу нововведень може бути потенційним або фактичним (реальним, комерційним), а науково-технічний, соціальний і екологічний ефекти можуть мати форму тільки потенційного економічного ефекту. По суті, якщо взяти до уваги тільки кінцеві результати впровадження або продажі нововведень, та будь-який вид інноваційної діяльності можна оцінити у вартісному вираженні. Критеріями кінцевої оцінки тут є: час одержання фактичного економічного ефекту і ступінь невизначеності його одержання (або рівень ризику вкладення інвестицій в інновації).

Не всі оцінки ефективності інновацій можна оцінити чітко. Наприклад, як можна визначити, що ступінь ризику вкладання коштів в деякий проект є високою і варто ще раз все зважити перед прийняттям рішення про його фінансування? Як раз в такому випадку можна скористатися теорією нечітких продукцій.

Розглянемо розробку нечіткої моделі для оцінювання ефективності фінансування інноваційного проекту.

8.2 Описання вхідних і вихідних змінних задач, що розглядається

Змістовна інтерпретація нечіткої моделі припускає вибір і специфікацію вхідних і вихідних змінних відповідної системи нечіткого виводу [1] (передбачається використовувати три вхідні змінні і одну вихідну змінну).

У якості першої вхідної змінної розглянемо ступінь ризику інноваційного проекту, що фінансується. Звичайно, чим вище ступінь ризику, тим більше втрачає сенс інвестувати даний проект. Друга вхідна змінна визначає соціальне значення нововведення, його вплив на життя суспільства. Іноді високе соціальне значення може «компенсувати» помірний і середній ризик чи невеликий економічний ефект проекту.

Третя вхідна змінна – кількісна оцінка економічної ефективності інноваційного проекту – чистий приведений дохід (ЧПД або NPV – Net Present Value), який при умові дискретного характеру параметра часу життєвого циклу може бути представлений формулою:

$$NPV = \sum_{t=0}^T \frac{P(t) - IV(t)}{(1+q)^t},$$

де $P(t)$ – функція чистого прибутку від діяльності по розробці та реалізації нововведення протягом часу життєвого циклу проекту $t \in [0, T]$, $IV(t)$ – функція витрат проекту, які пов'язані з погашенням інвестиційних боргів, q – ставка дисконту (знецінення) грошових одиниць протягом кроку часу.

Вважається, що чим більше додатне значення має величина ЧПД, тим ефективнішим є проект, що фінансується.

У якості вихідної змінної виступає оцінка проекту для інвестора з точки зору ефективності його фінансування, що є основою прийняття рішень про виділення кредитів для реалізації нововведення.

Базуючись на досвіді і отриманих знаннях про ринок інноваційних проектів, інвестор застосовує наступні евристичні правила:

1. Якщо ризик є допустимим, соціальне значення і ЧПД проекту є доволі високими, тоді приймається рішення про фінансування даного проекту. Можна вважати, що в такому випадку ступінь фінансування проекту є 100%.

2. Якщо ризик є допустимим, соціальне значення середнє, тоді міра прийняття рішення про фінансування проекту співпадає з нормованою величиною ЧПД, тобто відношенням ЧПД до свого максимально можливого значення: $\frac{NPV}{NPV_{\max}}$: {«0», «в 5 разів менша за ЧПД», «в 3 рази менша за ЧПД», «в

1.5 рази менша за ЧПД», «ЧПД», «1»}, чи в символічному виді: {P_0, P5, P3, P1_5, P1, P_1}.

3. Якщо ризик є малим, соціальне значення посереднє, тоді міра прийняття рішення про фінансування проекту менше ніж нормована міра ЧПД разів в три.

4. Якщо ризик є малим, а соціальне значення є високим, тоді міра прийняття рішення про фінансування проекту співпадає з нормованою величиною ЧПД.

5. Якщо ризик є малим, а соціальне значення є нижче за середній рівень, тоді міра прийняття рішення про фінансування проекту менше ніж нормована величина ЧПД в разів п'ять.

6. Якщо ризик є середнім, соціальне значення і ЧПД великі, тоді міра прийняття рішення про фінансування проекту в 1,5 рази менше за нормоване ЧПД.

7. Якщо ризик є середнім, соціальне значення середнє, тоді міра прийняття рішення про фінансування проекту в 1,5 рази менше за нормоване ЧПД.

8. Якщо ризик є малим, соціальне значення середнє, тоді міра прийняття рішення про фінансування проекту менше ніж нормована міра ЧПД в три рази.

9. Якщо ризик є малим, а соціальне значення є високим, тоді міра прийняття рішення про фінансування проекту співпадає з нормованою величиною ЧПД.

10. Якщо ризик є малим, а соціальне значення є нижче за середній рівень, тоді міра прийняття рішення про фінансування проекту менше ніж нормована величина ЧПД в разів п'ять.

11. Якщо ризик є середнім, соціальне значення і ЧПД проекту є малими, тоді міра прийняття рішення про фінансування проекту менше ніж нормована величина ЧПД в разів п'ять.

12. Якщо ризик є середнім, соціальне значення малим, тоді міра прийняття рішення про фінансування проекту менше ніж нормована величина ЧПД в разів п'ять.

13. Якщо ризик є середнім, соціальне значення середнє, тоді міра прийняття рішення про фінансування проекту менше ніж нормована міра ЧПД в три рази.

14. Якщо ризик є середнім, а соціальне значення є високим, тоді міра прийняття рішення про фінансування проекту в 1,5 рази менше за нормовану величину ЧПД.

15. Якщо ризик є середнім, а соціальне значення є нижче за середній рівень, тоді міра прийняття рішення про фінансування проекту менше ніж нормована величина ЧПД в разів п'ять.

16. Якщо ризик є середнім, соціальне значення і ЧПД проекту є низькими, тоді приймається рішення про не фінансування даного проекту. Тобто, можна вважати, що в такому «поганому» випадку ступінь фінансування проекту є 0%.

17. Якщо ризик є високим, соціальне значення середнє, ЧПД велике, тоді міра прийняття рішення про фінансування проекту в 1,5 рази нижче нормованої величини ЧПД.

18. Якщо ризик є високим, соціальне значення середнє, ЧПД середнє, тоді міра прийняття рішення про фінансування проекту менше ніж нормована міра ЧПД в три рази.

19. Якщо ризик є високим, а соціальне значення є високим, ЧПД високе, тоді міра прийняття рішення про фінансування проекту співпадає з нормованою величиною ЧПД. Для даного правила можна встановити ступінь істинності 0,75.

20. Якщо ризик є високим, а соціальне значення є нижче за середній рівень, ЧПД мале, тоді міра прийняття рішення про фінансування проекту 0%.

8.3 Фазифікація вхідних і вихідних змінних

Як терм-множину першої вхідної змінної "*Ступінь ризику*" (Risk) будемо використовувати множину $T1 = \{\text{«допустимий», «малий», «середній», «високий»}\}$ або в символічному виді $T1 = \{RD, RM, RS, RV\}$.

Як терм-множину другої вхідної змінної "*Соціальне значення*" (Social-meaning) будемо використовувати аналогічну множину $T2 = \{\text{«низьке», «нижче середнього», «середнє», «високе»}\}$ або в символічному виді $T2 = \{SN, SNS, SS, SV\}$.

Як терм-множину третьої лінгвістичної змінної "*ЧПД*" (NPV) будемо використовувати множину $T3 = \{\text{«мале», «середнє», «високе»}\}$ чи в символічному виді $T3 = \{CN, CS, CV\}$.

Як терм-множину вихідної лінгвістичної змінної "*Міра прийняття проекту на фінансування*" (Credit) будемо використовувати множину $T6 = \{\text{«0», «в 5 разів менша за ЧПД», «в 3 рази менша за ЧПД», «в 1.5 рази менша за ЧПД», «ЧПД», «1»}\}$ чи в символічному виді $T6 = \{P_0, P5, P3, P1_5, P1, P_1\}$.

8.4 Формування моделі системи нечіткого виводу

Наступний етап побудови моделі – побудова бази правил [22]. Використаємо сформульовані 20 правил нечітких продукцій (таблиця 8.1).

Правила нечітких продукцій для розглянутої системи нечіткого виводу

№ правила	Ступінь ризику	Міра соціального ефекту	ЧПД	Міра фінансування
1	RD	SV	CV	P1
2	RD	SS		P2
...
20	RM	SNS		P3

Далі необхідно визначити методи агрегування умов. Оскільки у всіх правилах 1 — 20 як логічне зв'язування для умов застосовується тільки нечітка кон'юнкція (операція "І"), то як метод агрегування використаємо операцію *мін-кон'юнкції*. Для акумуляції виводів правил використаємо метод *тах-диз'юнкції* (що також застосовується у випадку схеми нечіткого виводу методом Мамдані), а для дефазифікації – метод центра ваги [6, 10].

Розробку нечіткої моделі виконаємо з використанням графічних засобів системи MATLAB [3]. З цією метою в редакторі FIS визначимо 3 вхідні змінні з іменами *"ступінь ризику"* (Risk), *"соціальне значення"* (Social), *"ЧПД"* (NPV) і одну вихідну змінну з ім'ям *"міра фінансування"* (Credit). Вид графічного інтерфейсу редактора FIS для цих змінних зображений на рис.8.2.

Для вирішення поставленої задачі нечіткого моделювання використаємо систему нечіткого виводу типу Сугено. Залишимо без зміни параметри розроблювальної нечіткої моделі, запропоновані системою MATLAB. Визначимо функції приналежності термів для кожної з 3-х вхідних і вихідної змінних. Для цієї мети скористаємося редактором функцій приналежності системи MATLAB. Графічний інтерфейс редактора функцій приналежності для вихідної змінної *"міра фінансування проекту"* зображений на рис.8.3.

Далі задамо 20 правил для розроблюваної системи нечіткого виводу. Для цієї мети скористаємося редактором правил системи MATLAB. Вид графічного інтерфейсу редактора правил після завдання всіх 20 правил нечіткого виводу зображений на рис.8.4.

Оскільки в робочому вікні відображаються не всі змінні нечіткої моделі, для керування режимом відображення змінних правил варто скористатися спеціальними кнопками, розташованими в нижній правій частині редактора правил.

Тепер можна виконати аналіз побудованої системи нечіткого виводу для розглянутої задачі оцінки фінансової заможності клієнтів.

Відкриємо вікно перегляду правил системи MATLAB і введемо значення вхідних змінних для випадку, коли значення вхідної змінної *"ступінь ризику"* оцінюється в 0,08, значення вхідної змінної *"соціальне значення"* також оцінюється в 0,48 балів, значення вхідної змінної *"ЧПД"* оцінюється в 900. Це досить високі оцінки вхідних змінних, котрі навіть на інтуїтивному рівні свідчать на користь відповідного клієнта.

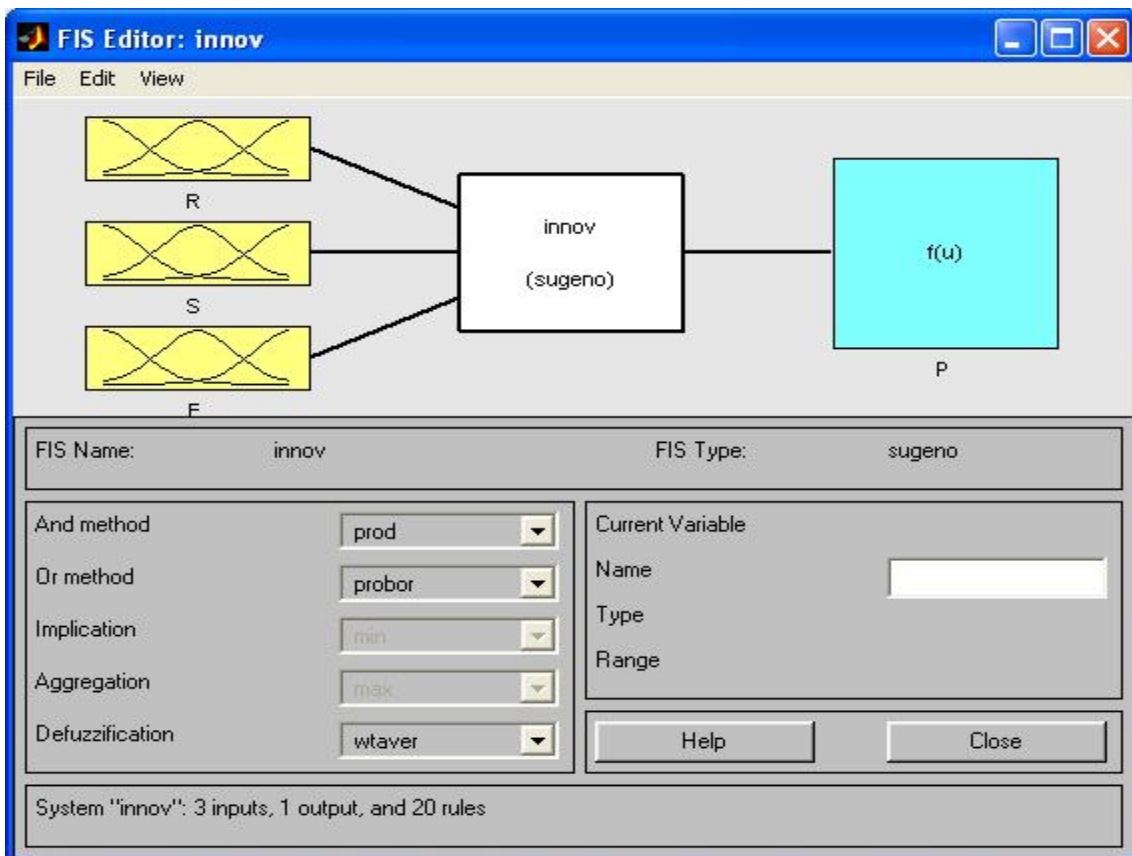


Рис. 8.2. Графічний інтерфейс редактора FIS після визначення входніх і вихідної змінних системи нечіткого виводу mortgage

