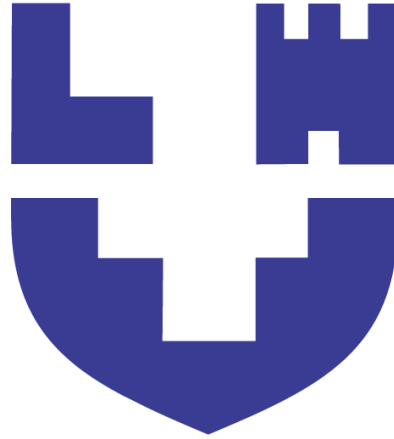


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



СИСТЕМНИЙ АНАЛІЗ ТА ТЕОРІЯ ПРИЙНЯТТЯ РІШЕНЬ

Методичні вказівки до лабораторних робіт
для здобувачів першого (бакалаврського) рівня
освітньо-професійної програми «Комп'ютерні науки»
галузі знань F(12) Інформаційні технології
спеціальності F3 (122) Комп'ютерні науки
денної та заочної форм навчання

УДК 004.41(07)
С 45

До друку

Голова вченої ради факультету комп'ютерних та інформаційних технологій _____

І. Кондіус

Електронна копія друкованого видання передана для внесення в репозитарій ЛНТУ

Директор бібліотеки _____ Н. Поліщук

Затверджено вченою радою факультету комп'ютерних та інформаційних технологій ЛНТУ,
протокол № ___ від «___» _____ 2026 року.

Розглянуто і схвалено на засіданні кафедри комп'ютерних наук ЛНТУ, протокол № ___ від
«___» _____ 2026 року

Завідувач кафедри КН _____ В. Ліщина

Укладачі: _____ В. Сачук, асистент кафедри комп'ютерних наук ЛНТУ.

_____ К. Вавринюк, асистент кафедри комп'ютерних наук ЛНТУ

Рецензент: _____ Н. Ліщина, кандидат технічних наук, доцент, завідувач кафедри
інженерії програмного забезпечення ЛНТУ.

Відповідальний за випуск: _____ В. Ліщина, к.т.н., доцент, завідувач кафедри
комп'ютерних наук ЛНТУ.

Системний аналіз та теорія прийняття рішень: методичні вказівки до лабораторних робіт для здобувачів першого (бакалаврського) рівня освітньо-професійної програми «Комп'ютерні науки» галузі знань F(12) Інформаційні технології спеціальності F3(122) Комп'ютерні науки денної та заочної форм навчання / уклад. В. Сачук, К. Вавринюк. Луцьк: ЛНТУ. 2026. 77 с.

ЗМІСТ

ВСТУП.....	4
ЛАБОРАТОРНА РОБОТА № 1.....	6
ЛАБОРАТОРНА РОБОТА №2.....	16
ЛАБОРАТОРНА РОБОТА №3-4	19
ЛАБОРАТОРНА РОБОТА №5-7	27
ЛАБОРАТОРНА РОБОТА №8.....	37
ЛАБОРАТОРНА РОБОТА №9.....	43
ЛАБОРАТОРНА РОБОТА № 10.....	48
ЛАБОРАТОРНА РОБОТА № 11.....	54
ЛАБОРАТОРНА РОБОТА №12.....	60
ЛАБОРАТОРНА РОБОТА №13.....	64
ЛАБОРАТОРНА РОБОТА №14.....	68
ЛАБОРАТОРНА РОБОТА №15.....	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	75

ВСТУП

Сучасний розвиток інформаційних технологій, ускладнення організаційних та виробничих систем, а також зростання обсягів даних, якими оперують підприємства й установи, зумовлюють нагальну потребу у фахівцях, здатних системно мислити, аналізувати складні проблеми та приймати обґрунтовані рішення. Саме цим визначається актуальність дисципліни «Системний аналіз та теорія прийняття рішень» у підготовці бакалаврів зі спеціальності F3(122) «Комп'ютерні науки».

Дисципліна охоплює широке коло питань: від базових понять теорії систем і методологічних засад системного підходу до математичних методів оптимізації та сучасних підходів до підтримки прийняття рішень в умовах визначеності, невизначеності й ризику. Вона формує у студентів уміння формалізовано описувати задачі, будувати математичні моделі реальних процесів і застосовувати ефективні методи їх розв'язання.

Методичні вказівки до лабораторних робіт підготовлені відповідно до робочої програми навчальної дисципліни, затвердженої кафедрою комп'ютерних наук Луцького національного технічного університету. Посібник охоплює п'ятнадцять лабораторних робіт, що відповідають трьом змістовим модулям: «Основи системного аналізу», «Математичні методи оптимізації» та «Теорія прийняття рішень».

У першому змістовому модулі студенти ознайомлюються з класифікацією та властивостями інформаційних систем, опановують технологію побудови інтелектуальних карт як інструменту структурування знань, а також набувають практичних навичок моделювання організаційних структур і бізнес-процесів у середовищі уEd Graph Editor із застосуванням методології ARIS. Вміння графічно відтворювати функціональні та процесні моделі є базою для подальшого проектування та аналізу складних інформаційних систем.

Другий змістовий модуль присвячений математичним методам оптимізації. Тут розглядаються задачі лінійного програмування, їх розв'язання симплекс-методом та методом штучного базису, транспортна задача і метод потенціалів, методи нелінійного програмування (зокрема метод множників Лагранжа), а також цілочисельне програмування з методом гілок і меж. Практичне засвоєння цих методів дозволяє розв'язувати реальні задачі оптимального управління ресурсами, плануванні виробництва та розподілі навантажень.

Третій змістовий модуль зосереджений на теорії прийняття рішень. Студенти вивчають експертні методи, багатокритеріальний аналіз альтернатив, метод аналізу ієрархій Т. Сааті, а також підходи до прийняття рішень в умовах невизначеності з використанням класичних критеріїв – Лапласа, Вальда, Севіджа та Гурвіца. Ці знання є надзвичайно важливими у практичній діяльності фахівців з комп'ютерних наук, оскільки реальні задачі управління й

проектування систем майже завжди виникають за умов неповної інформації та конкуруючих цілей.

Кожна лабораторна робота містить: мету роботи, стислі теоретичні відомості з посиланнями на основну літературу, завдання для самостійного виконання з варіантами, порядок виконання та контрольні запитання для перевірки засвоєного матеріалу. Такий структурований підхід сприяє послідовному та свідомому оволодінню навчальним матеріалом і формує вміння самостійно планувати та організовувати навчально-дослідницьку діяльність.

Для успішного виконання лабораторних робіт студенти повинні мати базові знання з дисциплін «Програмування», «Вища математика», «Теорія ймовірностей та математична статистика», «Чисельні методи». Водночас методичні вказівки побудовані таким чином, щоб кожна лабораторна робота була доступна для виконання навіть за умови самостійного опрацювання теоретичного матеріалу за рекомендованою літературою.

Методичні вказівки призначені для студентів денної та заочної форм навчання першого (бакалаврського) рівня вищої освіти зі спеціальності F3(122) «Комп'ютерні науки».

ЛАБОРАТОРНА РОБОТА № 1

ОПИС ТА МОДЕЛЮВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ

Мета лабораторної роботи: ознайомитися з класифікацією та властивостями інформаційних систем, засвоїти методи їх опису і моделювання.

Теоретичні відомості

Існує цілий ряд різних класифікацій систем, що відбивають ті чи інші їх властивості. Така різноманітність зумовлена розбіжністю цілей класифікації й багатогранністю властивостей систем.

Віднесення об'єкта до того чи іншого класу часто викликає проблеми, що можуть бути пов'язані з розмитістю меж класів, недостатньою вивченістю системи й іншими причинами. Одна з основних проблем класифікації – це проблема повноти. Тому часто вводять такі класи, як «змішаний», «усе інше» і т.п. У таблиці 1.1 наведено класифікацію систем за певними ознаками. Додамо, що перелік ознак може бути доповнений.

Таблиця 1.1 – Класифікація систем за певними ознаками

№	Класифікаційна ознака	Тип системи
1.	Природа елементів	Реальні (фізичні) / Абстрактні
2.	Походження	Природні / Штучні / Змішані
3.	Тип змінних	З якісними змінними З кількісними змінними Зі змішаним описом змінних
4.	Тип оператора системи	Чорний ящик Непараметричний клас Параметричний клас Білий ящик
5.	Термін існування	Сталі / Тимчасові
6.	Мінливість властивостей	Статичні / Динамічні
7.	Ступінь складності	Прості / Складні / Великі
8.	Відношення до зовнішнього середовища	Закриті / Відкриті
9.	Реакція на вплив, що збурює	Активні / Пасивні
10.	Характер управління	Керовані ззовні Самокеровані З комбінованим управлінням Без управління
11.	Ступінь організованості	Добре організовані Погано організовані (дифузні) Ті, що самоорганізуються
12.	Ступінь участі та впливу людини	Технічні Людино-машинні Організаційні

Розрізняють реальні та абстрактні системи. До реальних належать ті, що реально існують у природі, техніці або суспільстві. Наприклад, реальними є Сонячна система, Україна як держава. Прикладами абстрактних систем можуть бути ідеальні відображення реальних

систем та процесів (карти місцевості, технічні креслення тощо), а також інші ідеальні конструкції (системи рівнянь, алгоритми та інші) [1].

Для реальної системи може бути побудована певна множина моделей, що розрізняються за метою дослідження, необхідним ступенем деталізації та іншими ознаками.

Наприклад, реальна локальна обчислювальна система, з погляду системного адміністратора, є сукупністю програмного, математичного, інформаційного, лінгвістичного, технічного й іншого видів забезпечення, з погляду користувача, – це сукупність об’єктів, з якими можна обмінюватися інформацією, з погляду технічного обслуговування – сукупність справних та несправних засобів.

Природні та штучні об’єкти, що відображаються у свідомості людини, постають як абстракції та поняття, сформовані внаслідок мисленнєвого виокремлення й узагальнення певних їхніх властивостей або сторін. Такі утворення не мають безпосередніх аналогів у реальному світі. Абстрактні проекти створюваної системи втілюються в реально існуючу, яку можна відчувати, а при її вивченні знову відбити у формі абстрактної.

Розглядаючи системи за класифікаційною ознакою походження, можемо побачити, що у кожному типі систем можна виділити певні класи (рис. 1.1).

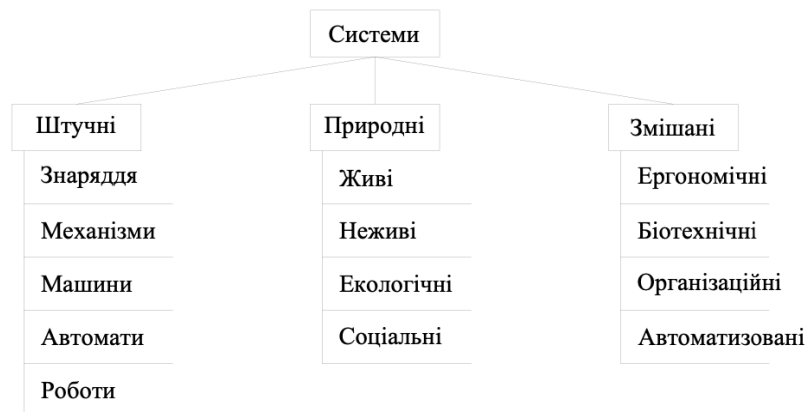


Рисунок 1.1 – Класифікація систем за класами

На рисунку 1.2 наведено схему класифікації за типом змінних, що описують вхід і вихід системи.

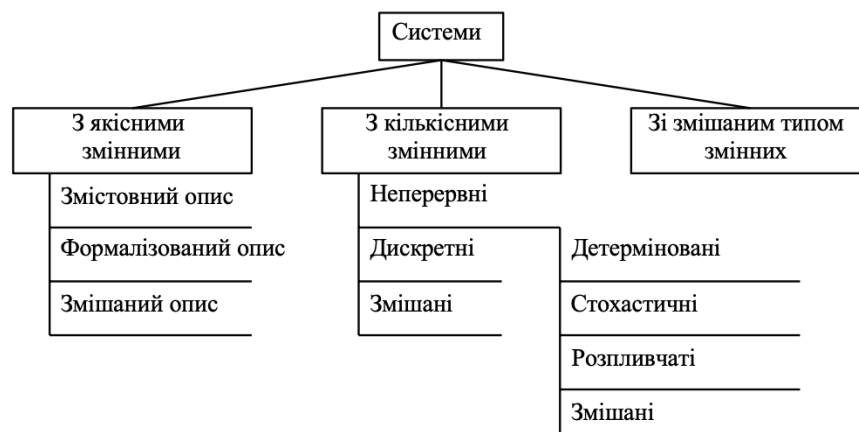


Рисунок 1.2 – Класифікація систем за типом змінних

Якісні й кількісні змінні потребують принципово різних підходів до їх обробки й аналізу. Клас систем зі змішаним описом змінних на першому рівні утворюють системи, для яких частина змінних має кількісний опис, а частина – якісний. На другому рівні до підкласу змішаних систем включено системи з якісними змінними, частина з яких описується змістовно, а частина – формалізовано. До аналогічного підкласу систем з кількісними змінними віднесено системи, у яких частина змінних є дискретними, а частина – неперервними. На третьому рівні до підкласу змішаних систем включено системи, у яких змінні принаймні двох з виділених типів (детерміновані, стохастичні чи розпливчасті) [1].

У наведеній класифікації другий рівень для систем з якісно-кількісним описом змінних є об'єднанням відповідних рівнів перших двох класів систем. Третій рівень є однаковим для всіх підкласів систем з якісним описом змінних.

Класифікація за типом оператора (рис. 1.3) поділяє системи за характером зв'язку між вхідними і вихідними змінними. На першому рівні характеристичною ознакою є ступінь повноти інформації про цей зв'язок.

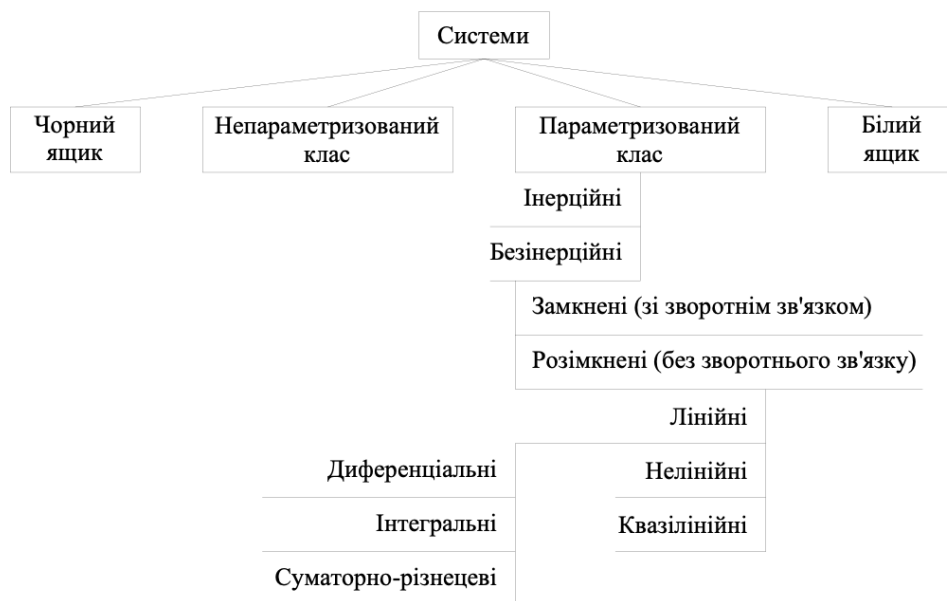


Рисунок 1.3 – Класифікація систем за типом оператора

Перший клас утворюють системи типу чорного ящика. Для таких систем немає ніякої інформації про характер зв'язку між вхідними і вихідними змінними. Для не параметризованих систем, що становлять другий клас, є уривчаста інформація про характер цього зв'язку.

Зокрема, може бути відомо, що оператор зв'язку має такі властивості, як гладкість, симетричність, неперервність, монотонність і т. п. До третього класу належать параметризовані системи. Для них оператор зв'язку відомий з точністю до значень параметрів, які до нього входять. Наприклад, може бути встановлено, що зв'язок між входом і виходом описується рівнянням виду $Y=AX + B$. Але при цьому є невідомими значення параметрів А і В. Нарешті, четвертий клас утворюють системи типу білого ящика, тобто такі, для яких зв'язок між входом і виходом є повністю описаним.

За терміном існування системи поділяють на сталі й тимчасові. До сталих зазвичай відносять ті, що зберігають свою цілісність протягом часу, який цікавить дослідника. Це

велика кількість природних (Всесвіт, Земля, стійкі елементарні частинки й атоми тощо) та штучних (ювелірні вироби, архітектурні та гідротехнічні споруди тощо) систем, які протягом певного часу функціонування практично не змінюються. Тимчасовими є системи, час існування яких є порівняним або навіть меншим, ніж час, потрібний для їх дослідження [1].

Спостереження мінливості властивостей дає змогу поділити системи на статичні та динамічні. До статичних відносять ті, при дослідженні яких можна нехтувати змінами їх характеристик та суттєвих властивостей у часі. Тобто у процесі дослідження систему можна вважати такою, що має лише один можливий стан.

На відміну від статичних, динамічні системи мають кілька можливих станів. З часом система може переходити з одного стану до іншого безупинно або тільки в деякі дискретні моменти.

Розподіл систем на прості, складні й великі підкреслює, що в системному аналізі розглядаються не будь-які, а саме складні системи великого масштабу. При цьому виділяють структурну й функціональну (обчислювальну) складність.

Загально визнаної межі, яка розділяла б прості, великі й складні системи, немає. Однак умовно будемо вважати, що складні характеризуються трьома основними ознаками: властивістю працеспроможності, наявністю неоднорідних зв'язків і емерджентністю.

Під працеспроможністю розуміють здатність системи зберігати часткову працездатність (ефективність) при відмові її окремих елементів чи підсистем. Вона зумовлюється функціональною надмірністю складної системи й виявляється в зміні ступеня деградації виконуваних функцій, що залежить від глибини збуджуючих дій. Проста система може знаходитися не більше ніж у двох станах: повної працездатності (справному) або повної непрацездатності (несправному).

У складних системах, крім значної кількості елементів, присутні численні й різні за типами (неоднорідні) зв'язки між елементами. Основними є такі види зв'язків: структурні (у тому числі ієрархічні), функціональні, казуальні (причинно-наслідкові, відносини істинності), інформаційні, просторово-часові. За цією ознакою будемо відрізняти складні системи від великих систем, що є сукупністю однорідних елементів, об'єднаних зв'язком одного типу.

Складна система має властивості, відсутні у кожного з її компонентів. Це називають інтегративністю (цілісністю), чи емерджентністю системи.

Одну з найбільш повних і цікавих класифікацій за рівнями складності запропонував К. Боулдінг. Виділені в ній рівні наведено в таблиці 1.2.

Таблиця 1.2 – Класифікація за рівнями складності К. Боулдінг

Тип системи	Рівень складності	Приклади
Неживі системи	Статичні структури або кістяки. Прості динамічні структури із заданим законом поведінки (спрямування). Кібернетичні системи з керованими циклами зворотного зв'язку	Кристали Годинниковий механізм Термостат

Тип системи	Рівень складності	Приклади
Живі системи	<p>Відкриті системи зі структурою, яка само зберігається (перший щабель, на котрому можливий поділ на живе і неживе).</p> <p>Живі організми з низькою здатністю сприймати інформацію.</p> <p>Живі організми з більш розвинутою здатністю сприймати інформацію, але такі, що не мають самосвідомості.</p> <p>Системи, що характеризуються самосвідомістю, мисленням і нетривіальною поведінкою.</p> <p>Соціальні системи</p> <p>Трансцендентні системи або системи, що є на сьогодні поза нашим пізнанням</p>	<p>Клітини, гомеостат</p> <p>Рослини</p> <p>Тварини</p> <p>Люди</p> <p>Соціальні організації</p>

У цій класифікації, як правило, кожен наступний клас містить у собі попередній, характеризується великим проявом властивостей відкритості та стохастичності поведінки, більш яскраво вираженими закономірностями ієрархічності й історичності (аналізованих нижче), хоча це не завжди зазначається, а також більш складними механізмами функціонування й розвитку [2].

При класифікації систем за способом управління (рис. 1.4) на першому рівні визначається місце розташування керуючої системи (поза чи всередині керованої системи). Можливий також випадок комбінованого управління, коли частина керуючої системи знаходиться всередині керованої системи, а інша частина – поза нею.

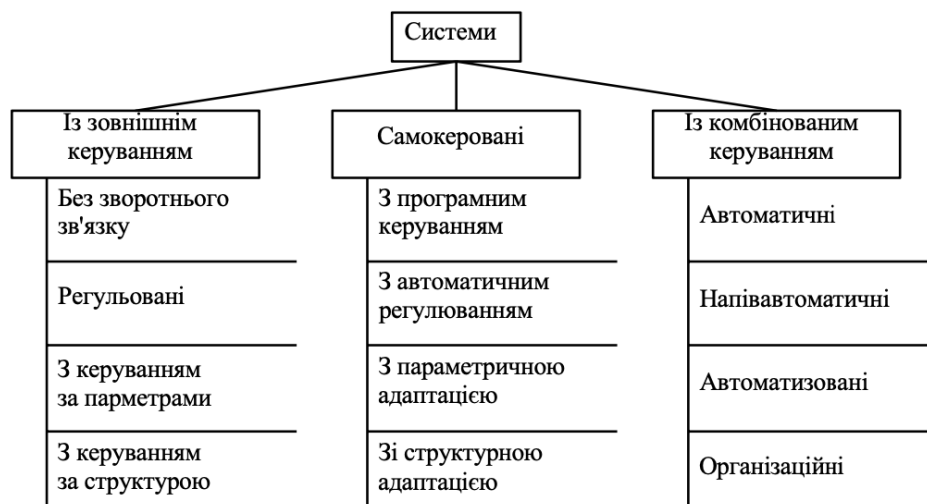


Рисунок 1.4 – Класифікація систем за способом управління

На другому рівні незалежно від місця розташування блоку управління розподіл систем здійснюється за ступенем визначеності траєкторії (у фазовому просторі), що приводить систему до мети, і здатності блоку управління утримувати систему на цій траєкторії. Перший підклас у всіх випадках утворюють системи, у яких точно відомо траєкторію, а також те, як треба впливати на систему для досягнення мети. Прикладами систем, що відносяться до цього підкласу (системи без зворотного зв'язку, самокеровані та автоматичні системи), можуть

служити телефон-автомат, робота ЕОМ за задалегідь заданою програмою, ріст зародка живого організму і т. п.

