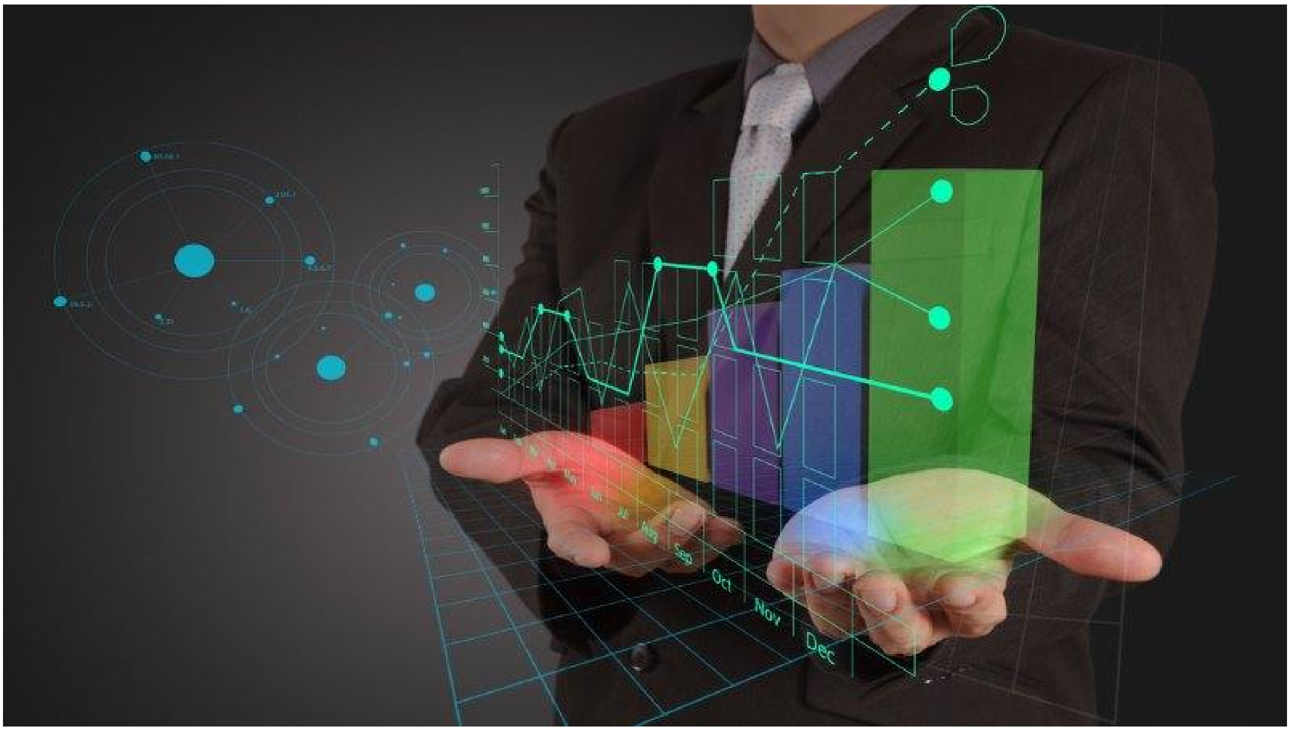
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬОТНА АКАДЕМІЯ
НАЦІОНАЛЬНОГО АВІАЦІЙНОГО УНІВЕРСИТЕТУ

Сікірда Ю.В.

Конспект лекцій
з навчальної дисципліни

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПРОФЕСІЙНИХ ЗАДАЧ

для здобувачів вищої освіти освітнього ступеня «магістр»
денної та заочної форм навчання
галузі знань 27 «Транспорт»
спеціальності 272 «Авіаційний транспорт»
освітньо-професійної програми
«Авіаційний транспорт»



Кропивницький
2022

УДК 656.7:658.5

У к л а д а ч:
Ю.В. Сікірда – кандидат технічних наук, доцент, доцент кафедри туризму та авіаційних перевезень.

Рецензенти:
О.В. Колотуха – завідувач кафедри туризму та авіаційних перевезень, доктор географічних наук, доцент;
Н.І. Легінькова – доцент кафедри менеджменту та економіки, кандидат економічних наук, доцент.

Сікірда Ю.В.
Математичне моделювання професійних задач : конспект лекцій. Кропивницький : ЛА НАУ, 2022. 109 с.

Конспект лекцій містить плани лекцій; мету та завдання лекцій; основні категорії, ключові поняття та визначення тем; тексти лекцій; висновки по лекціям.

Призначений для здобувачів вищої освіти освітнього ступеня «магістр» денної та заочної форм навчання галузі знань 27 «Транспорт» спеціальності 272 «Авіаційний транспорт» освітньо-професійної програми «Авіаційний транспорт».

УДК 656.7:658.5

Розглянуто та рекомендовано для видання і використання у освітньому процесі академії рішенням: Кафедри туризму та авіаційних перевезень, протокол від 02лютого 2022 року № 6; Науково-методичної ради академії, протокол від «12» травня 2022 року № 6.

ЗМІСТ

ВСТУП

Конспект лекцій складений згідно робочої програми навчальної дисципліни «Математичне моделювання професійних задач» відповідно до навчального плану підготовки магістрів зі спеціальності 272 «Авіаційний транспорт» за освітньо-професійною програмою «Авіаційний транспорт».

Програма навчальної дисципліни складається з таких змістових модулів:

1 Детерміновані математичні моделі виробничих процесів на авіаційному транспорті

2 Ймовірнісні математичні моделі виробничих процесів на авіаційному транспорті

Предметом вивчення навчальної дисципліни є: моделювання виробничих процесів на авіаційному транспорті для оптимізації виробництва і підвищення якості робіт.

Мета викладання навчальної дисципліни «Математичне моделювання професійних задач» – формування професійних знань та набуття практичних навичок в застосуванні математичних моделей для оптимізації виробничих процесів на авіаційному транспорті.

Завдання вивчення дисципліни «Математичне моделювання професійних задач» – освоєння і використання апарату математичного моделювання для оптимізації виробничих процесів на авіаційному транспорті на основі методів дослідження операцій; з'ясування ролі, стану і перспектив розвитку економіко-математичних методів оптимізації виробничих процесів на авіаційному транспорті в ринкових умовах з урахуванням трудових, матеріальних, техніко-експлуатаційних та організаційних обмежень; прищеплення здобувачам вищої освіти навичок дослідження і аналізу.

У результаті вивчення навчальної дисципліни здобувач вищої освіти повинен

знати:

- поняття та види моделей;
- сутність математичного моделювання;
- роль математичних методів в розв’язанні задач оптимізації виробничих процесів на авіаційному транспорті;

вміти:

- застосовувати результати математичного моделювання для підвищення ефективності виробничих процесів на авіаційному транспорті;
- використовувати сучасні інформаційні технології при моделюванні виробничих процесів на авіаційному транспорті;

бути ознайомленим:

- з специфікою математичного моделювання професійних задач в авіаційній галузі.

Змістовий модуль 1 ***Детерміновані математичні моделі виробничих процесів на авіаційному транспорті***

Тема 1 *Методологічні основи математичного моделювання виробничих процесів на авіаційному транспорті* (2 год.)

1.1 Мета та завдання лекції

Метою лекції є ознайомлення з сутністю математичного моделювання виробничих процесів на авіаційному транспорті.

Завдання лекції:

- розкрити місце математичного моделювання в кібернетиці, представити принципову схему процесу управління та розкрити основні поняття дослідження операцій;
- розглянути сутність моделювання та представити моделювання як природний процес пізнання;
- розкрити поняття моделі та охарактеризувати види моделей;
- розглянути основи побудови математичних моделей виробничих процесів;
- ознайомити з вимогами до інформаційного забезпечення моделей.

1.2 План лекції

1.1 Математичне моделювання – основний метод кібернетики. Принципова схема процесу управління. Основні поняття в дослідженні операцій.

1.2 Сутність моделювання. Моделювання як природний процес пізнання.

1.3 Поняття моделі та види моделей. Евристичні, фізичні та математичні моделі. Детерміновані і стохастичні моделі. Статичні та динамічні моделі. Дискретні та неперервні моделі.

1.4 Основи побудови математичних моделей виробничих процесів.

1.5 Інформаційне забезпечення моделей.

1.3 Основні категорії, ключові поняття та визначення теми

Дослідження операцій – дисципліна, яка займається розробкою та застосуванням методів знаходження оптимальних рішень на основі математичного моделювання у різних областях людської діяльності.

Інформаційне забезпечення **–** це сукупність інформації та способів її пошуку, обробки, накопичення, збереження, систематизації та узагальнення з метою використання в процесі математичного моделювання.

Інформація – сукупність відомостей (повідомлень, даних), яка визначає міру наших знань про ті чи інші явища, події та їх взаємозв'язки.

Кібернетика (від грец. kiber – над, nautis – моряк, тобто старший моряк, керманич, керуючий кермом, звідси – kybemetike – мистецтво (наука) управляти) – 1) наука про загальні принципи управління в комплексі складними системами різної природи походження (соціальними, технічними, біологічними, економічними тощо) на основі знань, які сформовані на зворотних зв'язках; 2) наука про загальні закономірності процесів управління та переробки інформації в живій та неживій природі, техніці та економіці; 3) сукупність інформаційних наук, у тому числі науки управління, що дозволяє на основі переходу від реальної системи до кібернетичного уявлення (моделі), здійснювати моделювання системи, досліджувати її характеристики та дати рекомендації щодо вдосконалення системи.

Математична модель – модель, що використовує для опису властивостей і характеристик досліджуваної системи математичні символи і методи.

Моделювання – це метод дослідження складних систем, заснований на тому, що розглянута система замінюється на модель і проводиться дослідження моделі з метою одержання інформації про досліджувану систему.

Модель в науці (від лат. modulus – зразок) – будь-який образ, аналог (уявний або умовний: зображення, визначення, схема, креслення, графік, карта, формула тощо) якого-небудь об'єкта, процесу або явища («оригіналу» цієї моделі).

Оптимізація – процес надання чому-небудь найвигідніших характеристик, співвідношень.

Система (від грец. system – ціле, складене з частин) – сукупність елементів, взаємодіючих один з одним, що утворюють певну цілісність, єдність.

Структура системи – мережа зв’язків або відношень між складовими частинами системи.

1.4 Текст лекції

Математичне моделювання – основний метод кібернетики. Принципова схема процесу управління. Основні поняття в дослідженні операцій

У сучасному розумінні кібернетика з'явилася в 1948 році, коли американський вчений Норберт Вінер (1894-1964) опублікував свою знамениту книгу "Кібернетика, або управління та зв'язок у тварині та машині".

У перекладі з грецького кібернетика означає "кормчий".

Грецький філософ Платон, який жив у Афінах у 4 в. е., спочатку використовував термін «кібернетика» у сенсі мистецтва управління кораблем, та був і суспільством.

У першій половині 19 століття Ампер, займаючись класифікацією наук, назвав кібернетикою науку, яка повинна досліджувати способи управління державою.

Основний висновок, зроблений Вінером за результатами дослідження полягав у тому, що всі системи управління побудовані на тих самих принципах.

Згідно з М. Вінером, кібернетика має ширший сенс, ніж управління державою, і визначається в такий спосіб.

Кібернетика (від грец. kiber – над, nautis – моряк, тобто старший моряк, керманич, керуючий кермом, звідси – kybemetike – мистецтво (наука) управляти) – 1) наука про загальні принципи управління в комплексі складними системами різної природи походження (соціальними, технічними, біологічними, економічними тощо) на основі знань, які сформовані на зворотних зв’язках; 2) наука про загальні закономірності процесів управління та переробки інформації в живій та неживій природі, техніці та економіці; 3) сукупність інформаційних наук, у тому числі науки управління, що дозволяє на основі переходу від реальної системи до кібернетичного уявлення (моделі), здійснювати моделювання системи, досліджувати її характеристики та дати рекомендації щодо вдосконалення системи.

Функція управління є універсальною, характерною для всіх галузей знань. Однак об'єкти управління відрізняються один від одного в залежності від знань. Відповідно різні і системи управління цими об'єктами.

У зв'язку зі специфікою управління різними класами систем виділяють прикладну кібернетику:

- технічну кібернетику;
- медичну кібернетику;
- соціальну кібернетику;
- біологічну кібернетику;
- військову кібернетику;
- економічну кібернетику тощо.

Будь-яке поняття можна визначити за допомогою ключових слів. **Кібернетика визначається п'ятьма ключовими словами:**

1 ***Система*** - сукупність елементів, взаємодіючих один з одним, що утворюють певну цілісність, єдність.

2 ***Інформація*** - у широкому розумінні - сукупність відомостей (повідомлень, даних), яка визначає міру наших знань про ті чи інші явища, події та їх взаємозв'язки, у вужчому - нові відомості.

3 ***Управління*** - цілеспрямований вплив на об'єкт з метою приведення його в необхідний стан.

4 ***Моделювання*** - основний метод кібернетики, що полягає в тому, що для отримання відомостей про об'єкт використовуються дослідження, виконані з його моделлю.

5 ***Штучний інтелект*** - науковий напрямок, метою якого є відтворення за допомогою комп'ютера інтелектуальних функцій, характерних для людини.

Роботи у сфері штучного інтелекту розпочалися 1943 року із зародженням нейрокібернетики. Сам термін «штучний інтелект» запропоновано 1956 року.

Слід особливо наголосити, що кібернетика розглядає лише інформаційну сторону процесів.

Кібернетика – це не окрема наука, а сукупність наук і наукових дисциплін. **До складу кібернетики входять п'ять фундаментальних наук**:

- теорія систем;
- теорія управління;
- теорія інформації;
- **моделювання;**
- штучний інтелект.

Велика кількість наукових дисциплін є складовими кібернетики. Серед них:

- теорія масового обслуговування;
- дослідження операцій;
- теорія ігор;
- теорія графів;
- теорія прийняття рішень;
- оптимізація систем і т.д.

Загальна мета вивчення сукупності дисциплін, що входять до складу наукового напряму, що називається «кібернетикою», полягає у формуванні кібернетичного світогляду. Ця думка полягає в тому, що фахівець-кібернетик вміє дивитися на навколишні системи, процеси та явища не тільки з позиції традиційних уявлень, але і з позицій кібернетичних уявлень.

Приклад переходу від звичайного представлення до кібернетичного показано в табл.1.1

Таблиця 1.1 – Приклад переходу від звичайного представлення до кібернетичного

Звичайне представлення	Кібернетичне представлення
Заклад вищої освіти	Система управління вищою освітою: ієрархічна, динамічна, замкнутого типу
Студенти	Об'єкт управління
Ректорат, деканат, викладацький склад	Система управління
Іспити, заліки	Зворотній зв'язок
Оцінки	Органи управління

Кожний об'єкт управління – це система, що складається із взаємозалежних елементів. Наприклад, народне господарство, його галузі, суб'єкти господарювання розглядаються як економічні системи. При цьому кожна система є одночасно елементом системи вищого рівня. Ієрархія систем веде як вгору до міністерства, так і вниз до виробничої одиниці. Здійснене відокремлення визначає межі виділеної системи. Відповідно все, що знаходиться поза досліджуваною системою, є ***зовнішнім середовищем***.

Водночас кожна система існує не відокремлено, а під дією як суміжних систем, так і навколишнього середовища. Кількість таких впливів безмежна, але враховуються тільки ті з них, які суттєво впливають на досліджувані параметри системи. Ці впливи називаються ***входами***. Впливи поділяють на керуючі та збурювальні. До ***керуючих*** впливів належать директиви, економічні нормативи, планові завдання, корективи обсягів робіт та ін.; до ***збурювальних*** – зриви у постачанні матеріалів (зовнішні), хвороби працівників, простої, поломки обладнання (внутрішні).

Оскільки всі системи - взаємозалежні, кожна з них, у свою чергу, впливає на зовнішнє середовище. Особливості цього впливу визначаються ***виходом*** системи.

Вихід і вхід системи є взаємозалежними, між ними існує прямий причинно-наслідковий зв'язок, що виявляється у функціонуванні системи.

Окрім вхідних і вихідних параметрів, система характеризується множиною змінних, які визначають ***внутрішній стан*** (рис. 1.1).

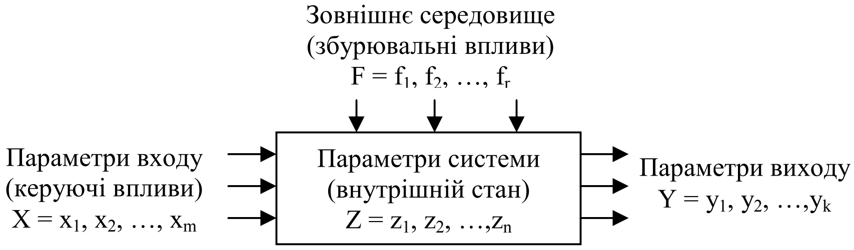


Рисунок1.1 – Основні характеристики системи

При дослідженні системи управління найбільший інтерес викликає залежність між її входом і виходом. Відповідно зміну вихідних параметрів під впливом вхідних кваліфікують як ***перетворення системи***.

Цілеспрямований вплив однієї системи (підсистеми) на іншу, який має на меті змінити її поведінку в певному напрямі (відповідно до заданої мети), є ***управлінням***. З цього випливає, що система, яка реалізує процес управління, складається як мінімум із двох частин (рис. 1.2): керованої (якою управляють) і керуючої (яка управляє).

Рисунок1.2 – Структура системи управління

Керована система – це виробничо-технічна система, а ***керуюча*** - це система вищого рівня (інформаційна). Механізми процесу управління дуже складні та важко доступні для розгляду. Розкрити їх зміст допомагає кібернетичний підхід, який розглядає тільки інформаційні процеси.

Система (від грец. system - ціле, складене з частин) - це сукупність елементів, взаємодіючих один з одним, що утворюють певну цілісність, єдність.

Системи можна класифікувати так:

1) Залежно від того, чи мета, яку реалізує система, знаходиться поза системою або всередині неї, розрізняють системи:

- ціленаправлені;
- цілеспрямовані.

2) За видом елементів:

- системи типу "об'єкт" - системи, елементами яких є предмети (двигун, машина, вертоліт);
- системи типу "процес" - системи, елементами яких є операції (виготовлення, перевезення, обслуговування).

3) Залежно від кількості рівнів управління:

- однорівневі;
- багаторівневі чи з ієрархічною структурою.

4) За ступенем складності:

- прості, які можна досліджувати як ціле, без розбиття на дрібніші системи (авторучка);
- великі, які можна поділити на окремі частини (літак);
- складні, які не можна розділити на окремі підсистеми (економіка, мозок людини).

5) За числом станів:
- динамічні - системи, стан яких змінюється зі зміною часу;
- статичні.
6) За типом станів:
- дискретні - якщо стан системи, тобто значення її атрибутів, змінюється стрибкоподібно, у дискретні моменти часу (людина ввійшла/вийшла у магазин);
- безперервні - якщо стан системи змінюється безперервно (температура закипаючого чайника);
- комбіновані (безперервно-дискретні) - якщо частина атрибутів, які описують стан системи, міняється безперервно, а частина дискретно (наприклад, поводження краплі з крана - "висить", "летить", "упала").
7) За умовами переходів з одного стану в інший:
- стохастичні (ймовірнісні, статистичні) - якщо при тих самих початкових умовах результати функціонування системи будуть розрізнятися під впливом випадкових факторів;
- детерміновані.
8) За походженням:
- природні;
- штучні.
9) За характером взаємодії із зовнішнім середовищем:
- відкриті, які взаємодіють із зовнішнім середовищем;
- закриті.
10) За наявністю циклів:
- із циклічними процесами;
- без циклів.
11) За розмірами:
- малі (мале підприємство);
- середні (фірма);
- великі (корпорація).
12) За характером змін внутрішніх властивостей системи:
- що самоорганізуються, із зміною структури;
- адаптивні, тобто, із зміною параметрів.
13) За наявністю зворотного зв'язку:
- без зворотних зв'язків;
- зі зворотними зв'язками.
14) За характером зворотного зв'язку:
- з негативним зворотним зв'язком;
- з позитивним зворотним зв'язком.

Авіація є системою цілеспрямованою, ієрархічною, складною, динамічною, штучною, як правило відкритою, статистичною, з циклічними процесами, з адаптацією та самоорганізацією, з позитивним зворотним зв'язком.

Процеси у відкритих системах значною мірою визначаються впливом довкілля і самі на нього впливають. Закрита система не взаємодіє із зовнішнім середовищем. Чіткого кордону між подібними системами не може бути. Тобто, реальні системи характеризуються різним ступенем відкритості. Прикладом закритої системи є годинник без заводу. Після того, як енергія заводу вичерпана, годинник зупиняється, тобто закрита система прагне дезорганізації. Такими є покинуті людиною міста.

Авіаційний транспорт має бути відкритою системою. Усі країни світу прагнуть збільшення відкритості для збільшення взаємозв'язків із зовнішнім світом.

Дослідження операцій – дисципліна, яка займається розробкою та застосуванням методів знаходження оптимальних рішень на основі математичного моделювання у різних областях людської діяльності.

Операцією називається будь-який захід (система дій), об'єднаний єдиним задумом і спрямований на досягнення певної мети. Операція є завжди керованим заходом, тобто, є можливість розпорядитися методом вибору деяких параметрів, що характеризують її організацію. Ці параметри називають ***керуючими змінними***.

Будь-який певний вибір таких змінних називається ***рішенням***. Рішення можуть бути вдалими та невдалими, розумними та нерозумними. ***Оптимальними*** називають такі рішення, які за деякими критеріями кращі за інших.

Мета дослідження операцій – попереднє кількісне обґрунтування оптимальних рішень, яких може бути більше ніж одне. Остаточний вибір рішення виходить за рамки дослідження операцій та проводиться засобами так званої ***теорії прийняття рішень***.

Будь-яке завдання дослідження операцій має початкові «дисциплінуючі» умови, тобто, такі вихідні дані, які фіксовані від початку і не можуть бути порушені. У своїй сукупності вони формують так звану безліч можливих рішень.

Щоб порівнювати між собою за ефективністю різні рішення, потрібно мати кількісний критерій, який називається ***показником (критерієм) ефективності (або цільовою функцією)***. Цей показник вибирається для того, щоб відображати цільову спрямованість операції.

Часто виконання операції супроводжується дією випадкових факторів. Тоді як показник ефективності береться не сама величина, яку хотілося б оптимізувати, а її середнє значення (або математичне очікування).

Іноді операція, яка супроводжується випадковими факторами, переслідує таку мету А, яка може бути досягнута повністю, або не досягнута зовсім (типу «так - ні»). Тоді як показник ефективності вибирають ймовірність досягнення цієї мети р(А). Якщо р(А) = 0 або 1, то приходимо до відомої в кібернетиці задачі «чорної скриньки».