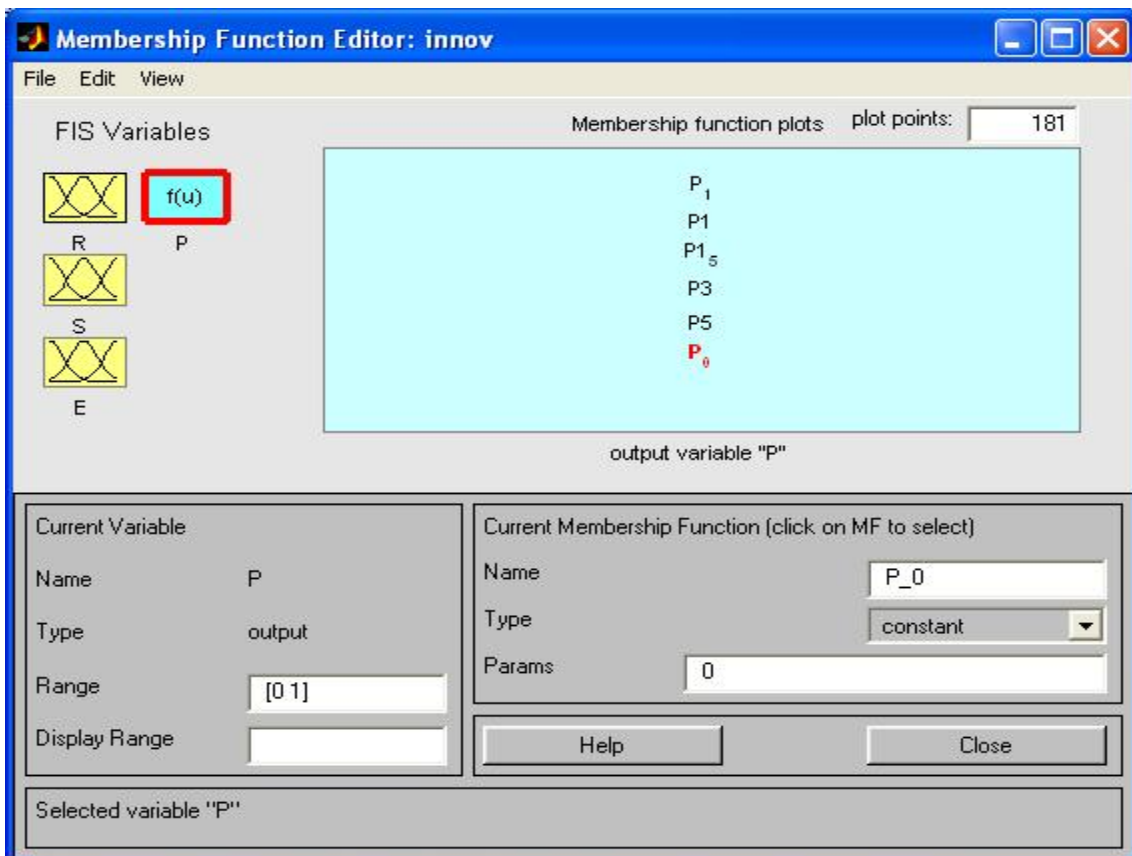


Рис. 8.3. Графічний інтерфейс редактора функцій належності для вихідної змінної "міра фінансування"

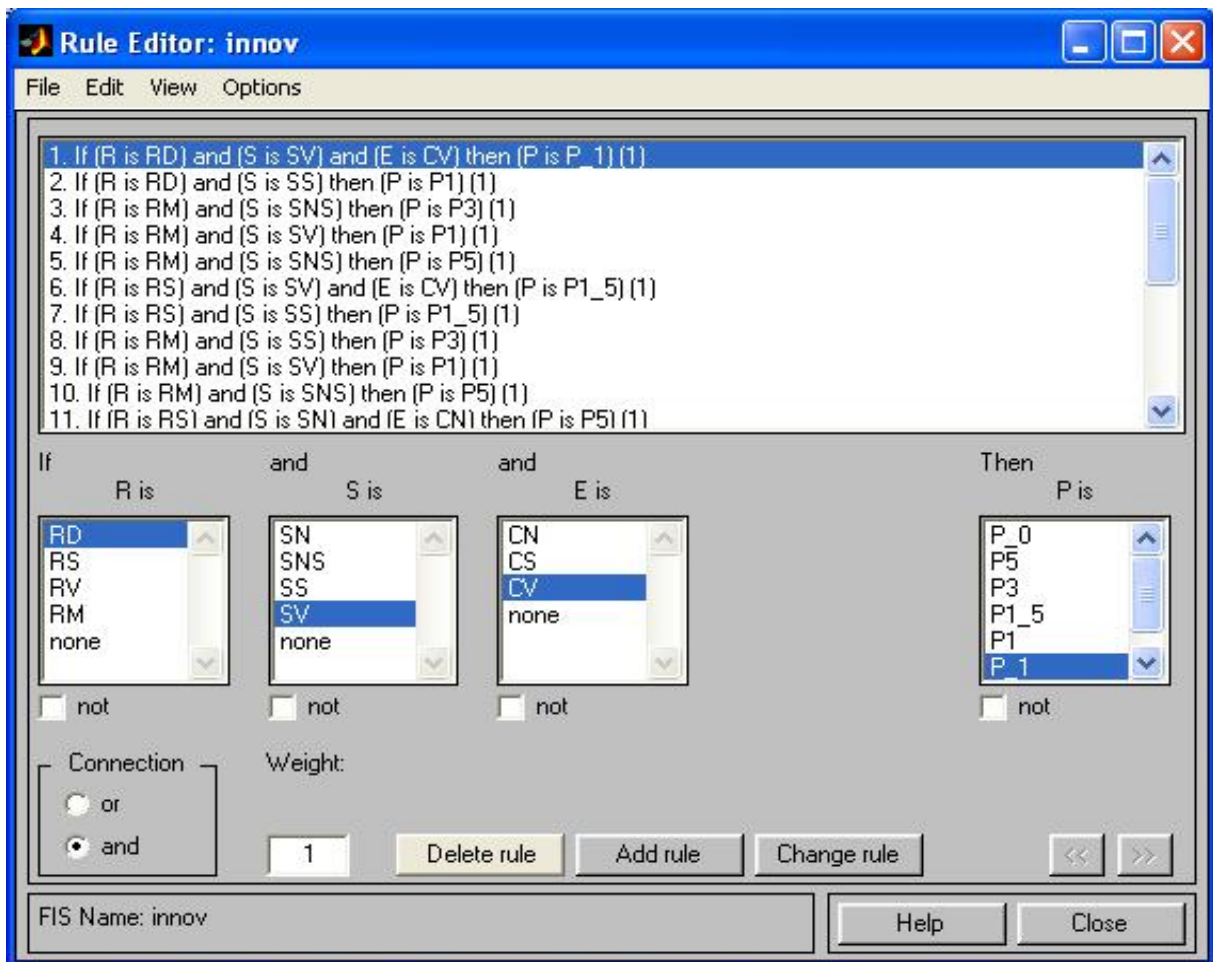


Рис. 8.4. Графічний інтерфейс введення правил

Процедура нечіткого виводу, виконана системою MATLAB для розробленої нечіткої моделі, видає в результаті значення вихідної змінної "міра фінансування проекту", рівне 50,6% (рис.8.5).

Це досить висока оцінка фінансової заможності потенційного клієнта, що може служити підставою для позитивного рішення з боку банку про надання кредиту під заставу. Як можна заключити, даний висновок цілком узгоджується з раніше висловленими інтуїтивними розуміннями.

Далі можна визначити міру фінансування проекту для інших варіантів значень вхідних змінних і проглянути ті поверхні, які системи пропонує як результат своєї роботи (рис. 8.6). Дана поверхня дозволяє установити залежність значень вихідної змінної від значень окремих вхідних змінних нечіткої моделі. Аналіз цих залежностей може служити підставою для зміни функцій належності вхідних змінних чи нечітких правил з метою підвищення адекватності системи нечіткого виводу для конкретних стратегій інвесторів.

Слід зазначити декілька спрощених характеристик моделі у порівнянні з реально використовуваною в процесі прийняття рішень керівництвом інвестиційного фонду. У той же час розглянута нечітка модель має досить високу адекватність, що обумовлює її успішне застосування в практиці фінансових операцій.

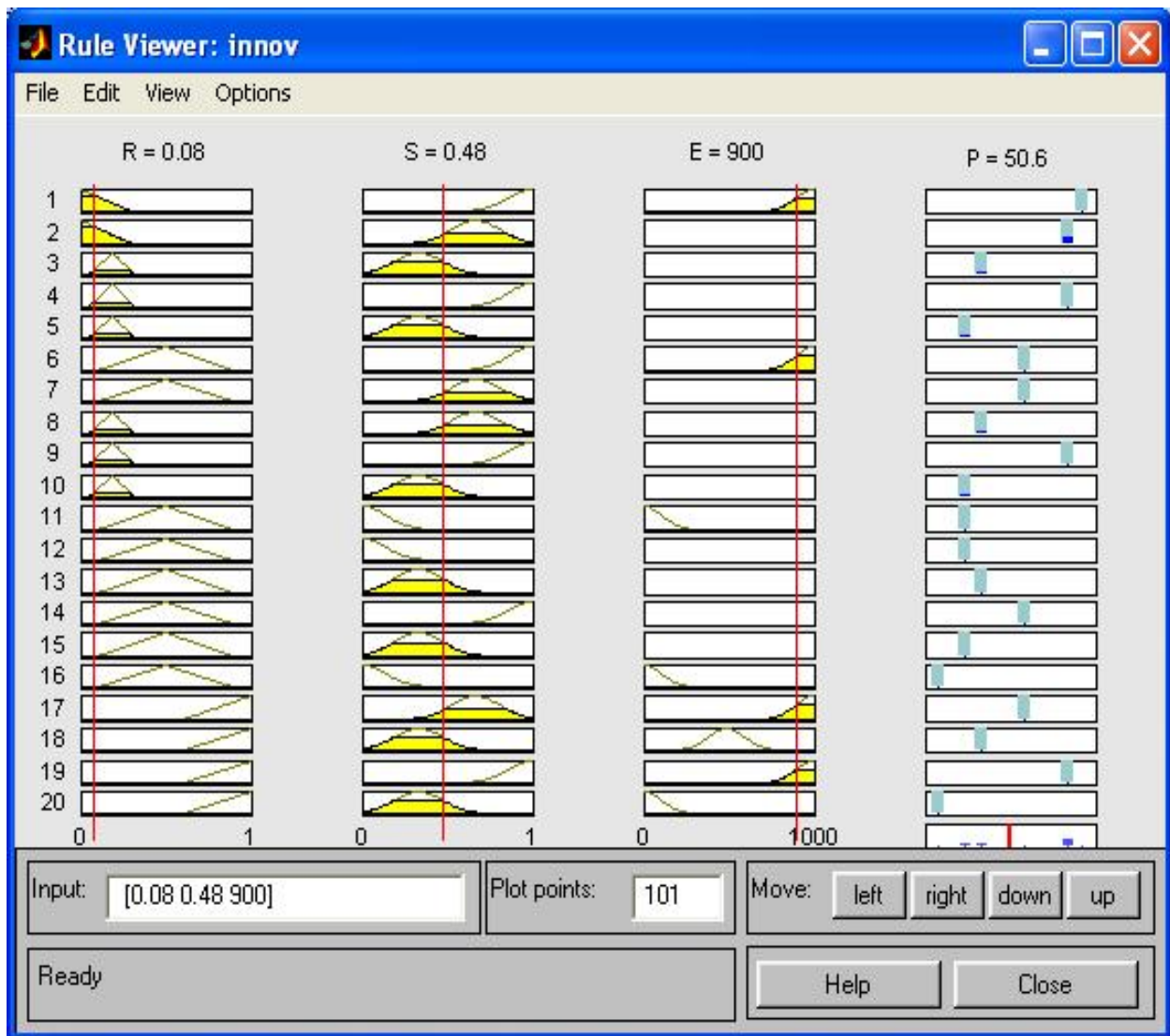


Рис. 8.5. Графічний інтерфейс програми перегляду правил після виконання процедури нечіткого виводу 2 варіанту