Набагато частіше зустрічається ситуація, коли система сходить із заданої траєкторії внаслідок того, що процеси на некерованих входах відрізняються від передбачуваних, або під впливом випадкових і неврахованих факторів. Якщо відхилення системи від заданої траєкторії невелике, то застосовується спосіб управління, названий регулюванням. У цьому разі знаходять різницю між поточною та заданою траєкторіями $y(t) - y_0(t)$ і визначають додаткові керівні впливи, здатні в найближчому майбутньому повернути виходи системи на задану траєкторію. Регульовані системи виділені в другий підклас усіх розглянутих класів. До них, зокрема, належать автопілоти літаків, рефлекторні реакції тварин, верстати, керовані операторами тощо.

Наступний спосіб управління застосовують, якщо відхилення від заданої траєкторії занадто великі, і повернути систему на цю траєкторію методами регулювання неможливо, а також у випадках, коли з якихось причин не можна задати опорну траєкторію. Управління в таких ситуаціях може полягати в тому, щоб спрогнозувати поточну траєкторію $y(t)$ і підібрати такі параметри системи, за яких прогнозована траєкторія в певний момент часу потрапить до заданої цільової області Y^* . Такому способу управління відповідає третій підклас. Прикладами його реалізації є процеси адаптації живих організмів до умов середовища, що змінюються, робота пілотів літаків, автоматизовані системи управління тощо [2].

Можлива також ситуація, коли траєкторія даної системи ні за яких значень параметрів не перетинає цільову область. У цьому разі ціль може бути досягнута шляхом зміни структури системи. Відповідний спосіб управління називають структурною адаптацією, а системи, у яких він реалізується, утворюють четвертий підклас. Прикладами можуть служити гнучкі автоматичні виробництва, обчислювальні мережі, сільськогосподарські машини зі змінними начіпними та причіпними пристроями, видоутворення в живій природі, утворення й розпад держав тощо.

Уперше поділ систем за ступенем організованості за аналогією до класифікації проблем Г. Саймона і А. Ньюелла (добре структуровані, погано структуровані та неструктуровані проблеми) запропонував В.В. Налімов, який виділив класи добре організованих (аналізованих іноді в літературі окремо як класи саморегульованих, самонавчальних, самоналагоджувальних і т. п. систем) та погано організованих (або дифузійних) систем. Якщо додати до цих двох класів ще системи, що самоорганізуються, то отримані класи можна достатньо чітко розмежувати за допомогою характерних для кожного з них ознак, які дають можливість поставити у відповідність класу певні методи формалізованого подання систем і засоби подання цілей у них. Виділені класи можна розглядати як підходи до відображення об'єкта або розв'язуваної задачі, що можуть вибиратися залежно від стадії пізнання об'єкта й можливості одержання інформації про нього.

Коротко охарактеризуємо ці класи.

Добре організовані системи. Відобразити аналізований об'єкт або процес ухвалення рішення у вигляді добре організованої системи означає визначити елементи системи та їх зв'язки між собою і з цілями системи. У цьому разі завдання вибору цілей і засобів їх досягнення (елементів, зв'язків) не розділяються. Проблемна ситуація може бути описана у вигляді виразу, що пов'язує ціль із засобами, тобто, у вигляді критерію або показника ефективності, критерію функціонування, цільової функції й т. п., що можуть бути подані складним рівнянням, формулою, системою рівнянь. Часто при цьому говорять, що ціль

рекомендується як критерій ефективності або критерій функціонування, хоча насправді в подібних виразах об'єднані й ціль, і засоби.

Більшість моделей фізики й технічних наук засновані на відображенні об'єктів і процесів як добре організованих систем. Наприклад, роботу складного механізму подають у вигляді спрощеної схеми або системи рівнянь, які враховують не всі, а найбільш істотні з погляду його призначення елементи та зв'язки між ними. Атом часто описують як планетарну систему, яка складається з ядра та електронів. Це суттєво спрощує реальну картину, але є достатнім для розуміння багатьох властивостей атома.

Неважко помітити, що для відображення об'єкта у вигляді добре організованої системи слід виділяти істотні та не зважати на несуттєві для конкретної цілі розгляду компоненти. За необхідності більш детального опису потрібно уточнити ціль, вказавши, з яким ступенем глибини нас цікавить досліджуваний об'єкт, і побудувати нову систему, що відображає його, беручи до уваги внесені корективи. Наприклад, при описі будови атома можна врахувати склад ядра, квантово-механічні закономірності поведінки електронів та ядер, магнітні властивості тощо [2].

Відображення об'єкта у формі добре організованої системи використовують у тому разі, коли можна запропонувати детермінований опис і експериментально обґрунтувати правомірність його вживання, тобто, адекватність моделі реальному об'єкту чи процесу. Спроби застосувати клас добре організованих систем для відображення складних багатокомпонентних об'єктів або для розв'язування багатокритеріальних задач, що виникають, наприклад, при удосконаленні управління й розробці АСУ, вдаються погано. Це не тільки потребує неприпустимо великих витрат часу на одержання й опрацювання моделей, а й часто практично не може бути реалізованим, оскільки не вдається поставити експеримент, що доводить правомірність застосування запропонованих аналітичних залежностей. Тому здебільшого при дослідженні складних об'єктів на початковому етапі постановки задач їх відображають класами, які характеризуються далі окремо.

Погано організовані, або дифузійні, системи. При відображенні об'єкта у вигляді погано організованої, або дифузійної, системи не ставиться завдання визначити всі компоненти, їх властивості та зв'язки між ними й цілями системи. Система характеризується певним набором макропараметрів та зв'язків між ними або закономірностями, що виявляються на основі дослідження визначеної за допомогою деяких правил достатньо представницької вибірки компонентів, що характеризують досліджуваний об'єкт або процес.

У першому випадку поведінка системи може бути детермінованою, тобто, знання певних характеристик системи дає змогу однозначно встановити її інші суттєві властивості. У другому випадку на основі вибіркового дослідження компонентів одержують їх статистичні характеристики або закономірності поведінки і поширюють отримані результати на всю систему в цілому. При цьому робляться відповідні застереження. Наприклад, при одержанні статистичних закономірностей їх поширюють на поведінку всієї системи, вказуючи рівень значущості (ймовірність помилки), довірчі межі тощо.

Прикладом дифузної системи є газ. Його поведінка не може бути визначена шляхом повного й точного опису поведінки всіх його молекул, тому його характеризують або макропараметрами (параметрами стану) – тиском, температурою, об'ємом тощо, або функціями розподілу мікропараметрів – швидкостей молекул, їх кінетичних та потенціальних енергій та інших. Базуючись на цих параметрах, розробляють прилади й устрої, що

використовують властивості газу, не досліджуючи при цьому поведінки кожної окремої молекули.

Відображення об'єктів у вигляді дифузійних систем знаходить широке застосування при визначенні пропускну здатності систем управління, кількості працівників в обслуговуючих, наприклад, ремонтних, цехах підприємств і в обслуговуючих установах (для вирішення подібних задач використовують методи теорії масового обслуговування), при дослідженні документальних потоків інформації тощо.

Системи із самоорганізацією. Відображення об'єктів у вигляді систем, що самоорганізуються, або розвиваються, дає змогу досліджувати найменш вивчені об'єкти та процеси з великою невизначеністю на початковому етапі постановки завдань. Такі системи мають ознаки, характерні для дифузійних систем: стохастичність поведінки, нестабільність окремих параметрів і, крім того, такі специфічні риси, як непередбачуваність поведінки; здатність адаптуватися до умов динамічного середовища; змінювати структуру, зберігаючи при цьому властивість цілісності; протистояти ентропійним тенденціям; формувати можливі варіанти поведінки й вибрати серед них найкращий; а також інші ознаки, що наближають їх до реальних об'єктів.

Сказане означає, що моделі систем, що самоорганізуються, або розвиваються, мають надавати можливість відображення розглянутих властивостей. При їх формуванні змінюється звичне уявлення про моделі, характерне для математичного моделювання та прикладної математики. Стають іншими також уявлення про доведення адекватності моделей.

Основну конструктивну ідею, завдяки якій можливо реалізувати відображення об'єкта класом систем, що самоорганізуються, можна сформулювати в такий спосіб. Розробляють знакову систему, за допомогою якої фіксують відомі на даний момент компоненти та зв'язки. Потім через перетворення отриманого відображення, використовуючи встановлені правила (правила структуризації, або декомпозиції, правила композиції), одержують нові, невідомі раніше взаємовідносини й залежності, що можуть або послужити основою прийнятих рішень, або підказати наступні кроки на шляху підготування рішення [2].

Таким чином, можна накопичувати інформацію про об'єкт, фіксуючи при цьому всі нові компоненти та зв'язки (правила взаємодії компонентів), і, застосовуючи їх, одержувати відображення послідовних станів системи, що розвивається, поступово створюючи все більш адекватну модель реального досліджуваного або проектного об'єкта. При цьому інформація може надходити від спеціалістів різноманітних галузей знань і накопичуватися. Адекватність моделі також доводиться послідовно шляхом оцінювання правильності відображення у знаковій моделі компонентів і зв'язків, необхідних для досягнення поставленої цілі дослідження або створення об'єкта.

Практична реалізація побудови такої моделі пов'язана з необхідністю розробки комп'ютеризованих систем автоматизованого проектування, підтримки прийняття рішень, управління тощо, а також відповідних мов. В основу останніх може бути покладений один із методів моделювання систем, наприклад, теорія множин, математична логіка, математична лінгвістика, імітаційне динамічне моделювання і т. п., але в міру розвитку моделі ці методи можуть змінюватися. При моделюванні найбільш складних процесів, таких як цілеутворення, удосконалення організаційних структур систем управління тощо, «механізм» розвитку (самоорганізації) може бути реалізований у формі відповідної методики системного аналізу.

Іноді аналізований клас систем розбивають на підкласи, виділяючи адаптивні системи, або системи, що самоприспосовуються, самонавчаються, самовідновлюються чи

самовідтворюються, й інші класи, які відповідають різноманітним властивостям системи, що розвивається. При поданні відображуваного об'єкта у вигляді системи, що самоорганізується, завдання визначення цілей і вибору засобів, як правило, відокремлюються. При цьому завдання вибору цілей, у свою чергу, може бути описане як система, що самоорганізується. Наприклад, при побудові автоматизованої системи управління її мета й завдання не є сталими, а змінюються з часом у міру розвитку системи, якою керують, технічних та програмних засобів управління тощо.

Розглянуті класи систем зручно використовувати як підходи на початковому етапі моделювання будь-якої задачі. Цим класам поставлені у відповідність методи формалізованого відображення систем.

Розглянемо приклад 1: характеристика системи – «Україна».

Класифікація системи: складна, багатопольярної рівноваги.

За структурою: мережева.

По відношенню до навколишнього середовища: відкрита.

По походженню: соціальна.

По опису змінних системи: зі змішаними змінними.

По типу опису закону функціонування системи: параметризована.

По способу керування системою (у системі): з комбінованим керуванням.

По залежності від передісторії: динамічна.

По залежності від часу: нестационарна.

За передбачуваністю поведінки: стохастична.

За режимом діяльності: що функціонує та розвивається, нестійка.

Завдання для самостійного виконання

1. Запропонувати та самостійно проаналізувати систему за схемою типового завдання (приклад 1).

№ варіанту		Тема:
1	15	Бібліотека
2	16	Кадровий облік
3	17	Робота куратора групи
4	18	Електронний довідник
5	19	Електронний навчально-методичний комплекс
6	20	Сайт-портфоліо
7	21	Облік комп'ютерних технічних засобів організації
8	22	Електронний посібник
9	23	База даних
10	24	Інтернет-магазин
11	25	Журнал успішності студентів
12	26	Моніторинг роботи користувачів локальної мережі
13	27	Облік та рейтингова оцінка діяльності студентів
14	28	Оптимізація розкладу

2. Описати підсистеми та визначити підцілі запропонованої системи.
3. Класифікувати систему згідно наступних параметрів:
 - природа елементів;
 - походження;
 - тип змінних;
 - тип оператора системи;
 - ступінь складності;
 - відношення до зовнішнього середовища;
 - реакція на вплив, що збурює;
 - характер управління;
 - ступінь організованості;
 - ступінь участі та впливу людини.

Контрольні запитання до лабораторної роботи

1. Дайте визначення поняття «система». Назвіть її основні ознаки.
2. Чим відрізняються реальні системи від абстрактних? Наведіть приклади.
3. Охарактеризуйте класифікацію систем за ознаками (детермінованість, відкритість тощо).
4. Що таке емерджентність системи? Чому вона є ключовою властивістю?
5. Охарактеризуйте класифікацію систем за рівнями складності К. Боулдінга.
6. Чим відрізняються добре організовані системи від дифузійних?

ЛАБОРАТОРНА РОБОТА №2 СТВОРЕННЯ ІНТЕЛЕКТУАЛЬНОЇ КАРТИ

Мета лабораторної роботи: освоєння технології побудови інтелектуальних карт. Створення інтелектуальної карти за допомогою редактора.

Теоретичні відомості

Інтелектуальні карти (або ментальні карти) є ефективним методом візуалізації інформації, що дозволяє структурувати думки в нелінійній формі для кращого сприйняття та запам'ятовування. Їхня технологія базується на принципах асоціативного мислення, де центральна ідея розгортається радіально через гілки з ключовими поняттями.

Інтелектуальна карта – це графічна схема, яка відображає зв'язки між ідеями, поняттями чи завданнями, з центральним вузлом у фокусі уваги. На відміну від традиційних лінійних конспектів, вона використовує радіальний запис, кольори, символи та малюнки для активації обох півкуль мозку, що полегшує творче мислення та аналіз складних тем (рис. 2.1).

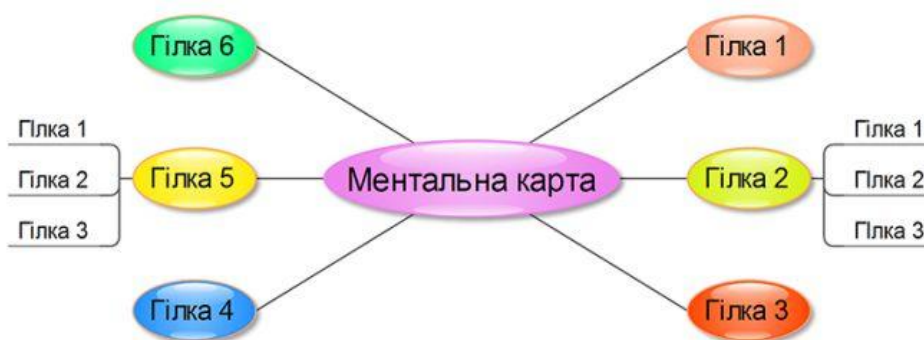


Рисунок 2.1 – Приклад інтелектуальної карти

Побудова карти базується на ієрархічній структурі. У центрі завжди розташовується головний образ або ключове поняття, від якого розходяться основні гілки (категорії першого рівня). Кожна основна гілка може мати необмежену кількість підгілок, формуючи логічне дерево знань [2].

Важливим елементом є використання ключових слів замість довгих речень. Це дозволяє мозку швидше «зчитувати» карту. Додатково застосовуються візуальні маркери (іконки, стрілки, різні кольори), які допомагають розрізнити смислові блоки (табл. 2.1).

Таблиця 2.1 – Ключові компоненти та їхнє призначення

Елемент карти	Роль у процесі сприйняття
Центральне ядро	Фокусує увагу на об'єкті вивчення.
Товсті гілки	Визначають стратегічні напрямки або розділи.
Тонкі гілки	Деталізують інформацію, уточнюють факти.
Колірне кодування	Групує ідеї за контекстом, полегшує навігацію.
Графічні образи	Заміняють цілі абзаци тексту, активуючи праву півкулю мозку.

Завдання для самостійного виконання

Створити інтелектуальну карту в середовищі редактора інтелектуальних карт. Перелік тем для створення карти відображений у таблиці 2.2.

Таблиця 2.2 – Теми для побудови інтелектуальних карт

№	Теми
1	Системність світу
2	Системний підхід
3	Системологія
4	Теорія систем
5	Кібернетика
6	Дослідження операцій
7	Системотехніка
8	Синергетика
9	Системний аналіз
10	Теорія систем

Сформуванати команду з трьох студентів для виконання завдання.

Використовуючи навчально-методичні матеріали дисципліни та ресурси мережі Інтернет, ознайомтеся з обраною темою, яку треба розкрити в інтелектуальній карті. Зберіть будь-які матеріали, інформацію, дані про досліджувану тему. Зафіксуйте адреси джерел. Приділіть увагу не тільки словам, а й візуальним образам (фотографіям, малюнкам, схемам).

Розбийте тему на основні підтеми. Кількість тем другого рівня може бути довільною (з урахування обмежень на сприйняття інформації людиною). Але для виконання роботи тему треба розбити на шість підтем, щоб кожний член команди працював над двома напрямками. Кожний напрям деталізуйте з необхідною кількістю ідей в ієрархічному порядку.

Створити інтелектуальну карту в редакторі.

Захист лабораторної роботи

1. Команда презентує розроблену карту в групі. Виступає кожний член команди (до 10 хвилин на команду).
2. Після презентації інші команди задають запитання стосовно змісту карти: по одному питанню від кожної команди.
3. Кожна команда та викладач за результатами презентації виставляють команді, що зробила презентацію, загальну оцінку та оцінку для кожного її учасника. Максимальна кількість балів – 5.
4. Кожний учасник команди отримує оцінку за виконану роботу як середнє арифметичне всіх отриманих оцінок.

Контрольні запитання до лабораторної роботи

1. Що таке інтелектуальна карта? Які ще назви вона має?
2. Назвіть основні правила побудови інтелектуальних карт.
3. Охарактеризуйте алгоритм побудови інтелектуальних карт.
4. Сформулюйте рекомендації до стилю інтелектуальної карти.
5. Який слід використовувати формат аркушу для карти?

6. Сформулюйте рекомендації для відображення центрального образу, назв, гілок.
7. Охарактеризуйте сфери та можливості застосування інтелектуальних карт.
8. Які існують інструментальні засоби для побудови інтелектуальних карт?

ЛАБОРАТОРНА РОБОТА №3-4
НАЛАГОДЖЕННЯ уEd Graph Editor
СТВОРЕННЯ МОДЕЛІ ОРГАНІЗАЦІЙНОЇ СТРУКТУРИ ПІДПРИЄМСТВА В уEd
Graph Editor

Мета лабораторної роботи: освоєння основних положень щодо моделювання в середовищі уEd Graph Editor, створення організаційної схеми підприємства дрібнооптової торгівлі.

Теоретичні відомості

Методологія ARIS (Architecture of Integrated Information Systems) – це методологія та середовище моделювання для аналізу бізнес-процесів, розроблена професором А.-В. Шесром. Вона заснована на теорії архітектури інтегрованих інформаційних систем і призначена для комплексного візуального аналізу та моделювання діяльності організацій.

Методологія ARIS припускає певний підхід до формалізації інформації про діяльність організації та подання її у вигляді графічних моделей, зручних для розуміння й аналізу. Створювані моделі є документованою сукупністю знань про систему управління, включаючи організаційну структуру, процеси, взаємодію між організацією та суб'єктами ринку, склад і структуру документів, послідовність етапів процесів [3].

Ключові переваги методології ARIS:

- можливість розглядати об'єкт з різних точок зору: організаційну, функціональну, процесну, інформаційну;
- різноманітність методів моделювання для відображення різних аспектів предметної області;
- створення інтегрованої та цілісної моделі: всі моделі та об'єкти зберігаються в єдиній базі проекту (репозиторії);
- можливість багаторазового застосування результатів моделювання та накопичення корпоративного знання;
- зберігання всієї інформації в єдиному репозиторії, що забезпечує цілісність і несуперечність процесу моделювання.

У цій лабораторній роботі класична методологія ARIS адаптована для виконання у середовищі уEd Graph Editor – безкоштовного кросплатформного редактора діаграм. Хоча уEd не має вбудованого репозиторію та автоматичної підтримки нотації ARIS, він дозволяє:

- створювати всі основні типи діаграм ARIS засобами стандартних форм вузлів.
- використовувати кольорове кодування для розрізнення типів об'єктів;
- застосовувати алгоритми автоматичного компоновання (Layout) для впорядкування схем;
- реалізувати зв'язки між діаграмами через URL-навігацію;
- зберігати всі діаграми у відкритому форматі GraphML.

Оригінальна методологія ARIS реалізована у професійних продуктах ARIS Express, ARIS Architect і ARIS Designer. У цій роботі ми відтворюємо лише візуальну нотацію та логіку побудови діаграм засобами уEd, без репозиторію та перевірки консистентності моделей. Це навчальна адаптація для освоєння принципів моделювання бізнес-процесів.

уEd Graph Editor – це безкоштовний кросплатформний редактор діаграм і графів, розроблений компанією уWorks GmbH. Програма написана на Java, що забезпечує її роботу на Windows, macOS та Linux без необхідності встановлення додаткових компонентів. Основне

призначення уEd – візуалізація структурованих даних у вигляді вузлів (nodes) та ребер (edges), тобто у вигляді орієнтованих або неорієнтованих графів.

Ключовою перевагою уEd над аналогічними інструментами є потужна система автоматичного компоунування (layout algorithms). Після введення структури даних користувач може одним натисканням застосувати один із вбудованих алгоритмів, і програма самостійно розмістить усі елементи у зручному для сприйняття вигляді. Такий підхід особливо цінний під час роботи з великими схемами – від десятків до тисяч вузлів [3].

Інтерфейс програми складається з кількох ключових зон, що розміщені навколо центрального полотна редагування (табл. 3.1).

Таблиця 3.1 – Елемент інтерфейсу уEd Graph Editor

Елемент інтерфейсу	Призначення
Palette	Бічна панель зі всіма доступними формами (шаблонами) вузлів, згрупованими за типами діаграм. Для org-chart тут зосереджено форми типу «Entity».
Properties View	Панель властивостей виділеного елемента: колір, розмір, шрифт, тип лінії тощо.
Overview	Мініатюрний перегляд усього полотна – корисний під час роботи з великими схемами, коли весь граф не вміщується на екрані.
Structure View	Деревовидний перелік усіх вузлів і зв'язків діаграми у вигляді ієрархічного списку.
Меню Tools → Layout	Доступ до алгоритмів автоматичного компоунування (Hierarchical, Organic, Tree, Circular тощо).