Неправильний вибір показника ефективності дуже небезпечний. Операції, організовані під кутом зору невдало обраного критерію, можуть призвести до невиправданих витрат та втрат.

Сутність моделювання. Моделювання як природний процес пізнання

Моделювання – основний метод дослідження складних систем.

Для однієї і тієї ж системи можливе існування багатьох моделей. Конкретний вид моделі визначається цілями досліджень та індивідуальними особливостями дослідника – його знаннями, досвідом, системою уподобань тощо.

Моделювання - це метод дослідження складних систем, заснований на тому, що розглянута система замінюється на модель і проводиться дослідження моделі з метою одержання інформації про досліджувану систему.

Моделювання дозволяє замінити експерименти у реальних умовах експериментами у штучному середовищі, зокрема, на комп'ютерах.

Моделювання є основним методом кібернетики. Слід зазначити, що моделювання – це наука та мистецтво.

Моделювання є методом пізнання навколишнього світу, що полягає у створенні моделей реальних об'єктів, групи чи системи об'єктів, що взаємодіють, явищ чи процесів.

Моделювання представляє собою **універсальний метод пізнання**, який використовувався ще в давнину, хоча і не усвідомлювався як особливий метод дослідження. Використання моделювання у науковому пізнанні диктується необхідністю розкрити такі сторони об'єктів, які або неможливо осягнути шляхом безпосереднього вивчення, або непродуктивно вивчати їх у силу якихось обмежень.

Цінність застосування методу моделювання полягає в наступному:

- експерименти на моделях замість експериментів із реальною системою дозволяють заощадити колосальні засоби;
- моделювання дозволяє досліджувати невідтворювані явища: катастрофи, землетруси, урагани, фінансові кризи тощо;
- моделювання широко застосовується на стадії проєктування, коли самої системи у кінцевому її вигляді ще не існує;
- моделювання дозволяє здійснювати навчання.

Необхідність застосування моделювання на авіаційному транспорті визначається такими факторами:

- складністю розв'язуваних завдань;
- високим ступенем ризику при прийнятті рішень;
- необхідністю дуже оперативного, швидкого прийняття рішень.

У моделюванні можна виділити ряд напрямків:

- фізичне моделювання;
- математичне моделювання;
- інтелектуальне моделювання;
- гібридне моделювання;
- напівнатурне моделювання.

Гібридне моделювання передбачає комбінацію кількох методів, наприклад, математичного та інтелектуального моделювання. Прикладами є різноманітні навчальні тренажери, які включають людей, і використовуються для навчання пілотів, диспетчерів тощо, автопілот літального апарату, головка самонаведення ракети і т.п.

Найдавніший вид моделювання – матеріальний, тобто, речові моделі. Проте відомі й ідеальні моделі, наприклад, геоцентрична модель світу Пталомея. (III ст. до н. е.).Відомі також моделі Коперника, Галілея, Дж. Бруно.

Винахідник першої парової машини Дж. Уатт використовував моделі технічних систем та модельні експерименти.

В історії розвитку моделювання можна виділити три напрямки:

- фізичне моделювання;
- математичне моделювання;
- інтелектуальне моделювання.

Фізичне моделювання розвивається приблизно у 30-40 рр. XX ст. Технічні об'єкти, що створюються в цей час, досягають такого рівня складності та економічної вартості, коли обходитися без попереднього дослідження на моделях стає неможливим. У цей час бурхливо розвивається авіаційна техніка і створювати авіацію без фізичного моделювання виявилося складним, тому будуються аеродинамічні труби, в яких здійснюються продування моделей літаків.

В електроенергетиці замість окремих електростанцій з'являються енергосистеми, що поєднують десятки, сотні електростанцій. Згодом вони об'єднуються у єдину енергетичну систему. Така система є складною динамічною системою, в якій з'являється можливість втрати стійкості. Для вирішення цієї проблеми було створено електродинамічні моделі. Із застосуванням моделювання знайшли рішення, які дозволили звести до мінімуму ймовірність порушення стійкості енергосистем.

Фізичні моделі – це подібні моделі. Вони характеризуються деякими масштабними змінами, що вибираються відповідно до критерію подібності. Один із прикладів фізичної моделі – глобус (фізична модель земної кулі).

Витоки **математичного моделювання** відносяться до XVIII ст. Перші моделі було запропоновано в 1759 р. економістом Квінсні. Першою спробою представити економіку як взаємозалежну статистичну модель було створення французьким ученим XVIII ст. Ф. Кене його "Економічної таблиці". Це перша в історії моделювання макроекономічна модель натуральних та грошових потоків. Вона стала прообразом моделі Леонтьєва. Ряд моделей попиту та пропозиції були розроблені у 1874р. французьким вченим Вальрас. Математичні моделі систем масового обслуговування розроблені Ерлангому 1909 р.

У 1943 р. народжується наука, що отримала назву "Дослідження операцій". Це наука про застосування математичних моделей для прийняття рішень. Розвивається наукова основа моделювання. Зокрема, розробляється лінійне програмування, основи якого було закладено у роботах Фон Неймана (1937 р.) і Конторовича (1939 р.).

Розвивається нелінійне, цілісне, динамічне програмування. Основи динамічного програмування закладені у роботах Маркова і розвинені американським ученим Беллманом. У 1944р. виходить книга "Теорія ігор та економічна поведінка" Фон Неймана та

Моргенштерна. Широке застосування математичного моделювання пов'язане зі створенням обчислювальних машин. У 1944р. у США побудовано першу обчислювальну машину ЕНІАК; в 1946 р. - обчислювальну машину ЮНІВАК; 1951р. у СРСР створено машину – БЕСМ-2. Обчислювальні машини забезпечили технічну основу реалізації математичних моделей.

Інтелектуальне моделювання засноване на використанні систем штучного інтелекту. Термін штучний інтелект запропонований у 1956р. До складу систем штучного інтелекту входять:

1 **Експертні системи** - це системи, які моделюють функції людини, та працюють на основі досвіду та знань людини у певній вузькій галузі знань, яка називається предметною областю.

2 **Нейронні мережі** - моделюють саму структуру мозку у вигляді мережі, що складається з вузлів - нейронів, та зв'язків між ними, які називаються синапсами.Вся інформація у нейронних мережах зберігається як коефіцієнти синаптичних зв'язків. Вважається, що всього головний мозок містить приблизно 1011 нейронів та 1015 синаптичних зв'язків.

3 **Нечіткі системи** - це системи, в яких як основні елементи використовуються не числа, а лінгвістичні змінні, прикладами яких можуть служити слова: високий, сильний, красивий, розумний тощо. Нечіткі системи наближають математичні міркування до реальних схем міркування, які використовують люди у повсякденній практиці.

4 **Генетичні алгоритми** - засновані на моделюванні біологічних процесів, як у популяціях.

Перевагами математичного моделювання проти інших видів моделювання є:

- економічність, збереження ресурсів реальної системи;
- можливість моделювання гіпотетичних, тобто, не реалізованих у природі об'єктів та систем;
- можливість реалізації режимів, небезпечних чи важковідтворюваних в реальності;
- можливість зміни масштабу часу;
- універсальність технічного та програмного забезпечення, наявність пакетів прикладних програм для проведення широкого кола робіт;
- можливість прогнозування та виявлення загальних закономірностей;
- можливості порівняно простого багатофакторного аналізу.

Поняття моделі та види моделей. Евристичні, фізичні та математичні моделі. Детерміновані і стохастичні моделі. Статичні та динамічні моделі. Дискретні та неперервні моделі

Модель в науці (від лат. modulus – зразок) – будь-який образ, аналог (уявний або умовний: зображення, визначення, схема, креслення, графік, карта, формула тощо) якого-небудь об'єкта, процесу або явища(«оригіналу» цієї моделі).

Існує велика кількість визначень поняття «модель». Наприклад:

- ***модель*** – деяке наближене відображення дійсності;
- ***модель*** – представлення системи у формі, яка відрізняється від форми реальної системи;
- ***модель*** – деяка інша система, що поводиться з погляду цілей дослідження аналогічно досліджуваній системі.

Існують два підходи до відображення системи у моделі:

- ізоморфний, тобто, один до одного;
- гомоморфний, тобто, багато в одному.

Майже кожна модель гомоморфна, тому що у ній відображаються не всі, а лише суттєві властивості системи-оригіналу.

Модель ніколи не є точною копією реального об'єкта. Модель завжди простіше. Вона містить лише головні риси об'єкта. Використання моделі дозволяє спростити вивчення системи.

При моделюванні складна реальна система заміняється деякою спрощеною схемою або копією, що відбиває якісь істотні риси, дозволяє розібратися в механізмі роботи системи, дає можливість передбачити її зміну. Одній і тій же системі можуть відповідати різні моделі.

За способом відображення дійсності розрізняють три основні види моделей:

1. *Евристична модель* - містить образи-об'єкти, що виникають в уяві людини, й описані словесно.

2. *Фізична (натурна) модель* - це сукупність фізичних (матеріальних) об'єктів.

3. *Математична модель* - описує реальний об'єкт, явище чи процес за допомогою математичних співвідношень між об'єктами математичної теорії.

Модель можна описати (подати) словесно, графічно чи за допомогою комп'ютерної програми. В останньому випадку говорять про *комп'ютерну модель*.

Особливу роль у науці грають **математичні моделі** (ММ).

При побудові ММ досліджуваної системи виділяють такі їх особливості:

1) містять більш-менш повну інформацію про систему;

2) допускають математичну формалізацію.

Математична формалізація означає, що особливостям системи можна поставити у відповідність придатні адекватні математичні поняття: числа, функції, матриці тощо. Тоді зв'язки і відносини, виявлені і передбачувані в досліджуваній системі між її складовими частинами можна записати за допомогою математичних відносин: рівностей, нерівностей, рівнянь. У результаті виходить математичний опис досліджуваної системи, тобто її математична модель.

Математична модель - модель, що використовує для опису властивостей і характеристик досліджуваної системи математичні символи і методи.

Математичне моделювання - процес вивчення об'єкта дослідження шляхом створення його математичної моделі і використання її для одержання корисної інформації.

Класифікація математичних моделей:

1 В залежності від методів їх реалізації:

1.1 Аналітичні - рішення виходять в абстрактному вигляді, тобто підстановка чисел замість символів виконується вже після того, як буде отримане рішення.

1.1.1 Алгебраїчні - моделі, у яких використовується кінцеве число арифметичних операцій, операцій зведення в цілочисельну ступінь і визначення кореня.

1.1.2 Наближені - моделі, що містять формули з показовими, логарифмічними, тригонометричними тощо функціями, що припускають наближені обчислення на ЕОМ шляхом розкладання в нескінченний ряд.

1.2 Алгоритмічні - використовуються для вивчення складних систем.

1.2.1 Чисельні - модель спрощується, наприклад, переходом від системи диференціальних рівнянь до системи лінійних рівнянь.

1.2.2 Імітаційні - математичні співвідношення для досліджуваної системи взагалі не записуються, а заміняються деяким алгоритмом, що моделює її поводження.

2В залежності від можливості дослідника керувати ними:

2.1 Керовані - надають досліднику можливість вибору не тільки варіанту функціонування, але і поставлених цілей (наприклад, максимізація прибутку, мінімізація витрат тощо).

2.1.1 Оптимізаційні - пошук управління, що забезпечує найкращі (оптимальні) значення обраного показника (критерію), при заданих обмеженнях.

2.1.2 Неоптимізаційні - пошук управління, що переводить систему з хиткого положення в стійке.

2.2 Що спостерігаються - дозволяють досліднику тільки фіксувати й аналізувати стан системи, але не впливати на нього.

3За наявністю можливості обліку випадкових факторів:

3.1 Детерміновані - моделі, у яких установлене взаємоднозначна відповідність між змінними, що описують систему.

3.2 Стохастичні - моделі, у яких зв'язок між змінними носить випадковий характер.

4За характером режимів:

4.1 Статична - модель, що включає опис зв'язків між основними змінними досліджуваної системи в сталому режимі без обліку зміни параметрів у часі.

4.2 Динамічна - модель, що описує зв'язки між основними змінними досліджуваної системи при переході від одного режиму до іншого.

5За типом станів:

5.1 Безперервна - модель, у якій змінні приймають значення з деякого проміжку.

5.2 Дискретна - змінні приймають ізольовані значення.

5.3 Змішана.

6 За характером врахування характеристик системи:

6.1 Лінійна - відношення між змінними, що описують поведінку системи, представляються у вигляді лінійної функції.

6.2 Нелінійна.

Основи побудови математичних моделей виробничих процесів

Розглянемо основні характеристики моделей.

По-перше, хороша модель дуже інформативна, і ця інформація представлена у стислому вигляді.

По-друге, модель ієрархічна – є моделі вищого рівня (наприклад, модель системи управління) і нижчого рівня (наприклад, моделі елементів систем управління).

По-третє, сама модель уточнюється та коригується у процесі моделювання, тобто,недоліки моделі не можна передбачити заздалегідь.

По-четверте, модель може бути як ідеал, що ідеалізує собою різні форми діяльності: управління, планування, прийняття рішень, прогнозування тощо. Наприклад, в адаптивних (які самоналаштовуються) системах управління технічними об'єктами реалізується принцип управління за еталонною моделлю.

Головний недолік методу моделювання полягає в тому, що при некоректному моделюванні можна отримати результати, які не відносяться до досліджуваних властивостей системи або неправильно відображають властивості реальної системи. У цьому є об'єктивна причина: модель відображає (не завжди точно) лише певні, але не всі властивості реального об'єкта.

І все ж переваг у методу моделювання більше, ніж недоліків**. Можна виділити такі переваги моделей:**

1 Моделі економні, оскільки економлять час, скорочують витрати й втрати матеріальних ресурсів у процесі дослідження чи проєктування технічного об'єкта.

2 Моделі практичні, вони завжди будуються так, щоб бути простішими та зручнішими для досліджень, ніж вхідні об'єкти. На моделях можна ставити такі експерименти, проведення яких на реальних об'єктах або занадто дорого, або небезпечно персоналу та довкіллю.

3 Деякі явища можна вивчати лише на їх моделях. Наприклад, ядерні вибухи, траєкторії космічних апаратів, електричні розряди блискавки, політ літака у разі розвитку аварійної ситуації на борту внаслідок відмов окремих функціональних підсистем тощо.

4 Моделі відтворюють лише основні, найважливіші для цього дослідження властивості досліджуваної системи. Звідси випливає, що з системи (об'єкта), що вивчається, можна отримати кілька (багато) моделей, кожна з яких відтворює (імітує) певний набір властивостей і характеристик. Так, наприклад, можна побудувати та використовувати модель, що описує динамічні (спрощено, швидкісні) властивості та характеристики літака. У той же час для визначення характеристик міцності, згинально-крутильних властивостей конструктивних елементів знадобиться зовсім інша модель.

5 Моделі дозволяють виявити механізм формування досліджуваних властивостей системи, навчитися прогнозувати ці властивості та цілеспрямовано їх змінювати у бажаний бік.

6 Дослідження, проведені із застосуванням моделей, можуть бути підставою для висновку про неспроможність деяких гіпотез або ідей.

7 При моделюванні систем можуть виникнути побічні ефекти. Наприклад, модель може відтворювати такі ознаки системи, які адекватні реальним властивостям, але дана модель не була призначена для цього. Цей ефект слід розглядати як виняток, а не як закономірність, хоча в історії науки є випадки, коли подібним чином робилися відкриття у сфері тонких фізичних явищ.

Переваги моделювання роблять його найефективнішим методом як наукових досліджень, так і практичної діяльності людини.

Загальні принципи моделювання:

1 Принцип інформаційної достатності. За повної відсутності інформації про систему, що досліджується, побудова її моделі неможлива, а за наявності повної інформації - недоцільна. Існує певний критичний рівень апріорних відомостей про систему (рівень інформаційної достатності), при досягненні якого може бути побудована адекватна модель.

2 Принцип доцільності. Модель створюється задля досягнення деяких цілей, визначених на початковому етапі формулювання проблеми моделювання.

3 Принцип здійсненності. Модель повинна забезпечувати досягнення поставленої мети дослідження з ймовірністю, що суттєво відрізняється від нуля, і за кінцевий час. Зазвичай задають деяке граничне значення P_0 ймовірності досягнення мети моделювання P(t), а також граничний час досягнення цієї мети t_0. Модель вважається здійсненною, якщо $P(t) \leq P_0$ та $t \in t_0$.

4 Принцип множинності моделей. Модель повинна насамперед відбивати властивості реальної системи (явища), які впливають на обраний показник ефективності. З використанням будь-якої конкретної моделі пізнаються лише певні сторони реальності. Для повнішого її дослідження необхідний ряд моделей, які дозволяють відображати досліджуваний процес з різних сторін і з різним ступенем деталізації.

5 Принцип агрегування. Складну систему здебільшого можна представити як таку, що складається з агрегатів (підсистем), для адекватного формального опису яких виявляються придатними деякі стандартні математичні схеми. Цей принцип дозволяє досить гнучко розбудовувати модель, залежно від завдань дослідження.

6 Принцип параметризації. У ряді випадків модельована система має у своєму складі деякі відносно ізольовані підсистеми, що характеризуються певним параметром, у тому числі векторним. Такі підсистеми можна замінювати на моделі відповідними числовими величинами, а не описувати процес їх функціонування. За необхідності залежність значень цих величин від ситуації можуть задаватися таблиці, графіки чи аналітичні вирази. Принцип параметризації дозволяє скоротити обсяг та тривалість моделювання, але при цьому параметризація знижує адекватність моделі.

У процесі моделювання взаємодіють дві сторони:

- замовник;
- система підготовки розв'язання.

Замовниками можуть бути: підприємці, органи державної влади, конструкторські бюро тощо. Замовник є особою, яка приймає рішення (ОПР). Інша сторона займається обґрунтуванням рішень та може називатися аналітиком, консультантом, дослідником операції, системою підготовки рішень (СПР). Як системи підготовки рішень можуть виступати окремі особи та організації.

Процес моделювання включає наступні етапи:

1 **Постановка задачі(побудова концептуальної моделі).** На цьому етапі здійснюється формулювання проблеми та її якісний аналіз. Головне на цьому етапі чітко визначити сутність проблеми, прийняті припущення, і питання, на які необхідно отримати відповідь. Велике значення на даному етапі має досвід та інтуїція дослідника, його знання сутності модельованого явища чи процесу. Дослідник повинен мати знання предметної області, методології математичного моделювання, а також мати хорошу математичну

підготовку. Водночас побудова моделі – це творчий процес. Побудова моделі – це наука та мистецтво.

На цьому етапі виявляються основні, істотні риси, властивості, відносини, закономірності явищ, що моделюються. Відокремлюються другорядні несуттєві зв'язки та фактори. Саме на цьому етапі закладається адекватність моделі.

2 **Побудова інформаційної моделі**, тобто, моделі інформаційних зв'язків між елементами моделювання. Тут вивчається структура об'єкта, виділяються його основні елементи та взаємозв'язки, вивчається процес його функціонування.

3 **Побудова математичної моделі**. Це етап формалізації проблеми. Тут вибирається тип моделі та здійснюється формування самої моделі, тобто, модель записується як математичні залежності.

4 **Математичний аналіз моделі**. Даний етап визначає, чи є можливість вирішення задачі аналітичними методами, чи потрібно переходити до наближених, чисельних методів дослідження, тобто, використовувати імітаційне моделювання.

5 **Отримання вихідної інформації**, яка визначає характеристики (параметри) моделей та характеристики вхідних і збурюючих впливів.

Розглянемо деякі **властивості математичних моделей**, які дозволяють тією чи іншою мірою або розрізняти, або ототожнювати модель з оригіналом (об'єктом, процесом). Прийнято виділяти такі властивості математичних моделей: цілеспрямованість, адекватність, замкнутість, коректність, простота і складність, м'якість і жорсткість, кінцевість, наближеність, економічність, істинність, інформативність, повнота, адаптивність, керованість, еволюціонованість.

1 Цілеспрямованість. Модель завжди відображає деяку систему, тобто має мету.

2 Адекватність. Під адекватністю моделі прийнято розуміти правильний якісний та кількісний опис об'єкту (процесу) за вибраною множиною характеристик з певним розумним ступенем точності. Адекватність є найважливішою вимогою до моделі, вона вимагає відповідності моделі її реальному об'єкту (процесу, системі тощо) щодо обраної множини його властивостей та характеристик. При цьому мають на увазі не адекватність взагалі, а адекватність за тими властивостями моделі, що є для дослідника істотними. Повна адекватність означає тотожність між моделлю та прототипом.

Математична модель може бути адекватна щодо одного класу ситуацій (стан системи + стан зовнішнього середовища) та неадекватна щодо іншого. Застосування неадекватної моделі може призвести або до суттєвого спотворення реального процесу або властивостей (характеристик) об'єкта, що вивчається, або до вивчення неіснуючих явищ, властивостей і характеристик. Можна запровадити поняття ступеня адекватності, який змінюватиметься від 0 (відсутність адекватності) до 1 (повна адекватність). Ступінь адекватності характеризує частку істинності моделі щодо обраної характеристики (властивості) об'єкта, що вивчається. У деяких простих ситуаціях чисельна оцінка ступеня адекватності не становить особливих труднощів. Труднощі оцінки ступеня адекватності виникає через неоднозначності та нечіткості самих критеріїв адекватності, і навіть через труднощі вибору тих ознак, властивостей і показників, якими оцінюється адекватність. Поняття адекватності є раціональним поняттям, тому підвищення її ступеня слід також здійснювати на раціональному рівні. Адекватність моделі повинна перевірятися, контролюватись, уточнюватися постійно в процесі дослідження за приватними прикладами, аналогіями, експериментами тощо. В результаті перевірки адекватності з'ясовують, до чого призводять зроблені припущення: чи до допустимої втрати точності, чи до втрати якості. При перевірці адекватності також можна обґрунтувати законність застосування прийнятих робочих гіпотез під час вирішення завдання чи проблеми.

3 Замкнутість. Математична модель є замкнутою, якщо вона враховує та відображає замкнуту (повну) систему необхідних гіпотез, зв'язків та відносин. Контроль математичної замкнутості полягає у перевірці того, що виписана система математичних співвідношень дає можливість однозначно вирішити поставлене математичне завдання. Наприклад, якщо завдання звелося до пошуку n невідомих із деякої системи алгебраїчних або трансцендентних рівнянь, то контроль замкнутості полягає у перевірці того факту, що кількість незалежних рівнянь має бути n. Якщо їх менше n, то треба встановити рівняння, що відсутні, а якщо їх більше n, то або рівняння залежні, або при їх складанні припущена помилка. Однак якщо рівняння виходять з експерименту або в результаті спостережень, можлива постановка задачі, при якій число рівнянь перевищує n, але самі рівняння задовольняються лише приблизно, а рішення шукають, наприклад, за методом найменших квадратів. Нерівностей серед умов також може бути будь-яке число, як це буває, наприклад, в задачах лінійного програмування. Властивість математичної замкнутості системи математичних співвідношень тісно пов'язане із запровадженим Ж. Адамаром поняттям коректно поставленого математичного завдання.