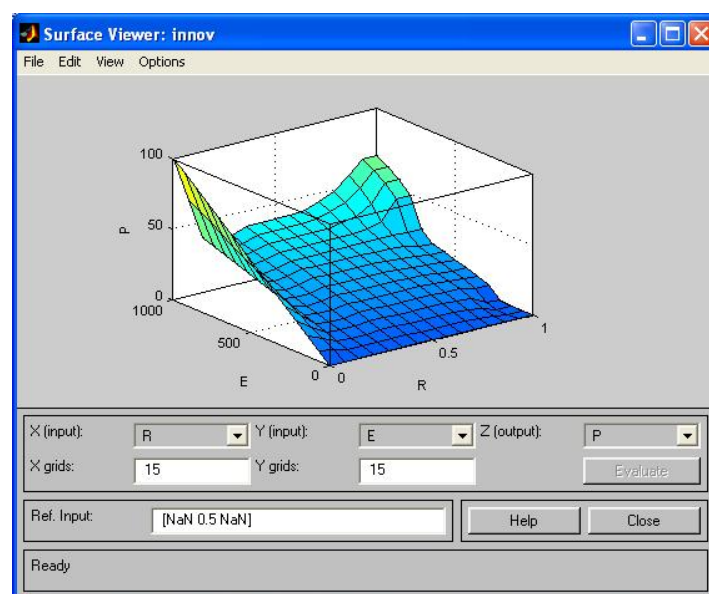


Рис. 8.6. Візуалізація поверхні нечіткого виводу розглянутої моделі для вхідних змінних "місцезнаходження" і "обробка"

9 ПРОГРАМНІ ПРОДУКТИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

9.1 СППР "Мислитель"

Головне призначення програми "Мислитель" (<http://spirit-prog.ru>) – полегшення вибору одного варіанта з декількох альтернативних. Вибір такого типу постійно зустрічається в житті кожної людини. Прикладами подібного вибору можуть служити вибір квартири, вибір машини, вибір місця для поїздки у відпустку, вибір банку для вкладення грошей і т.п. Нижче наведено послідовність дій під час роботи з програмою "Мислитель".

1. Запуск програми. Вікно вітання і стислої інформації про програмний продукт. (рис. 9.1).

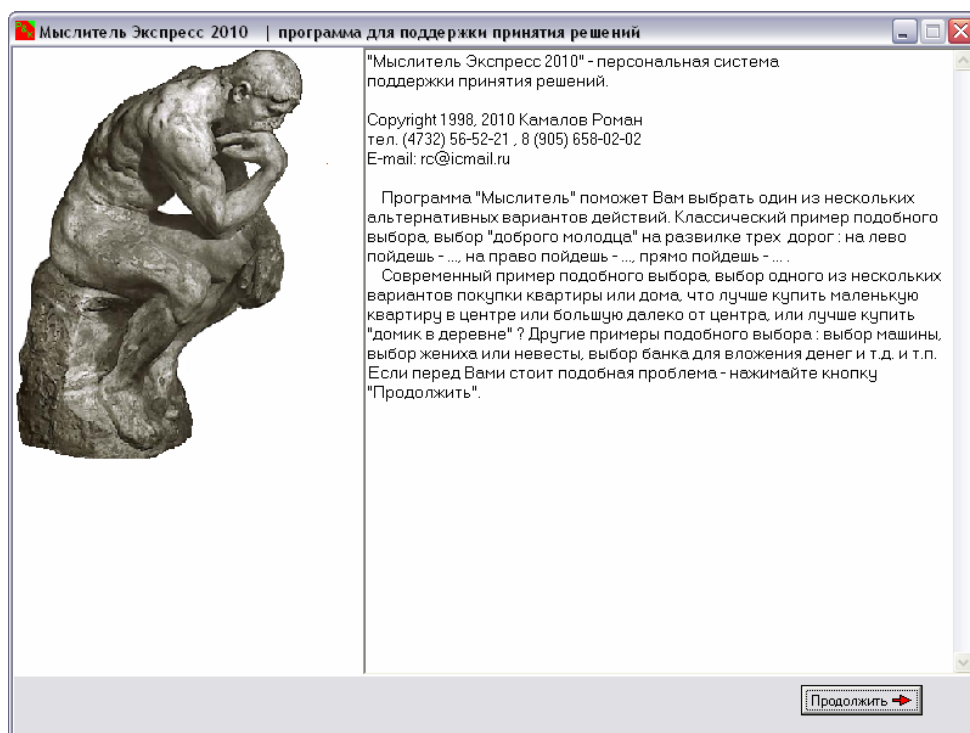


Рис. 9.1 Вікно запуску програми "Мислитель"

2. Після натиску кнопки "Продовжити" відкриється список завдань, які занесені розробником. Користувач може додавати власні завдання для вирішення і зберігати їх як шаблон. Для ознайомлення виберемо перше завдання, яке полягає у виборі квартири по 5 критеріям. Натискаємо кнопку «Продовжити» (рис. 9.2).

3. Відкриється список альтернатив, який може бути змінений. Мінімум дві альтернативи, з яких буде обрана найкраща, максимальна кількість альтернатив – 9. Будемо розглядати завдання з трьома альтернативами. Натискаємо кнопку «Продовжити» (рис. 9.3).

4. Далі необхідно заповнити список критеріїв, за якими буде визначатися найкраща альтернатива. Мінімальна кількість критеріїв – 2, максимальне – 9. Залишимо список критеріїв без зміни. Натискаємо кнопку «Продовжити» (рис. 9.4).

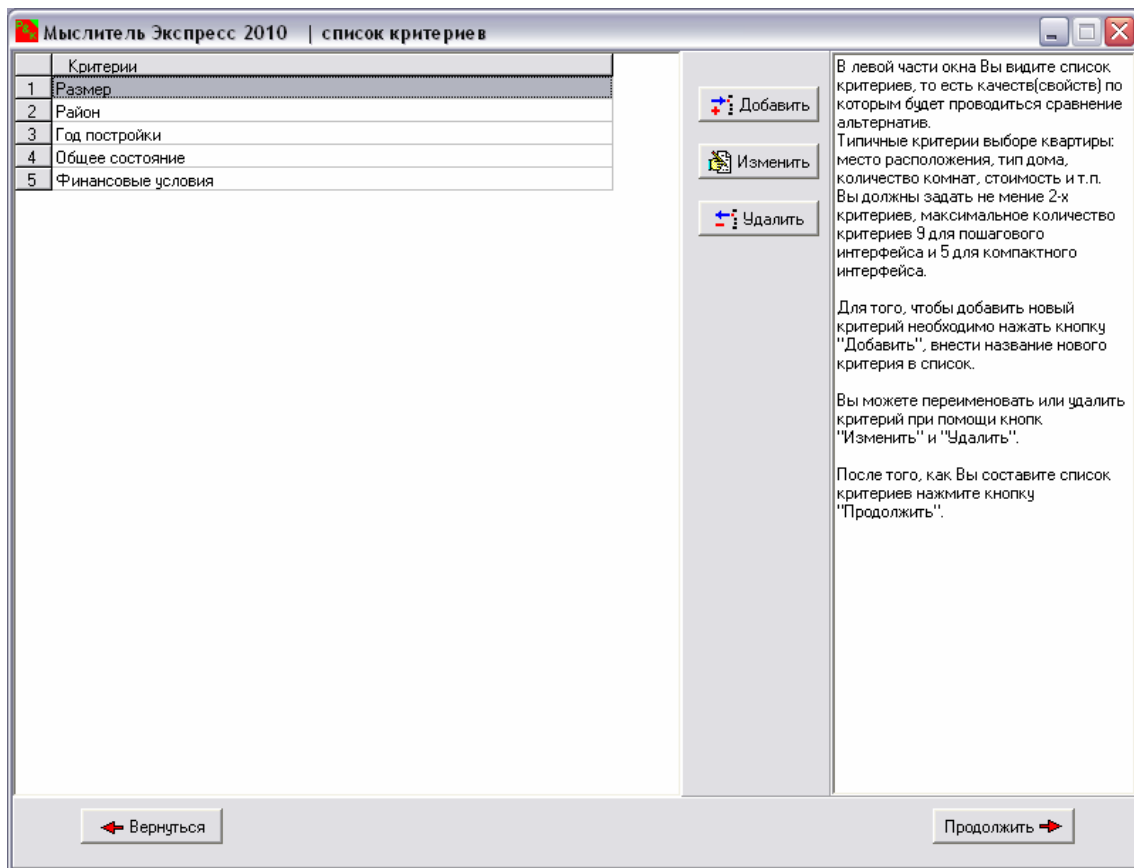


Рис. 9.2 Вікно списку завдань

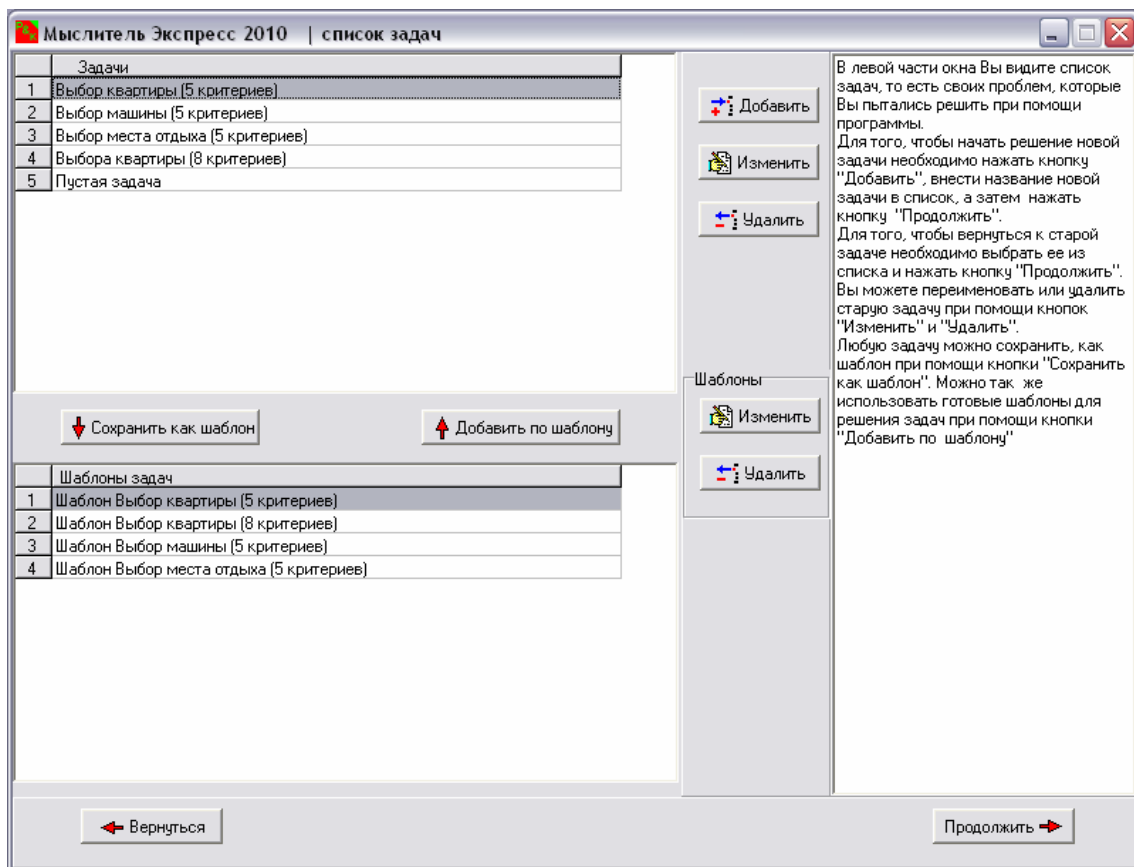


Рис. 9.3 Вікно списку альтернатив

5. Далі бачимо таблицю – матрицю порівняння, в яку необхідно ввести оцінку критеріїв для кожної альтернативи. Спочатку необхідно встановити важливість критеріїв, для цього в стовпці «Важливість критеріїв», який виділений зеленим кольором, необхідно встановити відповідну оцінку для кожного критерію. Найважливішому критерію відповідає 1, а найменш важливому оцінка 10. У даному прикладі для найбільш важливих критеріїв «Загальний стан», «Фінансові умови» ставимо в оцінці критеріїв значення 1. Наступний по значимості критерій – «Розмір», в якості оцінки ставимо значення 2 . Критерії «Район» і «Рік побудови» мають найменшу значимість, отже, як важливість критерію встановлюємо значення 3.