Файли зберігаються у відкритому форматі GraphML (XML-стандарт для зберігання графів), що дозволяє легко інтегрувати уEd у різні автоматизовані процеси або передавати схеми між різними програмами, що підтримують цей формат. Готові схеми можна експортувати у PNG, JPG, SVG, PDF та інші формати.

Організаційна структура

Організація співробітників підприємства вимагає впорядкування їх взаємин у процесі роботи. Організаційна структура підприємства, установи – одна з головних характеристик, що визначають оптимальність функціонування та використання виробничо-технологічного потенціалу.

Організаційна структура (organizational structure) – це розподіл відповідальності, повноважень і взаємин між працівниками. Розподіл зазвичай закріплений у розпорядчих документах.

Організаційна структура включає правила, що дозволяють провести статичну структурування підрозділів підприємства у вигляді схеми. Схеми організаційної структури, а також положення про відділення (департаменти), відділи, групи тощо використовують для аналізу організаційної структури.

Аналіз організаційної структури проводиться з метою виявлення:

- обгрунтованої кількості рівнів ієрархії;

- наявності надмірної кількості (більше 5-6) підлеглих підрозділів у одного керівника;
- наявності малої або надмірної кількості підлеглих співробітників у одного керівника;
- підпорядкування одних ланок (співробітників) різним керівникам.

Моделювання організаційної структури – стартова точка у створенні топології інформаційної системи, яка, як передбачається, підтримуватиме організаційну структуру найбільш оптимальним чином. З'єднання мережі та мережні вузли, розташовані в певних місцях компанії, є головними елементами топології мережі інформаційної системи.

Модель організаційної структури – одна з найважливіших моделей, оскільки вона описує суб'єкти, які визначають входи та виходи потоків ресурсів підприємства. У моделі організаційної структури доцільно показувати:

- підрозділи підприємства;
- посади та прізвища керівників підрозділів; фізичне розташування відділів на підприємстві.

Організаційна структура втілює розподіл економічного об'єкта (підприємства, установи) на підрозділи, відділення, відділи, цехи, лабораторії, ділянки, групи з метою впорядкування управління, налагодження взаємодії ланок, установлення підлеглості та підпорядкування, відповідальності.

Підрозділ (організаційна одиниця) – офіційно виділена частина організації та працівники, які входять до неї, виконують закріплені за ними роботи та відповідають за виконання покладених на них завдань. Підрозділи виділяються найчастіше за ознакою профілю доручених їм робіт. До підрозділів відносять: відділення (департаменти), відділи, цехи, склади, лабораторії, сектори тощо. Розрізняють функціональні та виробничі підрозділи [3].

Функціональний підрозділ – складова частина організації, що виконує конкретні завдання та досягає конкретних специфічних цілей.

Виробничий підрозділ – самостійна частина ланки управління, що виконує завдання оперативного управління виробництвом і забезпечує випуск продукції в матеріальній формі.

Департамент (відділення) – крупний структурний підрозділ, що спеціалізується на виконанні певних робіт в організації, наприклад: департаменти збуту та закупівель. У складі департаменту зазвичай виділяють дрібніші підрозділи – відділи, цехи тощо, наприклад, департамент збуту, департамент закупівель.

Відділ – структурний підрозділ управління організацією, що виконує певні функції управління, наприклад: відділи замовлень, збуту, закупівель, проектний, розрахунковий, конструкторський, маркетингу.

Цех – основний підрозділ виробничого підприємства, що виробляє напівфабрикати або кінцеву продукцію та здійснює відособлену частину технологічного циклу виготовлення продукції, наприклад: складальний і ливарний цехи.

Склад – приміщення, комплекс приміщень, призначений для зберігання матеріальних цінностей, наприклад: склади готової продукції і інструментальний.

Опис організаційної структури підприємства здійснюється зазвичай у вигляді схеми (діаграми) організаційної структури.

Організаційна схема – форма подання організаційних структур, яка описує організаційні одиниці й їх взаємозв'язки залежно від обраних критеріїв структуризації. Організаційна схема ілюструє розподіл завдань у компанії, а її об'єкти відображають

функціональні відповідальності.

Організаційні одиниці – це структурні підрозділи, що виконують завдання, які реалізуються для досягнення бізнес-цілей компанії.

Відносини – це зв'язки між організаційними одиницями. Для того щоб специфікувати відносини керівництва (підпорядкування), використовують різні типи з'єднань, що зв'язують організаційні одиниці, наприклад, «адміністративне керівництво», «функціональне керівництво», «складається з» тощо.

Посада – службове місце працівника, яке пов'язане з виконанням певних службових обов'язків і визначає коло його повноважень і відповідальності. Для опису окремих посад у компанії необхідний відповідний тип об'єкта – посада. Одна організаційна одиниця може бути пов'язана з кількома посадами. Змістовність з'єднання відповідає зв'язкам між ними.

Роль – нормативно закріплений очікуваний від працівника стандарт поведінки, чітко визначений його посадою в організації. З діяльністю організації можуть бути пов'язані не тільки її співробітники, а інші зацікавлені особи. У цьому сенсі роль – певний образ дій, який безпосередньо впливає з необхідності управління певним об'єктом, функція, виконувана співробітником або зацікавленою особою.

За допомогою зв'язку «займає» до посади (ролі) можуть бути прив'язані конкретні співробітники. Пов'язуючи співробітника з організаційною одиницею, вказують на те, що він є в її складі. Розташування визначає фізичне розташування організаційних одиниць, устаткування та технічних ресурсів компанії. Ними може бути регіон, місто, завод, будівля, кімната та навіть окреме робоче місце. Структура території може виявитися важливою для розподілених організацій, де територіальне розташування окремих підрозділів (філіалів) значною мірою впливає на особливості устрою системи управління, зокрема бізнес-процесами.

Розташування організаційної одиниці – це важливий зв'язок між організаційною структурою та топологією комп'ютерної мережі. Для кожної організаційної одиниці можна задати розташування, причому це повинно бути зроблено якомога раніше – на рівні визначення вимог до системи [3].

Розташування може бути пов'язане з організаційною одиницею на будь-якому рівні ієрархії. Воно може визначати як окрему будівлю, так і при детальному аналізі – окремий офіс або навіть одно робоче місце в кімнаті. Таким чином, на етапі специфікації проекту можна пов'язати вузли мережі з окремим робочим місцем організаційної одиниці. Характеристика основних об'єктів організаційної схеми подана в таблиці 3.2.

Таблиця 3.2 – Характеристика основних об'єктів організаційної схеми

Елемент	Вузол у уEd	Опис
Посада / відділ	Entity Node (прямокутник)	Базовий блок схеми. Містить назву посади або підрозділу. В уEd використовується стандартний прямокутний вузол або форми з розділу «Entity».
Підпорядкування	Directed Edge (стрілка)	Напрявлене ребро від керівника до підлеглого, що означає відносини прямого підпорядкування.
Функц. зв'язок	Dashed Edge (пунктир)	Пунктирне ребро для функціональних або консультативних зв'язків між підрозділами без прямого підпорядкування.

Елемент	Вузол у уEd	Опис
Підрозділ (група)	Group Node	Групуючий вузол у уEd, що об'єднує кілька посад усередині одного відділу. Дозволяє згорнути та розгорнути підрозділ на схемі.
Міжфайловий зв'язок	URL у властивостях вузла	Посилання зі схеми підприємства на окремий файл із деталізованою схемою підрозділу, реалізоване через властивість URL вузла.

Схема організаційної структури може бути подана такими рівнями ієрархії: очолює організацію директор (перший рівень ієрархії), якому підпорядковані заступники (другий рівень ієрархії), кожному з яких підпорядковані начальники відділень (третій рівень ієрархії), що мають у своєму підпорядкуванні декілька начальників відділів (четвертий рівень ієрархії). Схема організаційної структури будується ієрархічно – від верхнього рівня структури до нижнього рівня. У верхній рівень схеми включають самостійні підрозділи або вищих посадових осіб, які входять у структуру організації. Перший рівень деталізується на нижчі рівні – рівні вхідних структурних підрозділів або підлеглих посадових осіб.

Нижчим рівнем є опис підрозділів на рівні посад – штатних одиниць, займаних конкретними співробітниками. Для деталізації підрозділів до рівня співробітників доцільно повністю вказувати посаду у складі підрозділу. У випадку, якщо в одному підрозділі є декілька однакових посад, вони нумеруються.

Приклад організаційної схеми, виконаної в середовищі уEd Graph Editor, наведений на рисунку 3.1.

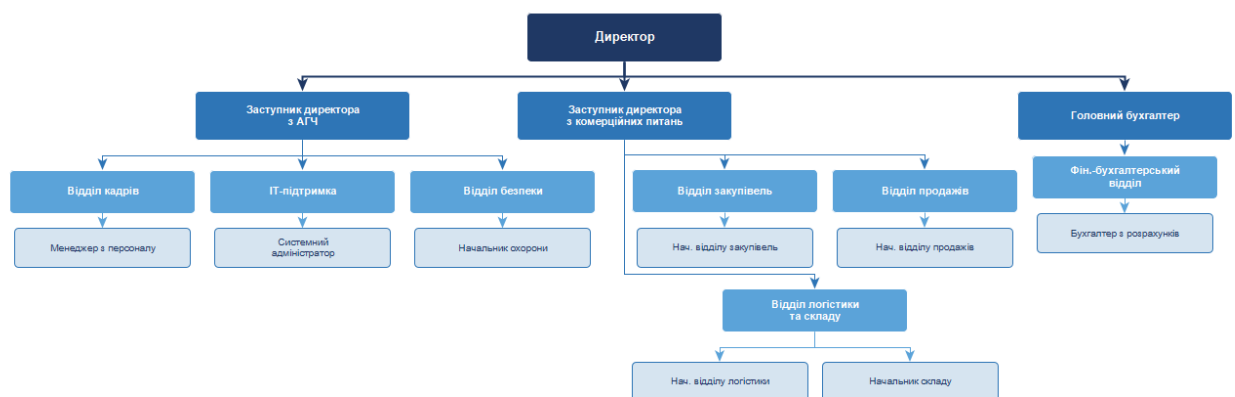


Рисунок 3.1 – Приклад організаційної схеми

Для побудови схеми організаційної структури допускається використання однотипних об'єктів. Виділяють такі діаграми з однотипними об'єктами:

- діаграма посадової підпорядкованості, що складається тільки з об'єктів типу «Посада» та показує взаємозв'язки між посадами;
- діаграма взаємозв'язків організаційних одиниць, що складається тільки з об'єктів типу «Організаційна одиниця» та показує взаємозв'язки між організаційними одиницями;
- діаграма розташування, що використовує тільки об'єкти типу «Розташування»

та показує зв'язки між місцями знаходження об'єктів.

Деталізація об'єктів організаційної схеми на моделі нижчого рівня повинна бути ретельно продумана й ухвалена учасниками моделювання.

Визначення глибини моделювання пов'язане по-перше з вибором ступеня узагальнення та деталізації розроблюваних моделей. По-друге кількості рівнів моделей, що є переходом від загальних моделей до деталізованих.

Перший пункт пов'язаний з цілями та завданнями моделювання. Наприклад, для впровадження інформаційної системи необхідний опис до рівня робочих місць або навіть до рівня операцій на робочих місцях з побудовою моделей структур даних і т. п.

Другий пункт пов'язаний з необхідністю отримання наочних моделей, не переобтяжених інформацією.

Як правило, діаграми більш ніж з чотирма або п'ятьма рівнями ієрархії стають погано читаними та важкими для сприйняття, а акуратне розміщення об'єктів у полі моделі – проблематичним.

Для зменшення кількості рівнів на одній діаграмі використовується механізм деталізації: об'єкти четвертого або п'ятого рівнів деталізують моделлю того ж типу, де й відображають наступні 4-5 рівнів ієрархії.

Проте деталізація буде створена не для кожного екземпляра об'єкта, а для його опису. Іншими словами, створюється деталізація одного з об'єктів моделі, яка буде проведена для всіх екземплярів, у якій би моделі об'єкт не був присутній.

Для об'єктів організаційної схеми необхідно задати властивості (рис. 3.2), опис яких наведений у таблиці 3.3.

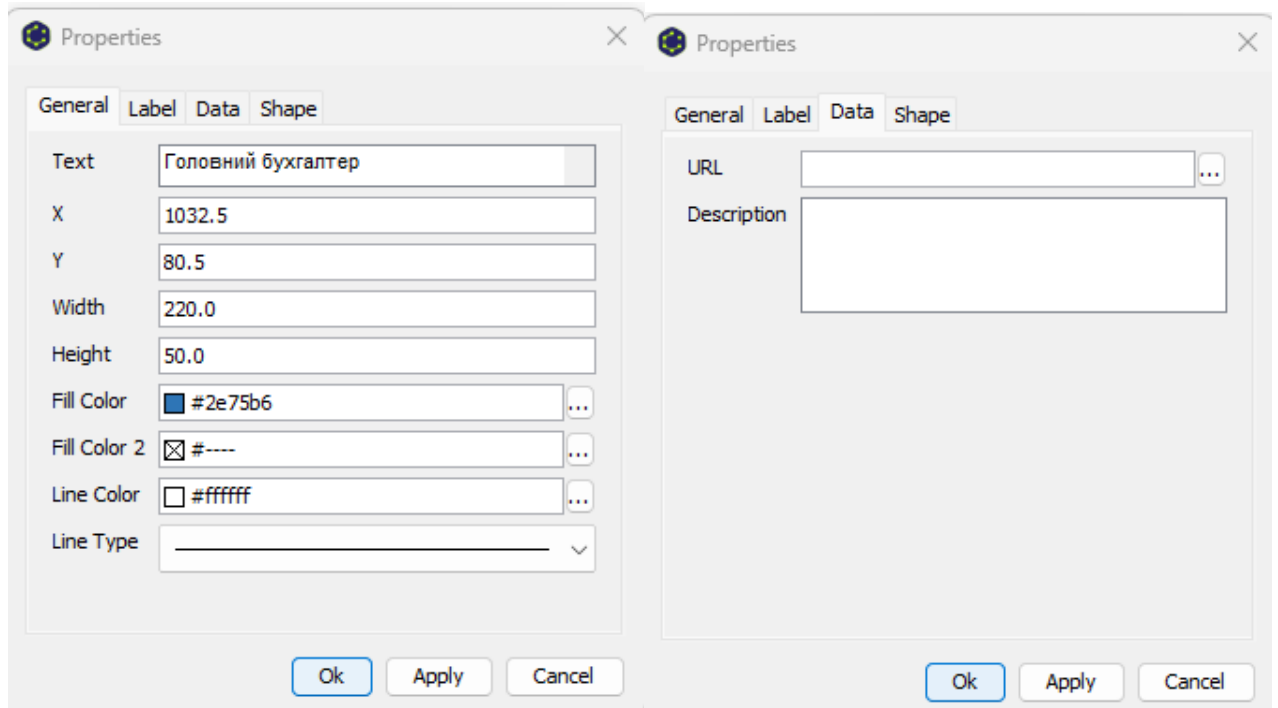


Рисунок 3.2 – Вікно Properties

Таблиця 3.3 – Властивості об'єктів організаційної схеми

Назва властивості	Опис властивості
Text/Текст	Дозволяє вказати імена для елементів. Унікальні імена допомагають ідентифікувати елементи. Цей атрибут може змінюватися безпосередньо на графічному зображенні об'єкта
Description/Опис	Використовується для словесного опису змісту об'єкта
Link/Посилання	Містить ярлик для документа або додатка, який може бути від критий або запущений для відповідного об'єкта. Якщо помістити ярлик у вікні моделі, символ посилання відображається поруч із ім'ям елемента. Ярлики відкривають прикладні системи або документи прикладних систем

Завдання для самостійного виконання

Завдання 3.1. Створити організаційну схему підприємства дрібнооптової торгівлі в уEd Graph Editor. Розмістіть, опишіть та зв'яжіть об'єкти на організаційній схемі підприємства.

Порядок виконання

1. Відкрийте нову схему: File → New, та збережіть файл одразу під назвою «enterprise_structure.graphml» у робочій папці.
2. Перетягніть з Palette (секція «Entity» або «Shape Nodes») прямокутні вузли для кожної посади. Розташуйте їх приблизно у потрібному порядку – алгоритм компоновання впорядковує їх пізніше.
3. Двічі клацніть на кожен вузол та введіть відповідну назву посади або підрозділу.
4. Для кожного вузла у Properties View налаштуйте колір: директор – темно-синій, керівники підрозділів – синій, рядові посади – блакитний.
5. Наведіть курсор на вузол-керівника до появи синього кружечка з'єднувача. Перетягніть ребро до вузла-підлеглого. Повторіть для всіх зв'язків підпорядкування.
6. Виділіть вузли одного підрозділу, натисніть ПКМ → «Group Selection». Введіть назву групи (наприклад, «Відділ продажу»).
7. Застосуйте автоматичне компоновання: меню Tools → Layout → Hierarchical → ОК. Схема автоматично впорядкується.
8. Збережіть файл (Ctrl+S).

Завдання 3.2. Створення організаційної схеми підрозділу підприємства.

У завданні 3.2 необхідно:

1. Ознайомитися з описом бізнес-процесу підприємства дрібнооптової торгівлі відповідно до варіанта, що виданий викладачем. Знайти на схемі організаційної структури підприємства відділ, у якому виконується даний процес і ознайомитися з його описом.
2. Створити детальну організаційну схему підрозділу, в якому виконуються бізнес-процеси, відповідно до варіанта завдання, в уEd Graph Editor.

Порядок виконання

1. Відкрийте новий файл: File → New. Збережіть під іменою «department.graphml».
2. Створіть вузли для всіх посад відділу.
3. Для кожної посади у Properties View (вкладка General) заповніть поле опису – вкажіть коротко основні обов'язки (наприклад: «Ведення клієнтської бази, виставлення рахунків»).

4. Додайте зв'язки підпорядкування та застосуйте компонування Tree або Hierarchical.

5. Збережіть файл.

Контрольні запитання до лабораторної роботи

1. Дайте визначення таких понять: організаційна структура, підрозділ, функціональний підрозділ, виробничий підрозділ, відділення, відділ, цех, склад, посада.

2. Що таке організаційна схема, які основні правила її побудови?

3. Охарактеризуйте основні об'єкти організаційної схеми в уEd Graph Editor.

4. У яких випадках використовують деталізацію організаційної схеми, як визначається глибина моделювання?

5. Які діаграми з однотипними об'єктами використовуються? Охарактеризуйте їх.

6. Як відобразити на схемі назву зв'язку?

7. Яким чином створюється зв'язок між основною та деталізованою схемам?

ЛАБОРАТОРНА РОБОТА №5-7 МОДЕЛЮВАННЯ ФУНКЦІЙ ТА БІЗНЕС-ПРОЦЕСІВ В ІНСТРУМЕНТАЛЬНІЙ СИСТЕМІ ARIS

Мета лабораторної роботи: освоєння основних положень з моделювання бізнес-процесів у нотації eEPC в середовищі уEd Graph Editor.

Теоретичні відомості

Процес – це сукупність взаємопов'язаних і взаємодіючих видів діяльності, що перетворюють вхід на вихід. Будь-який процес обов'язково повинен мати вхід і вихід, тобто отримувати будь-яку інформацію або ресурси від зовнішнього об'єкта або іншого процесу, перетворювати їх і передавати іншому об'єкту або процесу.

Бізнес-процес – це логічно впорядкована взаємопов'язана послідовність дій (робіт, операцій), виконуваних посадовими особами та підрозділами організації для отримання бажаного кінцевого результату. Процес не залежить від організаційної структури – він може проходити через різні підрозділи.

Будь-який процес обов'язково повинен мати вхід і вихід, тобто отримувати будь-яку інформацію або ресурси від зовнішнього об'єкта або іншого процесу, перетворювати їх і передавати іншому об'єкту або процесу [3].

Моделювання процесів відображає, як виконується певна функція. На початку проектування починають із Process Landscape / Value-Added Chain Diagram (VACD). Деталізацію процесів виконують за допомогою таких нотацій:

- Function Tree – дерево функцій (статичний ієрархічний опис);
- EPC (Event-driven Process Chain) – подійний ланцюжок процесів;
- FAD (Function Allocation Diagram) – діаграма оточення функції

Характеристика основних об'єктів, що використовуються для моделювання бізнес-процесів, подана в таблиці 5.1, а зв'язків між об'єктами в таблиці 5.2.

Таблиця 5.1 – Характеристика основних об'єктів моделей бізнес-процесів

Об'єкт ARIS	Форма	Реалізація в уEd	Правила іменування / колір
Функція (Activity / Function)	Прямокутник із заокругленими кутами	Shape Node → Round Rectangle; колір: #5BA3D9 (синій)	Ім'я починається з дієслова дії: «Оформити заявку», «Перевірити товар»
Подія (Event)	Шестикутник	Shape Node → Hexagon; колір: #00B050 (зелений)	Ім'я = об'єкт + стан: «Заявка отримана», «Товар відвантажений»
Інтерфейс процесу (Process Interface)	Прямокутник подвійної рамки	Shape Node → Rectangle (подвійна рамка); колір: #1F3864 (темно-синій)	Містить код і назву процесу: «Процес: Продаж товару»
Оператори AND / OR / XOR	Ромб із позначкою	Shape Node → Diamond; підпис: AND / OR / XOR; колір: #FF9900 (помаранчевий)	Не іменуються, лише позначка всередині ромба

Об'єкт ARIS	Форма	Реалізація в уEd	Правила іменування / колір
Документ (Document)	Прямокутник із загнутих кутами	Shape Node → Parallelogram або Rectangle; колір: #FFE699 (жовтий)	Назва документа: «Заявка на товар», «Рахунок-фактура»
Виконавець / Роль	Еліпс / овал	Shape Node → Ellipse; колір: #D6E4F0 (блакитний)	Назва посади або підрозділу: «Менеджер з продажів», «Відділ логістики»
ІТ-система (IT System)	Прямокутник зі скошеними кутами	Shape Node → Trapezoid або Rectangle зі скошенням; колір: #E2EFDA (зелений)	Назва системи: «1С: Підприємство», «CRM-система»
Ризик (Risk)	Трикутник / попередження	Shape Node → Triangle; колір: #FF0000 (червоний)	Опис небезпеки: «Клієнт не забирає товар»
База даних (Database)	Циліндр	Shape Node → Ellipse + підпис «БД»; або Custom shape; колір: #9DC3E6	Назва БД: «База клієнтів», «Склад_DB»

Таблиця 5.2 – Типи зв'язків між об'єктами

Назва зв'язку	Напрямок	Приклад / Реалізація в уEd
Активізує (Activates)	Подія → Функція; Функція → Подія; Функція → Функція	Основний зв'язок ланцюжка EPC. Стрілка суцільна, темно-синя (#1F3864).
Є входом для (Is input for)	Документ → Функція; Сутність → Функція	Документ або БД ліворуч від функції. Стрілка суцільна, помаранчева (#FF9900).
Має результат (Has output of)	Функція → Документ	Вихідний документ праворуч від функції. Стрілка суцільна, зелена (#00B050).
Виконує (Carries out)	Виконавець → Функція	Виконавець (еліпс) праворуч. Стрілка пунктирна, блакитна (#5BA3D9).
Пов'язаний ризик (Occurs at)	Функція → Ризик	Ризик (трикутник) поряд із функцією. Стрілка пунктирна, червона (#FF0000).
Виконується у системі (Is processed in)	Функція → ІТ-система	ІТ-система (трапеція) ліворуч від функції. Стрілка суцільна, зелена.