4 Коректність. Математична модель є коректною, якщо для неї здійснено та отримано позитивний результат усіх контрольних перевірок: розмірності, порядків, характеру залежностей, екстремальних ситуацій, початкових та граничних умов, фізичного сенсу та математичної замкнутості.

Математична задача є коректно поставленою, якщо її вирішення існує, воно єдине і безперервно залежить від вихідних даних. У цьому випадку рішення вважається безперервним, якщо малої зміни вихідних даних відповідає досить мала зміна рішення. Доказ коректності конкретної задачі часто є складною математичною проблемою. Математична модель є коректною, якщо для неї здійснено та отримано позитивний результат усіх контрольних перевірок: розмірності, порядків, характеру залежностей, екстремальних ситуацій, граничних умов, фізичного сенсу та математичної замкнутості.

5 Простота та складність. Одночасна вимога простоти та адекватності моделі є суперечливою. З погляду адекватності складні моделі є кращими за прості. У складних моделях можна врахувати більшу кількість факторів, що впливають на характеристики об'єктів, що вивчаються. Хоча складні моделі і більш точно відображають властивості оригіналу, які моделюються, але вони більш громіздкі, важкооглядові та незручні у використанні. Тому дослідник прагне спрощення моделі, оскільки простими моделями легше оперувати. При прагненні до побудови простої моделі повинен дотримуватися основний принцип спрощення моделі: спрощувати модель можна доти, доки зберігаються основні властивості, характеристики та закономірності, притаманні оригіналу. Цей принцип свідчить про межу спрощення. При цьому поняття простоти (або складності) моделі є відносним поняттям. Модель вважається досить простою, якщо сучасні засоби дослідження (математичні, інформаційні, фізичні) дають можливість провести якісний та кількісний аналіз з необхідною точністю. А оскільки можливості засобів досліджень безперервно зростають, то завдання, які раніше вважалися складними, тепер можуть бути віднесені до категорії простих. Найважчим завданням є забезпечення простоти/складності моделі складної системи, що складається з окремих підсистем, з'єднаних одна з одною у деяку ієрархічну та багатозв'язкову структуру. При цьому кожна підсистема та кожний рівень мають свої локальні критерії складності та адекватності, відмінні від глобальних критеріїв системи. З метою меншої втрати адекватності спрощення моделей доцільніше проводити: 1) на фізичному рівні із збереженням основних фізичних співвідношень; 2) на структурному рівні із збереженням основних системних властивостей. Спрощення моделей на математичному рівні може призвести до істотної втрати ступеня адекватності. Наприклад, усічення характеристичного рівняння високого порядку до 2–3-го порядку може призвести до абсолютно неправильних висновків про динамічні властивості системи. Прості моделі використовуються під час вирішення завдання синтезу, а складніші точні моделі – під час вирішення завдання аналізу.

6 Жорсткість та м'якість моделі. Прикладом жорсткої моделі є таблиця множення. Найпростіший приклад м'якої моделі – принцип "чим далі у ліс, тим більше дров". Можливість корисної математичної теорії м'яких моделей відкрита відносно недавно (Арнольд). Гармонійний осцилятор – приклад так званої «жорсткої» моделі. Вона отримана внаслідок сильної ідеалізації реальної фізичної системи. Для вирішення питання про її застосування необхідно зрозуміти, наскільки істотними є фактори, якими ми знехтували.

Іншими словами, потрібно досліджувати «м'яку» модель, що виходить малим збуренням «жорсткої». Якщо система зберігає свою якісну поведінку при малому збуренні, то кажуть, що вона структурно стійка. Гармонійний осцилятор – приклад структурно-нестійкої (негрубою) системи. Тим не менш, цю модель можна застосовувати для вивчення процесів на обмежених проміжках часу.

7 Кінцевість моделей. Відомо, що світ нескінченний, як і будь-який об'єкт, у просторі та часі, а також у структурі (будові), властивостях, відносинах з іншими об'єктами. Нескінченність проявляється в ієрархічній будові систем різної фізичної природи. Проте щодо об'єкта дослідник обмежується кінцевою кількістю його властивостей, зв'язків, використовуваних ресурсів тощо. Він хіба що «вирізує» з нескінченного світу певний кінцевий фрагмент конкретного об'єкта, системи, процесу тощо та намагається пізнати нескінченний світ через кінцеву модель цього фрагмента. Кінцевість моделей систем полягає, по-перше, в тому, що вони відображають оригінал в кінцевому числі відносин, тобто, з кінцевим числом зв'язків з іншими об'єктами, з кінцевою структурою і кінцевою кількістю властивостей на даному рівні вивчення, дослідження, опису ресурсів. По-друге, у тому, що ресурси (інформаційні, фінансові, енергетичні, тимчасові, технічні тощо) моделювання та наші знання як інтелектуальні ресурси є кінцевими, а тому об'єктивно обмежують можливості моделювання і сам процес пізнання світу через моделі. Тому дослідник (за рідкісним винятком) має справу з кінцевими моделями.

Вибір розмірності моделі (її ступеня свободи, змінних стану) тісно пов'язаний з класом розв'язуваних завдань. Збільшення розмірності моделі пов'язане з проблемами складності та адекватності. При цьому необхідно знати, яка функціональна залежність між ступенем складності та розмірністю моделі. Якщо ця залежність ступенева, проблема може бути вирішена за рахунок застосування обчислювальних систем. Якщо ж ця залежність експоненційна, то «прокляття розмірності» (Р. Калман) неминуче і позбутися його практично не вдається.

8 Наближеність моделей. Кінцевість і простота (спрощеність) моделі характеризують якісну різницю (на структурному рівні) між оригіналом і моделлю. Тоді наближеність моделі характеризуватиме кількісний бік цієї відмінності. Можна ввести кількісну міру наближеності шляхом порівняння, наприклад, грубої моделі з більш точною еталонною (повною, ідеальною) моделлю або з реальною моделлю. Наближеність моделі до оригіналу неминуча, існує об'єктивно, оскільки модель як інший об'єкт відбиває лише окремі властивості оригіналу. Тому рівень наближеності (близькості, точності) моделі до оригіналу визначається постановкою завдання, метою моделювання.

Надмірне прагнення до підвищеної точності моделі призводить до її значного ускладнення, і, отже, зниження її практичної цінності. При моделюванні складних (людино-машинних, організаційних) систем точність і практичний сенс не сумісні та виключають одне одного. Причина суперечливості та несумісності вимог точності та практичності моделі криється в невизначеності та нечіткості знань про сам оригінал – його поведінку, його властивості та характеристики, про поведінку навколишнього середовища, про механізми формування мети, шляхів та засоби її досягнення тощо.

9 Економічність моделей. Дана властивість математичних моделей визначається витратами ресурсів (людських, матеріальних, тимчасових, обчислювальних та ін.) на їх реалізацію та експлуатацію.

10 Істинність моделей. У кожній моделі є частка істини, тобто, будь-яка модель у чомусь правильно відображає оригінал. Ступінь істинності моделі виявляється лише за практичного порівняння її з оригіналом. Що ж до малих змінних, то ними нехтують зазвичай під час вирішення завдання синтезу, але намагаються врахувати їх вплив на властивості системи під час вирішення завдання аналізу. При моделюванні прагнуть по можливості виділити невелику кількість основних факторів, вплив яких одного порядку і не дуже складно описується математично, а вплив інших факторів можливо врахувати за допомогою середніх, інтегральних або "заморожених" характеристик.

З одного боку, у будь-який моделі міститься безумовно істинне, тобто, безумовно відоме та правильне. З іншого боку, модель містить і умовно істинне, тобто, правильне лише за певних умов. Типова помилка при моделюванні полягає в тому, що дослідники застосовують ті чи інші моделі без перевірки умов їх істинності, меж їх застосування. Такий підхід призводить свідомо до отримання неправильних результатів. У будь-якій моделі також міститься щось, що може бути в умовах невизначеності або вірним, або хибним. Тільки на практиці встановлюється фактичне співвідношення між істинним та хибним у конкретних умовах. Отже, під час аналізу рівня істинності моделі необхідно з'ясувати: 1) точні, достовірні знання; 2) знання, достовірні за певних умов; 3) знання, що оцінюються з деяким ступенем невизначеності; 4) знання, що не піддаються оцінці навіть із певним ступенем невизначеності; 5) незнання, тобто те, що невідомо. Таким чином, оцінка істинності моделі як форми знань зводиться до виявлення змісту в ній як об'єктивних достовірних знань, так і знань, що приблизно оцінюють оригінал, а також те, що становить незнання.

11 Інформативність. Модель повинна містити достатню інформацію про систему (в рамках гіпотез, прийнятих при побудові моделі) і давати можливість отримати нову інформацію.

12 Повнота. У моделі мають бути враховані всі основні зв'язки та відносини, необхідні для забезпечення досягнення мети моделювання.

13 Адаптивність. Модель має бути пристосована до різних вхідних параметрів, впливів зовнішнього середовища.

14 Керованість. Модель повинна мати хоча б один параметр, змінами якого можна імітувати поведінку моделі, що моделюється в різних умовах.

15 Еволюціонованість. Можливість розвитку моделей.

Метод математичного моделювання, який зводить дослідження явищ зовнішнього світу до математичних задач, посідає провідне місце серед інших методів досліджень, особливо завдяки наявності обчислювальної техніки. Він дозволяє проєктувати нові технічні засоби, що працюють в оптимальних режимах, для розв'язання складних задач науки і техніки та передбачати нові явища. Математичні моделі зарекомендували себе важливим засобом управління. Вони застосовуються у різних галузях знань, стали необхідним апаратом економічного планування і важливим елементом автоматизованих систем управління.

Інформаційне забезпечення моделей

Побудова математичних моделей вимагає відповідного інформаційного забезпечення. Забезпечення – це те, за допомогою чого створюються умови для математичного моделювання.

Інформаційне забезпечення **–** це сукупність інформації та способів її пошуку, обробки, накопичення, збереження, систематизації та узагальнення з метою використання в процесі математичного моделювання.

Завданням інформаційного забезпечення є інформування про стан об’єктів, що досліджуються.

Головним гальмом практичногозастосування математичного моделювання є наповнення розроблених моделей конкретною і якісною інформацією. Точність і повнота первинної інформації, реальні можливості її збору і обробки багато в чому визначають вибір типів прикладних моделей. З іншого боку, дослідження з моделювання висувають нові вимоги до системи інформації.

Інформація – сукупність відомостей (повідомлень, даних), яка визначає міру наших знань про ті чи інші явища, події та їх взаємозв'язки (рис. 1.3). Дане визначення використовується у широкому розумінні слова. У вузькому розумінні ***інформація*** - це відомості, які є об'єктом збору, обробки, передачі та зберігання.

Залежно від модельованих об'єктів і призначення моделей використовувана в них вихідна інформація має істотно різний характер і походження. Вона може бути розділена на

дві категорії: про минулий розвиток і сучасний стан об'єктів (спостереження та їх обробка) і про майбутній розвиток об'єктів, що включає дані про очікувані зміни їх внутрішніх параметрів і зовнішніх умов (прогнози). Друга категорія інформації є результатом самостійних досліджень, які також можуть виконуватися за допомогою моделювання.

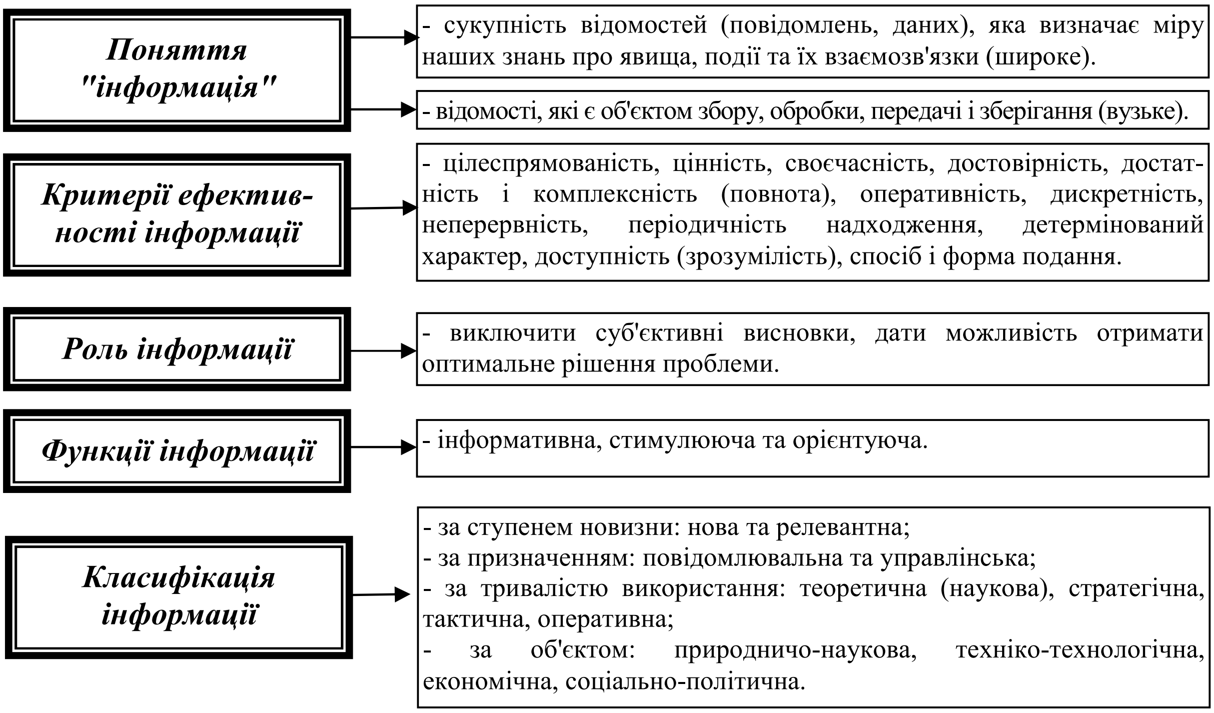


Рисунок 1.3 – Поняття, критерії ефективності, роль, функції та класифікація інформації

Якість та ефективність інформації визначається за такими ***критеріями***: цілеспрямованість, цінність, своєчасність, достовірність, достатність і комплексність (повнота), оперативність, дискретність, неперервність, періодичність надходження, детермінований характер, доступність (зрозумілість), спосіб і форма подання.

Основна роль інформації в дослідженнях полягає в тому, щоб виключити суб'єктивні висновки, дати можливість отримати оптимальне рішення проблеми. Рівень наукових досліджень залежить від достовірності, ступеня використання інформації і здатності дослідника переробити отриману інформацію. Детальніше дослідження цих зв'язків потребує вирішення питання про те, які ***функції повинна виконувати інформація***. Такими функціями є інформативна, стимулююча та орієнтуюча.

Суть ***інформативної функції*** полягає в тому, щоб дати знання, відомості про той чи інший об'єкт і предмет дослідження. Реалізація ***стимулюючої функції*** дозволяє привести дослідників до нової постановки питання, нового його вирішення, з тим, щоб вдосконалювати практику. ***Орієнтуюча функція*** відображається у положеннях, нормах, цільових настановах, які дослідники сприймають як обов'язкову суспільну регламентацію, щоб в найкоротший термін досягти необхідних наукових результатів. Всі функції інформації взаємопов'язані і в поєднанні сприяють розвитку творчості у дослідній діяльності.

Інформацію класифікують за різними ознаками:

За ступенем наукової новизни розрізняють:

а) *нову інформацію*, що відображає новизну запропонованого рішення теоретичного або практичного завдання;

б) *релевантну*, яка раніше містилась в аналогах (наприклад, в методичних вказівках).

За призначенням виділяють:

а) *повідомлювану інформацію*, що отримана в процесі дослідження;

б) *управлінську інформацію*, яка необхідна для прийняття управлінських рішень.

За тривалістю періоду, протягом якого інформація зберігає свою актуальність і використовується для прийняття рішень, інформацію класифікують на:

а) *теоретичну (наукову) інформацію* - це результати фундаментальних чи прикладних наукових досліджень в різних областях, які широко використовуються на авіаційному транспорті;

б) *стратегічну* - інформація, що зберігає актуальність протягом тривалих періодів (10-15 років): довготривалі плани і прогнози функціонування авіаційного транспорту, дані про повільно змінювані об'єкти авіаційного транспорту, проєктно-конструкторська документація;

в) *тактичну (кон'юнктурну)* - інформація з періодом актуальності 2-3 роки і менше;

г) *оперативну* - інформація, що актуальна в межах одного циклу оперативного управління на авіаційному транспорті.

Залежно від об'єкту, про який відображається інформація, вона буває:

а) *природничо-наукова* - характеризує зв'язки між природними об'єктами;

б) *техніко-технологічна* - відображає взаємозв'язки між предметами природи, які стосуються технології та технічних засобів;

в) *економічна* - розкриває відносини між людьми в процесі виробництва, розподілу, обміну і споживання;

г) *соціально-політична* - інформація про соціальні, політичні, ідеологічні відносини між людьми.

Одним із найважливіших видів джерел дослідження є літературні і, насамперед, наукові документи. ***Науковий документ*** - це матеріальний об'єкт, який містить наукову інформацію з певною логічною завершеністю і призначений для її зберігання, передачі і використання. Сукупність наукових документів складає ***науково-технічну літературу***. Це - матеріальна форма існування науки.

Основними літературними джерелами науково-технічної інформації є:

- книги (монографії, підручники, навчальні посібники),
- періодичні видання (журнали, бюлетені, праці інститутів, наукові збірники),
- матеріали конференцій (тези доповідей),
- нормативні документи (міжнародні конвенції та протоколи, стандарти і рекомендована практика ICAO (SARPs), стандарти Європейської об’єднаної авіаційної влади (JAR), документи Євроконтролю (ESARRs), Повітряний кодекс, Авіаційні правила України (АПУ), накази Державіаслужби тощо),
- каталоги і прейскуранти,
- патентна документація (патенти, авторські свідоцтва),
- звіти про науково-дослідні та дослідно-конструкторські роботи,
- інформаційні видання (збірники НТІ, аналітичні огляди, інформаційні листки, експрес - інформації, виставочні проспекти та ін.),
- переклади іноземної науково-технічної літератури,
- матеріали науково-технічних і виробничих нарад,
- дисертації та автореферати,
- виробничо-технічна документація організацій (звіти, акти приймання об'єктів в експлуатацію та ін.).

З метою удосконалення інформаційного забезпечення моделей доцільно використовувати відповідні автоматизовані системи.

1.5 Висновки по лекції.

Кібернетика є наукою про загальні принципи управління в комплексі складними системами різної природи походження (соціальними, технічними, біологічними, економічними тощо) на основі знань, які сформовані на зворотних зв'язках. Кожний об'єкт управління – це система, що складається із взаємозалежних елементів. **Системи класифікують** за метою, видом елементів, рівнями управління, ступенем складності, числом та типом станів, умовами переходів з одного стану в інший, походженням, характером взаємодії із зовнішнім середовищем, наявністю циклів, розмірами, характером змін внутрішніх властивостей системи, наявністю та характером зворотного зв'язку.

Дослідження операцій займається розробкою та застосуванням методів знаходження оптимальних рішень на основі математичного моделювання у різних областях людської діяльності. **Метою дослідження операцій** є попереднє кількісне обґрунтування оптимальних рішень, яких може бути більше ніж одне.

Моделювання - це метод дослідження складних систем, заснований на тому, що розглянута система заміняється на модель і проводиться дослідження моделі з метою одержання інформації про досліджувану систему. Під **моделлю** досліджуваної системи розуміється деяка інша система, що поводиться з погляду цілей дослідження аналогічно досліджуваній системі. За способом відображення дійсності розрізняють евристичні, фізичні (натурні) та математичні моделі.

Загальними принципами моделювання є принцип інформаційної достатності, принцип доцільності, принцип здійсненності, принцип множинності моделей, принцип агрегування, принцип параметризації. **Процес моделювання включає наступні етапи**: постановка задачі(побудова концептуальної моделі); побудова інформаційної моделі; побудова математичної моделі; математичний аналіз моделі; отримання вихідної моделі.

Математична модель використовує для опису властивостей і характеристик досліджуваної системи математичні символи і методи. Прийнято виділяти такі **властивості математичних моделей**: цілеспрямованість, адекватність, замкнутість, коректність, простота і складність, м'якість і жорсткість, кінцевість, наближеність, економічність, істинність, інформативність, повнота, адаптивність, керованість, еволюціонованість.

Побудова математичних моделей вимагає відповідного **інформаційного забезпечення –** сукупності інформації та способів її пошуку, обробки, накопичення, збереження, систематизації та узагальнення з метою використання в процесі математичного моделювання. ***Основна роль інформації*** в дослідженнях полягає в тому, щоб виключити суб'єктивні висновки, дати можливість отримати оптимальне рішення проблеми. ***Функції інформації:*** інформативна, стимулююча та орієнтуюча.

Тема 2 *Моделювання організації виробничих процесів на авіаційному транспорті методами лінійного програмування* (2 год.)

2.1 Мета та завдання лекції

Метою лекції є ознайомлення з роллю математичного та лінійного програмування в організації виробничих процесів на авіаційному транспорті.

Завдання лекції:

- ознайомити з загальною постановкою задачі та моделями математичного програмування;
- розглянути області застосування лінійного програмування при вирішенні задач оптимізації виробничого процесу;
- охарактеризувати завдання лінійного програмування, представити канонічну форму та геометричну інтерпретацію задачі лінійного програмування, розглянути симплекс-метод рішення задачі лінійного програмування;
- розглянути постановку, математичну модель, області застосування та метод потенціалів рішення транспортної задачі;
- розглянути задачу про призначення як приватний випадок транспортної задачі та приклади моделювання виробничих процесів.

2.2 План лекції

2.1 Загальна постановка задачі математичного програмування. Класифікація моделей математичного програмування.

2.2 Області застосування лінійного програмування при вирішенні задач оптимізації виробничого процесу на авіаційному транспорті.

2.3 Задача лінійного програмування. Канонічна форма задачі лінійного програмування. Геометрична інтерпретація задачі лінійного програмування. Симплекс-метод.

2.4 Постановка транспортної задачі лінійного програмування, її математична модель та області застосування. Рішення транспортної задачі методом потенціалів.