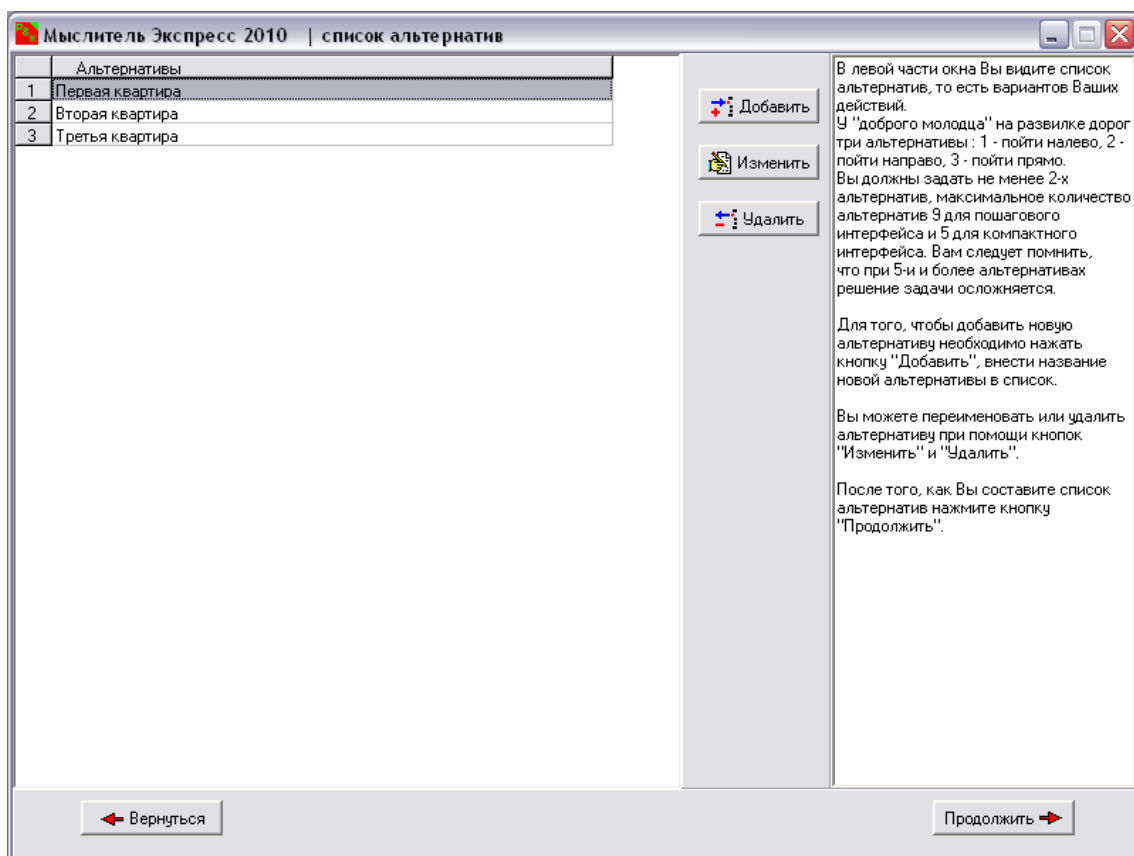


Рис. 9.4 Вікно списку критеріїв

6. Далі необхідно дати оцінку кожної альтернативи, квартири за встановленими критеріями. Для цього необхідно вносити значення в стовпці, що мають синій колір. Після зміни будь-якого поля, програма тут же перераховує результати, і видає рішення після кожної зміни вхідних даних. Результати – в правій частині вікна. Програма визначає найкращу альтернативу, а також показує процентне співвідношення всіх альтернатив. В результаті проведеного порівняння програма видає в якості найкращої альтернативи другу. Слід зазначити, що програма має дуже простий інтерфейс, а також миттєве визначення результату при будь-якій зміні даних, що дозволяє легко і просто зробити аналіз альтернатив і визначити оптимальну.

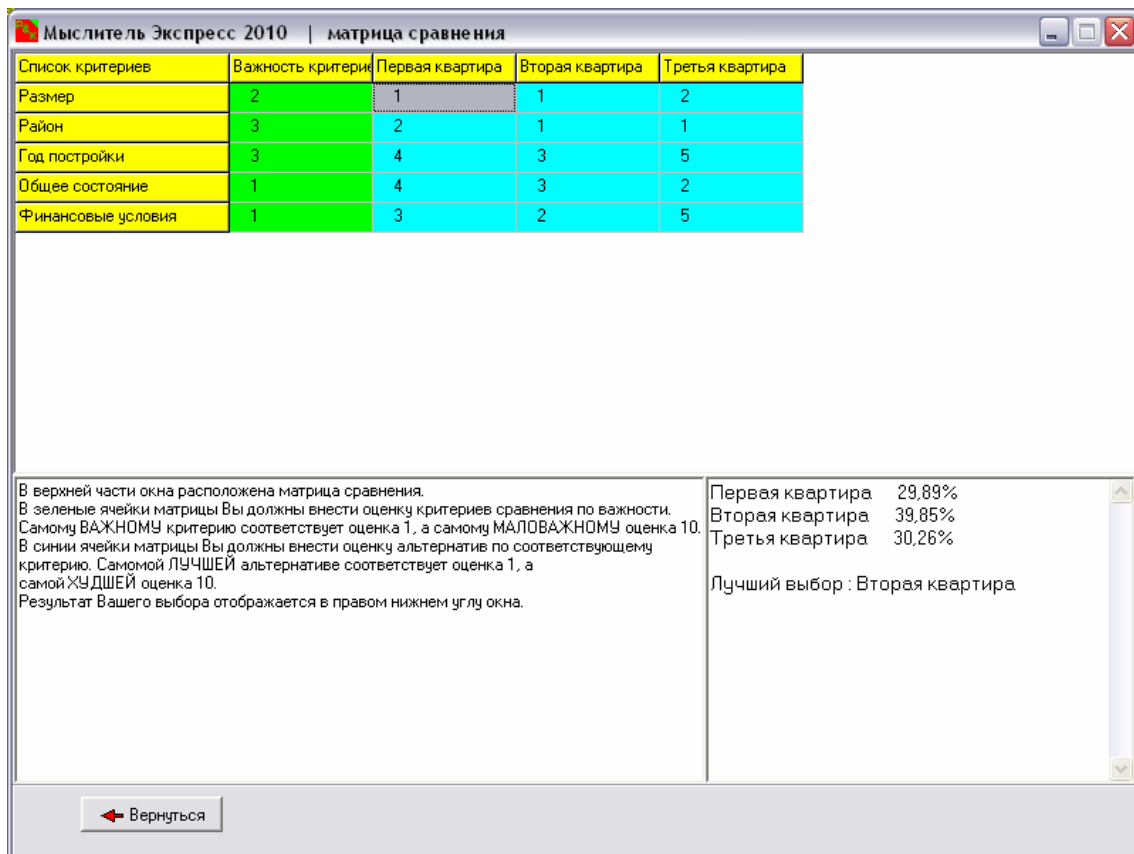


Рис. 9.5 Матрица порівняння альтернатив

9.2 СППР "Вибір"

Система підтримки прийняття рішень "Вибір" (<http://www.ciritas.ru>) – аналітична система, що базується на методі аналізу ієрархій [15], і є простим та зручним засобом, який допоможе:

- структурувати проблему;
- побудувати набір альтернатив;
- виділити фактори, що характеризують її;
- задати значимість цих факторів;
- оцінити альтернативи по кожному з факторів;
- знайти неточності і протиріччя в судженнях особи, що приймає рішення (ОПР) / експерта;
- проранжувати альтернативи;
- провести аналіз рішення і обґрунтувати отримані результати.

Запускаємо програму. Створюємо простий проект, щоб ознайомитись з основними можливостями програми. Створюємо три рівні: рівень мети, рівень критеріїв і рівень альтернатив.

Постановка завдання: вибрати машину по 3 критеріям. Автомобіль для міської їзди, отже, менша витрата палива має більшу перевагу перед великою потужністю двигуна. Вартість автомобіля відіграє вирішальну роль.

Отже, рівень мети – купити машину. Критерії, за якими буде проводитися вибір автомобіля: вартість, витрата палива, потужність двигуна.

В якості альтернатив маємо 3 автомобіля:

1. Kia Ceed

2. Mitsubishi Lancer X
3. Volkswagen Golf

Для більш достовірного аналізу візьмемо реальні дані з сайтів дистриб'юторів. Вихідні дані зведено в таблицю 9.1.

Таблиця 9.1

Прикладові дані – характеристики автомобілів

Характеристики автомобілів	Kia Ceed	Mitsubishi Lancer X	Volkswagen Golf
Вартість, \$	18 000	18 400	26 600
Витрата палива, л/100 км, Змішаний цикл	6.8	6.4	6.0
Потужність двигуна, л.с.	105	109	122

Створюємо 3 рівня у програмі «Вибір». Встановлюємо зв'язки. Мета залежить від 3 критеріїв. Кожен критерій впливає на кожну альтернативу. В результаті отримуємо 3 рівня, 7 вузлів, 12 зв'язків (рис. 9.6).

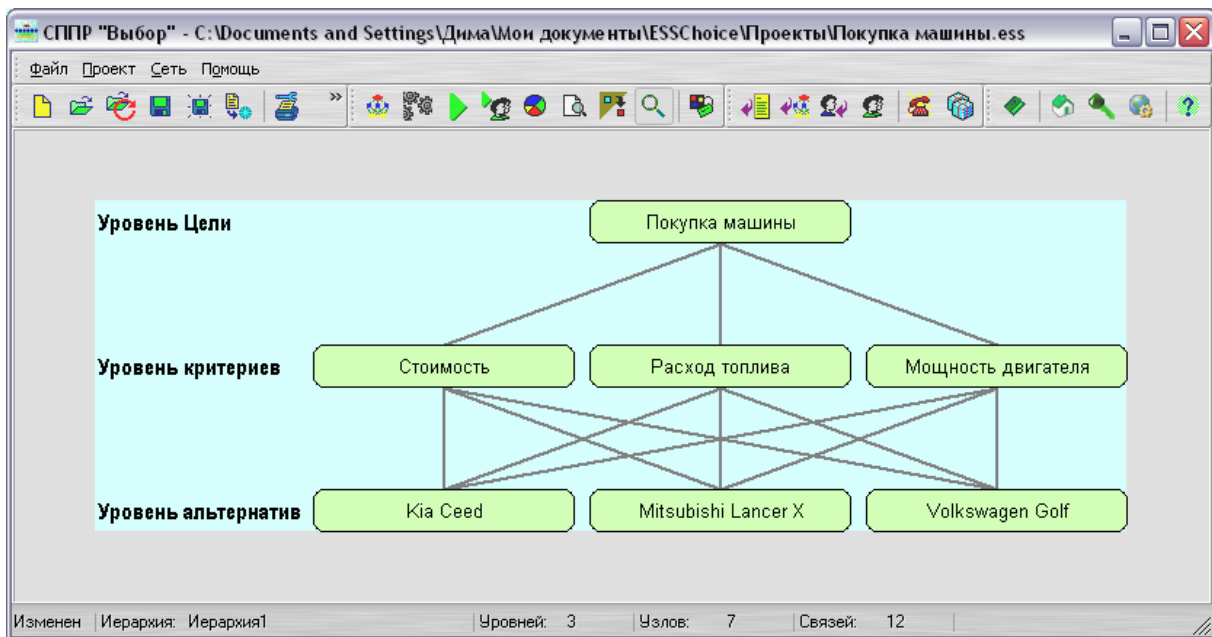


Рис. 9.6 Встановлення рівня мети, критеріїв та альтернатив у СППР "Вибір"

Отримання матриці парних порівнянь – основна дія при розрахунках методом аналізу ієрархій. Саме на підставі даної матриці і обчислюються ваги експертів, альтернатив.

Інструмент побудови матриці парних порівнянь для якого-небудь вузла ієрархії являє собою вікно, в лівій частині якого розташовано опис вузла, щодо якого необхідно провести порівняння факторів вказаного рівня, а також список факторів, між якими повинно проводитися порівняння (рис. 9.7).