Діаграма дерева функцій (Function Tree) в уEd (рис. 5.1).

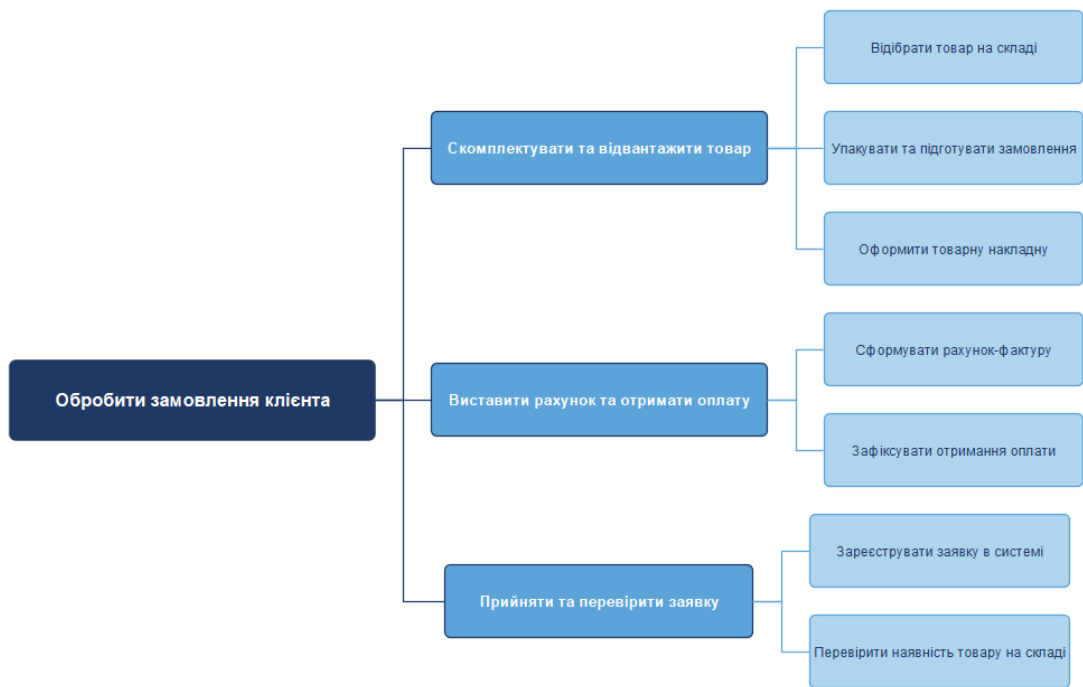


Рисунок 5.1 – Приклад діаграми дерева функцій

Функція – це опис елемента роботи, що є одним логічним етапом у рамках процесу. Базова функція – це функція найнижчого рівня, яку вже не можна розділити на складові елементи з метою аналізу бізнес-процесу.

Дерево функцій у уEd будується як ієрархічна діаграма типу «зверху вниз». На найвищому рівні розміщується головна функція (процес), нижче – підфункції, ще нижче – базові функції. Для автоматичного впорядкування ієрархії застосовується алгоритм компоновання Tools → Layout → Tree.

Критерії об'єднання функцій у дерево:

- процесний – приналежність до одного процесу;
- об'єктний – обробка одного й того самого об'єкта (наприклад, «заявка на товар»);
- операційний – виконання однакових операцій над різними об'єктами.

Правила іменування функцій: ім'я функції починається з дієслова дії або позначення процесу. Наприклад: «Оформити заявку на товар», «Перевірити залишок на складі», «Сформувати рахунок-фактуру». Уникайте іменників у називному відмінку без дієслова.

Діаграма EPC в уEd Graph Editor (рис. 5.2).

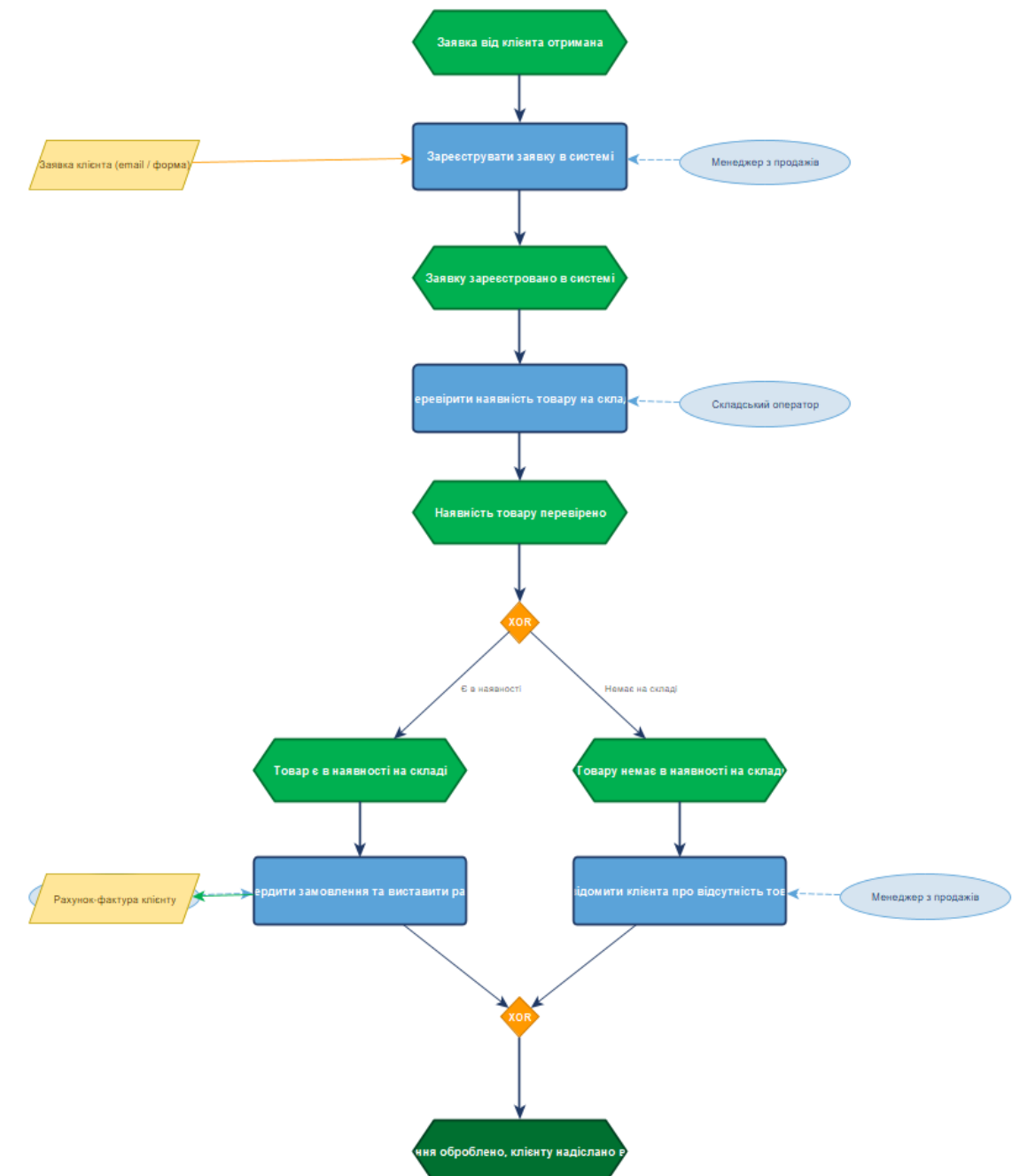


Рисунок 5.2 – Приклад діаграма EPC

Подійний ланцюжок процесу (Event-driven Process Chain, EPC) призначений для детального опису процесів, що виконуються підрозділами або конкретними співробітниками. Кожна функція в моделі може мати таке оточення: початкову та кінцеву події, відповідальних виконавців, матеріальні та документні потоки.

Подія – це стан, який є суттєвим для цілей управління та впливає на подальший хід виконання процесу. На відміну від функцій, події не мають тривалості та відбуваються миттєво. Ім'я події = об'єкт + стан: «Заявка сформована», «Оплата отримана».

Правила побудови EPC-діаграми в уEd:

- графічні елементи процесу розташовуються зверху донизу по вертикалі;
- модель починається однією стартовою подією (або інтерфейсом процесу) і завершується однією результативною подією;
- події і функції чергуються у ланцюжку;

- виконавці функцій (еліпси) розташовуються праворуч від функцій;
- вхідні документи розташовуються ліворуч від функцій; вихідні – теж ліворуч або праворуч залежно від напрямку потоку;
- логічні оператори (AND/OR/XOR) реалізуються через вузол-ромб із відповідним підписом.

Правила використання логічних операторів:

- шлях процесу завжди розділяється та об'єднується за допомогою оператора-ромба;
- рішення про подальший хід виконання процесу приймаються функціями, а не подіями;
- подія та функції, що настають за нею, НЕ повинні бути пов'язані операторами OR або XOR – це виключно прерогатива функцій;
- AND – усі гілки виконуються паралельно; OR – одна або більше гілок; XOR – виключно одна гілка.

Діаграма оточення функції (FAD) в уEd (рис. 5.3).

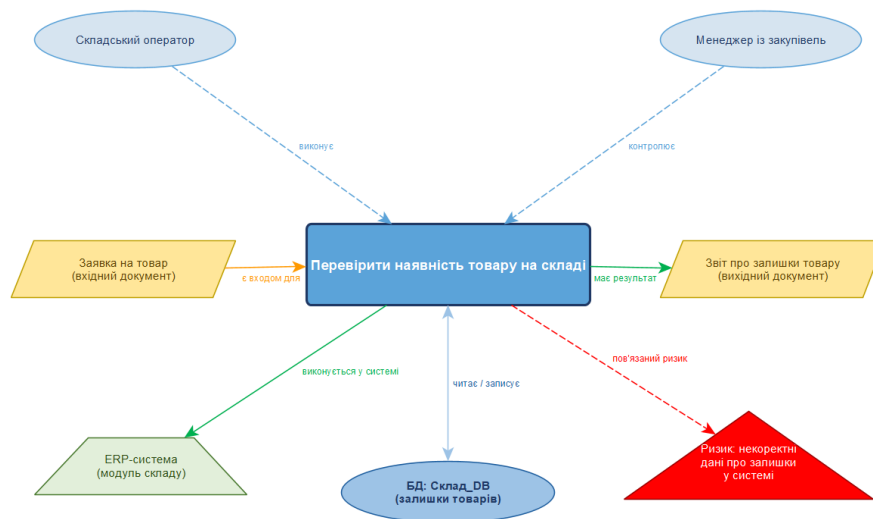


Рисунок 5.3 – Приклад діаграми оточення функції (FAD)

Діаграма оточення функції (Function Allocation Diagram, FAD) призначена для того, щоб описати всі об'єкти, що оточують функцію: виконавців, вхідні та вихідні потоки інформації, документи, матеріали, продукти/послуги, ІТ-системи, ризики, бази даних.

У уEd FAD будується як «зіркова» діаграма: центральний вузол – функція, навколо неї розташовані пов'язані об'єкти. Зв'язки позначаються підписами. Для автоматичного розташування об'єктів навколо центру застосовується Tools → Layout → Organic або Circular.

Завдання для самостійного виконання

Завдання 6.1. Створення моделі дерева функцій для процесу торговельного підприємства.

1. Ознайомитись з типами й операціями торговельних підприємств.
2. Ознайомтеся з описом процесу підприємства дрібнооптової торгівлі відповідно до варіанта.
3. Відкрийте уEd Graph Editor. Створіть нову діаграму: File → New.
4. Збережіть файл під назвою function_tree.graphml у робочій папці.

Порядок виконання

Крок 1. Налаштуйте Palette. У правій панелі Palette відкрийте розділ «Shape Nodes». Ви будете використовувати переважно Round Rectangle для функцій.

Крок 2. Додайте кореневий вузол (головна функція/процес). Перетягніть Round Rectangle на полотно. Двічі клацніть – введіть назву: наприклад, «Процес: Дрібнооптова торгівля». Задайте колір вузла (#1F3864, темно-синій): виділіть вузол → Properties View → Fill → виберіть колір.

Крок 3. Додайте функції першого рівня. Перетягніть ще 3-5 вузлів Round Rectangle нижче кореня. Задайте колір #2E75B6. Назвіть їх відповідно до варіанта, наприклад: «Прийом замовлень», «Формування поставки», «Відвантаження товару», «Облік та звітність».

Крок 4. Додайте базові функції (другий рівень). Для кожної функції першого рівня додайте 2-4 підфункції з кольором #5BA3D9. Наприклад, для «Прийом замовлень»: «Реєстрація замовлення», «Перевірка залишків», «Підтвердження замовлення».

Крок 5. З'єднайте вузли ребрами. Наведіть курсор на батьківський вузол до появи синього коннектора → перетягніть ребро до дочірнього вузла. Тип ребра: суцільна лінія без стрілки (тип «Hierarchic»).

Крок 6. Відобразіть назви зв'язків. Двічі клацніть на ребро → введіть підпис «is composed of» (складається з). Назви зв'язків розташуйте вздовж ребер.

Крок 7. Застосуйте автоматичне компонування: Tools → Layout → Tree → вкладка «General» → Direction: Top to Bottom → ОК. Схема впорядковується автоматично.

Крок 8. Збережіть модель (Ctrl+S).

Якщо дерево функцій виглядає надмірно широким, спробуйте змінити напрямок у налаштуваннях Layout Tree: Direction → Left to Right. Це дозволить краще використати простір листа при великій кількості гілок.

Завдання 6.2. Створення моделі ЕРС для опису процесу торгового підприємства.

1. Вивчіть основні положення щодо моделювання бізнес-процесів у нотації ЕРС.
2. За описом бізнес-процесу (відповідно до варіанта) визначте послідовність виконання функцій.
3. Для кожної функції визначте: події, що її ініціюють і завершують; виконавців; вхідні та вихідні документи.
4. Створіть модель ЕРС. Збережіть новий файл: erc_process.graphml.

Порядок виконання

Крок 1. Налаштуйте кольорову схему (відповідно до таблиці 1.2): функції – #5BA3D9 (Round Rectangle); події – #00B050 (Hexagon); оператори AND/OR/XOR – #FF9900 (Diamond); документи – #FFE699 (Parallelogram); виконавці – #D6E4F0 (Ellipse); ІТ-системи – #E2EFDA (Rectangle); ризики – #FF0000 (Triangle).

Крок 2. Розмістіть стартовий інтерфейс процесу або початкову подію вгорі полотна. Наприклад: вузол Rectangle (темно-синій) з підписом «Процес: Оформлення замовлення клієнта». Або подія Hexagon (зелений): «Замовлення від клієнта надійшло».

Крок 3. Додайте першу функцію нижче стартової події. Round Rectangle, синій: «Реєстрація замовлення». З'єднайте: Стартова подія → Функція (стрілка «Activates»).

Крок 4. Після кожної функції розміщуйте завершуючу подію (якщо вона значуща): «Замовлення зареєстровано» → наступна функція «Перевірка наявності товару». Чергуйте події та функції у вертикальному ланцюжку.

Крок 5. Де потрібне розгалуження (наприклад, товар є / товару немає), вставте оператор XOR (Diamond, помаранчевий). Від ромба проведіть дві стрілки: до функції «Підтвердження замовлення» та до функції «Повідомлення про відсутність товару». Підпишіть гілки (двічі клацніть на ребро): «Так» / «Ні» або «Товар є» / «Товару немає».

Крок 6. Для кожної функції додайте виконавця: перетягніть Ellipse (#D6E4F0) праворуч від функції → з'єднайте пунктирним ребром із підписом «Виконує». Наприклад: «Менеджер з продажів» → «Реєстрація замовлення».

Крок 7. Для кожної функції додайте вхідні документи ліворуч: Parallelogram (#FFE699). Поєднайте стрілкою «Is input for». Наприклад: «Заявка клієнта» → «Реєстрація замовлення».

Крок 8. Для ключових функцій додайте ІТ-системи (Rectangle зелений) ліворуч і вихідні документи. Наприклад: «1С: Підприємство» → функція «Формування накладної»; функція → «Видаткова накладна».

Крок 9. Завершіть модель кінцевою подією або інтерфейсом процесу: «Замовлення виконано» або «Процес: Доставка товару».

Крок 10. Застосуйте компонування: Tools → Layout → Hierarchical → ОК. Модель впорядкується у вертикальний ланцюжок.

Крок 11. Збережіть файл (Ctrl+S).

Зверніть увагу на оператори. Events не можуть ухвалювати рішення. Якщо після події є розгалуження – між подією та розгалуженням обов'язково має стояти функція. Оператор OR або XOR після події – помилка нотації EPC.

Орієнтовна структура EPC для дрібнооптової торгівлі

У таблиці 5.3 наведено мінімальний каркас EPC-діаграми, що відповідає типовому варіанту завдання

Таблиця 5.3 – Каркас EPC-діаграми

№	Подія / Функція	Тип об'єкта	Виконавець / Документ
1	Замовлення від клієнта надійшло	Подія (старт)	
2	Реєстрація замовлення	Функція	Менеджер з продажів; вхід: Заявка клієнта
3	Замовлення зареєстровано	Подія	
4	Перевірка наявності товару на складі	Функція	Комірник; вхід: Список залишків (БД Склад)
5	XOR – Товар є / Товару немає	Оператор XOR	
6(а)	Підтвердження замовлення клієнту	Функція (гілка «Є»)	Менеджер; вихід: Підтвердження замовлення
6(б)	Повідомлення про відсутність товару	Функція (гілка «Немає»)	Менеджер; ризик: Втрата клієнта
7	Формування видаткової накладної	Функція	Бухгалтер; ІТ: 1С; вихід: Видаткова накладна

№	Подія / Функція	Тип об'єкта	Виконавець / Документ
8	Відвантаження товару клієнту	Функція	Комірник / Водій-експедитор
9	Товар відвантажено. Процес завершено	Подія (фінал)	

Завдання 6.3. Створення моделі діаграми оточення функції для верхнього рівня процесу.

1. Вивчити основні положення з моделювання діаграми оточення функції.
2. Для функції бізнес-процесу верхнього рівня проаналізувати всі об'єкти, які її оточують: ризики, виконавці, відділи, розташування, вхідні і вихідні документи та сутності, ІТ-системи, продукти, БД.
3. Створити модель FAD для опису оточення верхнього рівня бізнес процесу в уEd Graph Editor..
4. Розмістити, описати та зв'язати об'єкти в моделі оточення процесу.

Порядок виконання

Крок 1. Розмістіть центральний вузол – функцію. Round Rectangle (#5BA3D9) в центрі полотна. Назва: наприклад, «Формування видаткової накладної». Розмір вузла зробіть дещо більшим (200×60 пікселів) у Properties View.

Крок 2. Розмістіть виконавців. Ліворуч або праворуч від функції розташуйте вузли Ellipse (#D6E4F0) для виконавців. Наприклад: «Бухгалтер», «Відділ продажів». З'єднайте пунктирними ребрами з підписом «Виконує».

Крок 3. Розмістіть вхідні документи. Ліворуч від функції – Parallelogram (#FFE699): «Підтверджене замовлення», «Прайс-лист». Стрілка від документа до функції, підпис «Is input for».

Крок 4. Розмістіть вихідні документи. Праворуч від функції – Parallelogram (#FFE699): «Видаткова накладна», «Рахунок-фактура». Стрілка від функції до документа, підпис «Has output of».

Крок 5. Розмістіть ІТ-системи. Нижче або ліворуч – Rectangle (#E2EFDA): «1С: Підприємство», «CRM-система». Стрілка від ІТ-системи до функції, підпис «Is processed in».

Крок 6. Розмістіть бази даних. Еліпс або прямокутник з підписом «БД» (#9DC3E6): «База клієнтів», «Склад_DB». Стрілка від БД до функції, підпис «Is input for».

Крок 7. Розмістіть ризики. Трикутник (#FF0000) поруч із функцією: «Помилка в реквізитах», «Системний збій 1С». Стрілка від функції до ризику, підпис «Occurs at».

Крок 8. Застосуйте компонування: Tools → Layout → Organic (або Circular) → ОК. Об'єкти розташовуються рівномірно навколо центральної функції.

Крок 9. Перевірте та вручну відкоригуйте позиції об'єктів за потреби. Збережіть файл (Ctrl+S).

Мінімальний склад FAD-діаграми: центральна функція + мінімум 2 виконавці + 2 вхідних документи + 2 вихідних документи + 1 ІТ-система + 1 база даних + 1–2 ризики. Загальна кількість вузлів – не менше 10.

Завдання 6.4. Створення зв'язку між діаграмами.

У уEd зв'язок між різними файлами-діаграмами реалізується через URL-властивість вузла. Клацнувши з утриманням Ctrl на такий вузол, уEd відкриває вказаний GraphML-файл у новій вкладці. Це відтворює функцію «Assigned diagram» з ARIS Express.

Порядок виконання

Зв'язок Function Tree → EPC

Крок 1. Відкрийте файл `function_tree.graphml`.

Крок 2. Виділіть вузол функції першого рівня (наприклад, «Прийом замовлень»).

Крок 3. У панелі Properties View знайдіть поле URL (у вкладці General).

Крок 4. Введіть шлях до файлу EPC: `file:///C:/Lab/epc_process.graphml` (або відносний шлях `./epc_process.graphml`, якщо обидва файли в одній папці).

Крок 5. Візуально позначте посилання: додайте до вузла невеликий значок «→» у правому нижньому куті, змінивши підпис вузла: «Прийом замовлень →».

Крок 6. Збережіть файл.

Крок 7. Перевірте навігацію: утримуйте Ctrl та двічі клацніть на вузол. Має відкритися файл `epc_process.graphml`.