2.5 Задача про призначення як приватний випадок транспортної задачі. Рішення задачі про призначення угорським методом. Приклади моделювання виробничих процесів на авіаційному транспорті у формі транспортної задачі.

2.3 Основні категорії, ключові поняття та визначення теми

Задача про призначення – одна з базових задач комбінаторної оптимізації, яка полягає в знаходженні парування мінімальної (або максимальної) ваги між елементами двох скінчених множин. Вона може бути подана як знаходження парування у зваженому дводольному графі.

Канонічна форма **–** така форма, що однозначно репрезентує об'єкт.

Критерій оптимізації – значення кількісного показника або правило (співвідношення), що характеризують екстремум (максимум або мінімум) цільової функції системи.

Лінійне програмування або ***лінійна оптимізація*** (англ. Linear Programming) – метод досягнення найліпшого виходу (такого, як найбільший прибуток або найменша вартість) у математичній моделі, чиї вимоги подані через лінійні відношення. Лінійне програмування є особливим випадком математичного програмування (математичної оптимізації).

Математичне програмування – це прикладна математична дисципліна, яка досліджує екстремум функції (задачі пошуку максимуму або мінімуму) і розробляє методи їх розв'язання. Такі задачі ще називають оптимізаційними.

Метод потенціалів **–** це метод послідовного покращення плану (перевезень) з використанням другої теореми двоїстості для перевірки оптимальності.

Опорний план – розв'язок системи лінійних обмежень взадачі лінійного програмування, який неможливо представити у вигляді лінійної комбінації будь-яких інших розв'язків.

Симплекс-метод **–** метод розв'язання задачі лінійного програмування, в якому здійснюється скерований рух за опорними планами до знаходження оптимального розв'язку; симплекс-метод також називають методом поступового покращення плану.

Транспортна задача **–** задача про оптимальний план перевезення продукції із пунктів відправлення до пунктів споживання.

Угорський метод **–** це метод послідовної побудови допустимого плану, який автоматично виявляється оптимальним. В основі угорського алгоритму лежить метод чергування ланцюгів.

2.4 Текст лекції

Загальна постановка задачі математичного програмування. Класифікація моделей математичного програмування

Математичне програмування досліджує *екстремальні задачі* (задачі пошуку *максимуму* або *мінімуму*) і розробляє *методи їх розв'язання*. Такі задачі ще називають *оптимізаційними*.

Математичне програмування дозволяє вирішувати багато управлінських, організаційних та інженерних задач *оптимальним чином.* Прикладом використання знань з математичного програмування може бути розв’язання таких виробничих задач:

- отримання *максимального прибутку* або випуску *максимального об’єму* продукції при заданих матеріальних, трудових, енергетичних або часових витратах;
- забезпечення планових показників підприємства при *мінімальному розмірі фінансових вкладень*;
- досягнення *максимально короткого терміну* виготовлення продукції, будівництва об'єкту, виробничого циклу і тому подібного при існуючих або заданих виробничих ресурсах;
- вибір параметрів об’єкту або процесу, при яких забезпечується його максимальна корисність.

В наведених прикладах *максимальний випуск* продукції, *максимальний прибуток*, *мінімальний розмір фінансових вкладень*, *максимально короткий термін, максимальна корисність* – це є *оптимуми* (*максимуми* або *мінімуми*), тобто результати, які при заданих умовах задачі неможливо перевершити.

В свою чергу, умови, які накладаються на можливі рішення задач (*задані матеріальні, трудові і часові витрати; виробничі ресурси; можливі діапазони значень параметрів або планових показників)*, називають *обмеженнями* задачі.

Оптимальне рішення задачі – це рішення, що обов’язково задовольняє обмеженням задачі.

Задачу математичного програмування можна подати в *змістовній (вербальній)* або *формальній постановці*.

Оптимізаційна задача в змістовній постановці

Змістовна постановка задачі – це її словесний опис. Розглянемо приклад оптимізаційної задачі в змістовній постановці.

Приклад 2.1. Для обшивки пасажирських крісел літака підприємство використовує шкірозамінник. Обшивка крісла в салоні економ-класу потребує 2 м^2 матеріалу, а в салоні першого класу – 4 м^2. Трудомісткість складає: крісло для салону економ-класу – 4 чол.-год., крісло для салону першого класу – 3 чол.-год. Прибуток від реалізації становить: крісло для салону економ-класу– 6000 грн., крісло для салону першого класу – 8000 грн. Підприємство для виготовлення крісел у своєму розпорядженні має 200 м^2 шкірозамінника та 600 чол.-год. Фонду робочого часу. Визначити, скільки крісел для салонів економ-класу та першого класу треба обшити, щоб прибуток від реалізації був максимальним.

Вихідні дані задачі доцільно звести в таблицю (табл. 2.1), що є зручною та наочною формою при розподілі або групуванні початкових даних, а в подальшому полегшує формування математичної моделі задачі.

Таблиця 2.1 – Вихідні дані прикладу2.1

Вид сировини	Норми витрат ресурсівна один виріб		Загальна кількість ресурсів
	Крісло для салону економ-класу	Крісло для салону першого класу	
Шкірозамінник,м^2	2	4	400
Трудомісткість, чол.-год.	4	3	600
Прибуток від реалізації одного крісла, грн.	6000	8000	

Оптимізаційна задача в формальній постановці

Змістовна постановка задачі повинна дозволяти переходити до строгої *математичної моделі*. У протилежному випадку необхідно пройти досить трудомісткі й кропіткі процеси математичного моделювання й ідентифікації, які в цьому курсі не розглядаються.

Розглянемо на прикладі 1 поетапний *процес побудови математичної моделі* задачі.

1. Визначимо *невідомі* задачі. Як правило, у якості невідомих виступають ті величини, які треба визначити за умовами задачі. В прикладі 1 такими величинами є *кількість шкірозамінника для крісел для салонів економ-класу та першого класу*. Позначимо ці кількості як x_1 і x_2відповідно.

2. Сформуємо *цільову функцію Y,* тобто функцію, оптимум якої треба встановити за вимогами задачі. Цільова функція це *критерій*, за яким визначається найкраще рішення. У даному випадку критерієм є функція, що виражає прибуток від обшивкихx_1кріселдля салону економ-класу та x_2кріселдля салону першого класу. Цільова функція для даного прикладу має вигляд (2.1).

$$Y(x_1,x_2)=6000x_1+8000x_2. (2.1)$$

Функція (1.1) визначає прибуток підприємства від реалізації всіх виробів.

3. Сформуємо математичну модель задачі без урахування обмежень задачі, так званої задачі безумовної оптимізації (2.2):

$Y(x_1,x_2)=6000x_1+8000x_2 \rightarrow \max$. (2.2)

Тут символи «$\rightarrow$ max» говорять, що треба знайти не аби які значення змінних x_1 і x_2 , а саме ті, що забезпечують оптимум цільової функції, у даному разі – її максимум. При цьому рішенням задачі безумовної оптимізації (2.2) може бути будь-яка точка двовимірного евклідового простору R^2(безмежної площини). Тому повний запис задачі (2.2) має вигляд

$$\underset{x_1,x_2\in R^2}{Y(x_1,x_2)=6000x_1+8000x_2 \rightarrow \max}.$$

4. Визначимо *обмеження* задачі Ω , тобто, область допустимих рішень.

По-перше, загальні витрати шкірозамінника на обшивку всіх крісел не можуть перевершувати 400 м^2 (2.3):

$$f_1(x_1,x_2) =2x_1+4x_2 \leq 400.(2.3)$$

По друге, загальні витрати робочого часу на обшивку всіх крісел не можуть перевершувати 600 чол.-год. (2.4):

$f_2(x_1,x_2)=4x_1+3x_2 \leq 600.$(2.4)

Крім того, кількості виробів кожного виду не можутьбути від'ємними,тобто (2.5):

$x_1, x_2 \geq 0.$(2.5)

До того ж невідомі (змінні) задачі за своєю суттю є цілочисловими величинами (2.6):

$x_1, x_2=\text{int}.$(2.6)

Вирази (2.3) – (2.6) враховують всі обмеження задачі, тому область допустимих рішень задачі визначається як:

$$\Omega: f_1(x_1,x_2)=2x_1+4x_2 \leq 400;$$
$$f_2(x_1,x_2)=4x_1+3x_2 \leq 600;\ x_1,\ x_2 \geq 0;$$
$$x_1,\ x_2= \text{int}.$$

Тут запис «Ω:» говорить, що далі (після символу «:») ідуть вирази, які визначають властивості кожного елементу (кожної точки) множини Ω.

Безумовно, множина точок Ω належить евклідовому простору R^2і складає тільки частину всіх точок простору $\Omega \subset R^2$.

5. Нарешті, сформуємо завершальну *математичну модель задачі* (з урахуванням обмежень) (2.7) – (2.11):

$\underset{x\in\Omega}{Y(x_1,x_2)=6000x_1+8000x_2 \rightarrow \max}$, (2.7)

Ω: $f_1(x_1,x_2)=2x_1+4x_2 \leq 400;$(2.8)

$f_2(x_1,x_2)=4x_1+3x_2 \leq 600;$(2.9)

$$x_1,\ x_2 \geq 0;\ (2.10)$$

x_1, x_2=int. (2.11)

Задача (2.7) – (2.11) є задачею <u>цілочислового лінійного програмування</u>. Процес рішення таких задач ми розглянемо пізніше, а поки дамо лише відповідь. Рішення задачі: x_1= 50, x_2= 150.

Якщо підприємство здійснить обшив 50 крісел для салону економ-класу і 150 крісел для салону першого класу, то вона отримує максимальний прибуток у розмірі $Y = 6000 \cdot 50 + 8000 \cdot 150 = 1500000$*грн*.

В задачі (2.7) – (2.11) пошук оптимального рішення здійснюється серед точок множини Ω, що є меншою в порівнянні з множиною R^2. Цю задачу, завдяки її малій вимірності та цілочислових змінних, можна вирішити шляхом *прямого перебору*: визначити усі точки простору Ω, знайти значення цільової функції в усіх точках простору Ω, порівняти їх та визначити найкраще значення цільової функції (у даному випадку максимум) і відповідну точку простору Ω.

Задача (2.7) – (2.11) в порівнянні з іншими задачами математичного програмування, що зустрічаються в практиці, має незначну складність. Але навіть для такої задачі пошук оптимального рішення методом повного перебору потребує багато зусиль і часу.

Предметом математичного програмування є способи математичного моделювання оптимізаційних задач, визначення необхідних і достатніх умов наявності екстремумів (оптимумів), розробка і дослідження методів визначення оптимальних рішень, які обминають пошук екстремальних рішень прямим перебором.

В загальному випадку математична постановка екстремальної задачі полягає в визначенні найменшого або найбільшого значення цільової функції $y(x_1, x_2, ..., x_n)$ при умовах $f_j(x_1, x_2, ..., x_n) \le b_j$ ($i = 1, n$; $j = 1, m$), $x^+ \le x \le x^{++}$, де $y(X)$ і $f_j(X)$– задані функції;а b, x^+, x^{++} – деякі дійсні числа;$X = x_1, x_2, ..., x_n$ – вектор змінних x.

Найбільш поширені поняття та визначення математичного програмування:

– *цільова функція, цільова квадратична форма, функція плану, критерій оптимізації* – функція, для якої треба визначити оптимальне рішення або знайти екстремальне значення;

– *оптимум (максимум або мінімум)* – найбільше (при максимізації) або найменше (при мінімізації) значення цільової функції y;

– *оптимальне рішення, оптимальний план, оптимальна точка* – значення змінних оптимізаційної задачі, при яких цільова функція набуває екстремального значення;

– *область допустимих рішень* – множина точок, серед яких шукають оптимальне рішення;

– *обмеження задачі у вигляді рівностей* – система рівностей або нерівностей, можливі рішення якої формують область допустимих рішень;

– *двобічна обмеженість змінних* – вираз, що визначає відрізок можливих значень змінних;

– *загальна задача математичного програмування* – задача пошуку оптимального рішення або оптимуму нелінійної цільової функції;

Залежно від властивостей функцій y та f_j математичне програмування розпадається на декілька самостійних дисциплін, що займаються дослідженням і розробкою методів розв'язання окремих класів задач. На рис. 2.1 подана **класифікація задач математичного програмування.**

Рисунок2.1 – Класифікація задач математичного програмування

Перед усім задачі математичного програмування поділяються на детерміновані задачі, задачі стохастичного та задачі динамічного програмування. **Динамічне програмування** – це розділ математичного програмування, що пов'язаний з вирішенням екстремальних задач спеціальної структури, а саме задач, в яких процес пошуку оптимального рішення є багатоетапним. **Стохастичне програмування** має справу з екстремальними задачами, в постановці яких присутні випадкові величини, залежні від різних факторів. **Детерміновані задачі** – це найбільш поширений клас задач математичного програмування. Вхідна інформація в таких задачах є повністю визначеною. Всі детерміновані задачі поділяються на задачі лінійного чи нелінійного програмування.

В **задачах нелінійного програмування** цільова функція та (або) обмеження є нелінійними функціями. В нелінійному програмуванні виділяють клас багатоекстремальних задач та клас задач опуклого програмування. В **багатоекстремальних задачах** цільова функція має декілька екстремумів. В **задачах опуклого програмування** – тільки один. Опукле програмування об'єднує три підкласи екстремальних задач: задачі при двобічних обмеженнях змінних і відсутності обмежень у вигляді рівнянь; задачі квадратичного програмування, які пов'язані з пошуком екстремуму квадратичної функції при лінійних обмеженнях; задачі в загальній постановці, тобто ті, що не належать до двох попередніх підкласів.

В **задачах лінійного програмування** цільова функція та всі обмеження є лінійними функціями. Лінійне програмування об'єднує наступні підкласи задач: задачі дискретного (цілочислового) програмування; задачі дрібно-лінійного програмування; задачі параметричного програмування; підклас транспортних задач. В **задачах дискретного (цілочислового) програмування** невідомі (змінні) можуть приймати тільки цілочислові значення. У **задачах дрібно-лінійного програмування** цільова функція представляє собою відношення двох лінійних функцій, а функції, що визначають область допустимих рішень, є звичайними лінійними функціями. У **задачах параметричного програмування** цільова функція або функції обмежень, або й те й інше залежать від деяких параметрів (коефіцієнти можуть змінюватися в деяких межах). Окремими класами лінійних задач є **транспортні задачі**, в яких змінні подаються у вигляді матриці, а коефіцієнти в обмеженнях можуть приймати тільки два значення: 1 або 0.

Області застосування лінійного програмування при вирішенні задач оптимізації виробничого процесу на авіаційному транспорті

До оптимізаційних задач можна віднести наступні класи задач:

– *задачі планування виробництва* (планування виконання авіаперевезень, завантаження обладнання авіаремонтного заводу, фінансування проектів будівництва терміналів, розподіл парку повітряних суден, календарне планування, мережеве планування);

– *задачі організації виробництва* (формування парку повітряних суден, про призначення авіаційного персоналу, про реконструкцію аеропорту, про розташування ангарів, про закриття авіакомпанії);

– *транспортні задачі* (перевезення вантажів з максимальним завантаженням літаків, максимальним об'ємом перевезень та мінімальними витратами, розподіл літаків за повітряними лініями, розміщення вантажного повітряного флоту);

– *комбінаторні задачі* (про завантаження літака(про рюкзак), про лінійний розкрій матеріалів на авіабудівному заводі, про розподіл пам'яті автоматизованої системи управління літаком, про розрахунок траєкторії літака (задача комівояжера)).

Задача лінійного програмування. Канонічна форма задачі лінійного програмування. Геометрична інтерпретація задачі лінійного програмування. Симплекс-метод

Лінійне програмування є найбільш розробленим розділом математичного програмування, пріоритет у якому належить радянському математикові Л.В. Канторовичу, який в 1937 році розглянув спеціальний клас задач лінійного програмування й запропонував метод їх розв’язання.

В задачі лінійного програмування всі функції $f_i(X)$ лінійні. Лінійність функції $f_i(X)$ означає, що, якщо вектор X містить тільки одну змінну, то ця функція на рисунку може бути зображена у вигляді прямої лінії, якщо змінних дві, то функція є площиною, при більшій кількості змінних функція є гіперплоскістю. Подібно до рівняння прямої на площині, лінійна функція може бути представлена у вигляді суми (2.12):

$$f(X) = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + \ldots + c_nx_n, \qquad (2.12)$$

де n – кількість змінних x у векторі X.

Загальна задача лінійного програмування(ЗЛП)формулюється в такий спосіб: знайти *оптимум лінійної функції цілі y(X), якщо обмеження $f_i(X)$лінійні* й *вектор змінних X невід’ємний.*

Аналітичний запис ЗЛП виходить із запису загальної задачі математичного програмування шляхом конкретизації виду функції *f(X)* (2.13) – (2.14):

$$y(X) = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + \ldots + c_nx_n \rightarrow \max (\min); \qquad (2.13)$$

$$\begin{aligned} f_1(X) &= a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n \le (\ge, =)\ b_1; \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ f_m(X) &= a_{1m}x_1 + a_{2m}x_2 + a_{3m}x_3 + \ldots + a_{nm}x_n \le (\ge, =)\ b_m, \end{aligned} \qquad (2.14)$$

де m – кількість обмежень, в кожному обмеженні може стояти або знак рівності, тоді це рівняння-обмеження, або знак нерівності, тоді це нерівність-обмеження;

коефіцієнти c_i, a_{ji}, b_j – деякі константи, що характеризують умови, в яких здійснюється пошук оптимальних рішень, такі як некеровані параметри процесу, наявні ресурси, значимість змінних тощо залежно від змісту задачі та змінних.

Можна використовувати більш лаконічну форму запису (2.15) – (2.17):

$$\sum_{i=1}^{a} c_ix_i \rightarrow \max(\min); \qquad (2.15)$$

$$\sum_{i=1}^{a} a_{ij}x_i \le (\ge, =)\ b_j,\ j = 1,2,3, \ldots, m; \qquad (2.16)$$

$$x_i \ge 0,\ i = 1, 2, 3, \ldots, n. \qquad (2.17)$$

Останній рядок у виразі є записом умови невід'ємності всіх змінних векторуX, тобто в ЗЛП вважається, що всі змінні х повинні бути більше або дорівнювати 0.

Щоб в безлічі обмежень була однаковість при вирішенні різних завдань лінійного програмування, їх доцільно привести до одного з двох видів – *стандартного*, коли всі обмеження записані у вигляді нерівностей з одним знаком, скажімо, зі знаком «≤»;

канонічного, коли всі обмеження є рівностями. Приведення до канонічного вигляду використовується, зокрема, при вирішенні ЗЛП симплекс-методом.

Перетворення одного виду запису в інший, тобто приведення до одного виду запису всіх обмежень, може бути виконано наступним чином.

1. Стандартного до канонічного – шляхом введення додаткових змінних: наприклад, якщо в ЗЛП з нерівностями типу $a_{j1}x_1 + a_{j2}x_2 + a_{j3}x_3 \le b_j$ додати змінну x_4 з коефіцієнтом a_{j4} = 1 (або «-1» у разі «≥»), то можна записати $a_{j1}x_1 + a_{j2}x_2 + a_{j3}x_3 + a_{j4}x_4 = b_j$.

2. Канонічного до стандартного – шляхом введення замість одної рівності двох нерівностей: замість $a_{j1}x_1 + a_{j2}x_2 + a_{j3}x_3 + a_{j4}x_4 = b_j$ можна записати $a_{j1}x_1 + a_{j2}x_2 + a_{j3}x_3 + a_{j4}x_4 \le b_j$, $a_{j1}x_1 + a_{j2}x_2 + a_{j3}x_3 + a_{j4}x_4 \ge b_j$.

Крім згаданих видів запису та їх взаємних перетворень можна перетворити задачу пошуку мінімуму в задачу пошуку максимуму (або навпаки) шляхом зміни знаків при коефіцієнті c_i в вираженні цільової функції. Наприклад, задача мінімізації $y(X)\rightarrow$minеквівалентна задачі максимізації $-y(X)\rightarrow$max.

Особливості задач лінійного програмування, на яких засновані методи вирішення ЗЛП, можна проілюструвати за допомогою графічної інтерпретації. **Графічний метод** є найбільш простим і наочним методом вирішення ЗЛП. Він застосовується для розв'язання задач з двома змінними, що подані в стандартній формі, або для задач з m =2 змінними, що подані в канонічній формі, де m – число лінійно незалежних обмежень-рівностей.

Допустима множина рішень ЗЛПΩ утворює опуклий багатокутник, на границі якого знаходиться оптимум функції цілі y^*. На цій множині можна розташувати безліч рівнів цільової функції, тобто ліній, в кожній точці яких цільова функція набуває однакових значень. Для лінійної цільової функції рівні утворюють множину паралельних прямих.

З геометричної точки зору при розв'язанні ЗЛП шукають таку кутову точку області Ω, в якій лінія рівня з найменшим (при мінімізації) або з найбільшим (при максимізації) значенням цільової функції торкається цієї області Ω.

Алгоритм розв'язання ЗЛП графічним методом у загальному випадку складається з таких етапів:

1. Приведення математичної моделі задачі до вигляду (2.13) –(2.14).
2. Побудова прямих, визначених рівняннями $a_{i1}x_1+a_{i2}x_2=b_i$, i=1,n; x_1= 0 таx_2= 0.

Для побудови i-ої прямої $a_{i1}x_1+a_{i2}x_2=b_i$знаходять пару точок (0; b_i/a_{i1}); (0; b_i/a_{i2}).

3. Знаходження напівплощин, обумовлених кожним з обмежень задачі.

Для кожної напівплощини беремо яку-небудь точкуX_0^T, наприклад X_0^T = [0 0], і перевіряємо відповідну нерівність $a_{i1}x_1+a_{i2}x_2\le b_i$. Якщо нерівність виконується, виходить, точкаX_0^Tналежить напівплощині, яку шукають, інакше – не належить. Знайдені напівплощини виділяємо будь-яким зручним способом.

4. Виділення багатокутника рішень.

Перетинання виділених напівплощин утворює багатокутник рішень, тобто, область допустимих рішень Ω.

5. Побудова прямої $y_0=c_1x_1+c_2x_2$, що проходить через багатокутник рішень. Тут y_0– константа, яка обирається довільно.
6. Побудова вектору$C^T = [c_1 c_2]$.
7. Переміщення прямої $y_0=c_1x_1+c_2x_2$ в напрямку вектору c до границі області Ω (задача максимізації) або у зворотному напрямку вектору c (задача мінімізації).
8. Визначення координат граничної точки шляхом розв'язання системи двох рівнянь. Рівняння системи визначають прямі, що перетинаються в точці X^*.
9. Обчислення значення цільової функції y^*в точці X^*.