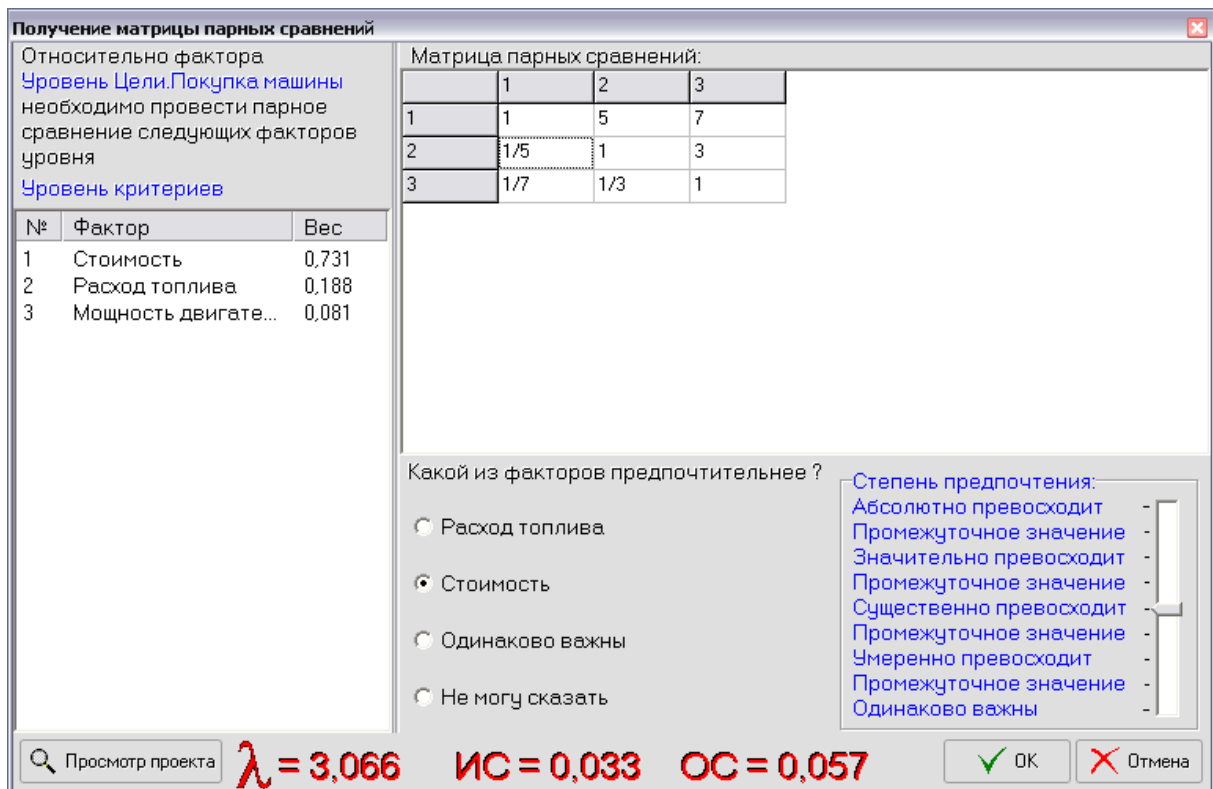


Рис. 9.7 Інструмент побудови матриці парних порівнянь у СППР "Вибір"

Права верхня частина вікна відображає саму матрицю парних порівнянь.

Діагональ матриці завжди містить одиниці (тому при порівнянні вузла з самим собою завжди має бути відповідь "Однаково важливі") і не доступна для редагування.

Матриця є зворотно-симетричною, тому досить заповнити частину матриці, яка розташована вище головної діагоналі, або нижче. Симетричний осередок матриці при цьому автоматично буде заповнена зворотним значенням.

Права нижня частина вікна містить інструменти, що допомагають заповнити матрицю:

– «Який з факторів кращий?» – Область відображення питань для заповнення матриці парних порівнянь. Для кожного осередку матриці в даній області вікна буде відображатися своє питання. Відповідь на питання здійснюється шляхом перемикавання на один із запропонованих варіантів відповідей. Наприклад, щодо фактора "Покупка машины" альтернатива "Вартість" буде переважніше альтернативи "Витрата палива";

– «Ступінь переваги» – область відображення інструменту для визначення ступеня переваги однієї альтернативи над іншою щодо зазначеного фактора. Для визначення ступеня переваги досить вибрати одне із запропонованих значень або ввести своє значення. У даному випадку вибираємо "Істотно перевершує". Нижня частина вікна відображає розрахункові дані, які необхідні для контролю узгодженості відповідей експертів.

Наприклад, індекс узгодженості матриці (IU) і ставлення узгодженості не повинно бути більше 0.1. Якщо $IU > 0.1$ або $OY > 0.1$, то це означає, що у своїх

відповідях експерт суперечить сам собі. Lambda – власне число матриці, не відображає міру правильності відповідей.

Приклад. Купівля машини

Для початку встановлюємо значення матриць парних порівнянь, для кожного рівня виходячи з власних побажань, встановлюємо ваги критеріїв (рис. 9.8).

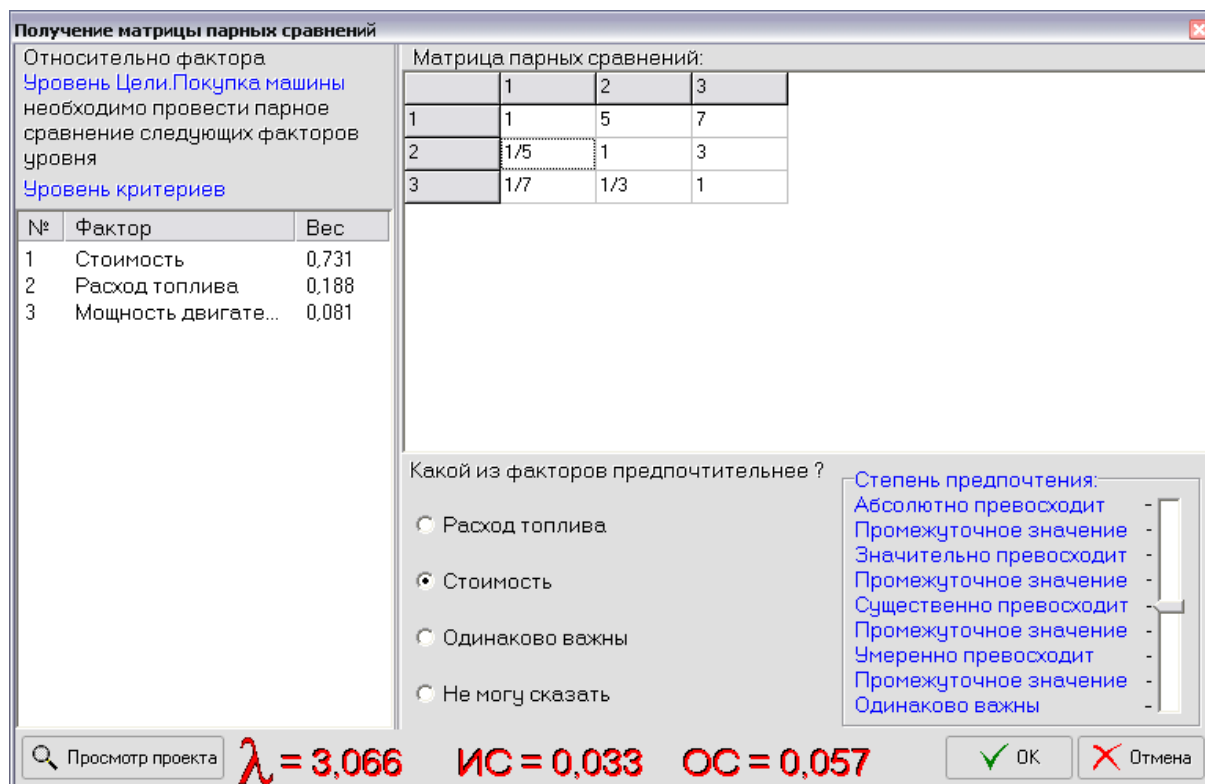


Рис. 9.8 Встановлення значень матриці парних порівнянь

В результаті отримали ваги критеріїв. Вартість – 0,731, Витрата палива – 0,188, Потужність двигуна – 0,081.

Натискаємо кнопку «ОК». Далі переходимо на рівень нижче. Тепер щодо кожного критерію: вартість, витрата палива, потужність двигуна, заповнюємо матриці парних порівнянь, в залежності від обраних альтернатив.

Отримано наступні результати:

– ваги альтернатив відносно критерія «Вартість»: *Kia Ceed* – 0,751, *Mitsubishi Lancer X* – 0,178, *Volkswagen Golf* – 0,070.

– ваги альтернатив відносно критерія «Витрата палива»: *Kia Ceed* – 0,059, *Mitsubishi Lancer X* – 0,162, *Volkswagen Golf* – 0,778.

– ваги альтернатив відносно критерія «Потужність двигуна»: *Kia Ceed* – 0,081, *Mitsubishi Lancer X* – 0,188, *Volkswagen Golf* – 0,731.

З урахуванням усіх матриць порівняння загальний індекс узгодження менший за 0,1, отже експерт не суперечить сам собі.

На рис. 9.9 представлено діаграму результатів.

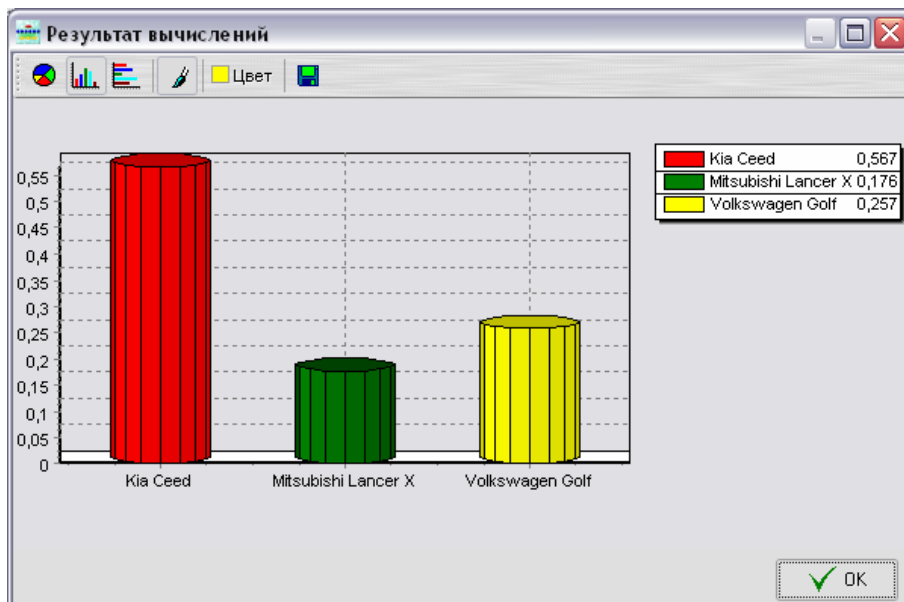


Рис. 9.9 Діаграма результатів

У результаті бачимо, що на підставі 3-х критеріїв оптимальним вибором буде Kia Ceed. Це найдешевший автомобіль з розглянутих альтернатив. Але бачимо, що найдорожчий з усіх Volkswagen Golf краще Mitsubishi Lancer X завдяки більш потужному і в той же час більш економічному двигуну.

Обидві розглянуті програми працюють на підставі одного алгоритму «Метод аналізу ієрархій». Перевагою першої програми є дуже простий і зручний інтерфейс, який дозволяє швидко розібратися в програмі і провести аналіз. Друга програма, «Вибір», є більш складною. У ній важче швидко зробити простий аналіз, але вона дозволяє будувати більш складні моделі, дозволяє проводити мережеву експертизу (участь віддалених експертів), що може стати ключовим моментом при виборі програмного забезпечення. Програма «Мислитель» позиціонується швидше як навчальний варіант реалізації алгоритму «Метод аналізу ієрархій», а програма «Вибір» націлена на організації, в яких необхідно забезпечити комп'ютерну підтримку прийняття рішень.

9.3 СППР Right Solution

У програмі Right Solution (<http://www.right-solution.com>) для прийняття рішення використовується метод аналізу ієрархій.

Метод аналізу ієрархій (MAI) – математичний інструмент системного підходу до складних проблем прийняття рішень. MAI не наказує особі, що приймає рішення (ОПР), будь-якого «правильного» рішення, а дозволяє йому в інтерактивному режимі знайти такий варіант (альтернативу), який найкращим чином узгоджується з його розумінням суті проблеми і вимогами до її вирішення. Цей метод розроблений американським математиком Томасом Сааті, який написав про нього книги, розробив програмні продукти і протягом 20 років проводить симпозиуми ISAHN (International Symposium on Analytic Hierarchy Process). MAI широко використовується на практиці і активно розвивається вченими всього світу.

У його основі поряд з математикою закладені і психологічні аспекти. МАІ дозволяє зрозумілим і раціональним чином структурувати складну проблему прийняття рішень у вигляді ієрархії, порівняти і виконати кількісну оцінку альтернативних варіантів рішення. Метод аналізу ієрархій використовується у всьому світі для прийняття рішень в різноманітних ситуаціях: від управління на міждержавному рівні до розв'язання галузевих і приватних проблем в бізнесі, промисловості, охороні здоров'я та освіті. Для комп'ютерної підтримки МАІ існують програмні продукти, розроблені різними компаніями.

Аналіз проблеми прийняття рішень в МАІ починається з побудови ієрархічної структури, яка включає мету, критерії, альтернативи та інші фактори, що впливають на вибір. Ця структура відображає розуміння проблеми ОПР. Кожен елемент ієрархії може представляти різні аспекти розв'язуваної задачі, причому до уваги можуть бути прийняті як матеріальні, так і нематеріальні чинники, вимірювані кількісні параметри та якісні характеристики, об'єктивні дані і суб'єктивні експертні оцінки. Аналіз ситуації вибору рішення в МАІ нагадує процедури і методи аргументації, які використовуються на інтуїтивному рівні. Наступним етапом аналізу є визначення пріоритетів, що представляють відносну важливість або перевагу елементів побудованої ієрархічної структури, за допомогою процедури парних порівнянь. Безрозмірні пріоритети дозволяють обґрунтовано порівнювати різнорідні фактори, що є відмінною рисою МАІ. На заключному етапі аналізу виконується синтез (лінійна згортка) пріоритетів на ієрархії, в результаті якої обчислюються пріоритети альтернативних рішень щодо головної мети. Кращою вважається альтернатива з максимальним значенням пріоритету.