Зв'язок EPC → FAD

Крок 1. Відкрийте файл `epc_process.graphml`.

Крок 2. Виділіть вузол функції верхнього рівня, для якої побудована FAD-діаграма.

Крок 3. У полі URL введіть шлях до файлу FAD: `./fad_function.graphml`.

Крок 4. Для зворотного зв'язку відкрийте `fad_function.graphml` → виділіть центральний вузол функції → у полі URL вкажіть `./epc_process.graphml`.

Крок 5. Збережіть обидва файли.

Крок 6. Перевірте навігацію в обох напрямках (Ctrl + подвійний клік).

Якщо навігація не працює:

Перевірте в налаштуваннях: Edit → Preferences → Navigation → увімкніть «Open URLs on Ctrl+Double Click». Також переконайтеся, що шляхи до файлів вказані коректно – з урахуванням регістру та роздільників шляху (/ на Linux/macOS, \\ на Windows).

Контрольні запитання до лабораторної роботи

1. Що таке підприємство, торговельне підприємство?
2. Охарактеризуйте групи торговельних підприємств.
3. Яким чином товар рухається від виробника до споживача?
4. Охарактеризуйте основні торговельні операції.
5. Охарактеризуйте функції процесу підприємства дрібнооптової торгівлі відповідно до варіанта.
6. Яке призначення моделі дерева функцій?
7. Охарактеризуйте поняття: «функція», «базова функція».
8. Які терміни разом з функцією можуть використовуватися для змістовного опису рівнів ієрархії в дереві функцій.
9. Охарактеризуйте об'єкт «функція» моделі дерева функцій за схемою: символ, назва, опис, правила іменування.
10. Які критерії можуть бути використані для об'єднання функцій у дерево функцій? Поясніть відповідь прикладами. Охарактеризуйте основні типи зв'язків між об'єктами моделі

дерева функцій відповідно до обраного критерію.

11. Яке призначення моделі подійного ланцюжка процесів?
12. Охарактеризуйте основні об'єкти моделі ЕРС за схемою: символ, назва, опис, правила іменування.
13. Охарактеризуйте основні типи зв'язків між об'єктами моделі ЕРС за схемою: символ – назва об'єкта – зв'язок – назва об'єкта – символ.
14. Наведіть основні правила розташування графічних символів на діаграмі ЕРС.
15. Наведіть правила використання логічних операторів на діаграмі ЕРС
16. Які існують правила розгалуження для подій?
17. Які існують правила розгалуження для функцій?
18. Яке призначення діаграми оточення функції?
19. Які об'єкти використовують у діаграмі оточення функції?

ЛАБОРАТОРНА РОБОТА №8

СИМПЛЕКСНИЙ МЕТОД РОЗВ'ЯЗКУ ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ. МЕТОД ШТУЧНОГО БАЗИСУ

Мета лабораторної роботи: засвоїти симплексний метод та метод штучного базису розв'язання задач лінійного програмування.

Теоретичні відомості

Задача лінійного програмування (ЗЛП) - це форма, в якій усі обмеження подані у вигляді рівностей, всі змінні є невід'ємними, і метою є максимізація або мінімізація лінійної цільової функції.

Канонічна форма ЗЛП (8.1):

$$F(X) = c_1x_1 + c_2x_2 + \dots + c_nx_n \rightarrow \max (\min) , \quad (8.1)$$

при обмеженнях: $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$, $a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$, ... $a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m$, $x_j \geq 0$, $j = 1, 2, \dots, n$, $b_i \geq 0$, $i = 1, 2, \dots, m$.

Основні характеристики канонічної форми:

- всі обмеження є рівностями;
- всі змінні невід'ємні;
- всі вільні члени b_i невід'ємні;
- система обмежень має ранг m ($m < n$).

Щоб привести задачу до канонічної форми, слід виконати такі кроки:

Перетворити всі нерівності типу \geq на \leq шляхом множення обмеження на (-1) .

Усі змінні повинні бути невід'ємними. Якщо є змінні, які можуть набувати негативних значень, їх треба замінити різницею двох нових невід'ємних змінних [4].

Якщо задача на мінімізацію, її потрібно перетворити на задачу максимізації шляхом зміни знака цільової функції $F(X) \rightarrow \min \Leftrightarrow -F(X) \rightarrow \max$.

Для перетворення обмежень-нерівностей типу « \leq » додаємо додаткову змінну: $a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b \Rightarrow a_1x_1 + a_2x_2 + \dots + a_nx_n + x_{n+1} = b$, $x_{n+1} \geq 0$.

При нерівності типа « \geq »: Віднімаємо додаткову змінну: $a_1x_1 + a_2x_2 + \dots + a_nx_n \geq b \Rightarrow a_1x_1 + a_2x_2 + \dots + a_nx_n - x_{n+1} = b$, $x_{n+1} \geq 0$.

Якщо змінна x_j не має обмеження на знак: $x_j = x_j' - x_j''$, де $x_j' \geq 0$, $x_j'' \geq 0$

Якщо змінна $x_j \leq 0$, вводимо $x_j' = -x_j$, де $x_j' \geq 0$.

Для перетворення вільних членів, якщо $b_i < 0$, множимо обидві частини i -го рівняння на (-1) : $a_1x_1 + a_2x_2 + \dots + a_nx_n = -b$ ($b > 0$) $\Rightarrow -a_1x_1 - a_2x_2 - \dots - a_nx_n = b$.

Симплексний метод – це ітераційний процес спрямованого перебору вершин багатогранника допустимих розв'язків, який забезпечує монотонне покращення значення цільової функції.

Основні етапи симплексного методу:

- вибір початкового базисного плану;
- обчислення коефіцієнтів цільової функції для кожної змінної (оцінка можливості покращення);
- визначення змінної, яка увійде до базису (найбільший потенціал покращення);
- визначення змінної, яка вийде з базису (обмежувальний рядок);

- оновлення симплексної таблиці;
- перевірка критерію оптимальності.

Початковий опорний план – це допустиме базисне рішення, яке задовольняє всі обмеження задачі. Для знаходження початкового плану часто використовують додаткові змінні або метод штучного базису [4].

Базисний розв’язок – розв’язок системи обмежень, в якому $(n - m)$ змінних дорівнюють нулю, а інші m змінних (базисні) визначаються однозначно.

Опорний план – базисний розв’язок, у якому всі базисні змінні невід’ємні.

Способи знаходження початкового опорного плану.

Спосіб 1. Наявність одиничної підматриці.

Якщо система обмежень містить одиничну підматрицю розміру $m \times m$, то відповідні змінні можна взяти як базисні:

Приклад (8.2):

$$\begin{aligned} x_1 + 2x_2 + x_3 &= 8, \\ 3x_1 + x_2 + x_4 &= 12, \\ 2x_1 + 3x_2 + x_5 &= 18. \end{aligned} \tag{8.2}$$

Базисні змінні: x_3, x_4, x_5 .

Початковий план: $X_0 = (0, 0, 8, 12, 18)$.

Спосіб 2. Метод штучного базису.

Даний спосіб використовується, якщо одинична підматриця відсутня. Для переходу від одного опорного плану до іншого створюється симплексна таблиця.

Стандартна форма симплексної таблиці:

Базис	x_1	x_2	...	x_n	b	Θ
x_{b1}	a_{11}	a_{12}	...	a_{1n}	b_1	b_1/a_{1k}
x_{b2}	a_{21}	a_{22}	...	a_{2n}	b_2	b_2/a_{2k}
...
x_{bm}	a_{m1}	a_{m2}	...	a_{mn}	b_m	b_m/a_{mk}
F	Δ_1	Δ_2	...	Δ_n	F_0	-

де:

- базис – стовпець базисних змінних;
- a_{ij} – коефіцієнти системи обмежень;
- b – стовпець вільних членів;
- Δ_j – оцінки (індексний рядок);
- Θ – симплексні відношення.

Наступним відбувається обчислення оцінок (індексного рядка).

Для задачі на максимум: $\Delta_j = \Sigma(c_{pi} \cdot a_{ij}) - c_j, j = 1, 2, \dots, n.$

де c_{pi} – коефіцієнти цільової функції для базисних змінних.

Для вільного члена: $F_0 = \Sigma(c_{pi} \cdot b_i).$

Алгоритм переходу до нового опорного плану складається з 5 кроків.

Крок 1. Вибір провідного стовпця. Для задачі на \max вибираємо стовпець з найбільшим додатним Δ_j . Для задачі на \min вибираємо стовпець з найбільшим від'ємним Δ_j . Індекс провідного стовпця – k .

Крок 2. Обчислення симплексних відношень: $\Theta_i = b_i/a_{ik},$ для $a_{ik} > 0.$

Крок 3. Вибір провідного рядка. Провідний рядок $r: \Theta_r = \min\{\Theta_i \mid a_{ik} > 0\},$ де елемент a_{rk} – провідний елемент.

Крок 4. Зміна базису. Змінна x_k входить в базис. Змінна $x_{\{br\}}$ виходить з базису.

Крок 5. Перетворення симплексної таблиці.

Згідно методу Жордана-Гаусса, для провідного рядку: $a'_{rj} = a_{rj}/a_{rk},$ для всіх $j, b'_r = b_r/a_{rk}.$

Інші рядки ($i \neq r$): $a'_{ij} = a_{ij} - a_{ik} \cdot a'_{rj}; b'_i = b_i - a_{ik} \cdot b'_r.$

Індексний рядок: $\Delta'_j = \Delta_j - \Delta_k \cdot a'_{rj}; F'_0 = F_0 - \Delta_k \cdot b'_r.$

Опорний план вважається оптимальним, якщо всі коефіцієнти оцінки можливості покращення у симплексній таблиці є невід'ємними для задачі на максимізацію (або неперезитивними для задачі на мінімізацію). Це означає, що більше немає можливості покращити значення цільової функції [4].

Метод штучного базису використовується тоді, коли немає очевидного початкового базисного плану. Суть методу полягає в тому, що до системи обмежень додаються фіктивні змінні, які утворюють початковий базис. Після цього, за допомогою симплекс-методу, поступово вилучають фіктивні змінні. Алгоритм методу складається з 5 кроків.

Крок 1. Модифікація цільової функції.

Для задачі на максимум: $F(X) = c_1x_1 + c_2x_2 + \dots + c_nx_n - M \cdot y_1 - M \cdot y_2 - \dots - M \cdot y_p \rightarrow \max,$ для задачі на мінімум: $F(X) = c_1x_1 + c_2x_2 + \dots + c_nx_n + M \cdot y_1 + M \cdot y_2 + \dots + M \cdot y_p \rightarrow \min,$ де M – дуже велике додатне число ($M \rightarrow +\infty$).

Крок 2. Додавання штучних змінних.

Розглянемо на прикладі.

Початкова система (8.3):

$$\begin{aligned}x_1 + 2x_2 - x_3 &= 5, \\3x_1 - x_2 &\geq 7.\end{aligned}\tag{8.3}$$

Після перетворення(8.4):

$$\begin{aligned}x_1 + 2x_2 - x_3 + y_1 &= 5, \\3x_1 - x_2 - x_4 + y_2 &= 7.\end{aligned}\tag{8.4}$$

Базис: $y_1, y_2.$

Крок 3. Формування початкової таблиці.

Обчислення модифікованих оцінок з урахуванням $M: \Delta_j = \Sigma(c_{pi} \cdot a_{ij}) - c_j.$

При цьому коефіцієнти записуються у вигляді: $A - M \cdot B.$

Крок 4. Застосування симплексного методу. Виконуються ітерації до отримання оптимального плану.

Крок 5. Аналіз результату

Задача має розв'язок:

- усі штучні змінні вийшли з базису ($y_i = 0$);
- отриманий план є оптимальним для вихідної задачі.

Задача не має розв'язку:

- хоча б одна штучна змінна залишилася в базисі ($y_i > 0$);
- система обмежень несумісна.

Властивості розв'язку задачі лінійного програмування симплекс-методом.

Монотонність – значення цільової функції на кожній ітерації не погіршується (для max – не зменшується, для min – не збільшується).

Скінченність – при відсутності виродженості алгоритм завершується за скінченне число кроків.

Оптимальність – якщо алгоритм завершився, то знайдений розв'язок є оптимальним.

Альтернативні розв'язки – якщо існує декілька оптимальних розв'язків, то їх множина опукла.

Завдання для самостійного виконання

Лінійне програмування, канонічна форма ЗЛП, допоміжні змінні, фіктивні змінні, симплекс-таблиця, розв'язковий елемент, правило чотирикутника, опорний план, індексна стрічка, симплекс перетворення, метод штучного базису.

Задача 1. Кондитерська фабрика для виготовлення трьох видів карамелі А, В і С використовує 3 види основної сировини: цукор, патоку і фруктове пюре. Норми витрат сировини кожного виду на виробництво 1т карамелі кожного виду наведені в таблиці. В ній же вказані обсяги сировини кожного виду, яка може бути використана фабрикою, а також прибуток від реалізації 1т карамелі даного виду.

Вид сировини	Норми витрат сировини (т) на 1т карамелі			Загальні запаси сировини, т
	А	В	С	
Цукор	0,8	0,5	0,6	800
Патока	0,2	0,4	0,3	600
Фруктове пюре	-	0,1	0,1	120
Прибуток від реалізації 1т продукції, грн	200	250	300	

Побудувати оптимізаційну модель і визначити такий план виробництва карамелі, який би забезпечував максимальний прибуток від її реалізації.

Задача 2. Конкуренція змушує торговельні підприємства займатися ще й випуском та реалізацією продукції власного виробництва, наприклад салатів, піци тощо. Норми витрат на виробництво різних видів піци, запаси ресурсів та вартість готової продукції наведені в таблиці.

Продукти	Норми витрат на виготовлення 100 шт. піци			Запаси продуктів, кг
	асорті	грибна	салями	
Гриби	6	7	2	40
Ковбаса	5	2	8	36
Тісто	10	8	6	50
Ціна за 100шт.,грн.	1000	800	700	

Визначити обсяги виробництва продукції кожного виду, які забезпечать підприємству отримання максимально можливого доходу (побудувати економіко-математичну модель і розв'язати її симплекс методом).

Задача 3. Фірма виготовляє і продає столи, шафи і стільці з деревини хвойних і широколистих порід. Витрати кожного виду деревини в кубометрах на одиницю виробу, запаси деревини та ціна реалізації одиниці виробу наведені в таблиці.

Виріб	Витрати сировини, м ³		Ціна виробу, грн.
	хвойні	широколисті	
Стіл	0,15	0,2	500
Шафа	0,3	0,1	800
Стілець	-	0,1	100
Запаси деревини, м ³	80	40	

Визначити оптимальну кількість продукції кожного виду, яку необхідно виготовляти і продавати для отримання максимального доходу.

Задача 4. Взуттєва фабрика може виготовляти три види взуття: чоловіче, жіноче і дитяче. На кожну пару чоловічого, жіночого і дитячого взуття відповідно необхідно клею 20, 20 і 10 г, шкіри 4, 2 і 1 дм². Вартість однієї пари чоловічого, жіночого та дитячого взуття складає 350, 400 і 200 грн. Запаси клею складають 3 т, шкіри – 4000 м². Розглянути наступні 2 операції: в першій операції всі наявні ресурси використовуються повністю, а в другій остання вимога не є обов'язковою. В обох операціях мета полягає у виборі такого плану виробництва взуття, при якому вартість виготовленої продукції буде максимальною. Знайти оптимальні стратегії в обох операціях і порівняти отримані операції.

Контрольні запитання до лабораторної роботи

1. Основні етапи побудови оптимізаційної економіко-математичної моделі.
2. З якою метою проводиться дослідження операцій в організаційних, економіко-виробничих системах?
3. Як звести задачу лінійного програмування до канонічної форми?
4. Які є форми запису задачі лінійного програмування?
5. Який розв'язок задачі лінійного програмування називається допустимим?
6. Основні аналітичні властивості розв'язків задачі лінійного програмування.
7. Для розв'язання яких математичних задач застосовують симплексний метод?
8. Суть алгоритму симплексного методу.

9. Умови оптимальності розв'язку задачі симплексним методом.
10. Як вибрати розв'язковий стовпець і розв'язковий рядок в симплекс таблиці?
11. Суть методу штучного базису.

ЛАБОРАТОРНА РОБОТА №9 ТРАНСПОРТНА ЗАДАЧА

Мета лабораторної роботи: ознайомитися з транспортною задачею та методами знаходження оптимального плану перевезень; навчитися будувати опорні плани різними методами та знаходити оптимальний розв'язок методом потенціалів.

Теоретичні відомості

Транспортна задача (ТЗ) – це задача оптимального планування перевезень однорідного вантажу від постачальників до споживачів з мінімізацією загальних витрат на транспортування.

Основні елементи задачі:

- постачальники (пункти відправлення): A_1, A_2, \dots, A_m (запаси вантажу: a_1, a_2, \dots, a_m);
- споживачі (пункти призначення): B_1, B_2, \dots, B_n (потреби у вантажі: b_1, b_2, \dots, b_n);
- тарифи перевезень: c_{ij} – вартість перевезення одиниці вантажу з A_i до B_j

Необхідно визначити: план перевезень, що забезпечує мінімальні сумарні витрати [4].

У математичній постановці невідомі: x_{ij} – кількість вантажу, що перевозиться з пункту A_i до пункту B_j ($i = 1, 2, \dots, m; j = 1, 2, \dots, n$).

Цільова функція (мінімізація витрат) (9.1):

$$F = \sum_{i=1}^m \sum_{j=1}^n c_{ij} \cdot x_{ij} \rightarrow \min. \quad (9.1)$$

Система обмежень.

Обмеження по запасах (від кожного постачальника вивозиться весь наявний вантаж) (9.2):

$$\sum_{j=1}^n x_{ij} = a_i, \quad i = 1, 2, \dots, m. \quad (9.2)$$

Обмеження по потребах (кожен споживач отримує необхідну кількість вантажу) (9.3):

$$\sum_{i=1}^m x_{ij} = b_j, \quad j = 1, 2, \dots, n. \quad (9.3)$$

Умова невід'ємності: $x_{ij} \geq 0$, для всіх i, j

Табличне подання транспортної задачі:

	B_1	B_2	...	B_n	Запаси
A_1	c_{11} x_{11}	c_{12} x_{12}	...	c_{1n} x_{1n}	a_1
A_2	c_{21} x_{21}	c_{22} x_{22}	...	c_{2n} x_{2n}	a_2
...
A_m	c_{m1} x_{m1}	c_{m2} x_{m2}	...	c_{mn} x_{mn}	a_m
Потреби	b_1	b_2	...	b_n	$\sum a_i = \sum b_j$

У кожній клітинці:

- верхній рядок (c_{ij}) – тариф;
- нижній рядок (x_{ij}) – обсяг перевезення;

Поняття закритої та відкритої транспортної задачі.

Закрита (збалансована) транспортна задача є за виконання умови, що сумарні запаси дорівнюють сумарним потребам (9.4):

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j. \quad (9.4)$$

Властивості:

- задача завжди має розв'язок;
- весь вантаж буде вивезений;
- всі потреби будуть задоволені;
- кількість базисних змінних: $m + n - 1$.

Теорема існування: закрита транспортна задача завжди має оптимальний розв'язок.

Відкритою (незбалансованою) транспортна задача вважається коли виникає надлишок запасів або надлишок потреб.

Випадок 1. Надлишок запасів (9.5):

$$\sum_{i=1}^m a_i > \sum_{j=1}^n b_j. \quad (9.5)$$

Для зведення до закритої моделі вводимо фіктивного споживача B_{n+1} з потребою (9.6):

$$b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j. \quad (9.6)$$

Тарифи до фіктивного споживача: $c_{i,n+1} = 0$ (для всіх i). Економічний зміст $x_{i,n+1}$ – залишки вантажу у постачальника A_i .

Випадок 2. Надлишок потреб (9.7):

$$\sum_{i=1}^m a_i < \sum_{j=1}^n b_j. \quad (9.7)$$

Метод зведення до закритої моделі: Вводимо фіктивного постачальника A_{m+1} із запасом (9.8):

$$a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i. \quad (9.8)$$

Тарифи від фіктивного постачальника: $c_{m+1,j} = 0$ (для всіх j). Економічний зміст $x_{m+1,j}$ – незадоволений попит споживача B_j .

Методи побудови опорного плану транспортної задачі.

Опорний план – допустимий базисний розв'язок транспортної задачі.

Вимоги до опорного плану: виконуються всі обмеження по запасах і потребах, всі $x_{ij} \geq 0$, кількість ненульових перевезень = $m + n - 1$.

Метод північно-західного кута.

Алгоритм: починаємо з лівої верхньої клітинки (північно-західний кут) таблиці [5].

Крок 1. Для поточної клітинки (i, j) : $x_{ij} = \min(a_i, b_j)$.

Крок 2. Якщо $a_i < b_j$: закриваємо рядок i , переходимо до $(i+1, j)$, то залишок потреби: $b_j := b_j - a_i$. Якщо $a_i > b_j$: закриваємо стовпець j , переходимо до $(i, j+1)$, то залишок запасу: $a_i := a_i - b_j$. Якщо $a_i = b_j$, то закриваємо рядок i стовпець j , переходимо до $(i+1, j+1)$.

Крок 3. Повторюємо до заповнення всієї таблиці.

Метод мінімального елемента (тарифу).

Крок 1. Знаходимо в таблиці мінімальний тариф c_{ij} .

Крок 2. Для клітинки з мінімальним тарифом: $x_{ij} = \min(a_i, b_j)$.

Крок 3. Якщо $a_i \leq b_j$: вилучаємо рядок i . Якщо $a_i \geq b_j$: вилучаємо стовпець j .

Крок 4: Повторюємо для залишкової таблиці.

Метод Фогеля (апроксимації). Основний принцип даного методу полягає у тому, що враховуємо не тільки мінімальні тарифи, а й штрафи за відхилення від оптимуму.

Крок 1. Для кожного рядка i стовпця обчислюємо штраф (різницю) (9.9):

$$\begin{aligned}\Delta_i &= c_i^{(2)} - c_i^{(1)} \text{ (для рядка } i\text{)}, \\ \Delta_j &= c_j^{(2)} - c_j^{(1)} \text{ (для стовпця } j\text{)}.\end{aligned}\tag{9.9}$$

де $c_i^{(1)}, c_i^{(2)}$ – два найменші тарифи в рядку (стовпці).

Крок 2. Вибираємо максимальний штраф $\max(\Delta_i, \Delta_j)$.

Крок 3. У відповідному рядку (стовпці) заповнюємо клітинку з мінімальним тарифом: $x_{ij} = \min(a_i, b_j)$.