При розв'язанні ЗЛП можливі чотири випадки щодо кількості та існування рішень: єдине рішення (рис. 2.2 (а)); незліченна множина рішень (відрізок AB на рис. 2.2 (б)); необмежена область допустимих рішень (рис. 2.2 (в)) і відсутність рішення через несумісність системи обмежень (рис.2.2 (г)).

На рис. 2.2 показано можливі випадки при пошуку максимуму цільової функції, аналогічні ситуації можуть виникати і при пошуку мінімуму.

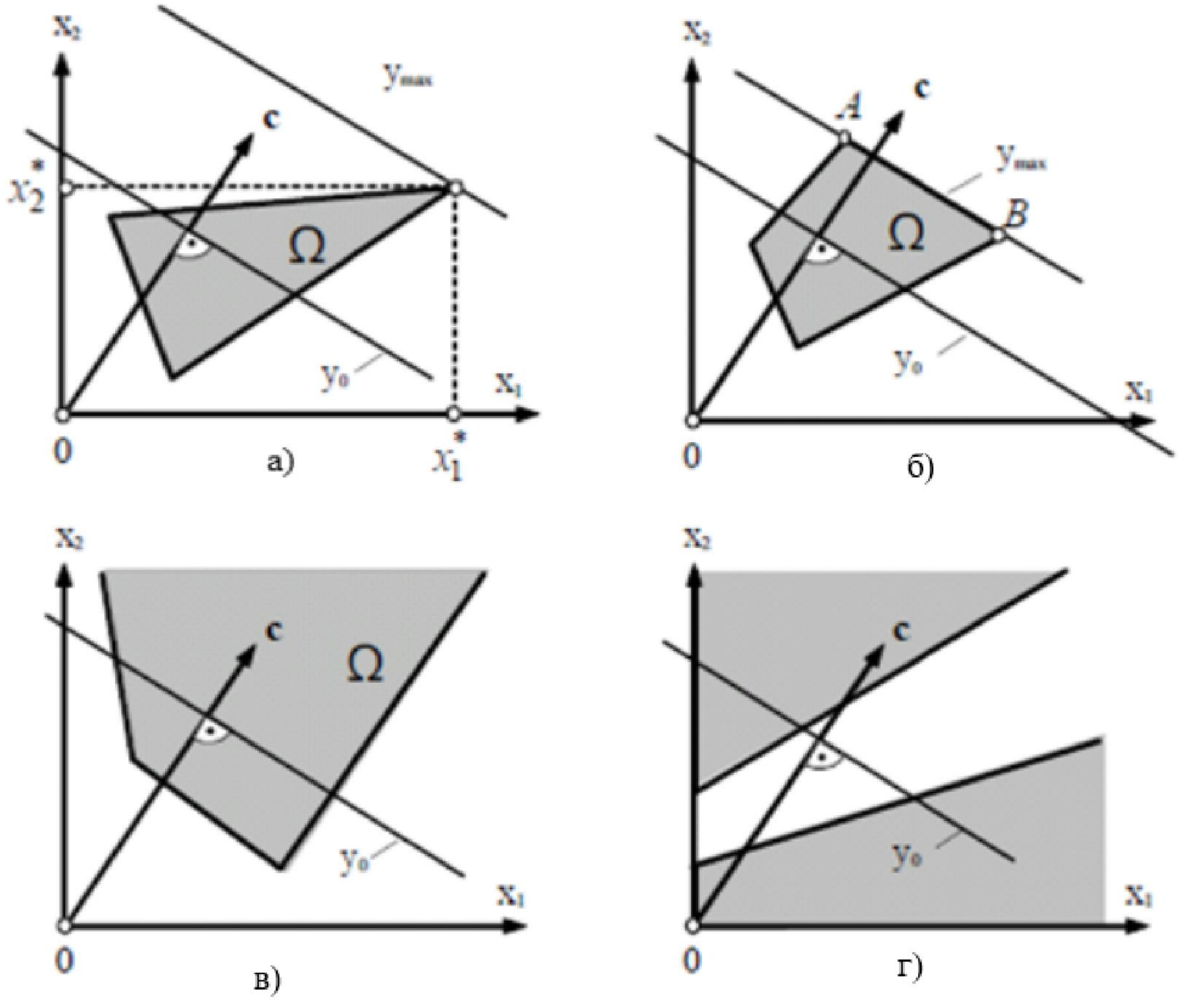


Рисунок 2.2 – Чотири випадки щодо кількості та існування рішень при розв’язанні ЗЛП: а)єдине рішення;б) незліченна множина рішень; в)необмежена область допустимих рішень;г)відсутність рішення через несумісність системи обмежень

Кожній задачі лінійного програмування можна певним чином поставити у відповідність іншу задачу лінійного програмування, що називають *двоїстою* стосовно вихідної (початкової) задачі. Вихідна та двоїста задачі тісно зв'язані між собою й утворюють єдину пару двоїстих задач, причому задача, двоїста стосовно двоїстої задачі, збігається з вихідної.

Вихідна задача	*Двоїста задача*
$y(\overline{x})=\sum_{j=1}^{n} c_j x_j \to max$	$d(\overline{z})=\sum_{i=1}^{m} b_i z_i \to min$
$\begin{cases} \sum_{j=1}^{n} a_{ij} x_j \le b_i, i=\overline{1,m}; \\ x_j \ge 0, j=\overline{1,n}. \end{cases}$	$\begin{cases} \sum_{i=1}^{m} a_{ji} z_i \ge c_j, j=\overline{1,n}; \\ z_i \ge 0, i=\overline{1,m}. \end{cases}$

Симплекс-метод розв’язання ЗЛП вважається найбільш поширеним методом у лінійному програмуванні. Метод був розроблений американським математиком Джорджем Данцігом в 1947 році.

Симплекс-метод – метод розв'язання задачі лінійного програмування, в якому здійснюється керований рух за опорними планами до знаходження оптимального розв'язання; симплекс-метод також називають методом поступового покращення плану.

Сутність його зводиться до наступного: за певними правилами при вирішенні задачі находяться так звані базисні рішення (симплекси). Ці базисні рішення збігаються з вершинами багатогранника області допустимих рішень. В процесі пошуку базисних рішень оцінюється їх оптимальність за правилами, що дозволяє визначити, чи можна поліпшити значення цільової функції, якщо вибрати якесь інше базисне рішення. Якщо можна – обирається нове базисне рішення (нова вершина багатогранника) і перевірка повторюється. Так робиться до тих пір, поки знайдене базисне рішення не виявиться оптимальним.

Симплекс-метод дозволяє знайти оптимальне рішення (якщо воно існує) за кінцеве число кроків. Існує велика множина модифікацій симплекс-методу, тут приводиться симплекс-метод, що дозволяє водночас вирішувати вихідну та двоїсту задачі.

Якщо вихідна задача лінійного програмування записана в стандартній формі, то її треба привести до канонічної форми, шляхом введення додаткових змінних $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$ у вихідну. Внаслідок приведення одержимо вихідну задачу (2.18) – (2.19):

$$y(\bar{x}) = \sum_{j=1}^{n} c_j x_j \to \max_{x \in \Omega}; \qquad (2.18)$$

$$\Omega: \begin{cases} f_i = \sum_{j=1}^{n} a_{ij} x_j + x_{n+1} = b_i, i = \overline{1,m} \\ x_j \geq 0, j = \overline{1, n+m} \end{cases}. \qquad (2.19)$$

У цьому випадку, змінні $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$ вихідної задачі вибираються в якості *залежних* (*Б- базисних*), а інші змінні $x_1, x_2, \ldots, x_n$ вихідної задачі є *незалежними* (*В- вільними*) і прирівнюються до нуля. Тоді перше опорне (базисне) рішення вихідної задачі буде:

$$X_0^T = (x_1=0, \ldots, x_n=0, x_{n+1}=b_1, x_{n+2}=b_2, \ldots, x_{n+m}=b_m).$$

Після визначення першого опорного рішення, перевіряють, чи є воно оптимальним. Якщо оптимум не досягнуто, то переходять до нового опорного рішення. Для цього потрібно визначити вільну змінну, яку потрібно ввести в базис, і базисну змінну, яку потрібно вивести з числа базисних змінних. Після одержання нового опорного рішення, його також перевіряють на оптимальність. Якщо критерій оптимуму досягнуто, то оптимальним рішенням задачі є останній опорний план (опорне рішення).

Всі розрахунки за симплекс-методом зручно виконувати за допомогою спеціальної симплекс-таблиці (табл. 2.2).

Критерієм оптимальності при мінімізації цільової функції є негативне значення всіх оцінок ($t_{m+1,j} \leq 0, j = \overline{1,n}$); при максимізації – невід’ємність всіх оцінок ($t_{m+1,j} \geq 0,\ j = \overline{1,n}$).

Таблиця 2.2 – Симплекс-таблиця

	Б (залежні)	z_{m+1}	z_{m+2}		z_{m+n}	$d(Z) \to \min$	Симпл. відн. θ
В(незалежні)	(незалежні) *Б* *В* (залежні)	x_1	x_2		x_n	Вільні	
$-z_1$	x_{n+1}	$t_{11} = a_{11}$	$t_{12} = a_{12}$		$t_{1n} = a_{1n}$	$t_{1,n+1} = b_1$	θ_1
$-z_2$	x_{n+2}	$t_{21} = a_{21}$	$t_{22} = a_{22}$		$t_{2n} = a_{2n}$	$t_{2,n+1} = b_2$	θ_2
...	...	...	...		...	...	...
$-z_m$	x_{n+m}	$t_{m1} = a_{m1}$	$t_{m2} = a_{m2}$		$t_{mn} = a_{mn}$	$t_{m,n+1} = b_m$	θ_m
Вільні	$y(X) \to max$	$t_{m+1,1} = -c_1$	$t_{m+1,2} = -c_2$		$t_{m+1,n} = -c_n$	$t_{m+1,n+1} = 0$	

Алгоритм симплекс-методу

Будемо вважати, що початкову таблицю симплекс-методу сформовано. Тоді послідовність розв'язання задачі симплекс-методом наступна:

1. Перевірити, чи виконується умова оптимуму (критерій оптимальності). Якщо виконується, то отримане опорне рішення є оптимальним. Якщо не виконується, перейти до наступного пункту.

2. Знайти *напрямний (r-й) стовпець*. Серед «хибних» оцінок ($t_{m+1,j} > 0$ – для задачі мінімізації,$t_{m+1,j} < 0$ – для задачі максимізації) обрати максимальну за модулем, тобто, оцінку $t_{m+1,r} = \max |t_{m+1,j}|,\ j = \overline{1,n}$.

3. Знайти *напрямний (k-й) рядок*. Для всіх $a_{ir} > 0$визначити симплексні відносини $\theta_i = b_i / a_{ir}$. Напрямним буде той рядокk, для якого θ_i буде мінімальним $\theta_k = \min\limits_{i,a_{ir}>0} \{\theta_i\}$. На перетині r-го стовпця та k-го рядка знаходиться *розв'язний елемент*t_{kr}.

Якщо всі компоненти напрямного стовпця t_{ir} непозитивні ($t_{ir} \le 0,\ i = \overline{1,m}$), лінійна форма задачі необмежена на багатокутнику рішень та розрахунки на цьому закінчуються. Задача не має рішення.

4. Виконати транспозицію змінних x_rта z_{m+r}, тобто, поміняти їх місцями, отримуючи нову пару змінних x_{n+k}та z_k. Сформувати нове опорне рішення з урахуванням транспозиції змінних. Перерахувати елементи таблиціt_{ij}($i = \overline{1,m+1},\ j = \overline{1,n+1}$) в елементи нової таблиці $t_{ij}^{н}$ ($i = \overline{1,m+1},\ j = \overline{1,n+1}$) за наступними правилами:

- розв'язний елемент $t_{kr}^{н} := 1 / t_{kr}$;
- елементи напрямного рядка $t_{kj}^{н} := t_{kj} / t_{kr},\ j \ne r$;
- елементи напрямного стовпця $t_{ir}^{н} := -t_{ir} / t_{kr},\ i \ne k$;
- інші елементи таблиці $t_{ij}^{н} := t_{ij} - (t_{ir} \cdot t_{kj}) / t_{kr},\ i \ne k\ ,\ j \ne r$.

5. Циклічно перейти до початкового пункту 1.

Постановка транспортної задачі лінійного програмування, її математична модель та області застосування. Рішення транспортної задачі методом потенціалів

Транспортна задача широко використовується в практиці планування. Це задача про знаходження найбільш раціонального з погляду витрат плану перевезень однорідного продукту *від постачальників до споживачів*. Для визначення оптимального плану транспортної задачі можна використати диференціальний алгоритм, симплекс-метод й інші універсальні методи. Однак через специфіку обмежень задачі, для визначення оптимального плану транспортної задачі доцільно застосовувати спеціально розроблені методи, наприклад метод потенціалів.

Проблема була вперше формалізована французьким математиком Гаспаром Монжем в 1781 році. Прогрес в рішенні проблеми був досягнутий під час великої Вітчизняної Війни радянським математиком і економістом Леонідом Канторовичем. Тому іноді цю проблему називають транспортною задачею Монжа – Канторовича.

Однорідний продукт, зосереджений у m пунктах відправлення в кількостях a_1, a_2, …, a_mодиниць відповідно, необхідно доставити в кожний з n пунктів призначення в кількостях b_1, b_2, …, b_nодиниць відповідно. Вартість (відстань) перевезення одиниці продукту з *i-го* пункту відправлення в *j-й* пункт призначення дорівнює c_{ij}і відома для кожного маршруту. Нехай x_{ij}– кількість продукту, перевезеного з *i-го* пункту відправлення в *j-й* пункт призначення. Задача полягає у визначенні таких величин x_{ij}для всіх маршрутів, при яких сумарна вартість або відстань перевезень були б мінімальними.

Тоді математична модель транспортної задачі про планування перевезень має вигляд (2.20) – (2.21):

$$y = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \to \min_{x_{ij}\in\Omega}; \qquad (2.20)$$

$$\Omega: \begin{cases} f_j = \sum_{i=1}^{m} x_{ij} = b_j,\ j = \overline{1,n} \\ f_{n+i} = \sum_{j=1}^{n} x_{ij} = a_i,\ i = \overline{1,m}. \\ x_{ij} \ge 0,\ i = \overline{1,m},\ j = \overline{1,n} \end{cases} \qquad (2.21)$$

де c_{ij}– тарифи (вартість, час, відстань) перевезення одиниці вантажу з *i-го*пункту відправлення в *j-й* пункт призначення;

a_i– запаси вантажу в *i*-му пункті відправлення;

b_j– потреба у вантажі в *j*-му пункті призначення;

x_{ij}– кількість одиниць вантажу, перевезеного з *i-го* пункту відправлення в *j-й* пункт призначення. Причому перевезений вантаж характеризується вагою, довжиною (погонні метри), площею (квадратні метри), об'ємом тощо. Наприклад, перевозиться рідина, сипучий матеріал, дрібні заготівлі або дрібна продукція.

Транспортну задачу можна представити у вигляді табл. 2.3.

Таблиця 2.3 – Транспортна таблиця

				Пункти призначення					
				1	2	…	*j*	…	*N*
				Потреби					
				b_1	b_2	…	b_1	…	b_n
Пункти відправлення	1	Запаси	a_1	c_{11} x_{11}	c_{12} x_{12}	…	c_{1j} x_{1j}	…	c_{1n} x_{1n}
	2		a_2	c_{21} x_{21}	c_{22} x_{22}	…	c_{2j} x_{2j}	…	c_{2n} x_{2n}
	…		…	…	…	…	…	…	…
	i		a_i	c_{i1} x_{i1}	c_{i2} x_{i2}	…	c_{ij} x_{ij}		c_{in} x_{in}
	…		…	…	…	…	…	…	…
	m		a_m	c_{m1} x_{m1}	c_{m2} x_{m2}	…	c_{mj} x_{mj}	…	c_{mn} x_{mn}

Теорема 2.1.Для *можливості розв'язання* транспортної задачі необхідно й достатньо, щоб *загальні запаси вантажу* в пунктах відправлення *дорівнювали загальним потребам* у вантажі в пунктах призначення, тобто щоб виконувалася рівність (2.25):

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad (2.25)$$

Якщо загальна потреба у вантажі в пунктах призначення дорівнює загальному запасу вантажу в пунктах відправлення, то модель такої транспортної задачі називається <u>закритою</u>. У протилежному випадку – <u>відкритою</u>.

Як і для всякої задачі лінійного програмування, оптимальний план транспортної задачі є й опорним планом.

Процес рішення транспортної задачі складається із двох етапів:

1. Побудови вихідного опорного плану перевезень.
2. Знаходження оптимального плану.

Побудова опорного плану

Існує кілька методів визначення вихідного опорного плану, серед яких можна відзначити метод *північно-західного кута*, метод *мінімальної вартості*, метод *подвійної переваги*.

План $X = [x_{ij}], i = \overline{1,m}, j = \overline{1,n}$ є *невиродженим* опорним планом, якщо в ньому кількість відмінних від нуля компонентів у точності дорівнює $m+n-1$, а якщо менше – те *виродженим*.

Найпростіший метод побудови опорного плану – це **метод північно-західного кута**. Він полягає в послідовному розподілі продукції споживачам з урахуванням можливостей постачальників, починаючи з лівої верхньої клітини таблиці транспортної задачі.

Пункти від-прав-лення	Запаси вантажу	Пункти призначення		
		1	2	3
		Потреба		
		80	120	100
1	100	2 **80**	1 **20**	3 –
2	50	1 –	4 **50**	2 –
3	150	2 –	3 **50**	1 **100**

Покажемо процес побудови опорного плану методом північно-західного кута на наступному прикладі.

Приклад 2.2. Методом **північно-західного кута** знайти опорне рішення транспортної задачі.

Три розчинобетонних заводи, які обслуговують злітно-посадкову смугу аеродрому, забезпечуються цементом із трьох складів. Попит заводів b_j відповідно дорівнює 80, 120 та 100 тис. т/міс. Пропускна здатність складів a_i відповідно дорівнює 100, 50 та 150 тис. т/міс. Відстань перевезення (у км) з i-го складу на j-й розчинобетонний завод представлена в матриці:

$$C = [c_{ij}] = \begin{bmatrix} 2 & 1 & 3 \\ 1 & 4 & 2 \\ 2 & 3 & 1 \end{bmatrix}.$$

Потрібно скласти план перевезень цементу зі складів на заводи, що задовольняв би пропускну здатність складів та потреби заводів, а сумарний пробіг вантажного транспорту був би мінімальним.

Рішення.

Розподіл кількості перевезень роблять без огляду на вартість перевезення одиниці продукції. Розподіл починається з визначення x_{11}. Для цього порівнюємо a_1=100 з $в_1$=80 і вибираємо менше,тоді x_{11}= 80. Оскільки потреби **першого пункту призначення** повністю **задоволені**, то в клітинах **21** й **31** ставимо **прочерки**. Запаси першого джерела ще не вичерпано й становлять 100-80=20 одиниць. Оскільки 20<120, тоді x_{12}= 20. Оскільки запаси першого джерела вже вичерпані, в клітину **13** ставимо**прочерк**.Середнезаповненихклітинрозглядаємоклітину22 (запас становить 50, а потреби 120-20),отже x_{22}= 50.

Клітина 23 – прочерк (запас вичерпаний).

Обчислимо x_{32}=120–20–50=50, тоді x_{33}= 150 – 50 = 100. Даний опорний план є невиродженим, оскільки кількість відмінних від нуля компонентів складає $n+m$-1=3+3-1=5. При цьому значення функції мети складе 80*2+20*1+50*4+50*3+100=630.

Метод мінімальної вартості полягає у тому, що спочатку заповнюються клітини з мінімальними вартостями. Продемонструємо його на прикладі.

Приклад 2.3. Методом **мінімальної вартості** знайти опорне рішення задачі із прикладу 2.2.

Рішення.

Спочатку знаходимо клітину з мінімальною вартістю, наприклад клітина12, і заповнюємо її мінімальним з 100 та 120. Тоді клітини 11та 13 заповнимо прочерками. Знаходимо наступну мінімальну клітину, це 21, і заповнюємо її мінімальним з 50та80.Відповідно,клітини 22 та 23заповнимо прочерками.При цьомуx_{31}=80–50=30, x_{32}= 120 – 100 = 20, а в клітину 33 помістимо 100. Значення функції мети складе 100*1+50*1+30*2+20*3+100*1=370. Це значення менше за знайдене методом північно-західного кута.

Пункти від-прав-лення	Запаси вантажу	Пункти призначення		
		1	2	3
		Потреба		
		80	120	100
1	100	2 –	1 **100**	3 –
2	50	1 **50**	4 –	2 –
3	150	2 **30**	3 **20**	1 **100**

Визначення оптимального опорного плану транспортної задачі

Найбільш популярний метод знаходження оптимального плану транспортної задачі – **метод потенціалів**. Метод припускає, що відомо який-небудь опорний план. Вихідний опорний план необхідно перевірити на оптимальність.

Теорема 2.2. Якщо для деякого опорного плану $X^* = \left[x^*_{ij}\right], i = \overline{1,m}, j = \overline{1,n}$ транспортної задачі із заданими тарифами перевезень c_{ij} існують такі числа u_i та v_j, що$u_i + v_j = c_{ij}$ при $x_{ij} > 0$ та $u_i + v_j \le c_{ij}$ при $x_{ij} = 0$ для всіх $i = \overline{1,m}, j = \overline{1,n}$, то $X^* = \left[x^*_{ij}\right]$ – оптимальний план.

Числа u_i та v_j називаються **потенціалами** відповідно пунктів відправлення та пунктів призначення.

Алгоритм методу потенціалів:

1-й етап. Для знайденого опорного невиродженого плану знаходять потенціали пунктів відправлення й призначення. Так як число заповнених клітин дорівнює n+m–1, то один з потенціалів дорівнюють до нуля.

2-й етап. Для кожної вільної клітки визначають числа $\Delta_{ij} = u_i + v_j - c_{ij}$. Якщо серед них немає позитивних, то отримано оптимальний план транспортної задачі. Якщо ждля деякої вільної клітини плану $\Delta_{ij} > 0$, то необхідно перейти до нового опорного плану.

3-й етап. Знаходять новий опорний план. Для цього розглядають всі вільні клітини, для яких Δ_{ij}>0, і вибирають ту, для якої число Δ_{ij}максимальне. Обрану клітину варто заповнити, її позначають знаком «+» і формують *цикл*, по якому необхідно змінити об'єми перевезень.

Циклом у таблиці транспортної задачі називається *замкнута ламана лінія, вершини якої розташовані в зайнятих клітинах таблиці*, а ланки – уздовж рядків і стовпців, причому в кожній вершині циклу зустрічаються рівно дві ланки, одна з яких перебуває в рядку, а інша – у стовпці.