Приклад прийняття рішення за допомогою програми Right Solution.

Відразу після запуску програма пропонує вам ввести рішення, яке потрібно прийняти (рис. 9.10).

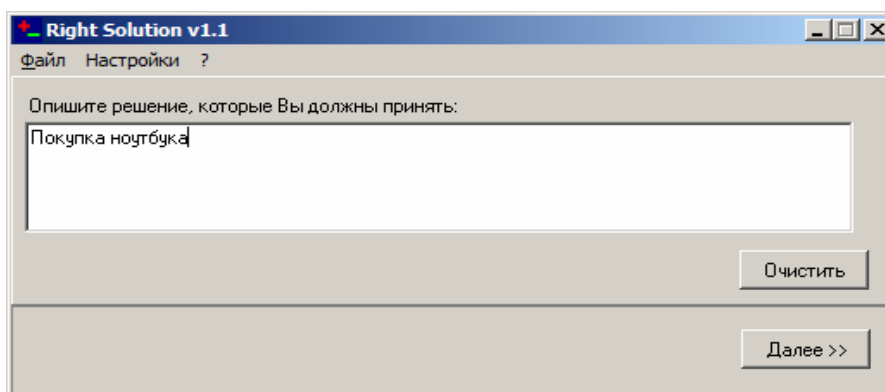


Рис. 9.10 Приклад постаановки задачі прийняття рішення за допомогою програми Right Solution

На першому кроці вказуємо позитивні сторони рішення, відсортувавши їх за пріоритетами так, щоб найбільш значущі виявилися зверху списку. Вводимо в список всі плюси ухвалення свого рішення, причому спочатку найбільші, потім не самі істотні. Тут же потрібно визначити пріоритети кожного пункту і оцінити його вірогідність. Пишемо текст, тиснемо кнопку [Додати]. Потім в списку напроти

пункту вибираємо його пріоритет і ймовірність (рис. 9.11). Можна вводити значення цих параметрів коефіцієнтом від 1 до 100, що показує, у скільки разів поточне значення є більш пріоритетним, ніж попереднє. Після визначення всіх позитивних сторін рішення тиснемо [Далі] і переходимо до наступного етапу.

Опишите положительные стороны решения, отсортировав их по приоритетам так, чтобы наиболее значимые "плюсы" оказались вверху списка. Для каждого пункта определите насколько он приоритетнее предыдущего. Приблизительно оцените, вероятность каждого пункта.

При задании приоритета можно либо выбрать значение из выпадающего списка, либо ввести свой коэффициент (1-100), показывающий во сколько раз текущее значение приоритетнее предыдущего. Аналогичным образом задается значение вероятности.

Добавить

Наименование	Приоритет	Вероятность
Возможность работы в дороге	значительный	достоверно
Независимость от электроэнергии	незначительны	20%
Портативность	существенный	достоверно
Экономия места	незначительны	достоверно
Персонализация работы	несущественны	50 на 50

Удалить

<< Назад Далее >>

Рис. 9.11 Встановлення пріоритетів рішень

На другому кроці – введення всіх негативних сторін рішення (так само, як і в попередньому кроці, з тією лише різницею, що вказуємо негативні сторони).

Третій крок – визначаємо, що переважає: найбільший плюс або найбільший мінус. Для цього треба ввести числове значення або вибрати приблизну оцінку в випадаючому списку (рис. 9.12).

Опишите отрицательные стороны решения, отсортировав их по приоритетам так, чтобы наиболее существенные "минусы" оказались вверху списка. Определите приоритетность каждого пункта относительно предыдущего. Приблизительно оцените, вероятность каждого пункта.

При задании приоритета можно либо выбрать значение из выпадающего списка, либо ввести свой коэффициент (1-100), показывающий во сколько раз текущее значение приоритетнее предыдущего. Аналогичным образом задается значение вероятности.

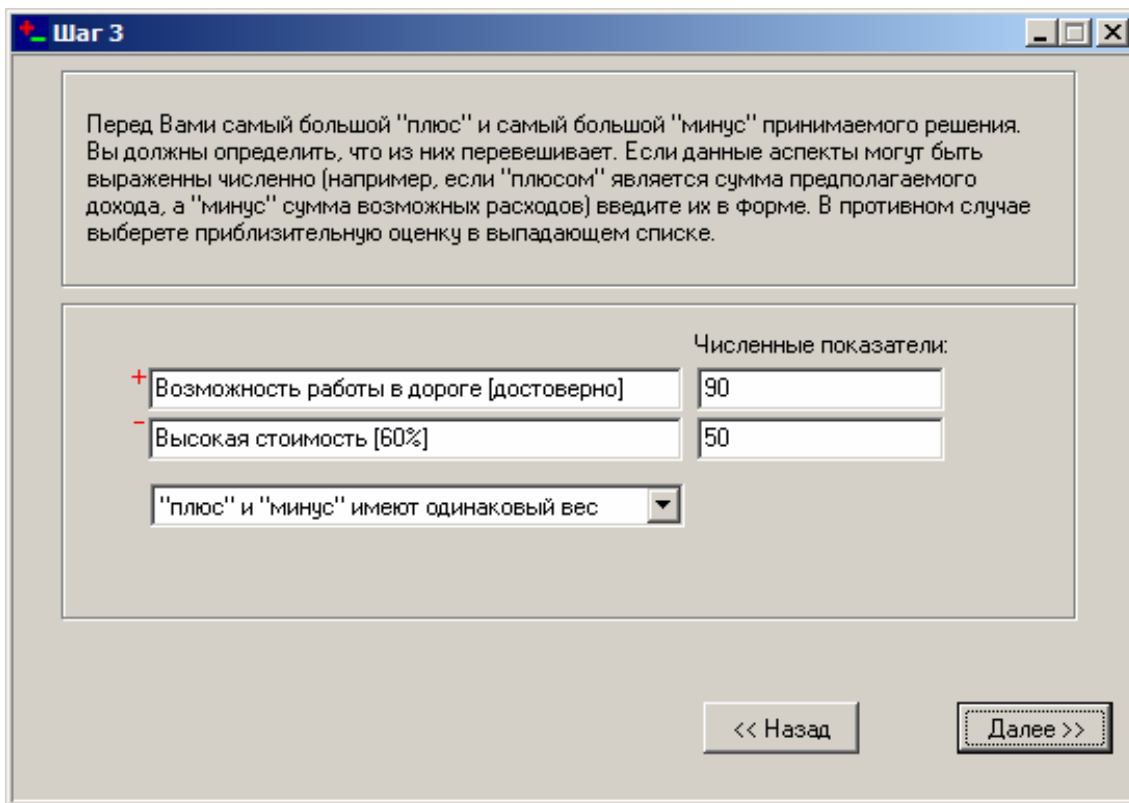
Добавить

Наименование	Приоритет	Вероятность
Высокая стоимость	существенный	60%
Невозможность модернизации	незначительны	высокая
Отсутствие модульности в случае поломки	незначительны	достоверно
Не большой экран	несущественны	высокая

Удалить

<< Назад Далее >>

Рис. 9.12 Вікна встановлення параметрів прийняття рішень у програмі Right Solution



Продовження рис. 9.12 Вікна встановлення параметрів прийняття рішень у програмі Right Solution

На останньому кроці відображається запропоноване програмою рішення. Для розглянутого вище прикладу варто ухвалити рішення про купівлю ноутбуку.

9.4 СППР FreeMind

FreeMind (<http://freemind.sourceforge.net>) – одна з найпоширеніших, безкоштовних програм для складання діаграм зв'язків. Діаграма зв'язків, відома також як карта знань (*Mind map*) або асоціативна карта – спосіб зображення процесу загального системного мислення за допомогою схем. Також може розглядатися як зручна техніка альтернативного запису.

Діаграма зв'язків реалізується у вигляді деревовидної схеми, на якій зображені слова, ідеї, завдання або інші поняття, пов'язані гілками, що відходять від центрального поняття або ідеї. В основі лежить принцип «радіантного мислення», що відноситься до асоціативних розумовим процесам, точкою прикладання яких є центральний об'єкт (радіант – точка небесної сфери, з якої як би виходять видимі шляхи тіл з однаково спрямованими швидкостями, наприклад, метеорів одного потоку). Це показує нескінченну різноманітність можливих асоціацій і невичерпність можливостей мозку. Подібний спосіб запису дозволяє діаграмі зв'язків необмежено рости і доповнюватися.

Діаграми зв'язків використовуються для створення, візуалізації, структуризації та класифікації ідей, а також як засіб для навчання, організації, вирішення завдань, прийняття рішень, при написанні статей.

Переваги програми FreeMind:

- інтуїтивно зрозуміле управління;

- наявність основних функціональних можливостей для побудови Mind Maps;
- можливість зберігати карту в різних форматах (jpeg, pdf, html та ін.).

Особливості програми:

- необхідно перед інсталяцією програми встановити Java;
- неможливо прикріплювати документи і файли до гілок;
- графічні елементи досить низької якості, проте можна прикріплювати власні.

Основні елементи головного вікна програми представлене на рис. 9.13.

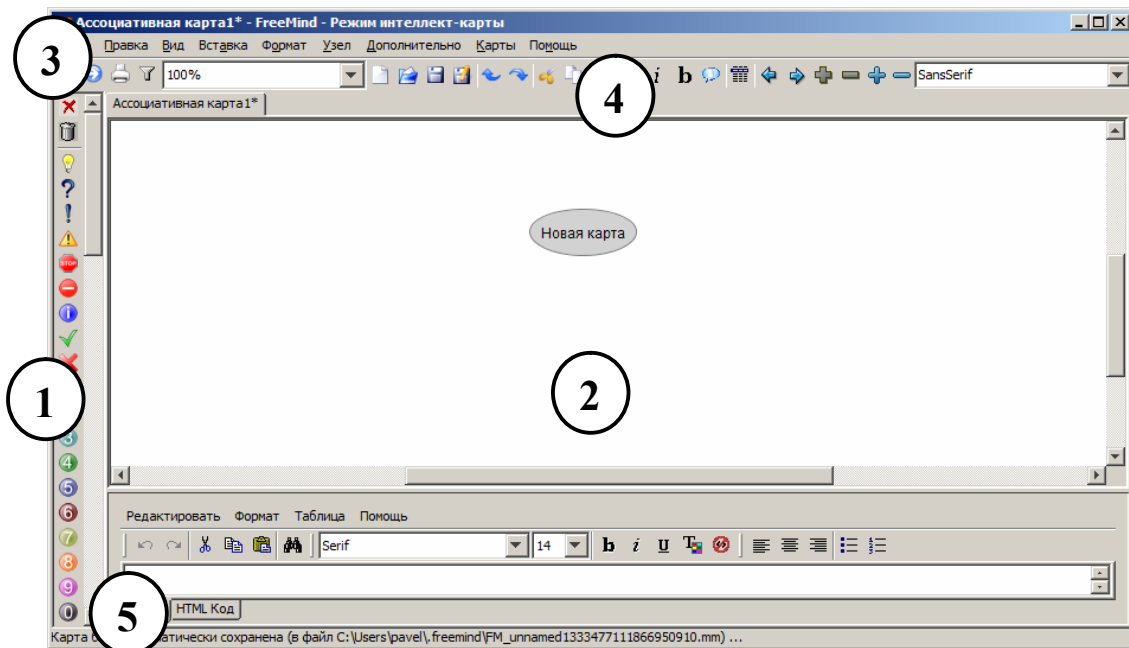


Рис. 9.13 Головне вікно програми FreeMind

Опис головного вікна: 1- набір піктограм для позначення вузлів карти; 2 - головна робоча область; 3 - меню програми; 4 - функції швидкого доступу до пунктів меню; 5 - вікно виводу програми.

Приклад побудови карти знань по темі «Написання дипломного проекту»

1. Після відкриття програми відображається робоча область, яка містить всього лише один вузол. Перейменуємо цей вузол в «написання дипломного проекту» і збережемо під ім'ям «Диплом_карта_знань».

2. Виділимо основні етапи написання диплома:

- формування теми;
- аналіз літератури на наявність існуючих аналогів;
- формування власного методу (створення алгоритму, написання проекту);
- реалізація методу;
- порівняння отриманих результатів з існуючими методами;
- написання пояснювальної записки.

Виділяючи центральний вузол і натискаючи клавішу [Insert], додамо 6 підлеглих вузлів і назвемо їх відповідно основним етапам (рис. 9.14).