Крок 4. Закриваємо відповідний рядок або стовпець.

Крок 5. Повторюємо для залишкової таблиці.

Метод подвійної переваги.

Крок 1. У кожному рядку позначаємо клітинку з мінімальним тарифом (одна позначка).

Крок 2. У кожному стовпці позначаємо клітинку з мінімальним тарифом (одна позначка).

Крок 3. Клітинки з двома позначками мають подвійну перевагу – заповнюємо їх першими: $x_{ij} = \min(a_i, b_j)$.

Крок 4. Заповнюємо клітинки з однією позначкою.

Крок 5. Решту клітинок заповнюємо методом мінімального тарифу.

Випадок виродженості опорного плану транспортної задачі.

Опорний план вважається невиродженим коли кількість ненульових перевезень $= m + n - 1$. Якщо кількість ненульових перевезень $< m + n - 1$ опорний план є виродженим..

Причинами виродженості може бути одночасне вичерпання запасу та задоволення потреби: $a_i = b_j$, при побудові початкового плану або утворення циклу з нульовим мінімальним перевезенням при перерахунку плану.

Метод потенціалів (МОДІ – Modified Distribution Method) – це алгоритм для знаходження оптимального плану транспортної задачі.

Крок 1. Отримання початкового опорного плану. Використовуємо один з методів (північно-західного кута, мінімального елемента, Фогеля).

Крок 2. Обчислення потенціалів:

- приймаємо $u_1 = 0$ (або будь-який інший потенціал);
- для базисних клітинок використовуємо умову: $u_i + v_j = c_{ij}$;
- послідовно знаходимо всі потенціали (9.10):

$$\begin{aligned} v_j &= c_{ij} - u_i \text{ (якщо } u_i \text{ відомо),} \\ u_i &= c_{ij} - v_j \text{ (якщо } v_j \text{ відомо).} \end{aligned} \quad (9.10)$$

Крок 3. Перевірка оптимальності, для кожної небазисної клітинки (i, j) обчислюємо оцінку: $\Delta_{ij} = c_{ij} - (u_i + v_j)$.

Критерій оптимальності:

- якщо $\Delta_{ij} \geq 0$ для всіх небазисних клітинок \rightarrow план оптимальний;
- якщо існує $\Delta_{ij} < 0 \rightarrow$ план можна покращити.

Крок 4. Побудова циклу перерахунку.

Вибираємо клітинку (k, l) з найменшою оцінкою: $\Delta_{kl} = \min \{ \Delta_{ij} < 0 \}$. Будуємо цикл – замкнений ламаний шлях:

- початок у клітинці (k, l) ;
- вершини тільки в базисних клітинках;
- повертається в (k, l) ;
- сторони паралельні осям таблиці.

Позначаємо вершини циклу знаками «+» та «-», починаючи з «+»:

- (k, l) : «+»;
- наступні вершини чергуються («-», «+», «-»,...). .

Крок 5. Перерахунок плану.

Знаходимо мінімальне перевезення серед клітинок зі знаком «-»: $\theta = \min \{ x_{ij} \mid \text{клітинка } (i, j) \text{ має знак «-»} \}$

Перераховуємо план:

- клітинки зі знаком «+»: $x'_{ij} = x_{ij} + \theta$;
- клітинки зі знаком «-»: $x'_{ij} = x_{ij} - \theta$;
- інші клітинки: без змін.

Обчислюємо нове значення F: $F' = F + \theta \cdot \Delta_{kl}$.

Крок 6: Повертаємося до кроку 2.

Завдання для самостійного виконання

Транспортна задача, план перевезень, матриця тарифів, метод потенціалів, метод північно-західного кута, метод мінімальної вартості, невироджений план, оптимальний план.

Задача 1. Чотири підприємства даного економічного району для виробництва продукції використовують три види сировини. Потреби в сировині кожного з підприємств відповідно дорівнюють 110, 70, 180 і 120 од. Сировина зосереджена в трьох місцях її отримання, а запаси відповідно дорівнюють 100, 130, 160 од. На кожне з підприємств сировина може завозитися з будь-якого пункту її отримання. Тарифи перевезень є відомими величинами і задаються матрицею C

Задача 2. На чотирьох складах компанії «Елітбуд» є цемент марки «400» в кількості відповідно 200, 150, 80 і 120 т. Цей цемент повинен бути доставлений на три бетонних заводи, відповідно в кількості 280, 90 і 180 т.:

- 1) побудувати математичну модель задачі;
- 2) сформулювати базове рішення, використовуючи метод північно-західного кута, визначити сукупні транспортні витрати за початкового плану перевезень;
- 3) сформулювати базове рішення, використовуючи метод найменшої вартості, визначити сукупні транспортні витрати за такого плану перевезень

продукції;

4) на основі плану перевезень, який забезпечує отримання менших транспортних витрат побудувати схему перевезення продукції за поліпшеним планом;

5) визначити план перевезення продукції, який забезпечує отримання мінімальної вартості транспортних витрат; 6) обґрунтувати отримані результати.

Вартість перевезення 1 т цементу зі складів до бетонних заводів, гр.од.

склад, i	завод, j		
1	1	5	3
2	6	8	9
3	2	7	4
4	4	1	11

Задача 3. П'ять учасників будівництва траси отримують пісок з двох кар'єрів. Обсяг виробництва піску в кар'єрах відповідно 160 і 220 м³. на добу. Додаткова потреба в піску учасників будівництва доріг 50, 50, 80, 100 і 100 м³. Відстань перевезень піску (в км) представлена в таблиці:

j \ i		1	2	3	4	5
		50	50	80	100	100
1	160	5	7	4	3	2
2	220	6	9	10	1	3

- встановити тип задачі;
- скласти табличну схему задачі;
- скласти математичну модель задачі;
- здійснити базове вирішення задачі;
- скласти вантажообіг поліпшеного плану;
- скласти план перевезень піску, що забезпечує найменший вантажообіг;
- обґрунтувати отримані результати.

Контрольні запитання до лабораторної роботи

1. Правила зведення відкритої транспортної задачі до закритої типу.
2. Критерій оптимальності опорного плану поставок.
3. Що таке опорний план транспортної задачі? Методи визначення початкового опорного плану.
4. Скільки елементів повинно бути в опорному плані? За якої умови транспортна задача називається виродженою?
5. Суть методу потенціалів розв'язування транспортної задачі.
6. За яких умов наявний план перевезення буде оптимальним?

ЛАБОРАТОРНА РОБОТА № 10

ОПТИМІЗАЦІЙНІ МЕТОДИ ТА МОДЕЛІ НЕЛІНІЙНОГО ПРОГРАМУВАННЯ

Мета роботи: систематизація знань про оптимізаційні методи і моделі нелінійного програмування та набуття навичок розв'язання оптимізаційних нелінійних задач методом множників Лагранжа, у тому числі в середовищі MS Excel

Теоретичні відомості

Побудова економіко-математичних моделей задач, розглянутих у першому модулі, ґрунтувалася на припущенні про лінійні залежності між показниками, що відображались лінійною цільовою функцією і лінійними обмеженнями. Водночас зв'язки між економічними показниками і факторами, які впливають на критерій оптимальності й обмеження, можуть бути нелінійними [5].

Загальну задачу математичного програмування можна сформулювати так: знайти такий план $X = (x_1, x_2, \dots, x_n)$, щоб цільова функція набувала оптимального (максимального чи мінімального) значення (10.1):

$$Z = f(x_1, x_2, \dots, x_n) \rightarrow \max(\min), \quad (10.1)$$

за обмежень (10.2):

$$x_j \geq 0, \quad j = 1; s, \quad s < n. \quad (10.2)$$

Якщо всі функції $f(x_1, x_2, \dots, x_n)$ та $q_i(x_1, x_2, \dots, x_n)$ ($i = 1; m$) лінійні, то задача – задача лінійного програмування; якщо хоча б одна з функцій нелінійна – це задача нелінійного програмування.

Особливості задач нелінійного програмування:

- не існує універсального методу розв'язання;
- існують кілька локальних оптимумів, що вимагає пошуку глобального оптимуму серед них;
- точка оптимуму не завжди є граничною, а може знаходитися й усередині допустимої області розв'язків;
- множина допустимих планів не завжди опукла, а може бути неопуклою або навіть складатися з незв'язаних між собою частин.

Задачі відшукування локального або глобального екстремуму функції за умови, що на змінні функції накладаються певні обмеження, називаються задачами на умовний екстремум функції.

Франко-італійський математик Жозеф-Луї Лагранж запропонував метод розв'язання задач на умовний екстремум – метод множників Лагранжа. Суть методу полягає у переході від задачі на умовний екстремум до задачі на безумовний екстремум шляхом заміни цільової функції функцією Лагранжа [5].

Алгоритм методу множників Лагранжа.

Крок 1. Побудова економіко-математичної моделі задачі

Нехай є задача у канонічній формі (10.3):

$$Z = f(x_1, x_2, \dots, x_n) \rightarrow \max(\min), \quad (10.3)$$

за обмежень (10.4):

$$\begin{aligned} q_i(x_1, x_2, \dots, x_n) &= b_i, \quad i = 1; m, \\ x_j &\geq 0, \quad j = 1; n, \end{aligned} \quad (10.4)$$

де функції f та q_i ($i = 1; m$) мають бути диференційованими.

Крок 2. Перехід до задачі безумовного екстремуму

Цільова функція (10.3) замінюється функцією Лагранжа (10.5):

$$L(x_1, x_2, \dots, x_n; \lambda_1, \lambda_2, \dots, \lambda_m) = f(x_1, \dots, x_n) + \sum \lambda_i (b_i - q_i(x_1, \dots, x_n)), \quad (10.5)$$

де λ_i – множники Лагранжа (невідомі змінні).

Крок 3. Визначення стаціонарних точок

Стаціонарні точки функції Лагранжа визначаються з необхідних умов існування екстремуму: знаходимо часткові похідні, прирівнюємо їх до нуля та розв'язуємо отриману систему.

Крок 4. Діагностування стаціонарних точок

Перевірка здійснюється за достатньою умовою екстремуму за допомогою матриці Гессе розмірністю $(m + n) \times (m + n)$.

Правила ідентифікації типу екстремуму.

Точка X^* є точкою максимуму, якщо, починаючи з головного мінору порядку $(m+1)$, наступні $(n-m)$ головних мінорів матриці Гессе утворюють знакозмінний ряд, знак першого члена якого визначається множителем $(-1)^{(m+1)}$.

Точка X^* є точкою мінімуму, якщо, починаючи з головного мінору порядку $(m+1)$, знак наступних $(n-m)$ головних мінорів матриці Гессе визначається множителем $(-1)^m$.

Розв'язання типових задач

Задача 1. Планування структури реалізації готової продукції.

Борошномельний завод реалізує борошно 2-ма способами: у роздріб через магазин і гуртом через торгових агентів. У разі продажу x_1 кг борошна через магазин видатки на реалізацію складають x_1^2 грн, а за продажу x_2 кг борошна гуртом – x_2^2 грн. Необхідно розробити план реалізації борошна, який забезпечує мінімальні витрати, якщо добові потужності заводу 5000 кг борошна.

Розв'язання

Економіко-математична модель (10.6):

$$\begin{aligned} Z &= x_1^2 + x_2^2 \rightarrow \min, \\ x_1 + x_2 &= 5000, \\ x_1 &\geq 0, \quad x_2 \geq 0. \end{aligned} \quad (10.6)$$

Функція Лагранжа (10.7):

$$L(x_1, x_2, \lambda) = x_1^2 + x_2^2 + \lambda(5000 - x_1 - x_2). \quad (10.7)$$

Стаціонарні точки (система рівнянь) (10.8):

$$\begin{aligned} \partial L / \partial x_1 &= 2x_1 - \lambda = 0, \\ \partial L / \partial x_2 &= 2x_2 - \lambda = 0, \\ \partial L / \partial \lambda &= 5000 - x_1 - x_2 = 0. \end{aligned} \quad (10.8)$$

Розв'язавши систему: $x_1^* = x_2^* = 2500$, $\lambda^* = 5000$.

Розв'язання в MS Excel (надбудова Пошук рішення):

Модель задачі нелінійна, тому у діалоговому вікні «Параметри пошуку рішення» у полі «Оберіть метод рішення» необхідно вибрати Пошук рішення нелінійних задач методом ОПГ (метод зведеного градієнта).

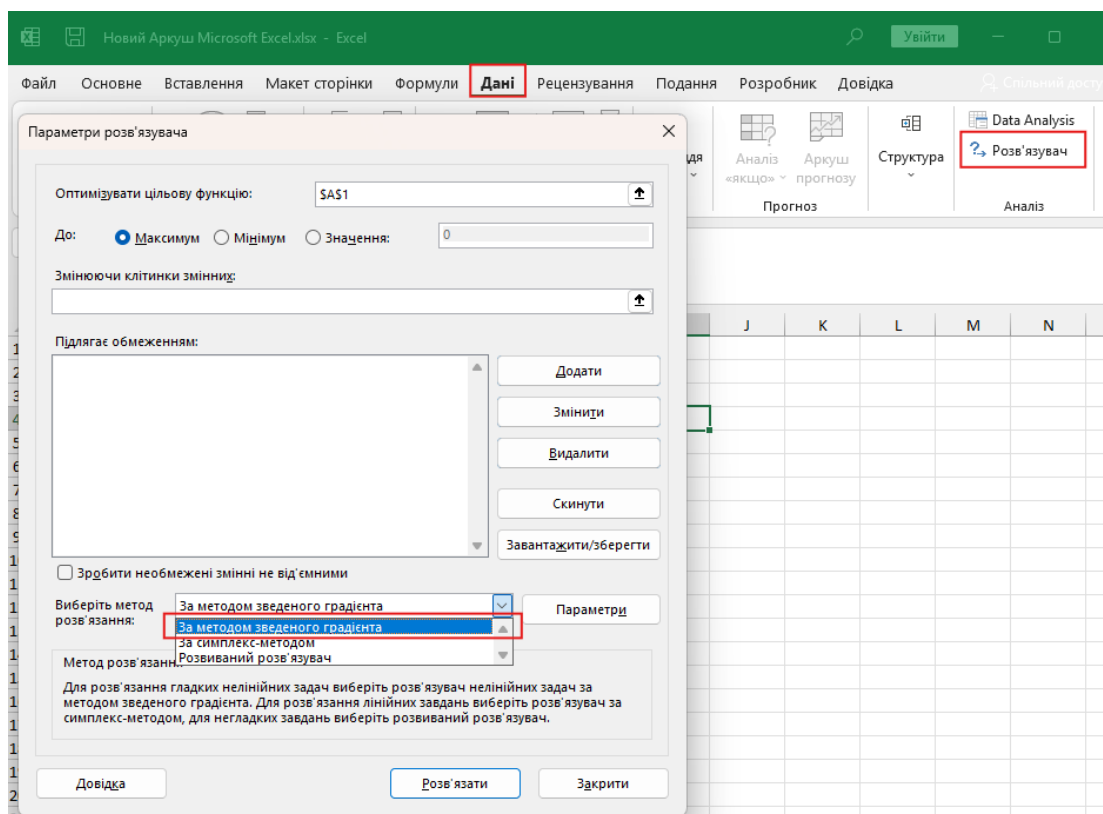


Рисунок 10.1 – Надбудова «Пошук рішення»

Задача 2. Формування портфеля цінних паперів.

Інвестор доручив брокерській конторі купити на 1 млн. грн акції трьох компаній. Угода укладається на рік. Інвестор зацікавлений у максимізації середнього прибутку та мінімізації ризику. Ризик враховується з вагою втричі більшою, ніж прибуток.

Фінансові аналітики прогнозують:

Показник	Компанія	Значення
Середній річний прибуток (μ_1, μ_2, μ_3)	1, 2, 3	8%, 10%, 13%
Дисперсія ($\sigma_1^2, \sigma_2^2, \sigma_3^2$)	1, 2, 3	0.1, 0.15, 0.19
Коваріація σ_{12}	1–2	0.01
Коваріація σ_{13}	1–3	0.02
Коваріація σ_{23}	2–3	0.03

Економіко-математична модель (10.9):

$$\begin{aligned}
 Z(x_1, x_2, x_3) = & 0,08x_1 + 0,1x_2 + 0,13x_3 - \\
 & - 0,3 \cdot x_1^2 - 0,45 \cdot x_2^2 - 0,57 \cdot x_3^2 - \\
 & - 0,06 \cdot x_1x_2 - 0,12 \cdot x_1x_3 - 0,18 \cdot x_2x_3 \rightarrow \max, \\
 & x_1 + x_2 + x_3 = 1000, \\
 & x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.
 \end{aligned}
 \tag{10.9}$$

Функція Лагранжа (10.10):

$$\begin{aligned}
 L(x_1, x_2, x_3, \lambda) = & 0,08x_1 + 0,1x_2 + 0,13x_3 - \\
 & - 0,3 \cdot x_1^2 - 0,45 \cdot x_2^2 - 0,57 \cdot x_3^2 - 0,06 \cdot x_1x_2 - 0,12 \cdot x_1x_3 - 0,18 \cdot x_2x_3 + \\
 & + \lambda(1000 - x_1 - x_2 - x_3).
 \end{aligned}
 \tag{10.10}$$

Система рівнянь для стаціонарних точок (10.11):

$$\begin{aligned}
 \partial L / \partial x_1 = & 0,08 - 0,6x_1 - 0,06x_2 - 0,12x_3 - \lambda = 0, \\
 \partial L / \partial x_2 = & 0,1 - 0,9x_2 - 0,06x_1 - 0,18x_3 - \lambda = 0, \\
 \partial L / \partial x_3 = & 0,13 - 1,14x_3 - 0,12x_1 - 0,18x_2 - \lambda = 0, \\
 \partial L / \partial \lambda = & 1000 - x_1 - x_2 - x_3 = 0.
 \end{aligned}
 \tag{10.11}$$

Координати стаціонарної точки:

Змінна	Значення (тис. грн)
x_1^*	496,8088
x_2^*	305,2632
x_3^*	197,9281
λ^*	-340,072

Матриця Гессе та тип екстремуму. Оскільки у задачі $m = 1$ обмеження, $n = 3$ змінні, то матриця Гессе має розмірність $(m + n) \times (m + n) = 4 \times 4$.

Головні мінори утворюють знакозмінний ряд та, починаючи з головного мінору 2-го порядку, наступний мінор визначається знаком $(-1)^{(m+1)} = (-1)^2$, тобто X^* є точкою максимуму.

$Z_{\max} = 169\,988$ тис. грн – глобальний максимум (функція строго увігнута).

Економічний аналіз розв'язку. За реалізації найбільш песимістичного сценарію, коли ризик втрати з вагою втричі більший за інвестиційний прибуток, максимально можливе значення критерію ризику портфеля $-169\,988$ тис. грн забезпечує такий план інвестування:

- в акції 1-го типу – 496,809 тис. грн;
- в акції 2-го типу – 305,263 тис. грн;
- в акції 3-го типу – 197,928 тис. грн.

Двоїста оцінка обмеження $\lambda = 340,073$, тому за збільшення суми, яка інвестується, на 0,2% (тобто на 2 тис. грн), значення критерію зменшиться на 681 тис. грн.

Завдання для самостійного виконання

За планом виробництва продукції підприємству необхідно виготовити n виробів. Ці вироби можуть бути зроблені двома технологічними способами, відомі витрати на виробництво x_1 виробів першим способом і x_2 виробів другим способом (значення за варіантами наведено у табл. 10.1). Необхідно визначити оптимальне число виробів, що виготовляються різними способами, так щоб загальні витрати на виробництво продукції були мінімальними. Обчислити економію, одержувану від такого розв'язку задачі оптимізації.

Вказівка. Побудувати економіко-математичну модель задачі і знайти оптимальний план методом множників Лагранжа.

Таблиця 10.1 – Оптимізаційні методи та моделі нелінійного програмування

Варіант	Виробничі витрати, грн		Загальний обсяг виробництва, шт.
	I спосіб	II спосіб	
1	$x_1^2+2x_1$	$x_2^2+16x_2$	100
2	$x_1^2+25x_1$	$x_2^2+5x_2$	150
3	$x_1^2+2x_1$	$x_2^2+10x_2$	68
4	$x_1^2+2x_1$	$x_2^2+22x_2$	75
5	$x_1^2+3x_1$	$x_2^2+21x_2$	50
6	$x_1^2+4x_1$	$x_2^2+24x_2$	90
7	$x_1^2+3x_1$	$x_2^2+12x_2$	37
8	$x_1^2+15x_1$	$x_2^2+5x_2$	45
9	$x_1^2+3x_1$	$x_2^2+9x_2$	90
10	$x_1^2+9x_1$	$x_2^2+27x_2$	38
11	$x_1^2+2x_1$	$x_2^2+14x_2$	137
12	$x_1^2+4x_1$	$x_2^2+16x_2$	77
13	$x_1^2+2x_1$	$x_2^2+10x_2$	44
14	$x_1^2+12x_1$	$x_2^2+2x_2$	83
15	$x_1^2+4x_1$	$x_2^2+20x_2$	150
16	$x_1^2+3x_1$	$x_2^2+12x_2$	100
17	$x_1^2+2x_1$	$x_2^2+18x_2$	90
18	$x_1^2+5x_1$	$x_2^2+25x_2$	155
19	$x_1^2+6x_1$	$x_2^2+18x_2$	130
20	$x_1^2+21x_1$	$x_2^2+7x_2$	75
21	$x_1^2+12x_1$	$x_2^2+3x_2$	56
22	$x_1^2+4x_1$	$x_2^2+24x_2$	97
23	$x_1^2+6x_1$	$x_2^2+30x_2$	95
24	$x_1^2+14x_1$	$x_2^2+2x_2$	138
25	$x_1^2+5x_1$	$x_2^2+25x_2$	140

Контрольні запитання до лабораторної роботи

1. Дайте означення задачі нелінійного програмування.
2. Чи існує універсальний метод розв'язання задач нелінійного програмування? Обґрунтуйте свою відповідь.
3. У якому випадку задача називається задачею на пошук умовного екстремуму?
4. У чому полягає суть методу множників Лагранжа?
5. З яких кроків складається алгоритм методу множників Лагранжа?
6. Як будується функція Лагранжа?
7. Як визначаються стаціонарні точки функції Лагранжа?

8. З якою метою будується матриця Гессе?
9. Як ідентифікується тип екстремуму функції?
10. Множники Лагранжа мають економічний зміст чи це суто технічні коефіцієнти?
11. У чому полягає ключова відмінність розв'язання задач лінійного і нелінійного програмування з використанням надбудови Пошук рішення?
12. Який звіт надбудови Пошук рішення містить значення множників Лагранжа?