При правильній побудові опорного плану для будь-якої вільної клітини можна побудувати лише один цикл. Вільна клітина позначається «+», потім знаки чергуються «–», «+», «–», «+», …. У вільну клітину переносять найменше із чисел x_{ij}, що знаходяться в «мінусових» клітках, і одночасно це число додають до чисел, що знаходяться в «плюсових» клітках. Клітина, що раніше була вільною, стає зайнятою, а «мінусова» клітина, у якій стояло мінімальне число x_{ij}, стає вільною. Далі переходять до 1-го етапу.

Приклад 2.4. Методом потенціалів знайти оптимальне рішення задачі із прикладу2.2.

Рішення.

1-й етап. Знаходимо потенціалипунктів відправлення та призначення.

Нехай $u_3 = 0$, тоді $v_3 = 1$, $v_2 = 3$. Потім $u_2 = 4 - 3 = 1$, u_1=1–3=–2,і нарешті, $v_1 = 2 + 2 = 4$.

2-й етап. Перевірка опорного рішення на оптимальність. Для перевірки плану на оптимальність знайдемо Δ_{ij}для вільних клітин.

Δ_{13} = 1-2-3=-4<0.

Δ_{21} = 4+1-1=4>0.

Δ_{23} = 1+1-2=0.

Δ_{31} =4+0-2=2>0.

Рішення не є оптимальним.

Пункти від-прав-лення	Запаси грузу	Пункти призначення			u_i
		1	2	3	
		Потреби			
		80	120	100	
1	100	2 80	1 20	3 –	-2
2	50	1 –	4 50	2 –	1
3	150	2 –	3 50	1 100	0
v_j		4	3	1	

Пункти від-прав-лення	Запаси грузу	Пункти призначення			u_i
		1	2	3	
		Потреби			
		80	120	100	
1	100	– 2 80	+ 1 20	3 –	-2
2	50	+ 1 –	– 4 50	2 –	1
3	150	2 –	3 50	1 100	0
v_j		4	3	1	

Пункти від-прав-лення	Запаси грузу	Пункти призначення			u_i
		1	2	3	
		Потреби			
		80	120	100	
1	100	– 2 30	+ 1 70	3 –	-2
2	50	1 50	4 –	2 –	-3
3	150	+ 2 –	– 3 50	1 100	0
v_j		4	3	1	

3-й етап. Знаходження нового опорного плану. Будуємо цикл. Клітина 21 позначається «+», відповідно клітини 11 та 22, як «-», і клітина 12 – як «+». Оскільки 50<80, здійснимо перекидання за циклом 50 одиниць продукції та знайдемо нове опорне рішення. І повернемося до етапу 1. Знайдемо значення цільової функції 30*2+70*1++50*1+50*3+100=430. Значення цільової функції зменшилось. Одержимо нові потенціали.

Перевіримо рішення на оптимальність.

Δ_{13} = 1-2-3=-4<0.

Δ_{22} = 3-3-4=-4<0.

Δ_{23} = 1-3-2=-4<0.

Δ_{31} = 4+0-2=2>0.

Рішення не оптимальне. Знайдемо нове опорне рішення. Клітину 31позначимо «+».

Виконаємо перекидання за циклом 30<50. Одержимо нове опорне рішення. Знайдемо потенціали й перевіримо на оптимальність.

Δ_{11} = 2-2-2=-2<0.

Δ_{13} = 1-2-3=-4<0.

Δ_{22} = 3-1-4=-2<0.

Δ_{23} = 1-1-2=-2<0.

Пункти від-прав-лення	Запаси грузу	Пункти призначення			u_i
		1	2	3	
		Потреби			
		80	120	100	
1	100	2 –	1 100	3 –	-2
2	50	1 50	4 –	2 –	-1
3	150	2 30	3 20	1 100	0
v_j		2	3	1	

Всі оцінки від'ємні, значить знайдено оптимальне рішення. Мінімум складе100+50+60+60+100=370.

Якщо ми маємо справу із транспортною задачею відкритого типу, то вводять або фіктивного постачальника, або фіктивного споживача та вирішують як закриту задачу. У випадку перевищення запасу над потребою, тобто $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$, уводиться фіктивний ($n$+1)-й пункт призначення з потребою $b_{n+1} = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j$. При цьому відповідні тарифи вважаються рівними нулю: $c_{i,n+1} = 0$ ($i = \overline{1,m}$).

Аналогічно, при $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$ вводиться фіктивний (m+1)-й пункт відправлення із запасом вантажу $a_{m+1} = \sum_{j=1}^{n} b_j - \sum_{i=1}^{m} a_i$. При цьому відповідні тарифи вважаються рівними нулю: $c_{m+1,j} = 0$ ($j = \overline{1,n}$).

Задача про призначення як приватний випадок транспортної задачі. Рішення задачі про призначення угорським методом. Приклади моделювання виробничих процесів на авіаційному транспорті у формі транспортної задачі

У процесі управління виробництвом часто виникають **задачі призначення** виконавців на різні види робіт, наприклад: підбор кадрів і призначення кандидатів на вакантні посади, розподіл джерел капітальних вкладенні між різними проєктами науково-технічного розвитку, розподіл екіпажів літаків між авіалініями.

Задача про призначення є приватним випадком **транспортної задачі**, в якій число пунктів відправлення дорівнює числу пунктів призначення, тобто, транспортна таблиця має форму квадрата. Крім того, всі величини попиту й величини пропозиції рівні (у кожному пункті призначення обсяг попиту дорівнює 1, і величина пропозиції кожного пункту відправлення також дорівнює 1). Задача про призначення може бути розв'язана з використанням методів лінійного програмування або алгоритму рішення транспортної задачі. Однак через особливу структуру даної задачі був розроблений спеціальний алгоритм, що одержав назву **угорського методу**.

Ідея цього методу розв'язання транспортної задачі вперше була запропонована угорським математиком Є.Егерварі у 1931 році, тобто ще до розроблення загальної теорії лінійного програмування. Спочатку цей метод був розроблений для розв'язання специфічного виду транспортної задачі, а згодом узагальнений. Нині угорський метод є одним з найпоширеніших методів розв'язання транспортних задач.

Він досить простий з погляду обчислень і може застосовуватися без упереджень навіть у разі виродженості плану.

При розгляді задачі про призначення **в стандартній формі** передбачається, що кількість працівників *дорівнює* кількості робіт.

Задачі про призначення **у відкритій формі** виникають тоді, коли кількість працівників *не дорівнює* кількості робіт. У цих випадках задача може бути перетворена в задачу, сформульовану в стандартній формі.

Математична модель задачі про призначення(2.26)–(2.27):

- цільова функція:

$$L = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to min, n = m\,; \tag{2.26}$$

- обмеження:

$$\sum_{j=1}^{n} x_{ij} = 1\,,\ i = \overline{1,m}\,;$$

$$\sum_{i=1}^{m} x_{ij} = 1\,,\ \ j = \overline{1,n}\,; \tag{2.27}$$

$$x_{ij} = \begin{cases} 1 \\ 0 \end{cases},\ i = \overline{1,m},\ j = \overline{1,n}\,;$$

$$x_{ij} \ge 0,\ i = \overline{1,m},\ j = \overline{1,n}\,,$$

де c_{ij}– показник ефективності призначення i-го працівника на j-ту роботу, наприклад витрати виконання i-м працівником j-ої роботи;

x_{ij}–змінна моделі (x_{ij}= 1, якщо i-й працівник використовується на j-й роботі, і x_{ij} = 0 у противному випадку).

Приклад 2.5. В авіакомпанії організовується фінансово-економічний відділ і необхідно розподілити працівників за вакантними посадами. Кожний працівник може виконувати будь-яку роботу, але з різним ступенем майстерності. Якщо на певну посаду призначити працівника саме тої кваліфікації, яка необхідна для роботи на ній, то вартість виконання роботи буде нижча, ніж при призначенні на дану посаду працівника невідповідної кваліфікації. Компетентна комісія оцінила витрати, пов'язані з майбутнім розподілом працівників. В табл. 2.4 наводяться умовні оцінки вартості c_{ij} призначення i-го працівника на j-ту посаду. Змінна x_{ij} =1 у випадку призначення і-го працівника на j-ту посаду, або x_{ij} =0 – в іншому випадку. Необхідно знайти оптимальний розподіл працівників за всіма посадами, що забезпечує мінімальну вартість виконання робіт.

Таблиця 2.4 – Оцінки вартості призначення працівників на певні посади

Працівники	Посади			
	Економіст	Бухгалтер	Касир	Менеджер
Іванов І.П.	10 x_{11}	40 x_{12}	60 x_{13}	30 x_{14}
Петров С.М.	90 x_{21}	70 x_{22}	100 x_{23}	90 x_{24}
Сидоров В.Н.	40 x_{31}	50 x_{32}	110 x_{33}	70 x_{34}
Федотов Р.Д.	80 x_{41}	70 x_{42}	80 x_{43}	50 x_{44}

Для прикладу маємо наступну модель:

$$L = 10x_{11} + 40x_{12} + 60x_{13} + 30x_{14} + 90x_{21} + 70x_{22} + 100x_{23} + 90x_{24} + 40x_{31} +$$
$$+ 50x_{32} + 110x_{33} + 70x_{34} + 80x_{41} + 70x_{42} + 80x_{43} + 50x_{44} \to min;$$

$$\begin{cases} x_{11} + x_{12} + x_{13} + x_{14} = 1; \\ x_{21} + x_{22} + x_{23} + x_{24} = 1; \\ x_{31} + x_{32} + x_{33} + x_{34} = 1; \\ x_{41} + x_{42} + x_{43} + x_{44} = 1; \\ x_{11} + x_{21} + x_{31} + x_{41} = 1; \\ x_{12} + x_{22} + x_{32} + x_{42} = 1; \\ x_{13} + x_{23} + x_{33} + x_{43} = 1; \\ x_{14} + x_{24} + x_{34} + x_{44} = 1; \end{cases}$$

$$\begin{array}{llll} x_{11} \geq 0; & x_{21} \geq 0; & x_{31} \geq 0; & x_{41} \geq 0; \\ x_{12} \geq 0; & x_{22} \geq 0; & x_{32} \geq 0; & x_{42} \geq 0; \\ x_{13} \geq 0; & x_{23} \geq 0; & x_{33} \geq 0; & x_{43} \geq 0; \\ x_{14} \geq 0; & x_{24} \geq 0; & x_{34} \geq 0; & x_{44} \geq 0. \end{array}$$

Знаходимо в кожній строчці мінімальний елемент і віднімаємо його від кожного елементу рядка (табл. 2.5).

Таблиця 2.5– Знаходження мінімальних елементів в рядках та їх віднімання

Працівники	Посади				
	Економіст	Бухгалтер	Касир	Менеджер	min
Іванов І.П.	$\boxed{0}$ x_{11}	$\boxed{30}$ x_{12}	$\boxed{50}$ x_{13}	$\boxed{20}$ x_{14}	*10*
Петров С.М.	$\boxed{20}$ x_{21}	$\boxed{0}$ x_{22}	$\boxed{30}$ x_{23}	$\boxed{20}$ x_{24}	*70*
Сидоров В.Н.	$\boxed{0}$ x_{31}	$\boxed{10}$ x_{32}	$\boxed{70}$ x_{33}	$\boxed{30}$ x_{34}	*40*
Федотов Р.Д.	$\boxed{30}$ x_{41}	$\boxed{20}$ x_{42}	$\boxed{30}$ x_{43}	$\boxed{0}$ x_{44}	*50*

Знаходимо в кожному стовпці мінімальний елемент і віднімаємо його від кожного елементу стовпця (табл. 2.6).

Таблиця 2.6– Знаходження мінімальних елементів в стовпцях та їх віднімання

Працівники	Посади			
	Економіст	Бухгалтер	Касир	Менеджер
Іванов І.П.	$\boxed{0}$ x_{11}	$\boxed{30}$ x_{12}	$\boxed{20}$ x_{13}	$\boxed{20}$ x_{14}
Петров С.М.	$\boxed{20}$ x_{21}	$\boxed{0}$ x_{22}	$\boxed{0}$ x_{23}	$\boxed{20}$ x_{24}
Сидоров В.Н.	$\boxed{0}$ x_{31}	$\boxed{10}$ x_{32}	$\boxed{40}$ x_{33}	$\boxed{30}$ x_{34}
Федотов Р.Д.	$\boxed{30}$ x_{41}	$\boxed{20}$ x_{42}	$\boxed{0}$ x_{43}	$\boxed{0}$ x_{44}
min	*0*	*0*	*30*	*0*

Якщо в кожній строчці та кожному стовпці знаходиться один „0”, тобто, кожний працівник призначається тільки на одну посаду, то оптимальне рішення знайдене. В іншому випадку:

– проводимо мінімальне число прямих через строки та стовпці, щоб всі „0” були викресленими (табл. 2.7);

– обираємо мінімальний невикреслений елемент (=10), віднімаємо його від кожного невикресленого елементу і додаємо до кожного елементу, що стоїть на перетині проведених прямих (табл. 2.8).

Якщо на останньому кроці оптимальне рішення не знайдене, то процедуру проведення прямих необхідно повторювати до тих пір, поки не буде отримане допустиме рішення.

Таблиця 2.7– Викреслювання нулів

Працівники	Посади			
	Економіст	Бухгалтер	Касир	Менеджер
Іванов І.П.	0 x_{11}	30 x_{12}	20 x_{13}	20 x_{14}
Петров С.М.	20 x_{21}	0 x_{22}	0 x_{23}	20 x_{24}
Сидоров В.Н.	0 x_{31}	10 x_{32}	40 x_{33}	30 x_{34}
Федотов Р.Д.	30 x_{41}	20 x_{42}	0 x_{43}	0 x_{44}

Таблиця 2.8– Знаходження оптимального рішення

Працівники	Посади			
	Економіст	Бухгалтер	Касир	Менеджер
Іванов І.П.	0 x_{11}	20 x_{12}	10 x_{13}	10 x_{14}
Петров С.М.	30 x_{21}	0 x_{22}	0 x_{23}	20 x_{24}
Сидоров В.Н.	0 x_{31}	0 x_{32}	30 x_{33}	20 x_{34}
Федотов Р.Д.	40 x_{41}	20 x_{42}	0 x_{43}	0 x_{44}

Таким чином, мінімальна сумарна вартість виконання робіт працівниками відділу складає:

L = 10 + 100 + 50 + 50 = 210 у.о.

Оптимальним рішенням є призначення Іванова І.П. на посаду економіста, Петрова С.М. – на посаду касира, Сидорова В.Н. – на посаду бухгалтера, Федотова Р.Д. – на посаду менеджера.

Задані обмеження щодо призначення кожного працівника тільки на одну посаду та забезпечення всіх посад працівниками виконуються.

2.5 Висновки по лекції

Математичне програмування досліджує екстремальні задачі (задачі пошуку максимуму або мінімуму, або оптимізаційні задачі) і розробляє методи їх розв'язання. **Задачі математичного програмування** поділяються на детерміновані задачі, задачі стохастичного та задачі динамічного програмування.

Всі детерміновані задачі поділяються на задачі лінійного чи нелінійного програмування. В **задачах лінійного програмування** цільова функція та всі обмеження є лінійними функціями. **Графічний метод** є найбільш простим і наочним методом вирішення ЗЛП. **Симплекс-метод** розв’язання ЗЛП, в якому здійснюється скерований рух за опорними планами до знаходження оптимального розв'язку, вважається найбільш поширеним методом у лінійному програмуванні.

Транспортна задача полягає в знаходженні найбільш раціонального з погляду витрат плану перевезень однорідного продукту від постачальників до споживачів. Для визначення вихідного опорного плану можна відзначити **метод північно-західного кута** та **метод мінімальної вартості**. Для знаходження оптимального плану використовується **метод потенціалів.**

Задача про призначення є приватним випадком транспортної задачі, в якій число пунктів відправлення дорівнює числу пунктів призначення, тобто, транспортна таблиця має форму квадрата. Через особливу структуру даної задачі був розроблений спеціальний алгоритм, що одержав назву **угорського методу**.

Тема 3 *Графічне моделювання організації виробничих процесів на авіаційному транспорті* (2 год.)

3.1 Мета та завдання лекції

Метою лекції є ознайомлення з роллю графічного моделювання в організації виробничих процесів на авіаційному транспорті.

Завдання лекції:

– розкрити поняття елементів теорії графів, в т.ч. поняття вузлів (вершин), ребер та дуг, петель та ланцюгів, орієнтованих мереж, зв'язаних мереж;

– ознайомити з застосуванням мережевого планування при розробці проєктів виконання різних комплексів робіт з організації виробничого процесу;

– ознайомити з побудовою мережі проєкту, поняттям критичних робіт, з методом критичного шляху, часовим графіком;

– розглянути задачу про найкоротший маршрут, алгоритми Дейкстри та Флойда;

– розглянути задачу про максимальний потік, розкрити сутність розрізу та перебору розрізів.

3.2 План лекції

3.1 Елементи теорії графів. Вузли (вершини), ребра та дуги. Петлі та ланцюги. Орієнтовані мережі. Зв'язані мережі.

3.2 Застосування мережевого планування при розробці проєктів виконання різних комплексів робіт з організації виробничого процесу на авіаційному транспорті.

3.3 Побудова мережі проєкту. Критичні роботи. Метод критичного шляху. Часовий графік.

3.4 Задача про найкоротший маршрут. Алгоритм Дейкстри. Алгоритм Флойда.

3.5 Задача про максимальний потік. Розріз. Перебір розрізів.

3.3 Основні категорії, ключові поняття та визначення теми

Алгоритм Флойда-Воршелла – це алгоритм динамічного програмування для знаходження найкоротших відстаней між усіма вершинами зваженого орієнтованого графа.

Вершина (вузол)– точка, де можуть сходитися/розходитися ребра та/або дуги.

Граф –абстрактний математичний об'єкт, який представляє собою безліч вершин графа і набір ребер, тобто з'єднань між парами вершин.

Зв'язна мережа – така мережа, в якій будь-які два вузли зв'язані принаймні одним шляхом.

Критичні роботи –роботи та події, які лежать на критичному шляху.

Ланцюг –незамкнений маршрут, у якого немає ребер (дуг), що повторюються, для неорієнтованого графа.

Мережеве планування– графічне зображення певного комплексу виконуваних робіт, яке відображає їх логічну послідовність, існуючий взаємозв'язок і заплановану тривалість, та забезпечує подальшу оптимізацію розробленого графіка на основі економіко-математичних методів і комп'ютерної техніки з метою його використання для поточного управління ходом робіт.

Петляу графі – це ребро, інцидентне одній і тій же самій вершині.

Ребро графа – лінія, яка з'єднує пару суміжних вершин графа.

Теорія графів–розділ математики, який вивчає властивості графів.

3.4 Текст лекції

Елементи теорії графів. Вузли (вершини), ребра та дуги. Петлі та ланцюги. Орієнтовані мережі. Зв'язані мережі

Теорія графів – розділ математики, що вивчає властивості графів. Наочно граф можна представити як геометричну конфігурацію, яка складається з точок (вершин) сполучених лініями (ребрами). У строгому визначенні ***графом*** називається така пара множин $G = (V, E)$, де V є підмножина будь-якої зліченної множини, а E– підмножина $V \times V$.

Теорія графів має потужний апарат рішення прикладних задач у самих різних сферах науки. До яких відносяться, наприклад теорія зв'язку, аналіз систем, проєктування обчислювальних машин, архітектура, дослідження операцій, генетика, психологія, соціологія, економіка і т.д. Теорія графів також тісно пов'язана з такими розділами математики, як теорія множин, теорія матриць, математична логіка та теорія ймовірності. В усіх цих розділах графи застосовуються для представлення різних об'єктів.

Скінченна сукупність точок, деякі з яких з'єднані одна з одною лініями, називається ***графом***. Ці точки називаються вершинами графа, а вказані лінії – ребрами графа.

Вершина (вузол)– точка, де можуть сходитися/розходитися ребра та/або дуги.

Ребро графа (дуга графа)– лінія, що з'єднує пару суміжних вершин графа.

Дуга – орієнтоване ребро.

Вершини графа зазвичай нумерують десятковими числами, але можна використовувати і будь-які інші знаки. Якщо вершини пронумеровані, то ребра означають неупорядковані пари номерів вершин. Кожну пару утворюють номери тих вершин, які з'єднані ребром.

Вершини, які не належать кінцям жодного з ребер у графі, називаються **ізольованими**. Граф, який складається лише з ізольованих вершин, називається **нуль-графом.** Граф, у якому будь-яка пара вершин з'єднана ребрами,називається **повним.** Якщо всі вершини і ребра графа знаходяться в одній площині, то він називається **плоским**, у протилежному випадку – **просторовим.**

Якщо ребро з'єднує дві вершини, то кажуть, що воно **інцидентне** цим вершинам, а вершини, які з'єднані таким ребром, називаються **суміжними**. Якщо кінці ребра належать одній вершині, то таке ребро називається **петлею**.

Ланцюг – така послідовність ребер неорієнтованого графа, в якій кінець одного ребра є початком іншого, а початкова і кінцева вершини не збігаються. Аналогічно визначається шлях на орграфі. Перша у ланцюгу чи шляху вершина називається початковою, остання – кінцевою, всі інші – проміжними. Маршрут М називається **ланцюгом**, якщо всі його ребра різні. Маршрут М називається **простим ланцюгом**, якщо всі його вершини різні. Замкнений ланцюг називається **циклом**. Замкнений простий ланцюг називається **простим циклом**. Граф без циклів називають **ациклічним.**

Ребро називається **спрямованим**, або **орієнтованим** (у цьому разі ребро будемо називати **дугою**), якщо в одному напрямку можливий тільки потік додатних значень, а в протилежному – тільки нульовий. В **орієнтованій мережі** всі ребра орієнтовані.

Шляхом називається послідовність різних ребер, що з'єднують два вузли незалежно від напрямку потоку в кожнім ребрі. Шлях формує **цикл**, якщо початковий і кінцевий вузли збігаються. **Орієнтований цикл**– це цикл, у якому всі дуги орієнтовані у визначеному напрямку.

Такий граф, у якому для всіх ребер вказано напрям, називається **орієнтованим**, або **орграфом.** *Для орграфів ланцюг* називається **шляхом**, а *цикл* – **контуром**.

Граф називається **зв'язаним**, якщо для будь-яких двох його вершин існує шлях, що їх з'єднує; в іншому випадку граф називається **незв'язаним**.