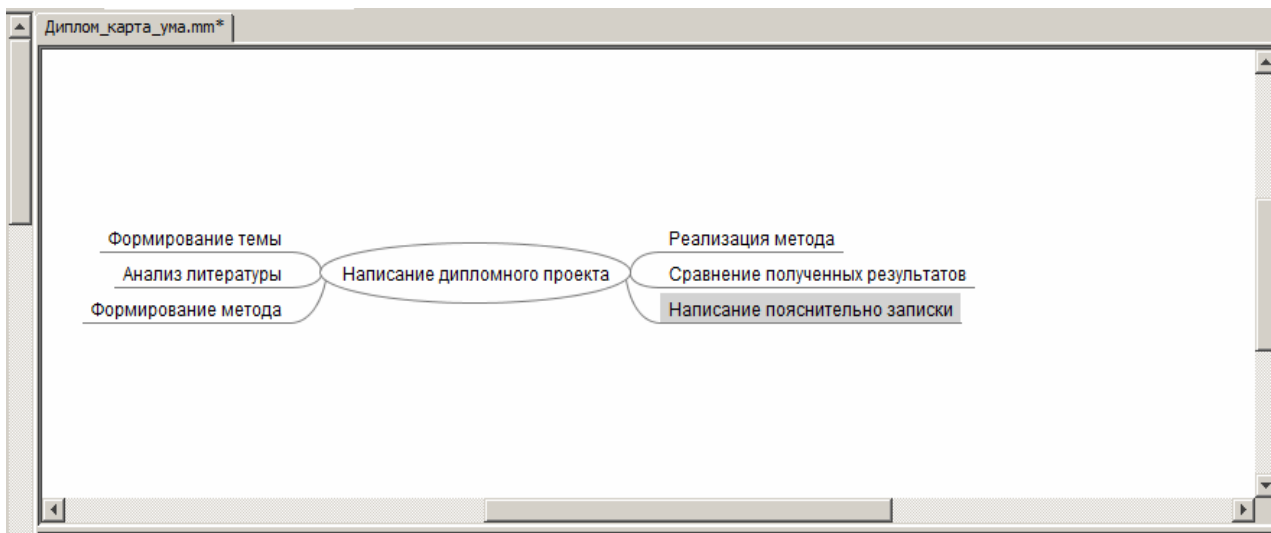


Рис. 9.14 Вказання основних етапів написання диплому

3. Після чого виконують дану процедуру для кожного дочірнього вузла, додавши для кожного з них свої підвузли (рис. 9.15).



Рис. 9.15 Приклад завдання вузлів

4. Оформимо отриману карту знань. Для цього скористаємося пунктом меню Формат. Виділяючи необхідні вузли із затиснутою клавішею [ctrl].

Змінимо типи вузлів на овал, і кольору фону вузлів. Так само додамо іконки в ті пункти, на які варто особливо звернути увагу. Для цього виділимо потрібний вузол і клікнемо на необхідній іконці з лівого меню.

ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ

1. У зв'язку з чим з'явився новий клас обчислювальних систем – системи підтримки прийняття рішень (СППР)?
2. Як можна реалізувати генерацію можливих рішень в системі підтримки прийняття рішень (СППР)?
3. Які методи узгодження рішень спираються на комп'ютерні процедури і є людино-машинними?
4. Для виявлення структури яких причинних зв'язків призначена когнітивна карта?
5. Хто будує когнітивну карту – особа, що приймає рішення, чи група експертів?
6. Елементи яких двох типів містить когнітивна карта?
7. При від'ємному значенні відношення причинності збільшення значення однієї із змінних, пов'язаних цим відношенням, викликає зменшення чи збільшення значення іншої?
8. Визначте, яким є (додатне, від'ємне або нульове) причинне відношення: число заводів → споживана енергія.
9. Визначте, яким є (додатне, від'ємне або нульове) причинне відношення: ціна на енергію → кількість використаної енергії.
10. Наведіть приклад знакового графа.
11. Чи може *додатний зворотний зв'язок* значно посилити початкову змінну?
12. Якщо у знаковому графі є кілька додатних циклів, це може привести до нестабільності системи?
13. Як визначити в графі ситуацію, коли незначна початкова зміна може привести до непередбаченого значного стрибка?
14. Як визначити від'ємний або додатний зворотний зв'язок в системі на основі аналізу когнітивної карти?
15. Підкресліть, з чим треба визначитися для побудови когнітивної карти:
 - зі списком концептів (вершин знакового графа),
 - зі списком відношень причинності (дуг графа),
 - зі списком значень відношень причинності кожної дуги.
16. В чому полягає перевага методу побудови когнітивної карти перед іншими методами?
17. Які підсистеми є для експертної системи обов'язковими?
18. Хто створює базу знань експертної системи?
19. Сформулюйте правило розподілу ресурсів.
20. Наведіть структуру системи підтримки переговорів та охарактеризуйте основні її складові.
21. Які існують методи оцінки варіантів рішень СППР?
22. Як здійснюється формування моделі системи нечіткого виводу?
23. Наведіть програмні продукти підтримки прийняття рішень.

ЗАВДАННЯ ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ

1. Визначити основні фактори, що впливають на вирішення проблеми і їх причинно-наслідкові зв'язки.

2. Побудувати когнітивну карту (КК).

Для побудови КК необхідно визначити:

- 1) список концептів (вершин знакового графа),
- 2) список відношень причинності (дуг графа),
- 3) список значень відношень причинності кожної дуги.

3. Виконати аналіз когнітивної карти.

1) Визначити, чи утворюються додатні і від'ємні зворотні зв'язки.

Додатний або від'ємний зворотний зв'язок в системі може бути легко визначений: в циклі існує додатний зворотний зв'язок тоді і тільки тоді, якщо в ній парне число знаків мінус (або коли знаків мінус немає).

Загальне правило поведінки шляхів і циклів наступне:

– зворотній зв'язок у циклі додатний, якщо число дуг зі знаком мінус парне (або їх немає), і від'ємний, якщо число дуг зі знаком мінус – непарне;

– загальний зворотний зв'язок вершини А на вершину В (можливо на ту ж вершину А) є сума зворотних зв'язків в циклах на всьому шляху від А до В (можливо від А до А).

Якщо зворотні зв'язки у всіх циклах додатні, то сума зворотних зв'язків додатна. Якщо сума зворотних зв'язків у всіх циклах від'ємна, то сума від'ємна. В шляху або циклі з парним числом дуг зі знаком мінус початкова зміна посилюється. В шляху або циклі з непарним числом дуг зі знаком мінус первісним змінам чиниться протидія.

2) Визначити результати (якісні) різних "стратегій", тобто змін, які ми хочемо (або змушені) здійснити в досліджуваній системі (ці зміни можуть відбуватися або відбуваються незалежно від нашої волі, і ми в цьому випадку намагаємось тільки оцінити їх наслідки). Отримати оцінку стабільності системи, оцінити результати можливих змін, визначити стабільні і нестабільні змінні.

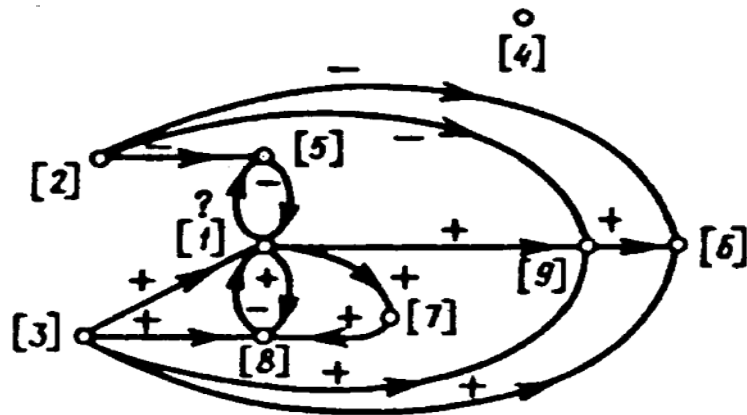
Запропоновані до розгляду проблеми, які потребують підтримки прийняття рішень:

- 1) вивчення внутрішньоміських поїздок на роботу;
- 2) аналіз кадрової політики у вузі;
- 3) вибір типу перевезень;
- 4) очищення прибережної зони;
- 5) розподіл ресурсів на медичні потреби;
- 6) "наука і суспільство";
- 7) студентське харчування.

4. Виконати декомпозицію графа, розбивши його на тісно пов'язані області.

5. За когнітивною картою визначити додатні та від'ємні зворотні зв'язки між концептами. Чи є система, що описана цією когнітивною картою, стійкою?

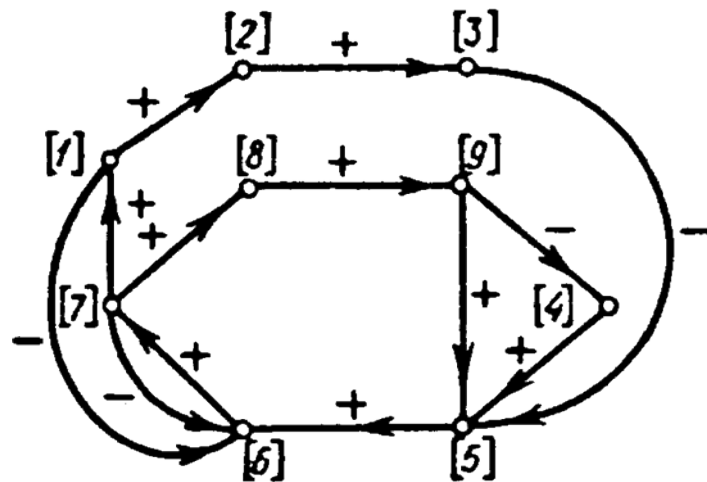
«Вивчення внутрішньоміських поїздок на роботу»



[1] – протяжність поїздки; [2] – економія пального; [3] – чисельність населення; [4] – вартість автомобіля; [5] – вартість проїзного квитка; [6] – забруднення атмосфери; [7] – число нещасних випадків; [8] – ймовірність запізнення; [9] – витрата пального. Знак дуги [1] – [5] не визначено.

6. За когнітивною картою побудувати контекстно-залежну формальну граматику

«Наука і суспільство»

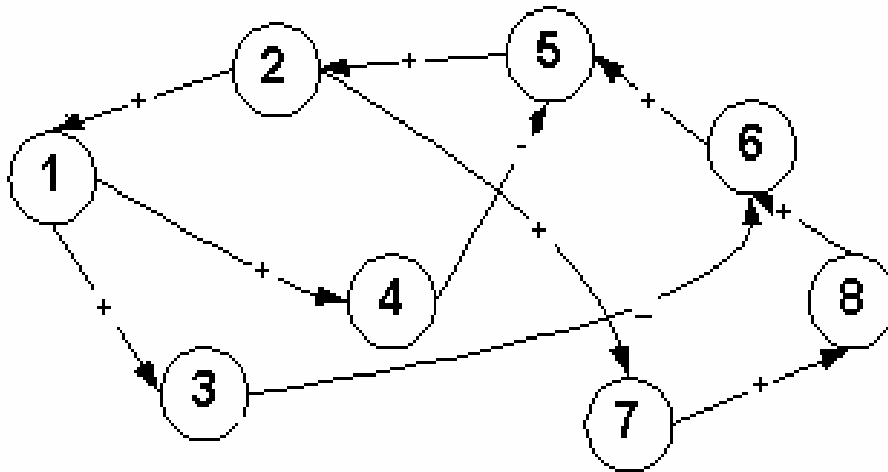


[1] – число робочих місць для науковців; [2] – число слабо підготовлених дослідників; [3] – частка "поганої" наукової продукції або шкідливі наслідки використання результатів науково-технічних досліджень; [4] – зовнішні та внутрішні загрози суспільству, для подолання яких потрібне застосування досягнень науки і техніки; [5] – громадська думка на користь розвитку наукових досліджень; [6] – бюджетні обмеження; [7] – державний бюджет наукових досліджень; [8] – число добре підготовлених дослідників; [9] – частка

"добротної" наукової продукції або позитивні наслідки використання досягнень науки і техніки.

7. За когнітивною картою побудувати контекстно-вільну формальну граматику

«Економічна ситуація фірми»



[1] – кількість фірм-виробників; [2] – рівень прибутку; [3] – обсяг виробленої продукції; [4] – рівень конкуренції; [5] – ціна на продукцію; [6] – дефіцит продукції; [7] – рівень доходу споживачів; [8] – рівень попиту.

8. Дана граMATИКА $G = (V_T, V_N, P, C)$, в якій

$V_T = \{a, b, c, d, e\}$, $V_N = \{A, B, C, D, E\}$,

$P = \{A \rightarrow ed, B \rightarrow Ab, C \rightarrow dD, D \rightarrow Ae, E \rightarrow bc\}$.

Визначити, чи виводиться в цій граMATИЦІ ланцюг **eadabcb**.

9. Є чотири споживача ресурсу. Попарна домовленість між ними задана наступними співвідношеннями за частками отриманого ресурсу:

$C_1=6C_2, C_1=7C_3, C_1=5C_4, C_2=6C_3, C_2=4C_4, C_3=5C_4$.

Знайти пріоритети (або ваги) споживачів, які стали б базою для подальшого узгодження рішень.