ЛАБОРАТОРНА РОБОТА № 11 ЦІЛОЧИСЕЛЬНЕ ПРОГРАМУВАННЯ

Мета лабораторної роботи: засвоїти метод ланцюгів і границь для розв'язання задач цілочисельного лінійного програмування.

Теоретичні відомості

Часто в задачі лінійного програмування (ЗЛП) потрібно знайти оптимум цільової функції (11.1):

$$L = c_1x_1 + c_2x_2 + \dots + c_nx_n \Rightarrow \max, \quad (11.1)$$

за обмежень (11.2):

$$\begin{aligned} a_{i1}x_1 + \dots + a_{in}x_n &\leq b_i, \quad \forall i = 1, \bar{n}, \\ x_j &\geq 0, \quad \forall j = 1, \bar{n}. \end{aligned} \quad (11.2)$$

потрібно одержати розв'язок, у якому деякі або всі компоненти мають бути цілими числами. Для цього використовують метод ланцюгів і границь.

Алгоритм методу ланцюгів і границь. Розв'язуємо ЗЛП (11.1), (11.2) за допомогою симплекс-методу без умови цілочисельності змінних. Якщо змінні – цілі числа, задача вважається розв'язаною. Нехай змінна x_k набула не цілого значення $x_k = \alpha_k$, де α_k має дробову складову.

Розв'язуємо дві підзадачі:

а) задача (11.1), (11.2) за умови $x_k \geq [\alpha_k] + 1$;

б) задача (11.1), (11.2) за умови $x_k \leq [\alpha_k]$, де значок $[\cdot]$ означає цілу частину числа, що в ньому міститься.

У випадку цілих розв'язків задач а) і б) порівнюємо одержані значення функцій L . Більше з них – оптимальне значення $L = L^{\text{opt}}$, а змінні, за яких воно досягається, – розв'язок задачі [6].

Якщо знайдеться таке x_1 , що не відповідає умові цілочисельності, повторюємо п. 2, замінивши x_k на x_1 . Процедуру повторюємо доти, доки всі потрібні змінні не стануть цілими.

Приклад розв'язання ЗЛП в цілих числах (11.3):

$$\begin{aligned} L = 2x_1 + 3x_2 &\Rightarrow \max, \\ x_1 + 4x_2 &\leq 14, \\ 2x_1 + 3x_2 &\leq 12, \\ x_1, x_2 &\geq 0; \quad x_1, x_2 - \text{цілі числа.} \end{aligned} \quad (11.3)$$

Крок 1. Розв'язання без умови цілочисельності

Виконуємо п. 1 відповідно до симплекс-процедури розподілу:

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3

	b	x₁	y₁
L'	-21/2	5/4	-3/4
x ₂	14/4	1/4	1/4
y ₂	3/2	5/4	-3/4

Другий крок:

	b	y₂	y₁
L'	-12	-1	0
x ₂	16/5	-1/5	3/5
x ₁	3/2 ÷ 5/4	4/5	-3/4 ÷ 5/4

Розв'язок досягається при $x_1 = 6/5$, $x_2 = 16/5$. Оскільки x_1 та x_2 – не цілі числа, переходимо до виконання п. 2.

Крок 2а.

Підзадача а): $x_2 \geq [16/5] + 1 = 4$, $x_1 \geq 0$.

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3
y ₃	-4	0	-1

	b	x₁	y₁
L'	-21/2	5/4	-3/4
x ₂	14/4	1/4	1/4
y ₂	3/2	5/4	-3/4
y ₃	-1/2	1/4	1/4

Оскільки в рядку, де стоїть від'ємний елемент, немає від'ємних чисел, задача розв'язку не має, допустима область порожня. Це означає: при $x_2 \geq 4$ розв'язку даної задачі не існує.

Крок 2б. Підзадача б): $x_2 \leq [16/5] = 3$

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3
y ₃	3	0	1

	b	x₁	y₃
L'	-9	2	-3
y ₁	2	1	-4
y ₂	3	2	-3
x ₂	3	0	1

	b	y₂	y₃
L'	-12	-1	0
y ₁	1/2	-1/2	-5/2
x ₁	3/2	1/2	-3/2
x ₂	3	0	1

Відповідь: $L = 11$, $x_1 = 1$, $x_2 = 3$.

Крок 3. Задача б) – другий рівень ($x_1 \leq 1$).

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3
y ₃	-2	-1	0
y ₄	3	0	1

	b	y₃	x₂
L'	-4	2	3
x ₂	12	1	4
y ₂	8	2	3
x ₁	2	-1	0
y ₄	3	0	1

	b	y₃	y₂
L'	-12	-6	-1
y ₁	4/3	-2/3	-4/3
x ₂	8/3	2/3	1/3
x ₁	2	1	0
y ₄	1/3	-2/3	-1/3

Відповідь: $L^{\text{opt}} = 12$, $x_1^{\text{opt}} = 2$, $x_2^{\text{opt}} = 8/3$.

Задача знову розпадається на дві підзадачі:

а) $x_1 \geq 2$, $x_2 \leq 2$;

б) $x_1 \geq 2$, $x_2 \geq 3$.

Крок 4а. Підзадача а): $x_1 \geq 2$, $x_2 \leq 2$.

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3
y ₃	-2	-1	0
y ₄	2	0	1

	b	y₃	x₂
L'	-4	2	3
x ₂	12	1	4
y ₂	8	2	3
x ₁	2	-1	0
y ₄	2	0	1

	b	y₃	y₄
L'	-10	2	-3
y ₁	4	1	-4
y ₂	2	2	-3
x ₁	2	-1	0
x ₂	-2	0	1

	b	y₂	y₄
L'	-12	-1	0
y ₁	3	-1/2	-5/2
y ₃	2/2	1/2	-3/2
x ₁	3	1/2	-3/2
x ₂	2	0	1

Відповідь: $L = 12$, $x_1 = 3$, $x_2 = 2$.

Крок 4б. Підзадача б): $x_1 \geq 2$, $x_2 \geq 3$.

	b	x₁	x₂
L'	0	2	3
y ₁	14	1	4
y ₂	12	2	3
y ₃	-2	-1	0
y ₄	-3	0	-1

	b	x₁	y₄
L'	-9	2	3
y ₁	0	1	4
y ₂	3	2	3
y ₃	-2	-1	0
x ₂	3	0	1

	b	y₁	y₄
L'	-9	-2	5
x ₁	0	1	4
y ₂	3	-2	-5
y ₃	-2	1	4
x ₂	3	0	-1

Відповідь: задача розв'язку не має. Область порожня.

Загальний результат (порівняння всіх розглянутих випадків) (11.4):

$$L_{\text{opt}} = 12, \quad x_{1\text{opt}} = 3, \quad x_{2\text{opt}} = 2. \quad (11.4)$$

Завдання для самостійного виконання

Розв'язати задачі у цілих числах або довести, що вони не мають розв'язку.

1	$F = 3x_1 - 2x_2 \rightarrow \max;$ $2x_1 + x_2 \leq 11,$ $-3x_1 + 2x_2 \leq 10,$ $3x_1 + 4x_2 \geq 20,$ $x_1 \geq 0, \quad x_2 \geq 0.$	2	$F = 2x_1 + 2x_2 \rightarrow \max;$ $3x_1 - 2x_2 \geq -6,$ $x_1 + x_2 \geq 3,$ $0 \leq x_1 \leq 3,$ $0 \leq x_2 \leq 5.$
3	$F = 3x_1 + 3x_2 \rightarrow \max;$ $x_1 + x_2 \leq 4,$ $3x_1 + x_2 \geq 4,$ $x_1 + 5x_2 \geq 4,$ $0 \leq x_1 \leq 3,$ $0 \leq x_2 \leq 3.$	4	$F = 2x_1 - x_2 \rightarrow \max;$ $x_1 - 2x_2 \geq 4,$ $5x_1 + 2x_2 \geq 10,$ $4x_1 - 3x_2 \leq 12,$ $7x_1 + 4x_2 \leq 28,$ $x_1, x_2 \geq 0.$
5	$F = x_1 + 2x_2 \rightarrow \max;$ $5x_1 - 2x_2 \leq 4,$ $x_1 - 2x_2 \geq -4,$ $x_1 + x_2 \geq 4,$ $x_1, x_2 \geq 0.$	6	$F = -3x_1 + 2x_2 \rightarrow \min;$ $x_1 + 2x_2 \geq 10,$ $3x_1 + x_2 \geq 15,$ $x_1, x_2 \geq 0.$
7	$F = 2x_1 - 4x_2 \rightarrow \min;$ $8x_1 - 5x_2 \leq 16,$ $x_1 + 3x_2 \geq 2,$ $2x_1 + 7x_2 \leq 9,$ $x_1, x_2 \geq 0.$	8	$F = -2x_1 + x_2 \rightarrow \max;$ $2x_1 + x_2 \leq 8,$ $x_1 + x_2 \leq 6,$ $-3x_1 + 2x_2 \geq 3,$ $x_1, x_2 \geq 0.$
9	$F = 2x_1 - 4x_2 \rightarrow \max;$ $8x_1 - 5x_2 \leq 16,$ $x_1 + 3x_2 \leq 2,$ $2x_1 + 7x_2 \geq 9,$ $x_1, x_2 \geq 0.$	10	$F = x_1 + 2x_2 \rightarrow \max;$ $5x_1 - 2x_2 \leq 4,$ $x_1 - 2x_2 \geq -4,$ $x_1 + x_2 \geq 4,$ $x_1, x_2 \geq 0.$
11	$F = 3x_1 + 3x_2 \rightarrow \max;$ $x_1 - 4x_2 \leq 4,$ $3x_1 + 2x_2 \leq 6,$ $-x_1 + x_2 \leq 1,$ $x_1 + 2x_2 \geq 2,$ $x_1, x_2 \geq 0.$	12	$F = 7x_1 - 2x_2 \rightarrow \max;$ $5x_1 - 2x_2 \leq 3,$ $x_1 + x_2 \geq 1,$ $-3x_1 + x_2 \leq 3,$ $2x_1 + x_2 \leq 4,$ $x_1, x_2 \geq 0.$
13	$F = -2x_1 + x_2 \rightarrow \min;$ $2x_1 + x_2 \leq 8,$ $x_1 + 3x_2 \geq 6,$ $3x_1 + x_2 \geq 3,$ $x_1, x_2 \geq 0.$	14	$F = x_1 + 2x_2 \rightarrow \max;$ $2x_1 + x_2 \leq 14,$ $-3x_1 + 2x_2 \leq 9,$ $3x_1 + 4x_2 \leq 27,$ $x_1, x_2 \geq 0.$
15	$F = x_1 + 3x_2 \rightarrow \max;$	16	$F = -3x_1 + 2x_2 \rightarrow \min;$

$$\begin{aligned} -x_1 + x_2 &\leq 3, \\ 4x_1 + 3x_2 &\geq 20, \\ x_1, x_2 &\geq 0. \end{aligned}$$

$$\begin{aligned} 2x_1 + 3x_2 &\geq 15, \\ x_1 + 2x_2 &\geq 10, \\ x_1, x_2 &\geq 0. \end{aligned}$$

Контрольні запитання

1. Що таке задача цілочисельного лінійного програмування (ЦЗЛП)?
2. У чому полягає суть методу ланцюгів і границь для розв'язання ЦЗЛП?
3. Опишіть схему розгалуження при розв'язанні ЦЗЛП.
4. Коли підзадача у методі ланцюгів і границь відкидається?
5. Що означає «порожня допустима область» у контексті ЦЗЛП?
6. Як визначити оптимальний цілочисельний розв'язок після повного розгалуження?

Після розставлення чинників роблять оцінку рангу. Сума рангів усіх чинників повинна дорівнювати одиниці, тобто сума по стовпцю 3 має дорівнювати одиниці.

Далі усі рішення оцінюються по кожному з чинників (критеріїв) оцінки. З цією метою кожному чиннику виставляються бали (гр. 4-8).

Максимальний бал по будь-якому з чинників для рішення дорівнює 100, мінімальний – 0.

Після перемножування вагових коефіцієнтів кожного чинника (ст.3) і значення оцінки по кожному з рішень (ст.4-8) отримуємо інтегральну експертну оцінку, яку записуємо в ст. 9-13. інтегральна експертна оцінка пріоритетності варіантів рішення визначається як сума по ст. 9-13.

Цей метод можна застосовувати для попереднього відбору найбільш перспективних варіантів рішень. Для подальшого розгляду залишаються альтернативи, що отримали найвищі результати. Недоліком розглянутого методу є те, що в ньому не відбивається, як враховуються думки кожного експерта. Наприклад, на першому етапі при виборі чинників, які можуть значною мірою вплинути на вибір рішення, або на другому, коли ці чинники ранжуються і так далі. Цей недолік усувається за допомогою методу «збалансоване голосування» (таблиця 12.2).

Таблиця 12.2 – Форма для збалансованого голосування

	Рішення №1	Рішення №2	...	Рішення №n
Експерт №1				
Експерт №2				
Експерт №3				
...				
Всього				

Дозволяє за відсутності досить точних цифрових даних вибрати серед декількох рішень (чинників) те, яке здається найбільш важливим або таким, що краще всього адаптується до ситуації. Цей засіб, зокрема, корисно для групи експертів, де всі члени мають різні думки. При збалансованому голосуванні кожен позитивно вносить свій вклад у вибір рішення, що не дає приводу для дискусії.

Суть системи полягає в тому, що кожен експерт має в розпорядженні кількістю балів, які відповідають кількості рішень. Необхідно розподілити бали між рішеннями залежно від думки експерта: максимальний бал присвоюється найкращому рішенню і так далі. Перевагою є те, що кожен учасник бере участь в голосуванні і у виборі рішень на відміну від голосування «так-ні», що породжує незадоволення у учасників, вибір яких не підтримується.

Потім підраховують суму балів по кожному рішенню. Це дозволяє кількісно оцінити рішення (чинники), визначити його пріоритет, тобто ранжувати усі рішення [6].

Метод «Вибір по багатьом критеріям» вирішує ті ж завдання, що і у разі збалансованого голосування, але вибір за багатьма критеріями дозволяє, крім того, виявити ефективність рішення, виходячи з важливості критеріїв, які визначаються групою експертів (таблиця 12.3).

Група проводить збалансоване голосування, щоб вибрати 3 з усіх запропонованих критеріїв для аналізу рішень. Залежно від отриманих сум, критерії ранжуються. Перший вибирають критерії з максимальною сумою і записують до графі 1. Потім знову проводять збалансоване голосування по кожному з рішень, але з урахуванням обраних критеріїв. Отриману часткову суму множать на коефіцієнт ваговитості відповідного критерію.

Таблиця 12.3 – Форма для засобу «Вибір за багатьма критеріями»

		Рішення №1	Рішення №2	...	Рішення №n
Критерій №1	Експерт №1				
	Експерт №2				
	...				
	Експерт №n				
	$3 \times \sum$				
Критерій №2	Експерт №1				
	Експерт №2				
	...				
	Експерт №n				
	$2 \times \sum$				
Критерій №3	Експерт №1				
	Експерт №2				
	...				
	Експерт №n				
	$1 \times \sum$				
Разом					

Надалі обчислюються підсумкові суми по кожному рішенню, що дозволяє визначити пріоритетність того або іншого рішення.

Запропоновані системи дозволяють залучити в процес ухвалення рішення широкий круг експертів. При цьому зважає на думку кожного з них, яке не протиставляється іншим думкам. Це підвищує якість прийнятого рішення, особливо за відсутності достатньої кількості точних цифрових даних.

Розглянемо організацію роботи групи експертів і обробки результатів групової експертизи на прикладі вибору автомобіля.

Завдання для самостійного виконання

- отримати у викладача індивідуальний варіант початкових даних;
- провести експертне оцінювання альтернатив свого варіанту завдання;
- результати роботи оформити у вигляді таблиці;
- після закінчення роботи скласти індивідуальний звіт.

Контрольні запитання до лабораторної роботи

1. Що таке метод експертних оцінок? Яка його роль у прийнятті рішень?
2. Чим відрізняються індивідуальні та колективні експертні оцінки?
3. Охарактеризуйте метод збалансованого голосування.
4. Що таке метод «Вибір за багатьма критеріями»? Яка його відмінність від збалансованого голосування?

5. Як здійснюється обробка результатів групової експертизи?
6. Якими є переваги групового експертного оцінювання порівняно з індивідуальним?

ЛАБОРАТОРНА РОБОТА №13 БАГАТОКРИТЕРІАЛЬНИЙ ВИБІР АВТОМОБІЛЯ

Мета лабораторної роботи: вивчення методів рішення багатокритеріальних задач прийняття рішень. Отримання навичок вибору рішень в різних ситуаціях інформованості ОПР про важливість критеріїв.

Теоретичні відомості

Для забезпечення якості рішень, що приймаються, при проектуванні різноманітних систем необхідно вибрати і обґрунтувати методи їх багатокритеріальної оцінки і оптимізації.

Найбільш поширений підхід до рішення багатокритеріальної задачі – її зведення до однокритеріальної. Основою такого підходу є теорія корисності, згідно якої передбачається, що існує деяка узагальнена оцінка цінності або корисності будь-якого рішення $x \in X$. В цьому випадку формування схеми компромісу пов'язане з вибором виду функції корисності [7] (13.1).

$$R = R[k_1(x), k_2(x), \dots, k_j(x)]. \quad (13.1)$$

Для вирішення задачі потрібно обґрунтувати вид функції корисності часткових критеріїв $R_j(k_j)$, $j = \underline{1, J}$, яка була б універсальною і відображала особливості конкретних систем, їх цілей і критеріїв.

Функція корисності часткових критеріїв повинна задовольняти наступним умовам: мати єдиний інтервал змін $[0,1]$; бути безрозмірною, інваріантною до виду екстремуму часткового критерію (min і max), тобто найкращому значенню відповідає 1, а найгіршому – 0. Вказаним вимогам задовольняє функція виду (13.2):

$$R_j(k_j) = \frac{k_j - k_{j\text{НГ}}}{k_{j\text{НК}} - k_{j\text{НГ}}}, j = \underline{1, J}, \quad (13.2)$$

де k_j , $k_{j\text{НГ}}$, $k_{j\text{НК}}$ – поточне, найгірше (гранично допустиме) і найкраще значення j -го приватного критерію, що відповідають межах області його зміни (наближеній області компромісів).

Функція корисності (13.2) характеризує міру наближення до локального оптимуму за критерієм k_j . Іноді зручніше використовувати поняття функції втрати корисності (шкідливості) (13.3):

$$\underline{R}_j(k_j) = 1 - R_j(k_j) = 1 - \frac{k_j - k_{j\text{НГ}}}{k_{j\text{НК}} - k_{j\text{НГ}}} = \frac{k_{j\text{НК}} - k_j}{k_{j\text{НК}} - k_{j\text{НГ}}}. \quad (13.3)$$

Оскільки аргумент функцій (13.2), (13.3) у свою чергу залежить від рішення x , то надалі для цих функцій буде використано позначення $R_j(x)$ і $\underline{R}_j(x)$.

Для визначення меж наближеної області компромісів по j -му частковому критерію $[k_{j\text{НГ}}, k_{j\text{НК}}]$ можна скористатися еталонною базою оцінки.

На множині допустимих рішень по кожному з часткових критеріїв проводиться оптимізація і знаходиться рішення: $x_j^0 = \arg \text{extr } k_j(x) \mid x \in X$, $j = \underline{1, J}$ і відповідні йому значення всіх часткових критеріїв $k_i(x_j^0)$, $i \neq j$. Тоді $k_{j\text{НК}} = k_j(x_j^0)$, а $\{ \max k_j(x^0_i) \mid k_j(x) \rightarrow \min \}$ (13.4):

$$k_{jHG} = \begin{cases} \max_i k_j(x_i^0) | k_j(x) \rightarrow \min, \\ \min_i k_j(x_i^0) | k_j(x) \rightarrow \max. \end{cases} \quad (13.4)$$

Інтервал $[k_{jHG}, k_{jHK}]$ містить точки екстремумів усіх часткових критеріїв. Значення k_{jHK}, k_{jHG} ($j = \underline{1}, J$) є межами відображення наближеної області компромісів K^P на простір критеріїв. Вона включає і область компромісів K^C (область Парето), оскільки для неї виконується необхідна умова – досягнення локальних екстремумів часткових критеріїв [7].

Схема максимальної аддитивної корисності. При відомих значеннях вагових коефіцієнтів λ_j ($j = \underline{1}, J$) часткових критеріїв і їх функцій корисності $R_j(x)$ оцінка проектних рішень $x \in X$ і вибір найкращого x^0 проводиться за узагальненим критерієм (13.5):

$$W'(x^0) = \max \sum_{j=1}^J \lambda_j R_j(x) \mid \sum_{j=1}^J \lambda_j = 1, \quad (13.5)$$

де $\sum_{j=1}^J \lambda_j = 1, \lambda_j \in [0,1]$ ($j = \underline{1}, \bar{J}$).

Схема лексикографічного впорядкування. Якщо кількісні значення λ_j невідомі, часткові критерії ранжировані по важливості (13.6):

$$k_1(x) > k_2(x) > \dots > k_j(x), \quad (13.6)$$

то оцінка рішень і вибір найкращого здійснюється за послідовно вживаними критеріями. Рішення $x_1 \in X$ прийнятніше за рішення $x_2 \in X$ при виконанні умов (13.7):

$$k_j(x_1) = k_j(x_2); k_i(x_1) > k_i(x_2); j = \underline{1}, \bar{J}; i = j + \underline{1}, J. \quad (13.7)$$

Формальна схема вибору найкращого рішення за послідовно вживаними критеріями (13.8):

$$\begin{aligned} W''(x^0) &= \max R_j(x), \\ x &\in X; R_i(x) = R_{iHL}, \\ j &= \underline{1}, J; i = \underline{j + 1}, J. \end{aligned} \quad (13.8)$$

де R_{iHK} – найкраще значення функції корисності i -го критерію.