Найчастіше використовуються два види графів: **дерево і мережа**. **Дерево** представляє собою зв'язаний граф без циклів, що має вихідну вершину (корінь) і крайні вершини; шляхи від вихідної вершини до крайніх вершин називаються **гілками**. **Остове дерево –** це дерево, що містить *усі* вузли графа.

Мережа – це орієнтований кінцевий зв'язаний граф, що має початкову вершину (джерело) і кінцеву вершину (стік). Таким чином, *мережева модель*представляє собою граф виду «мережа».

Застосування мережевого планування при розробці проєктів виконання різних комплексів робіт з організації виробничого процесу на авіаційних підприємствах

Мережеве планування – це метод наукового планування та управління складними виробничими процесами; одна з форм графічного відображення змісту робіт і тривалості виконання стратегічних планів і довгострокових комплексів проєктних, планових, організаційних та інших видів діяльності підприємства.

Під ***мережевим плануванням*** прийнято розуміти графічне зображення певного комплексу виконуваних робіт, що відображає їх логічну послідовність, існуючий взаємозв'язок і заплановану тривалість та забезпечує подальшу оптимізацію розробленого графіка на основі економіко-математичних методів і комп'ютерної техніки з метою його використання для поточного управління ходом робіт.

Початкові ідеї мережевого планування були розроблені в кінці 1950-х років в США і реалізовані у вигляді трьох систем мережевого аналізу – CPM (Critical Path Method – метод критичного шляху), PERT (Program Evaluation and Review Technique – програмна оцінка і аналіз) і GERT (Graphic Evaluation and Review Technique – графічна оцінка і аналіз).

Вперше плани-графіки виконання виробничих процесів були застосовані на американських фірмах Г. Рантьє. На лінійних або стрічкових графіках по горизонтальній осі в обраному масштабі часу відкладається тривалість робіт за всіма стадіями, етапами виробництва. Зміст циклів робіт зображується по вертикальній осі з необхідним ступенем їх розчленування на окремі частини або елементи. Циклові або лінійні графіки звичайно застосовуються на вітчизняних підприємствах у процесі короткострокового чи оперативного планування виробничої діяльності. Основним недоліком таких планів-графіків є відсутність можливості тісної взаємозв’язки окремих робіт в єдину виробничу систему або загальний процес досягнення запланованих кінцевих цілей підприємства. На відміну від лінійних графіків, мережеве планування служить основою економічних та інженерних розрахунків, графічних і аналітичних обчислень, організаційних і управлінських рішень, оперативних і стратегічних планів, що забезпечують не тільки зображення, а й моделювання, аналіз і оптимізацію проєктів виконання складних технічних об'єктів і конструкторських розробок і т.д.

В Україні роботи з мережевого планування почалися в 1960-х роках. Тоді методи мережевого планування управління знайшли застосування в будівництві та наукових розробках. Надалі мережеві методи стали широко застосовуватися і в інших галузях народного господарства,в тому числі і на авіаційному транспорті. На авіаційному транспорті методами мережевого планування описуються технологічні процеси пілотування, управління повітряним рухом, технічного обслуговування і ремонту літаків, авіаперевізні процеси тощо.

Технологія роботи авіаційного спеціаліста (пілота, диспетчера) відповідає чіткому алгоритму дій, що прописані у нормативних та регламентуючих документах, тому для моделювання дій авіаспеціаліста, наприклад, в особливих випадках в польоті (ОВП), можна застосовувати детерміновані моделі. Оскільки ОВП – це не одномоментна подія, а подія, що розвивається в часі, то для моделювання прийняття рішень людиною-оператором (Л-О) відповідно до алгоритму дій у разі виникнення ОВП доцільно користуватися мережевими графіками. Мережеве планування – визначення оптимальної послідовності виконання операцій.

Мережевий графік виконання операційних процедур (дій) оператором у разі виникнення аварійних чи непередбачуваних операцій – це орієнтований граф без контурів, який має вузли і дуги. Вузли графу відповідають події на початку (наприкінці) операційної процедури (дії) авіадиспетчера (пілота, членів екіпажу повітряного судна), наприклад, у разі виникнення ОВП. Дуги інтерпретують суть операційних процедур (дій) відповідно до технології (інструкції, керівництва з льотної експлуатації цього типу повітряного судна (ПС)).

Формалізація дій оператора (пілота, диспетчера) в ОВП за допомогою апарату мережевого планування і управління дозволяє визначити:

1) оптимальну послідовність операційних процедур (дій) Л-О в разі виникнення ОВП;

2) критичний (оптимальний) час виконання операційних процедур (дій) Л-О на парирування ОВП;

3) максимальний (мінімальний) час виконання операційних процедур (дій) Л-О в разі виникнення ОВП;

4) резерви часу для парирування ОВП тощо.

Мережеве планування має ряд переваг:

- забезпечує наочність технологічної послідовності робіт;
- дозволяє скласти оперативні та поточні плани, а також прогнозувати складні процеси;
- дозволяє виявити приховані ресурси часу і матеріальних засобів при виконанні виробничих процесів.

Щоб приступити до мережевого планування (моделювання) виробничого процесу, необхідно мати перелік і тривалість виконання операцій, відповідних виробничому процесу. Мережеве планування супроводжується побудовою структурно-часових таблиць і мережевих графіків.

Побудова мережі проєкту. Критичні роботи. Метод критичного шляху. Часовий графік

Основними поняттями мережевого планування є наступні: робота, подія, шлях. На рис. 3.1 графічно представлена мережева модель, яка складається з 5 подій (кружечки) і 6 робіт (стрілки); тривалість виконання робіт в деяких одиницях часу вказана над стрілками.

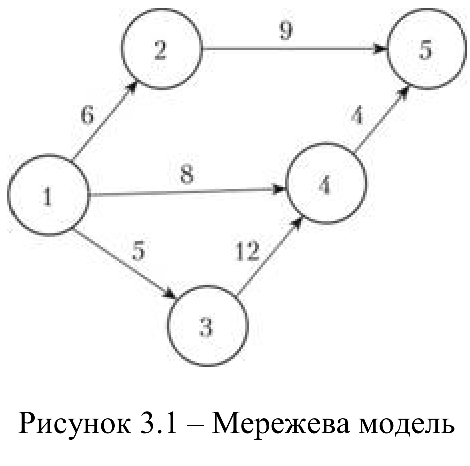


Рисунок 3.1 – Мережева модель

На мережевому графіку **події** зображуються *вершинами*, а **роботи** – *дугами*.

Робота характеризує матеріальну дію, що вимагає використання ресурсів, або логічну дію, яка вимагає лише взаємозв'язку подій. При графічному поданні робота зображується стрілкою, яка з'єднує дві події. Вона позначається парою укладених у дужки чисел *(i, j)*, де *i*– номер події, з якої робота виходить, а *j*– номер події, в яку вона входить. Робота не може початися раніше, ніж здійсниться подія, з якої вона виходить. Кожна робота має певну тривалість *t(i, j)*. Наприклад, запис *t*(2,5)=4 означає, що робота (2,5) має тривалість 4 одиниці. До робіт відносяться також такі процеси, які не вимагають ні ресурсів, ні часу виконання. Вони полягають у встановленні логічного взаємозв'язку робіт і показують, що одна з них безпосередньо залежить від іншого і не може виконуватися, перш ніж ця інша буде завершена; такі роботи називаються **фіктивними** і на графіку зображуються пунктирними стрілками.

Подіями називаються результати виконання однієї або декількох робіт. Вони не мають протяжності в часі. Подія здійснюється в той момент, коли закінчується остання з робіт, що входить до неї. Події позначаються одним числом і при графічному представленні мережевої моделі зображуються колом (чи іншої геометричної фігурою), усередині якого проставляється його порядковий номер ($i=1, 2, ..., n$).

У мережевій моделі є початкова подія (з номером 1), з якої роботи тільки виходять, і кінцева подія (з номером N), в яку роботи тільки входять.

Шлях – це ланцюжок наступних одна за одною робіт, що з'єднують початкову і кінцеву вершини (рис. 3.2-3.3). Шлях, що має максимальну довжину, називають ***критичним*** і позначають $L_{кр}$, а його тривалість – $t_{кр.}$ Роботи, що належать критичному шляху, називаються ***критичними.*** Їх несвоєчасне виконання веде до зриву термінів всього комплексу робіт.

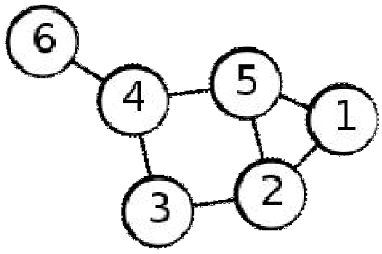


Рисунок 3.2 – (6, 4, 5, 1) і (6, 4, 3, 2, 1) є шляхами між вершинами 6 та 1

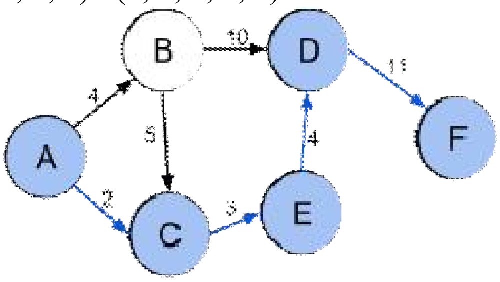


Рисунок 3.3 – Найкоротший шлях (A, C, E, D, F) між вершинами A та F у зваженому орієнтованому графі

При побудові мережі проєкту необхідно дотримуватись таких правил:

1) в мережі, крім завершальної події, не повинно бути тупикових подій, з яких не виходить жодна робота;

2) в мережі, крім вихідної події, не повинно бути хвостових подій, яким не передує жодна робота;

3) в мережі не повинно бути замкнутих контурів (циклів) та петель;

4) будь-які дві події повинні бути зв’язані лише однією роботою;

5) в мережі повинні бути лише одна вхідна та одна завершальна події.

Наявність в мережі тупикових та хвостових подій, а також циклів вказує на необхідність проведення більш ретельного аналізу робіт та взаємозв’язків між ними. Якщо в проєкті потрібно виконати декілька робіт при одних і тих же початковій та кінцевій подіях, то в таких випадках необхідно ввести фіктивні події та роботи.

Графічно правила побудови мережі наведено на рис. 3.4.

Якщо процес реально можна почати або завершити з кількох робіт, то необхідно ввести фіктивні вихідну та завершальну події.

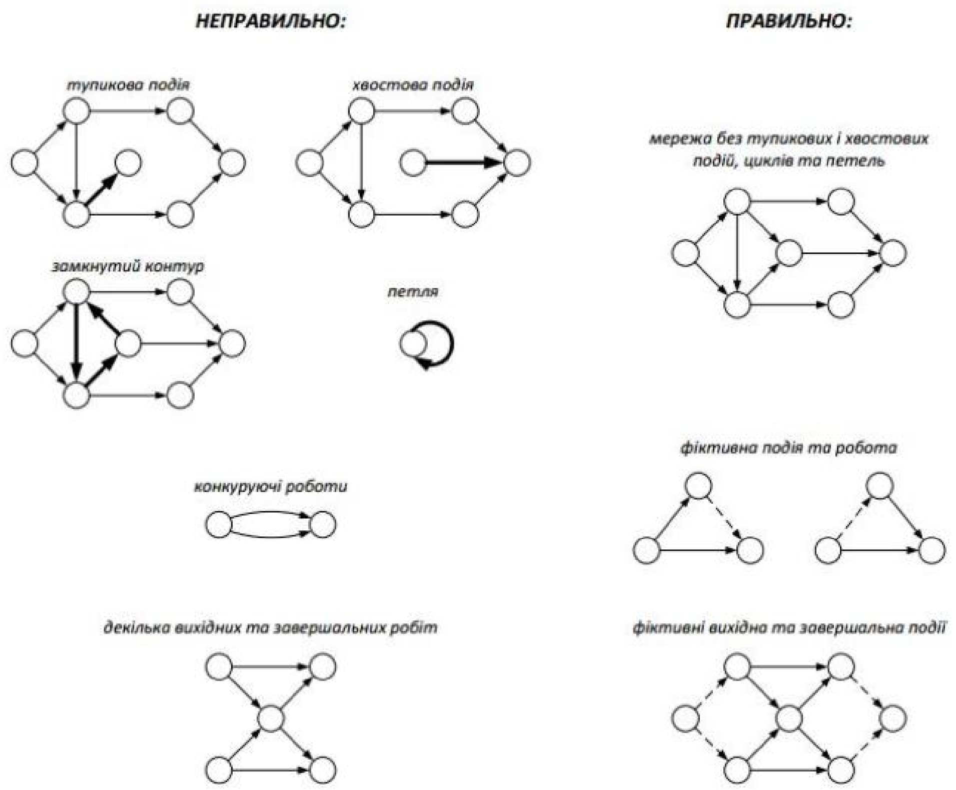


Рисунок 3.4 – Правила побудови мережі проєкту

Припустимо, що чотири роботи повинні задовольняти таким правилам:

- робота С повинна початись одразу після завершення робіт А і В;
- робота Е повинна початись одразу після завершення роботи В.

На рис. 3.5, а наведено неправильне зображення робіт, оскільки виходить, що робота Е повинна початись як після завершення як роботи В, так і роботи А. На рис. 3.2, б наведено, як за допомогою фіктивної роботи D вирішити цю проблему.

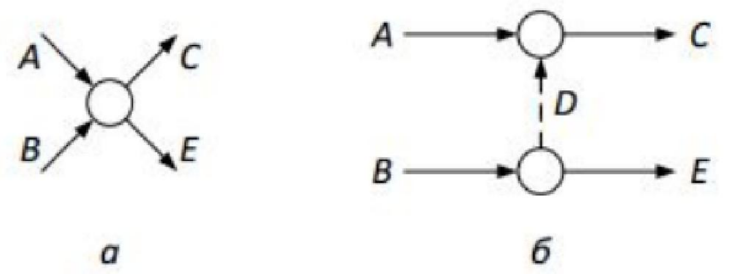


Рисунок 3.5 – Використання фіктивної роботи для правильного відображення послідовності робіт

Вершини мережі можна нумерувати довільно, але для розв’язування задач зручно використовувати так звану правильну нумерацію подій – нумерацію події, за якої номер кожної наступної події більший, ніж номер будь-якої попередньої.

Процес мережевого планування і управління (МПУ) включає в себе чотири взаємопов'язаних етапи:

1. Опис комплексу робіт, визначення їх тривалості та послідовності (побудова табл. 3.1).

Таблиця 3.1 – Структурно-часова таблиця (перелік робіт у складі мережевого графіка)

№ з/п	Зміст робіт	Позначення робіт	Спираються на роботи	Тривалість, год.

2. Побудова мережевого графіка.

3. Розрахунок і аналіз параметрів мережевого графіка.

4. Оптимізація мережевого графіка, контроль і оперативне управління ходом виконання комплексом робіт.

Метод критичного шляху (CPM) передбачає, що тривалість виконання кожної роботи відома. В результаті використання методу CPM можна визначити:

- мінімальний час виконання проєкту;
- множину критичних та некритичних робіт;
- час початку та закінчення виконання окремих робіт.

Критичний шлях – найдовший шлях (максимальної довжини) від вхідної до завершальної події. Критичний шлях характеризує мінімальну тривалість виконання всього комплексу робіт. Роботи та події, що лежать на критичному шляху, називаються **критичними**. Їх несвоєчасне виконання веде до зриву термінів всього комплексу робіт.

Всі події характеризуються:

- раннім терміном настання події – E_i;
- пізнім терміном настання події – L_i.

Всі роботи характеризуються:

- тривалістю роботи – t_{ij};
- раннім терміном початку роботи $ES_{ij} = E_i$;
- пізнім терміном закінчення роботи $LF_{ij} = L_j$;
- пізнім терміном початку роботи $LS_{ij} = LF_{ij} – t_{ij}$;
- раннім терміном закінчення роботи $EF_{ij} = ES_{ij} + t_{ij}$;
- повним резервом часу $R_{ij} = LS_{ij} - ES_{ij}$(проміжок часу між раннім та пізнім термінами початку роботи);
- вільним резервом часу $r_{ij} = E_j - E_i – t_{ij}$ (проміжок часу між раннім терміном настання події та раннім терміном настання подіїj).

Для критичної роботи (i, j) виконуються умови:

$$E_i = L_i;$$
$$E_j = L_j;$$
$$L_i - L_j = E_j - E_i = t_{ij}.$$

Повний резерв часу R_{ij} визначає час, на який може бути відкладена робота без збільшення тривалості виконання проєкту.

Вільний резерв часу r_{ij} визначає час, на який може бути відкладена робота без збільшення раннього терміну настання наступної події.

Якщо $R_{ij} = r_{ij}$, то робота може початись в будь-який час в межах від раннього до пізнього терміну початку роботи без збільшення тривалості проєкту. Якщо $R_{ij}< r_{ij}$, то зсув початку роботи на величину, що є в межах від r_{ij} до $R_{ij},$ супроводжується зсувом всіх наступних некритичних робіт без збільшення тривалості проєкту.

Критичні роботи резерву часу не мають.

Визначення критичного шляху та часових параметрів мережевого графіка виконують в два проходи. При проході вперед визначаються ранні терміни настання подій, при проході назад – пізні терміни настання тих же подій.

Прохід вперед (з вихідної до завершальної події):

- *початковий крок*: приймаємо $E_0 = 0$, тобто проєкт починається в нульовий момент часу;
- *основний крок*: для вузла j визначаємо вузли $p, q, ..., v$, що безпосередньо зв'язані з вузлом j роботами $(p, j), (q, j), ... , (v, j),$ для яких вже обчислено ранні терміни настання подій; ранній термін настання події j визначається за формулою:
- $E_j = max\{Ep + t_{pj} ; E_q + t_{qj}; ... ; E_v + t_{vj}\}$;
- прохід завершується, коли буде визначено ранній початок завершальної події E_n.

Прохід назад (із завершальної до вихідної події):

- *початковий крок*: приймаємо $L_n = E_n$, тобто ранній та пізній терміни настання завершальної події співпадають;

- *основний крок*: для вузла *j* визначаємо вузли *p, q, ..., v*, що безпосередньо зв’язані з вузлом *j* роботами *(j, p), (j, q), ..., (j, v)*, для яких вже обчислено пізні терміни настання подій; пізній термін настання події *j* визначається за формулою:

- $L_j = min\{L_p + t_{pj}; L_q + t_{qj}; ...; L_v + t_{vj}\}$;

- прохід завершується, коли буде визначено пізній початок початкової події E_0.

У реальних проєктах кожна робота характеризується не тільки часом, але і вартістю виконання. У цьому випадку повна вартість проєкту дорівнює сумі вартостей всіх назв робіт.

Кінцевим результатом застосування CPM є побудова **часового графіка виконання проєкту**. Для цього проводяться спеціальні обчислення, в результаті чого отримують наступну інформацію.

- загальна тривалість виконання проєкту;
- поділ множини процесів, що становлять проєкт, на критичні та некритичні.

Процес є критичним, якщо він не має зазору для часу свого початку і завершення. Таким чином, щоб весь проєкт завершився без затримок, необхідно, щоб всі критичні процеси починалися і закінчувалися в строго визначений час. Для некритичного процесу можливий деякий дрейф часу його початку, але в певних межах, коли час його початку не впливає на тривалість виконання всього проєкту.

Мережевий графік проєкту – мережа, накреслена без масштабу часу. Через це мережевий графік хоч і дає уявлення про порядок виконання робіт, але недостатньо наглядний для визначення робіт, які виконуються в кожний момент часу. Тому, крім мережевого графіка проєкту, будують також **часовий графік(діаграма Гантта)**. На осі абсцис відкладається час, на осі ординат – роботи. Кожна робота зображується у вигляді паралельного до осі часу відрізку, довжина якого дорівнює тривалості роботи (рис. 3.6).

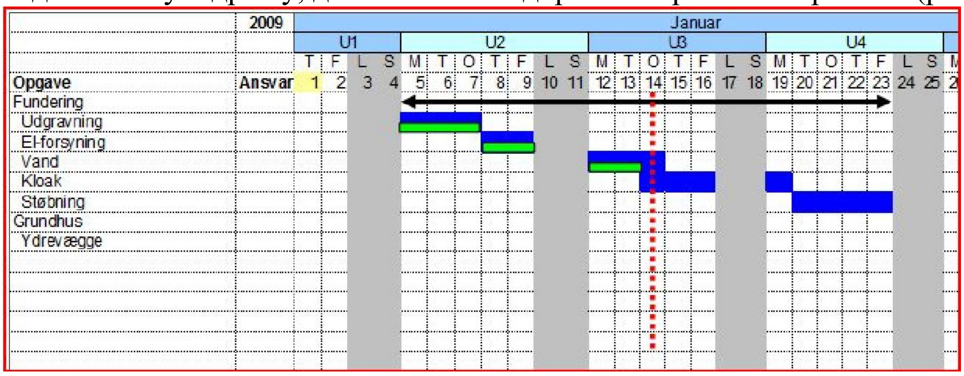


Рисунок 3.6 – Діаграма Гантта

Фіктивна робота нульової тривалості зображується точкою. Критичні роботи розташовуються послідовно без часового зазору і перекриття. Некритичні роботи подаються максимальними інтервалами часу виконання, які перевищують тривалість виконання цих робіт. Як правило, некритичні процеси починають якомога раніше. В даному випадку залишається запас часу, який можна використати для вирішення проблем, що неочікувано виникли під час виконання роботи. Разом з тим за необхідності можна перенести початок виконання якої-небудь роботи. Припустимо, що для двох некритичних робіт, часові інтервали яких перекриваються, використовується одне і те ж обладнання, причому в кожний момент часу його можна задіяти лише для однієї роботи. В цьому випадку потрібно виключити часове накладання таких робіт.

Приклад. *Планування виконання дій авіадиспетчера ОВП – моделювання прийняття рішення Л-О у разі виникнення ОВП на етапі зльоту – зіткнення ПС з птахом.*

Побудуємо структурно-часову таблицю виконання дій авіадиспетчером у разі потрапляння птаха у двигун ПС (табл.3.2). Час, необхідний для виконання дій, спрямованих на парирування ОВП, визначений за допомогою методу експертних оцінок.

Таблиця 3.2 – Структурно-часова таблиця виконання дій авіадиспетчером при попаданні птаха у двигун ПС

№ з/п	Робота	Опис роботи	Час виконання роботи, t, с
1	a_1	Отримати від командира ПС повідомлення про зіткнення з птахом	5
2	a_2	Повідомити командира ПС місцезнаходження, зафіксувати час	10
3	a_3	Перевірити встановлення командиром ПС сигналу «Лихо»	5
4	a_4	Доповісти керівнику польотів про виникнення ОВП на борту ПС	10
5	a_5	При необхідності ввести режим радіомовчання	10
6	a_6	Уточнити подальші наміри командира ПС щодо негайного заходу на посадку на аеродромі вильоту або необхідності зливу палива	10
7	a_7	При необхідності сформувати зону зливу палива і направити до неї ПС	15
8	a_8	Передати інформацію про аварійний ПС	5
9	a_9	Запросити у АМСЦ погоду для посадки	10
10	a_{10}	Отримати доповідь від командира ПС про завершення зливу палива	5
11	a_{11}	Дати вказівки екіпажу ПС для заходу на посадку	10
12	a_{12}	Передати ПС на управління диспетчеру АДВ	5

За даними табл.3.2 будуємо мережевий графік виконання комплексу дій диспетчером (рис. 3.7).