10. Провести аналіз альтернатив за допомогою СППР «Мислитель». Створити власну задачу. Зберегти її як шаблон. Створити три альтернативи, серед яких буде вибиратися найкраща. Створити 5 критеріїв. Заповнити матрицю порівняння, вказуючи важливості критеріїв і даючи оцінку альтернативам за кожним критерієм. Описати власне завдання. Пояснити, чому саме це завдання вибрано. Привести в прикладі адекватні альтернативи, із зазначенням стану за кожним критерієм. Описати критерій, на підставі якого і виставити важливість критеріїв.

11. Провести за допомогою програми «Вибір» аналіз альтернатив і вибрати найкращу. Вирішити завдання вибору ноутбука (з 3-х ноутбуків за 5 параметрами, які відіграють вирішальну роль при покупці).

Варіант	Альтернативи	Розмір екрану, дюйм	Частота процесора, ГГц	Розмір оперативної пам'яті, Гб	Розмір жорсткого диску, Гб	Вага
1	1 ноутбук	15,4	2,4	2	320	3,5
	2 ноутбук	17	2,2	2	250	4,5
	3 ноутбук	15,4	2,0	4	500	3,5
2	1 ноутбук	16	2,0	2	250	3,8
	2 ноутбук	15,4	2,0	4	320	3,6
	3 ноутбук	15,6	2,2	4	320	3,0
3	1 ноутбук	15,4	2,4	2	500	3,2
	2 ноутбук	17	2,6	3	250	4,0
	3 ноутбук	16	2,8	4	250	3,5
4	1 ноутбук	15,4	2,0	4	320	2,8
	2 ноутбук	15,4	2,6	4	500	3,2
	3 ноутбук	15,6	2,6	2	160	3,0
5	1 ноутбук	17	2,0	2	320	4,0
	2 ноутбук	17	2,2	4	320	3,8
	3 ноутбук	15,4	2,6	8	500	2,9
6	1 ноутбук	16	2,8	2	160	3,5
	2 ноутбук	18	2,8	4	320	4,0
	3 ноутбук	14,1	2,6	4	320	2,5
7	1 ноутбук	15,4	2,2	2	250	3,0
	2 ноутбук	15,6	2,6	3	250	3,2
	3 ноутбук	18	2,0	4	320	4,0
8	1 ноутбук	14,1	2,2	2	320	2,5
	2 ноутбук	15,6	2,8	8	500	3,5
	3 ноутбук	15,6	2,8	8	500	3,8
9	1 ноутбук	15,4	2,0	2	160	3,4
	2 ноутбук	15,4	2,0	4	320	3,6
	3 ноутбук	18	2,2	4	320	4,2
10	1 ноутбук	14,1	2,4	2	160	2,7
	2 ноутбук	14,1	2,4	4	320	2,9
	3 ноутбук	17	2,4	4	320	3,9

Завдання виконується за номером комп'ютера. Слід зазначити, що при виборі ноутбука, який вибираємо для навчання, слід враховувати середню величину екрану. Наприклад, будемо вважати, що найкращим вибором є ноутбук, з розміром екрану рівним 15,4 дюймів. Отже, необхідно враховувати, що якщо екран більше, або менше, то це менш бажаний варіант. Всі інші критерії, максимальні, чим більше, тим краще. Тільки вага, звичайно, повинна бути мінімальна.

Необхідно визначити оптимальний ноутбук.

При виконанні завдання необхідно описувати дії і логіку студента, чому цей критерій має більшу вагу, аргументувати ступінь переваги.

12. За допомогою програми **Right Solution** виконати процедуру прийняття рішення. Завдання для прийняття рішення вказані в таблиці.

Варіант	Рішення
1	Купівля персонального комп'ютера
2	Купівля автомобіля
3	Внесок грошей у банк
4	Інвестування в проект
5	Купівля плеєра
6	Поїздка за кордон на відпочинок
7	Покупка квартири
8	Придбання домашньої тварини
9	Проведення ремонту
10	Запис на курси водіння
11	Купівля мікрохвильовій печі
12	Поїздка на навчання за кордон
13	Участь у випускному балі НГУ
14	Заняття спортом
15	Прочитання книги

13. За допомогою **СППР FreeMind** виконати процедуру побудови діаграми зв'язку. В якості розв'язуваної задачі для побудови карти знань взяти тему свого дипломного проекту. Вузли оформити різними стилями.

ВИСНОВКИ

1. Очікувані навчальні результати

У результаті засвоєння матеріалу навчального посібника студенти повинні отримати наступні уміння та навички:

- здійснювати обробку, аналіз, систематизацію науково-технічної інформації, узагальнювати передовий вітчизняний та зарубіжний досвід з питань розробки інтелектуальних систем підтримки прийняття рішень, використовуючи сучасні методи наукових досліджень;

- розробляти концептуальні моделі системи, використовуючи процедури формалізованого уявлення про систему;

- знаходити найкращі рішення за допомогою методів прийняття рішень.

- здійснювати систематизацію даних про об'єкт автоматизації і види його діяльності в умовах обстеження об'єкта управління за допомогою експертів, анкет, документів господарської діяльності об'єкта, використовуючи методи усного опитування, анкетування, експертних оцінок, системного спостереження, вивчення документів;

- розробляти альтернативні варіанти концепції систем підтримки прийняття рішень з оцінкою необхідних ресурсів на їх реалізацію в процесі передпроектного обстеження, використовуючи методики функціонально-вартісного аналізу, розрахунку ефективності упровадження систем тощо;

- прогнозувати наслідки впровадження до експлуатації інформаційних технологій, систем підтримки прийняття рішень, автоматизованих робочих місць тощо в умовах дослідження об'єкта управління, побудови інформаційної, функціональної, математичної моделі за допомогою програмного і технічного забезпечення, використовуючи методи довгострокового та короткострокового прогнозування;

- розробляти та використовувати моделі прийняття рішень в умовах невизначеності за допомогою сучасних програмних середовищ для математичного моделювання (FUZZY MATLAB та ін.);

- вибирати формальний апарат для представлення знань в умовах розробки експертних систем за допомогою аналізу формальних граматики, використовуючи мови представлення знань;

- розробляти механізм логічних висновків, що відповідає формальному апарату представлення знань.

2. Оцінювання успішності студентів

Для перевірки повноти та рівня засвоєння студентами знань, умінь та навичок з навчальної дисципліни рекомендованим методом контролю є письмова робота. Вона призначена для з'ясування у письмовій формі ступеня оволодіння студентами знаннями та визначення їх якості – правильності, точності, усвідомленості, вміння застосувати на практиці.

Проміжна письмова перевірка здійснюється у формі модульної контрольної роботи, що складається з теоретичної частини та практичних завдань. Теоретичні питання повинні виявляти знання фактичного матеріалу. Практичні завдання містять розрахункові задачі, приклади вирішення яких наведені у навчальному посібнику. Практична перевірка передбачає контроль оволодіння студентами системою професійних умінь та навичок. Вона дає змогу виявити, якою мірою студент усвідомив теоретичні основи цих дій.

За результатами засвоєння матеріалу навчального посібника студенти складають іспит за екзаменаційними білетами, затвердженими кафедрою.

Критеріями оцінювання успішності студентів з огляду на досягнення запланованих навчальних результатів є наступні:

- обсяг та характер засвоєння знань про об'єкт вивчення, а саме: повнота, правильність, логічність, рівень усвідомленість, міцність запам'ятовування, точність, вміння застосовувати знання тощо;
- якість виявленого студентом знання, логіка мислення, аргументація, послідовність і самостійність викладу;
- вміння студента порівнювати, абстрагувати, класифікувати, узагальнювати інформацію з дисципліни, навички аналізувати, планувати, організовувати, контролювати процес і результати виконання завдання тощо.

3. Значення матеріалу навчального посібника для формування компетенцій фахівця

Матеріал навчального посібника спрямований на формування наступних навчальних та професійних компетенцій:

- визначення цілей проектування об'єктів професійної діяльності, критеріїв ефективності проектних рішень, обмежень;
- системний аналіз об'єкта проектування і предметної області, їх взаємозв'язків;
- розробка вимог і специфікацій об'єктів професійної діяльності на основі аналізу запитів користувачів, моделей предметної області і можливостей технічних засобів;
- розробка складових систем підтримки прийняття рішень у наукових дослідженнях, проектно-конструкторській діяльності, керуванні технологічними, економічними, соціальними системами й у гуманітарних областях діяльності людини;
- аналіз, теоретичне й експериментальне дослідження систем підтримки прийняття рішень;
- розробка й удосконалювання формальних моделей, що застосовані при створенні об'єктів професійної діяльності;
- знаходження компромісу між різними вимогами і пошук прийнятних рішень;
- супровід систем підтримки прийняття рішень та ін.

БІБЛІОГРАФІЧНИЙ СПИСОК

1. Борисов А.Н. Обработка нечеткой информации в системах принятия решений / А.Н. Борисов, А.В. Алексеев, Г.В. Меркурьева, Н.Н. Слядзь, В.И. Глушков // М.: Радио и связь, 1989. – 304 с.
2. Венцель Е.С. Исследование операций. Задачи, принципы, методология / Е.С. Венцель // М.: Наука, 1988. – 208 с.
3. Дьяконов В. MATLAB 6: учебный курс / В. Дьяконов // СПб.: Питер, 2001. – 592 с.
4. Зайченко Ю.П. Исследование операций. Нечеткая оптимизация / Ю.П. Зайченко // Киев: Вища школа, 1991. – 191 с.
5. Кини Р.Л. Принятие решений при многих критериях: предпочтения и замещения / Р.Л. Кини, Х. Райфа // М.: Радио и связь, 1984. – 560 с.
6. Кофман А. Введение в теорию нечетких множеств // М.: Радио и связь, 1982. – 432с.
7. Ларичев О.И. Качественные методы принятия решений / О.И. Ларичев, Е.М. Мошкович // М.: Наука. – Физматлит, 1996. – 208 с.
8. Мелихов А.Н. Ситуационные советующие системы с нечеткой логикой / А.Н. Мелихов, Л.С. Бернштейн, С.Л. Коровин // М.: Наука, 1990. – 272 с.
9. Миркин Б.Г. Проблема группового выбора / Б.Г. Миркин // М.: Наука, 1974. – 256 с.
10. Нечеткие множества в моделях управления и искусственного интеллекта / Под ред. Д.А. Поспелова - М.: Наука, 1986. – 311с.
11. Нечеткие множества и теория возможностей. Последние достижения / Под. ред. Р. Ягера - М.: Радио и связь, 1986. – 391с.
12. Орловский С.А. Проблемы принятия решений при нечеткой исходной информации / С.А. Орловский // М.: Наука, 1981. – 206 с.
13. Прикладные нечеткие системы / Под ред. Т. Тэрano – М: Мир, 1993. – 512 с.
14. Робертс Ф.С. Дискретные модели с приложениями к социальным биологическим и экологическим задачам / Ф.С. Робертс // М.: Наука, 1986. – 496 с.
15. Саати Т. Принятие решений. Метод анализа анархий / Т. Саати // М.: Радио и связь, 1993. – 320 с.
16. Силов В.Б. Принятие стратегических решений в нечеткой обстановке / В.Б. Силов // М. ИНПРО-РЕС, 1995. – 228 с.
17. Таха Х. Введение в исследование операций: В 2-х книгах. Кн.1 / Х. Таха // М.: Мир, 1985. – 479 с.
18. Таха Х. Введение в исследование операций: В 2-х книгах. Кн.2 / Х. Таха // М.: Мир, 1985. – 496 с.

19. Трахтенгерц Э.А. Современные компьютерные технологии управления информационно-аналитической деятельностью / Э.А. Трахтенгерц // М.: СИНТЕГ, 2007. – 372 с.

20. Трахтенгерц Э.А. Компьютерная поддержка принятия решений: Научно-практическое издание. Серия “Информатизация России на пороге XXI века” / Э.А. Трахтенгерц // М.: СИНТЕГ, 1998. – 376 с.

21. Трахтенгерц Э.А. Компьютерная поддержка переговоров при согласовании управленческих решений / Э.А. Трахтенгерц // М., 2003. – 181 с.

22. Штовба С.Д. Проектирование нечетких систем средствами MATLAB / С.Д. Штовба // М.: Горячая линия – Телеком, 2007. – 288 с.

Навчальне видання

Саричева Людмила Віссаріонівна
Сергєєва Катерина Леонідівна

КОМП'ЮТЕРНА ПІДТРИМКА ПРИЙНЯТТЯ РІШЕНЬ

Навчальний посібник

Підписано до друку 31.10.2016. Формат 30 × 42/4.
Папір офсетний. Ризографія. Ум. друк. арк. 5,5.
Обл.-вид. арк. 5,5. Тираж 30 пр. Зам. №

Підготовлено до друку та видруковано
у Державному вищому навчальному закладі
«Національний гірничий університет».
Свідоцтво про внесення до Державного реєстру ДК № 1842 від 11.06.2004

49005, м. Дніпро, просп. Д. Яворницького, 19