Схема справедливої поступки на першому етапі визначається оптимальне за найбільш важливим частковим критерієм рішення, після чого призначається допустимий рівень зниження корисності цього часткового критерію ΔR_i . Потім в цій області визначається оптимальне рішення за другим по важливості частковим критерієм і так далі. Найкращим вважається те рішення, яке задовольняє наступній формальній схемі (13.9):

$$W'''(x^0) = \max R_j(x) \mid x \in X; R_i(x) \geq R_{iHL} - \Delta R_i, j = \underline{1}, J, i = \underline{1}, j - 1. \quad (13.9)$$

Максимінна та мінімаксна схеми. Коли немає ні кількісної, ні якісної інформації про пріоритети часткових критеріїв, доцільно використовувати максимінну або мінімаксну схему компромісу:

$$W^*(x^0) = \max_{x \in X} \min_{j=1, J} R_j(x) \quad (13.10)$$

$$W^{**}(x^0) = \min_{x \in X} \max_{j=1, J} \bar{R}_j(x) \quad (13.11)$$

У цих схемах компромісу забезпечується вирівнювання значень функцій корисності приватних критеріїв.

Узагальнений критерій оцінки (універсальна форма). В якості універсальної форми узагальненого критерію оцінки, що враховує усі розглянуті схеми компромісу, пропонується (13.12):

$$W(x^0) = \max_{x \in X} \left\{ \sum_{j=1}^J [\lambda_j R_j(x)]^\beta \right\}^{1/\beta}, \quad (13.12)$$

або (13.13):

$$\bar{W}(x^0) = \min_{x \in X} \left\{ \sum_{j=1}^J [\lambda_j \bar{R}_j(x)]^\beta \right\}^{1/\beta}. \quad (13.13)$$

Достоїнством узагальненого критерію оцінки (13.12) є те, що залежно від значення коефіцієнта β реалізується широкий клас схем компромісу, цілей системи, що відбивають особливості. При $\beta=1$ реалізується схема максимальної аддитивної корисності (13.5), при $\beta \rightarrow -\infty$ – максимінна схема компромісу (12.10), а при $\beta \rightarrow \infty$ – мінімаксна (13.11). У випадку $\beta=1$, приймаючи в порядку убуття важливості критеріїв $\lambda_j = 1; j = 1, J; \lambda_i = 0; i = j+1, J$; і додаючи на кожному етапі додаткове обмеження по j -у частковому критерію, можна реалізувати схему оцінки і вибору рішення за послідовно вживаними критеріями (13.8) або справедливої поступки (13.9).

Розглянуті методи рішення багатокритеріальних задач є основними. Незважаючи на наявність великої кількості модифікацій, які враховують конкретні особливості ситуації прийняття рішення, у будь-якому випадку схема оцінювання зводиться або до однієї з вказаних, або до їх комбінації.

Завдання для самостійного виконання

- отримати у викладача індивідуальний варіант початкових даних;
- кожен характеристику альтернатив рішення по кожному критерію виразити кількісною оцінкою;
- вчислити функції корисності множини альтернатив;
- знайти рішення свого варіанту завдання за допомогою програмного забезпечення ПК;
- результати оформити у вигляді таблиці;
- привести аналіз результатів, зробити висновки, оформити індивідуальний звіт про роботу.

Контрольні запитання

1. Дати визначення поняттю надійності.
2. Основні властивості надійності. Їхня характеристика.

3. Перелічити критерії надійності.
4. Порівняйте класифікаційні ознаки відмов.
5. Поясніть визначення збереженості. Приведіть приклад.
6. У чому відмінності відновлюваних і невідновлюваних об'єктів. Приведіть приклад відновлювального об'єкта.

ЛАБОРАТОРНА РОБОТА №14 МЕТОД АНАЛІЗА ІЄРАРХІЙ Т.СААТІ

Мета лабораторної роботи: вивчення методу аналізу ієрархій (МАІ) для розв'язання багатокритеріальних задач прийняття рішень. Набуття практичних навичок побудови матриць попарних порівнянь та визначення вагових коефіцієнтів критеріїв і альтернатив.

Теоретичні відомості

Метод аналізу ієрархій (МАІ), розроблений Томасом Сааті, є систематичною процедурою для ієрархічного подання елементів, що визначають суть будь-якої проблеми. Метод полягає у декомпозиції проблеми на більш прості складові частини й подальшій обробці послідовності суджень особи, що приймає рішення (ОПР), за методом попарних порівнянь.

Метод включає процедури синтезу множинних суджень, одержання пріоритетності критеріїв і знаходження альтернативних рішень. Відносна міра або пріоритет кожного окремого об'єкта чи властивості може бути виведена і обґрунтована [8].

Структура МАІ – ієрархія з трьох рівнів:

- рівень 1 – мета (загальна ціль прийняття рішення);
- рівень 2 – критерії (фактори оцінювання альтернатив);
- рівень 3 – альтернативи (варіанти рішень).

Закон ієрархічної безперервності потребує, щоб елементи нижнього рівня ієрархії були порівнянні попарно стосовно елементів наступного рівня і т.д. аж до вершини ієрархії.

Метою побудов є одержання пріоритетів чи вагових коефіцієнтів елементів на нижньому рівні, які щонайкраще відбивають відносний вплив на вершину ієрархії.

Матриця попарних порівнянь.

Нехай $O_1, O_2, O_3, \dots, O_n$ – множина з n елементів і $\omega_1, \omega_2, \omega_3, \dots, \omega_n$ – відповідно їхні реальні значення важливості або пріоритетності. Порівнюємо важливість кожного елемента з важливістю елемента множини стосовно загальної для них властивості або мети (табл. 14.1). Таблиця 14.1 – Порівняння елементів «О» за їхньою важливістю чи пріоритетністю щодо «Мети»

Мета	O_1	O_2	...	O_n
O_1	ω_1/ω_1	ω_1/ω_2	...	ω_1/ω_n
O_2	ω_2/ω_1	ω_2/ω_2	...	ω_2/ω_n
...
O_n	ω_n/ω_1	ω_n/ω_2	...	ω_n/ω_n

Тоді реально матриця порівнянь має вигляд (14.1) :

$$O = \|\alpha_{ij}\|_{n,n}, \quad (14.1)$$

причому $\alpha_{ii} = 1, \alpha_{ji} = 1 / \alpha_{ij},$ де $i = 1, \bar{n}, j = 1, \bar{n}.$

На практиці елементи в матриці порівнянь визначаються експертами у відповідності до шкали відносної важливості (пріоритетності), запропонованої Т. Сааті (табл. 14.2).

Таблиця 14.2 – Шкала відносної важливості (пріоритетності)

Інтенсивність відносної важливості	Визначення	Пояснення
1	Елементи однаково важливі (пріоритетні)	Рівний внесок двох елементів у досягнення мети
3	Незначна перевага одного над іншим	Є умови, що надають легку перевагу одного над іншим
5	Істотна перевага	Існують вагомі факти, що один істотно важливіший від іншого
7	Явна перевага одного над іншим	Є беззаперечні факти переваги одного над іншим
9	Дуже сильна перевага	Очевидність переваги одного над іншим не викликає сумнівів
2, 4, 6, 8	Проміжний результат рішення між двома сусідніми міркуваннями	Застосовується у компромісному випадку
Обернені розміри	Якщо при порівнянні одного елемента з іншим отримано одне з вище зазначених чисел (наприклад, 3), то при зворотному порівнянні елементів одержимо зворотне число (тобто 1/3)	

Коли проблема подана ієрархічно, матриця складається для порівняння відносної важливості чи пріоритетності критеріїв на другому рівні стосовно загальної мети на першому рівні. Подібні матриці повинні бути побудовані для парних порівнянь кожного елемента на третьому рівні стосовно критеріїв (елементів) другого рівня.

Необхідно знайти власне значення та власний вектор матриці O , використовуючи наближені методи. Далі вирішується задача знаходження вектора $\omega = [\omega_1, \omega_2, \dots, \omega_n]^T$.

Перший можливий алгоритм передбачає застосування методу рядкових сум, де на першому кроці додаємо елементи кожного рядка матриці і результати підсумовування передаємо у вигляді матриці-стовпчика такої ж розмірності, що й у матриці. Другим кроком є додавання разом усіх елементів одержаного вектора-стовпчика, а на третьому кроці ділимо кожний із елементів матриці-стовпчика на знайдену суму.

Другим алгоритмом є метод зворотних сум стовпчиків, що базується на обчисленні сум елементів кожного стовпчика початкової матриці з наступним визначенням зворотних до них величин. Отримані значення підсумовують, після чого кожен елемент зворотного стовпця ділять на загальну суму для отримання нормованих ваг [8].

Третій алгоритм це метод нормалізації стовпчиків, він передбачає попереднє підсумовування елементів за кожним стовпчиком матриці A . На наступному етапі всі елементи матриці ділять на суми відповідних стовпчиків, що призводить до їх нормалізації. Далі обчислюють суму елементів кожного рядка отриманої матриці та записують ці значення у вектор-стовпець. Кінцевий вектор пріоритетів визначають шляхом ділення кожного елемента цього стовпця на порядок матриці n . Алгоритм 4 (Метод геометричного середнього) вважається найбільш точним серед наближених підходів. Процес обчислення починається з перемноження елементів кожного рядка матриці, після чого з кожного отриманого добутку

витають корінь n-го степеня. Усі знайдені корені додають, а потім кожен окремий корінь ділять на отриману загальну суму, що дозволяє сформувати нормований вектор пріоритетів.

Важливим елементом даної моделі є знаходження індексу узгодженості (ІУ), який дає інформацію про порушення числової та транзитивної матриці порівнянь. Тому цей індекс можна розглядати як показник «близькості до узгодженості». Похибки співвідношень (14.2):

$$\alpha_{ik} = \alpha_{ij} \cdot \alpha_{jk}, \quad k = 1, \bar{n}, \quad i = 1, \bar{n}, \quad j = 1, \bar{n}. \quad (14.2)$$

Для цього, використовуючи відхилення максимального власного числа від розмірності матриці λ_{\max} , будемо величину, звану індексом узгодженості (14.3):

$$IU = (\lambda_{\max} - n) / (n - 1), \quad (14.3)$$

де n – число порівнюваних елементів.

Потім порівнюємо ІУ з відповідним індексом, отриманим для матриці, побудованої випадковим чином, і отримуємо відношення узгодженості $ВУ = IU / ВУ$. Випадкові узгодженості для матриць різного порядку вибираються з таблиці 14.3.

Таблиця 14.3 – Значення середньої випадкової узгодженості для випадкових назад симетричних матриць різного порядку

Розмір матриці $n \cdot n$	3	4	5	6	7	8	9	10
Середня випадкова узгодженість	0,58	0,90	1,12	1,24	1,32	1,41	1,45	1,49

Прийнятною є ВУ не більше 10%. У деяких випадках можна допустити 20%, але не більш. Інакше необхідно провести переоцінку відповідної матриці [8].

Подібні обчислення проводяться для всіх рівнів і груп в ієрархії.

Останнім кроком є обчислення загальної ваги варіанту вирішення шляхом послідовного зважування матриці-строки ваг рівня (варіантів рішень), що пролягає нижче, компонентами матриці-строки ваг вищерозміщеного рівня (характеристик).

При участі в роботі над проектом декількох експертів можна використовувати геометричне середнє судження, якщо учасники не хочуть дебатів. Інакше можна одержати індивідуальні матриці-строки ваг і за відповідь взяти їх геометричне середнє.

Завдання для самостійного виконання

1. Отримати у викладача індивідуальний варіант початкових даних.
2. Побудувати ієрархічну структуру задачі (мета → критерії → альтернативи).
3. Скласти матрицю попарних порівнянь критеріїв відносно мети.
4. Скласти матриці попарних порівнянь альтернатив відносно кожного критерію.
5. Знайти вектори пріоритетів для кожної матриці (одним із чотирьох алгоритмів).
6. Обчислити індекс узгодженості (ІУ) та відношення узгодженості (ВУ) для кожної матриці.
7. Обчислити загальні пріоритети альтернатив і обрати найкращу.

8. Результати оформити у вигляді таблиць. Зробити висновки.

Контрольні запитання

1. Що таке метод аналізу ієрархій (МАІ)?
2. Опишіть структуру ієрархії в МАІ. Що розміщується на кожному рівні?
3. Що таке матриця попарних порівнянь? Які властивості вона має?
4. Яку шкалу відносної важливості використовують при побудові матриці порівнянь?
5. Опишіть алгоритм 4 (метод геометричного середнього) знаходження вектора пріоритетів.
6. Що таке індекс узгодженості (ІУ) і як він обчислюється?
7. Що таке відношення узгодженості (ВУ)? Яке його прийнятне значення?
8. Як обчислюється загальна вага (пріоритет) альтернативи в МАІ?
9. Що робити, якщо ВУ перевищує допустиме значення?
10. Як враховуються судження декількох експертів у МАІ?

ЛАБОРАТОРНА РОБОТА №15 ПРИЙНЯТТЯ РІШЕНЬ ЗА УМОВ НЕВИЗНАЧЕНОСТІ

Мета лабораторної роботи: набуття практичних навичок прийняття рішень в умовах невизначеності при відомих ймовірності станів.

Теоретичні відомості

Задача прийняття рішення (ЗПР) за умов невизначеності полягає у виборі оптимальної стратегії, успіх реалізації якої залежить також від деяких невизначених факторів, що не підвладні ОПР та невідомі в момент прийняття рішення. Розрізняють невизначеності не стохастичної та стохастичної природи.

Так, невизначеності не стохастичної природи можуть спричинятися дією таких факторів:

Стратегічні невизначеності – зумовлені протидією кількох активних учасників, що мають різні цілі (наприклад, діями конкурентів). Тут невизначеність зумовлена тим, що ОПР приймає рішення за умов, коли невідомі майбутні дії або стратегії інших учасників (у термінах теорії ігор – гравців).

Концептуальні невизначеності – невизначені фактори, що зумовлені прийняттям особливо складних рішень, рішень, що мають довгострокові наслідки або можуть бути пов'язані з нечітким усвідомленням ОПР як власних цілей та можливостей, так і інших гравців. Крім цього, концептуальні невизначеності можуть бути пов'язані із труднощами кількісної оцінки складних цілей та якісних критеріїв, що важко формалізуються [9].

ЗПР з невизначеністю не стохастичного типу розв'язують методами теорії ігор та теорії мінімаксу .

Невизначеності стохастичного типу зумовлені об'єктивною дійсністю, яку називають природою. Природа розглядається як не зацікавлена сторона. У такому разі ЗПР розв'язуються за допомогою теорії статистичних рішень.

Прийняття рішень в умовах невизначеності, як і в умовах ризику, вимагає визначення альтернативних дій, яким відповідають платежі, залежні від (випадкових) станів природи. Матрицю платежів в задачі прийняття рішень з m можливими діями і n станами природи можна представити таким чином (рис. 15.1).

	z_1	z_2	...	z_n
d_1	$v(d_1, z_1)$	$v(d_1, z_2)$...	$v(d_1, z_n)$
d_2	$v(d_2, z_1)$	$v(d_2, z_2)$...	$v(d_2, z_n)$
...
d_m	$v(d_m, z_1)$	$v(d_m, z_2)$...	$v(d_m, z_n)$

Рисунок 15.1 – Матриця платежів

Елемент d_i представляє i -е можливе рішення, а елемент z_j – j -й стан природи. Плата (або дохід), пов'язана з рішенням d_i і станом z_j , дорівнює $v(d_i, z_j)$. Відмінність між прийняттям

рішень в умовах ризику і невизначеності полягає в тому, що в умовах невизначеності імовірнісний розподіл, відповідний станам z_j , $j=1,2 \dots, n$, або невідомий, або не може бути визначений. Цей недолік інформації зумовив розвиток наступних критеріїв для аналізу ситуації, пов'язаної з прийняттям рішень:

- критерій Лапласа;
- максимінний Вальда;
- критерій Севіджа;
- критерій Гурвіца.

Ці критерії відрізняються по ступеню консерватизму, який проявляє особа, що приймає рішення, перед лицем невизначеності [9].

Критерій Лапласа спирається на принцип, який свідчить, що, оскільки розподіл вірогідності станів $P(z_j)$ невідомий, немає причин вважати різними. Отже, використовується оптимістичне припущення, що вірогідність всіх станів природи рівна між собою, тобто $P(z_1) = P(z_2) = \dots = P(z_n) = 1/n$.

Якщо при цьому $v(d_i, z_j)$ представляє отриманий прибуток, то найкращим рішенням є те, яке забезпечує (15.1):

$$\max_{a_i} \left\{ \frac{1}{n} \sum_{j=1}^n v(d_i, z_j) \right\} \quad (15.1)$$

Якщо величина $v(d_i, z_j)$ представляє витрати ОПР, то оператор «max» замінюється на «min».

Максимінний (мінімаксний) критерій Вальда заснований на консервативній обережній поведінці ОПР, і зводиться до вибору найкращої альтернативи з найгірших. Якщо величина $v(d_i, z_j)$ представляє отриманий прибуток, то відповідно до максимінного критерію як оптимальне вибирається рішення, що забезпечує (15.2):

$$\max_{a_i} \left\{ \min_{s_j} v(d_i, z_j) \right\} \quad (15.2)$$

Якщо величина $v(d_i, z_j)$ представляє втрати, використовується мінімаксний критерій, який визначається наступним співвідношенням (15.3):

$$\min_{a_i} \left\{ \max_{s_j} v(d_i, z_j) \right\} \quad (15.3)$$

Критерій Севіджа прагне пом'якшити консерватизм максимінного (мінімаксного) критерію шляхом заміни матриці платежів (виграшів або програшів) $v(d_i, z_j)$ матрицею втрат $r(a_i, s_j)$, яка визначається таким чином (15.4):

$$r(d_i, z_j) = \begin{cases} \max_{d_i} \{v(d_i, z_j)\} - v(d_i, z_j), & \text{якщо } v - \text{дохід} \\ v(d_i, z_j) - \min_{d_i} \{v(d_i, z_j)\}, & \text{якщо } v - \text{втрати} \end{cases} \quad (15.4)$$

Розглянемо тепер критерій Гурвіца. Цей критерій охоплює ряд різних підходів до прийняття рішень від найбільш оптимістичного до найбільш песимістичного (консервативного). Нехай $0 \leq \beta \leq 1$ і величини $v(d_i, z_j)$ представляють доходи.

Тоді рішення, вибраному по критерію Гурвіца, відповідає (15.5):

$$\max_{d_i} \left\{ \beta \max_{z_j} v(d_i, z_j) + (1 - \beta) \min_{z_j} v(d_i, z_j) \right\} \quad (15.5)$$

Параметр β – показник оптимізму. Якщо $\beta = 0$, критерій Гурвіца стає консервативним, оскільки його застосування еквівалентне застосуванню звичайного максимінного критерію. Якщо $\beta = 1$, критерій Гурвіца стає дуже оптимістичним, бо розраховує на найкращих з найкращих умов. Ми можемо конкретизувати ступінь оптимізму (або песимізму) належним вибором величини β з інтервалу $[0,1]$. За відсутності яскраво вираженої схильності до оптимізму або песимізму вибір $\beta = 0.5$ представляється найбільш розумним [9].

Якщо величини $v(d_i, z_j)$ представляють втрати, то критерій приймає наступний вигляд (15.6):

$$\min_{d_i} \left\{ \beta \min_{z_j} v(d_i, z_j) + (1 - \beta) \max_{z_j} v(d_i, z_j) \right\} \quad (15.6)$$

Процес прийняття рішення за умов невизначеності стохастичного типу є певною мірою суб'єктивним, але в будь-якому разі є сенс проаналізувати ситуацію з погляду різних критеріїв.

Завдання для самостійного виконання

- отримати у викладача індивідуальний варіант початкових даних;
- знайти рішення свого варіанту завдання, за допомогою програмного забезпечення ПК;
- результати оформити у вигляді таблиці;
- після закінчення роботи скласти індивідуальний звіт.

Контрольні запитання до лабораторної роботи

1. Що таке багатокритеріальна задача прийняття рішень?
2. Поясніть поняття функції корисності часткового критерію.
3. Яка роль матриці попарних порівнянь у методі аналізу ієрархій?
4. Охарактеризуйте максимінну та мінімаксу схеми компромісу.
5. Що таке індекс узгодженості та відношення узгодженості у МАІ?
6. Що таке схема справедливої поступки?

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Теорія систем і системний аналіз: навчальний посібник / О.А. Балтовський, К.Ю. Ісмайлов, О.І. Сіфоров, Г.В. Форос, О.М. Заєць; за заг. ред. Балтовського О.А. Одеса: РВВ ОДУВС, 2021. 156 с.
2. Творошенко І.С. Технології прийняття рішень в інформаційних системах : навч. посіб. Харків : ХНУРЕ, 2021. 118 с.
3. Системний аналіз та теорія прийняття рішень : конспект лекцій для здобувачів першого (бакалаврського) рівня вищої освіти освітньої програми «Комп'ютерні науки» галузі знань 12 Інформаційні технології спеціальності 122 Комп'ютерні науки денної та заочної форм навчання/ уклад. В.О. Ліщина, Н.М. Ліщина. Луцьк : Луцький НТУ, 2021. 116 с.
4. Згуровський М.З., Панкратова Н.Д. Основи системного аналізу : підручник. Київ : ВПЦ «Київський університет», 2021. 544 с.
5. Бідюк П.І., Коршевнік Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень : навч. посіб. Київ : НТУУ «КПІ», 2022. 340 с.
6. Катренко А. В. Дослідження операцій : підруч. Львів : Магнолія Плюс, 2021. 352 с.
7. Сачук В., Вавринюк К., Хиць Р. Застосування методів багатокритеріального аналізу для підтримки прийняття управлінських рішень в умовах невизначеності. Herald of Khmelnytskyi National University. Technical Sciences, 2025. Vol. 359, № 6.2. P. 142-148.
8. Raskin L., Sira O. Method of solving fuzzy problems of mathematical programming. Eastern-European Journal of Enterprise Technologies. 2021. Vol. 1, № 4 (109). P. 6-14.
9. Kuchuk N., Mozhaiev O., Semenov S. Optimization of the information security system of a critical infrastructure facility. Advanced Information Systems. 2021. Vol. 5, № 3. P. 117-123.

Системний аналіз та теорія прийняття рішень: методичні вказівки до лабораторних робіт для здобувачів першого (бакалаврського) рівня освітньо-професійної програми «Комп'ютерні науки» галузі знань F(12) Інформаційні технології спеціальності F3(122) Комп'ютерні науки денної та заочної форм навчання / уклад. В.О. Сачук, К.В. Вавринюк, Луцьк : ЛНТУ, 2026. 77 с.

Комп'ютерний набір

К.В. Вавринюк

Редактор

В.О. Сачук

Підп. до друку «__» ____2026 р. Формат 60x84/16. Папір офс.

Гарн. Таймс. Ум. друк. арк. 2,5.

Інформаційно-видавничий відділ

Луцького національного технічного університету

43018 м. Луцьк, вул. Львівська, 75

Друк – ІВВ ЛНТУ