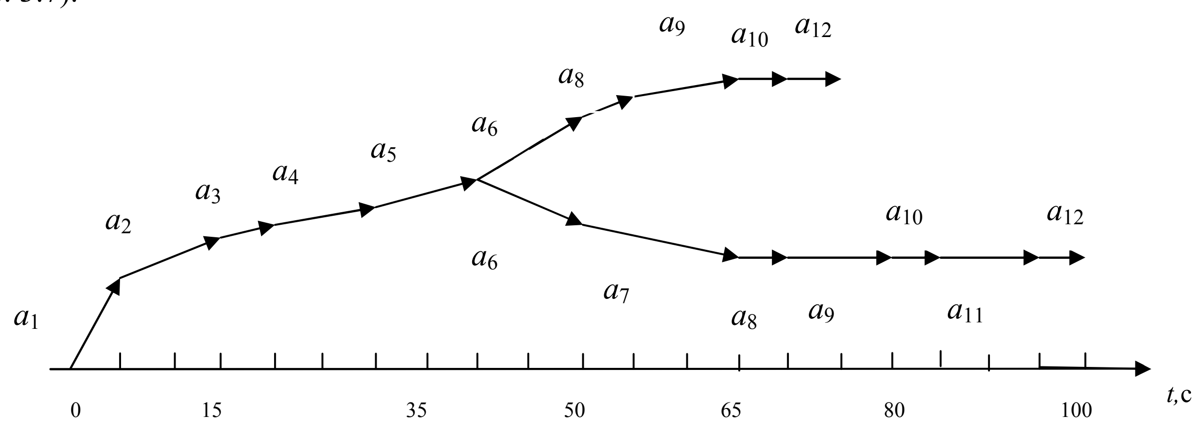


Рисунок3.7 – Мережевий графік виконання дій, спрямованих на парирування ОВП – зіткнення з птахом на етапі зльоту

Наведений мережевий графік прийняття рішень Л-О при виникненні ОВП (зіткнення ПС з птахом) дозволяє визначити критичний час залежно від рішення, прийнятого командиром ПС (здійснювати посадку на аеродромі вильоту негайно або спочатку злити паливо), який становить $T_{крит1}$= 75сі$T_{крит2}$=100 с відповідно.

Метод оцінки та перегляду планів (PERT) є різновидом аналізу за методом критичного шляху з більш критичною оцінкою тривалості кожного етапу проєкту. При використанні цього методу необхідно оцінити найменш можливу тривалість виконання кожної роботи, найбільш ймовірну тривалість і найбільшу тривалість на той випадок, якщо тривалість виконання цієї роботи буде більше очікуваної. Метод PERT допускає невизначеність тривалості операцій і аналізує вплив цієї невизначеності на тривалість робіт по проєкту в цілому.

Цей метод використовується, коли для операції складно задати і визначити точну тривалість.

Особливість методу PERT полягає в можливості обліку імовірнісного характеру тривалостей всіх або деяких робіт при розрахунку параметрів часу на мережевій моделі. Він дозволяє визначати ймовірності закінчення проєкту в задані періоди часу і до заданих термінів.

Метод PERT відрізняється від CPM тим, що тут тривалість процесів характеризується трьома оцінками:

а) оптимістична оцінка часу *a*, коли передбачається, що виконання процесу буде відбуватися максимально швидко;

б) найбільш ймовірна оцінка часу *m*, коли передбачається, що виконання процесу буде відбуватися нормально;

в) песимістична оцінка часу *b*, коли передбачається, що виконання процесу буде відбуватися дуже повільно.

Будь-яка можлива оцінка часу виконання процесу буде лежати в інтервалі *(a, b)*. Тому оцінка часу *m* також повинна лежати в цьому інтервалі.

Метод графічної оцінки й аналізу (GERT) застосовується в тих випадках організації робіт, коли наступні завдання можуть починатися після завершення тільки деякого числа з попередніх завдань, причому не всі завдання, представлені на мережевій моделі, повинні бути виконані для завершення проєкту.

Основу застосування методу GERT становить використання альтернативних мереж, званих в термінах даного методу GERT-мережами.

По суті, GERT-мережі дозволяють більш адекватно описувати складні технологічні процеси в тих випадках, коли важко або неможливо (з об'єктивних причин) однозначно визначити, які саме роботи і в якій послідовності повинні бути виконані для досягнення наміченого результату (тобто, існує багатоваріантність реалізації проєкту).

Слід зазначити, що ручний розрахунок GERT-мереж, що моделюють реальні процеси, надзвичайно складний, однак програмне забезпечення для обчислення мережевих моделей такого типу в даний час, на жаль, не поширене.

Задача про найкоротший маршрут. Алгоритм Дейкстри. Алгоритм Флойда

Задача про найкоротший шлях(маршрут) полягає у знаходженні найкоротшого шляху від заданої початкової вершини до заданої кінцевої вершини. Прикладом може бути знаходження найкоротшого шляху між двома пунктами на аеронавігаційній мапі; в цьому випадку, вершинами є пункти, а ребрами – відрізки дороги із вагами, рівними часу, необхідному для подолання цього відрізку.

Формулювання задач про знаходження відстаней таке:

- для заданої початкової вершини *a* знайти найкоротші шляхи від *a* до всіх інших вершин;
- знайти найкоротші шляхи між усіма парами вершин.

Найважливіші алгоритми для розв'язання цієї задачі:

- алгоритм Дейкстри розв'язує задачі з однією парою, одним входом і одним виходом;
- алгоритм Беллмана-Форда розв'язує задачі з одним входом, якщо ваги ребер можуть бути від'ємні;
- алгоритм пошуку *A** розв'язує задачу для однієї пари із використанням евристики в спробі пришвидшити пошук;
- алгоритм Флойда-Воршелла розв'язує задачу для всіх пар;
- алгоритм Джонсона розв'язує задачу для всіх пар, і може бути швидшим за алгоритм Флойда-Воршелла на розріджених графах;
- теорія збурень знаходить (в найгіршому випадку) локально найкоротший шлях.

Алгоритм Дейкстри – найефективніший алгоритм на графах, відкритий нідерландським вченим Е.Дейкстрою у 1959році. Цей алгоритм знаходить найкоротшу відстань від однієї вершини графу до всіх інших. Алгоритм працює лише для графів, у яких ребра не мають від'ємної ваги. Алгоритм широко застосовується в програмуванні і технологіях, наприклад, його використовують протоколи маршрутизації OSPF та ISIS.

Нехай $G = (V, E)$ – зважений орієнтований граф, $w(v_i, v_j)$ – вага дуги (v_i, v_j). Почавши з вершини a, знаходимо відстань від a до кожної із суміжних із нею вершин. Вибираємо вершину, відстань від якої до вершини a найменша; нехай це буде вершина v^*. Далі знаходимо відстані від вершини a до кожної вершини, суміжної з v^* вздовж шляху, який проходить через вершину v^*. Якщо для якоїсь із таких вершин ця відстань менша від поточної, то замінюємо нею поточну відстань. Знову вибираємо вершину, найближчу до a та не вибрану раніше; повторюємо процес.

Описаний процес зручно виконувати за допомогою присвоювання вершинам міток. Є мітки двох типів: тимчасові та постійні. Вершини з постійними мітками групуються у множину M, яку називають **множиною позначених вершин**. Решта вершин має тимчасові мітки, і множину таких вершин позначимо як T, $T = V \setminus M$. Позначатимемо мітку (тимчасову чи постійну) вершини v як $l(v)$. Значення постійної мітки $l(v)$ дорівнює довжині найкоротшого шляху від вершини a до вершини v, тимчасової – довжині найкоротшого шляху, який проходить лише через вершини з постійними мітками. Фіксованою початковою вершиною вважаємо вершину a; довжину найкоротшого шляху шукаємо до вершини z (або до всіх вершин графа). Тепер формально опишемо алгоритм Дейкстри.

Алгоритм Дейкстри:

1 Присвоювання початкових значень. Виконати $l(a) = 0$ та вважати цю мітку постійною. Виконати $l(v) = \infty$ для всіх $v \neq a$ й уважати ці мітки тимчасовими. Виконати $x = a$, $M = \{a\}$.

2 Оновлення міток. Для кожної вершини $v \in \Gamma(x) \setminus M$ замінити мітки:

$l(v) = min\{l(v), l(x) + w(x, v)\}$, тобто оновлювати тимчасові мітки вершин, у які з вершини x іде дуга.

3 Перетворення мітки в постійну. Серед усіх вершин із тимчасовими мітками знайти вершину з мінімальною міткою, тобто знайти вершину v^* з умови $l(v^*)= min\{l(v)\}, v\in T,$ де $T = V \setminus M$.

4 Уважати мітку вершини v* постійною й виконати $M = M \cup \{v^*\}; x = v^*$ (вершину v^* включено в множину M).

5 а) Для пошуку шляху від a до z: якщо $x = z$, то $l(z)$ – довжина найкоротшого шляху від a до z, зупинитись; якщо $a \neq z$, то перейти до кроку 2.

б) Для пошуку шляхів від a до всіх вершин: якщо всі вершини отримали постійні мітки (включені в множину M), то ці мітки дорівнюють довжинам найкоротших шляхів, зупинитись; якщо деякі вершини мають тимчасові мітки, то перейти до кроку 2.

Алгоритм Флойда-Воршелла – це алгоритм динамічного програмування для знаходження найкоротших відстаней між усіма вершинами зваженого орієнтованого графа. Розроблений в 1962 році Робертом Флойдом і Стівеном Воршеллом.

Для знаходження найкоротших шляхів між всіма вершинами графа використовується не перебір всіх можливостей, що приведе до використання значного часу та витрат більшого об'єму пам'яті, а динамічне програмування, тобто ті задачі, які будуть потрібні для вирішення головної задачі, прораховуються завчасно і потім використовуються.

Цей алгоритм більш загальний у порівнянні з алгоритмом Дейкстри, тому що він знаходить найкоротші шляхи між **будь-якими** двома вузлами мережі. У цьому алгоритмі мережа представлена у вигляді квадратної матриці з n рядками та n стовпцями. Елемент (i, j) дорівнює відстані d_{ij}від вузла i до вузла j, що має кінцеве значення, якщо існує дуга (i, j), і дорівнює нескінченності в противному випадку.

Покажемо спочатку основну ідею методу Флойда. Нехай є три вузли i, j та k і задані відстані між ними (рис. 3.6).

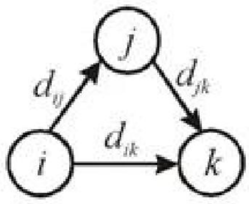


Рисунок 3.6 – Визначення трикутного оператора

Якщо виконується нерівність $d_{ij} + d_{jk} < d_{ik}$, то доцільно замінити шлях $i \to k$ шляхом $i \to j \to k$. Така заміна (далі її будемо умовно називати **трикутним** оператором) виконується систематично в процесі виконання алгоритму Флойда.

Алгоритм Флойда вимагає виконання наступних дій.

Крок 0. Визначаємо початкову матрицю відстаней D_0 і матрицю послідовності вузлів S_0. Діагональні елементи обох матриць (рис. 3.7) позначаються знаком "–", що показує, що ці елементи в обчисленнях не беруть участь. Вважаємо $k = 1$.

Основний крок *k*. Задаємо рядок k і стовпець k як **ведучий рядок** і **ведучий стовпець**. Розглядаємо можливість застосування **трикутного оператора** до всіх елементів d_{ij} матриці
D_{k-1}.

Якщо виконується нерівність $d_{ik}+d_{kj}< d_{ij}(i\neq k, j\neq k$ та $i\neq j)$, тоді виконуємо наступні дії:

а) створюємо матрицю D_k шляхом заміни в матриці D_{k-1} елемента d_{ij} на суму $d_{ik}+d_{kj}$;

b) створюємо матрицю S_k шляхом заміни в матриці S_{k-1} елемента s_{ij}на k. Вважаємо $k = k +1$ і повторюємо крок k.

D_0 =

	1	2	…	j	…	n
1	—	d_{12}	…	d_{1j}	…	d_{1n}
2	d_{21}	—	…	d_{2j}	…	d_{2n}
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
i	d_{i1}	d_{i2}	…	d_{ij}	…	d_{in}
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
n	d_{n1}	d_{n2}	…	d_{nj}	…	—

S_0 =

	1	2	…	j	…	n
1	—	2	…	j	…	n
2	1	—	…	j	…	n
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
i	1	2	…	j	…	n
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
n	1	2	…	j	…	—

Рисунок 3.7 – Початкова ситуація

Пояснимо дії, виконувані на k-му кроці алгоритму, представивши матрицю D_{k-1} так, як вона показана на рис. 3.8. На цьому рисунку рядок k і стовпець k є ведучими. Рядок i – будь-який рядок з номером від 1 до $k–1$, а рядок p – довільний рядок з номером від $k + 1$ до n. Аналогічно стовпець j представляє будь-який стовпець із номером від 1 до $k–1$, а стовпець q – довільний стовпець із номером від $k + 1$ до n.

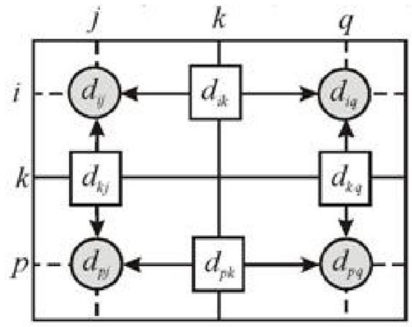


Рисунок 3.8 – Фрагмент матриці D_{k-1}

Трикутний оператор виконується в такий спосіб. Якщо сума елементів ведучого рядка та стовпця (показаних у квадратиках) менше елементів, що перебувають на перетині стовпця та рядка (показані в кружках), які відповідають розглянутим ведучим елементам, то відстань (елемент у кружку) заміняється на суму відстаней, представлених ведучими елементами.

Після реалізації n кроків алгоритму визначення за матрицями D_n іS_n найкоротшого шляху між вузлами i і j виконується за наступними правилами:

1 Відстань між вузлами i і j дорівнює елементу d_{ij} у матриці D_n.

2 Проміжні вузли шляху від вузла i до вузла j визначаємо за матрицею S_n. Нехай $s_{ij} = k$, тоді маємо шлях $i \rightarrow k \rightarrow j$. Якщо далі $s_{ik} = k$ та $s_{kj} = j$, тоді вважаємо, що весь шлях визначений, тому що знайдені всі проміжні вузли. У противному випадку повторюємо описану процедуру для шляхів від вузла i до вузла k і від вузла k до вузла j.

Задача про максимальний потік. Розріз. Перебір розрізів

В теорії оптимізації та теорії графів, ***задача про максимальний потік*** полягає у знаходженні такого потоку за транспортною мережею, щоб сума потоків з джерела, або, що означає те ж саме, сума потоків до стоку, була максимальна.

Алгоритм знаходження максимального потоку

Ідея даного алгоритму полягає в пошуку наскрізних шляхів з позитивними потоками від джерела до стоку.

Розглянемо ребро (i, j) з (початковою) пропускною здатністю (C_{ij}, C_{ji}). У процесі виконання алгоритму частини цих пропускних здатностей "забираються" потоками, що проходять через дане ребро, в результаті кожне ребро буде мати залишкову пропускну здатність.

Будемо використовувати запис (C_{ij}, C_{ji}) для представлення залишкових пропускних здатностей. Мережа, в якій всі ребра мають залишкову пропускну здатність, називається залишковою.

Для довільного вузла j, який одержує потік від вузла i, визначимо мітку [a, i], де a_j – величина потоку, що протікає від вузла j до вузла i.

Щоб знайти максимальний потік, виконаємо наступні дії.

Етап 1. Для всіх ребер (i, j)приймемо залишкову пропускну здатність рівну первісній пропускній здатності. Призначимо $a_1 = \infty$ і помітимо вузол 1 міткою [∞, -]. Вважаємо $i= 1$ і переходимо до другого етапу.

Етап 2. Визначаємо множину S як множину вузлів j, в які можна перейти з вузла i по ребру з позитивною залишкової пропускною здатністю. Якщо $S \neq \emptyset$ виконуємо третій етап, в іншому випадку переходимо до етапу 4.

Етап 3. У множині S знаходимо вузол K, такий, що $C_{ik} = \max\{c_{ij}\}$. Приймемо $a_k = c_{ik}$ і помітимо вузол K міткою [a_k, i]. Якщо останній міткою позначений вузол стоку (тобто якщо $k = n$), наскрізний шлях знайдений, і ми переходимо до п'ятого етапу. В іншому випадку вважаємо $i = k$ і повертаємося до етапу 2.

Етап 4. Відкат назад. Якщо i= 1, наскрізний шлях неможливий, і ми переходимо до етапу 6. Якщо $i \neq 1$, знаходимо позначений вузол r, безпосередньо передуючий вузлу i, та видаляємо вузол i з безлічі вузлів, суміжних з вузлом r. Вважаємо $i = r$ і повертаємося до другого етапу.

Етап 5. Визначення залишкової мережі. Позначимо через $N_p = \{1, k_1, k_2, ..., n\}$множину вузлів, через які проходить p-й знайдений наскрізний шлях від вузла джерела (вузол 1) до вузла стоку (вузол n). Тоді максимальний потік, що проходить цим шляхом, обчислюється як f_p=min$\{a_1, a_{k1}, ..., a_n\}$.

Залишкові пропускні здатності ребер, що складають наскрізний шлях, зменшуються на величину f_p в напрямку руху потоку і збільшуються на цю ж величину в протилежному напрямку.

Таким чином, для ребра (i, j), що входить в наскрізний шлях, поточні залишкові пропускні здатності (C_{ij}, C_{ji}) зміняться таким чином:

$(C_{ij}-f_p,\ C_{ji}+f_p)$, якщо потік йде від вузла i до вузла j;

$(C_{ij}+f_p,\ C_{ij}-f_p)$, якщо потік йде від вузла j до вузла i.

Далі відновлюємо всі вузли, вилучені на етапі 4. Вважаємо i= 1 і повертаємося до другого етапу для пошуку нового наскрізного шляху.

Етап 6. Рішення.

a) При m знайдених наскрізних шляхах максимальний потік обчислюється за формулою $F=f_1+f_2, ..., +f_m$.

b) Маючи значення початкових і кінцевих пропускних здатностей ребра (i, j), можна обчислити оптимальний потік через це ребро таким чином. Приймемо$(a, \beta) = (C_{ij}-c_{ij}, C_{ji}-c_{ji})$. Якщо a>0, потік, що проходить через ребро (i, j), дорівнює a. Якщо ж $\beta>0$, тоді потік дорівнює β. Випадок, коли одночасно $a>0$ і $\beta>0$, неможливий.

Розріз визначає безліч ребер, при видаленні яких з мережі повністю припиняється потік від джерела до стоку. ***Пропускна здатність розрізу*** дорівнює сумі пропускних здатностей "розрізаних" ребер. Серед всіх розрізів мережі розріз *з мінімальною пропускною здатністю* визначає максимальний потік в мережі.

Якщо на мережі заданий деякий потік, то ребро називають *насиченим*, якщо величина потоку по цьому ребру збігається з його пропускною здатністю, і *ненасиченим*, якщо потік менше його пропускної здатності.

3.5 Висновки по лекції

Теорія графів– розділ математики, що вивчає властивості графів. Наочно **граф** можна представити як геометричну конфігурацію, яка складається з точок (вершини) сполучених лініями (ребрами). **Елементами теорії графів** є вершини (вузли), ребра, дуги, ланцюги, петлі.

Мережеве планування – це одна з форм графічного відображення змісту робіт і тривалості виконання стратегічних планів і довгострокових комплексів проєктних, планових, організаційних та інших видів діяльності підприємства. Мережеве планування реалізовані у вигляді **трьох систем мережевого аналізу** – CPM (Critical Path Method – метод критичного шляху), PERT (Program Evaluation and Review Technique – програмна оцінка і аналіз) і GERT (Graphic Evaluation and Review Technique – графічна оцінка і аналіз).

Критичний шлях – найдовший шлях від вхідної до завершальної події. Він характеризує мінімальну тривалість виконання всього комплексу робіт. Роботи та події, що лежать на критичному шляху, називаються критичними.Їх несвоєчасне виконання веде до зриву термінів всього комплексу робіт.

Задача про найкоротший шлях полягає у знаходженні найкоротшого шляху від заданої початкової вершини до заданої кінцевої вершини. **Алгоритм Дейкстри**знаходить найкоротшу відстань від однієї вершини графу до всіх інших. **Алгоритм Флойда-Воршелла** знаходить найкоротших відстаней між усіма вершинами зваженого орієнтованого графа.

Задача про максимальний потік полягає у знаходженні такого потоку за транспортною мережею, щоб сума потоків з джерела, або, що означає те ж саме, сума потоків до стоку була максимальна. **Розріз** визначає безліч ребер, при видаленні яких з мережі повністю припиняється потік від джерела до стоку. **Пропускна здатність розрізу**

дорівнює сумі пропускних здатностей "розрізаних" ребер. Серед всіх розрізів мережі розріз з мінімальною пропускною спроможністю визначає максимальний потік в мережі.

Тема 4 *Моделювання виробничих процесів на авіаційному транспорті методами динамічного програмування*

4.1 Мета та завдання лекції

Метою лекції є ознайомлення з роллю динамічного програмування в моделюванні виробничих процесів на авіаційному транспорті.

Завдання лекції:

- розкрити поняття динамічного програмування;
- охарактеризувати рекурентну природу розрахунків в динамічному програмуванні;
- представити розподіл задач динамічного програмування на етапи;
- ознайомити з застосуванням динамічного програмування при вирішенні задач оптимізації виробничого процесу;
- навести приклади розв'язання задачі розрахунку траєкторії літака, задачі про рюкзак (завантаження транспортного засобу).

4.2 План лекції

4.1 Поняття динамічного програмування.

4.2 Рекурентна природа розрахунків в динамічному програмуванні.

4.3 Розподіл задач динамічного програмування на етапи.

4.4 Застосування динамічного програмування при вирішенні задач оптимізації виробничого процесу на авіаційному транспорті.

4.5 Задачі динамічного програмування про оптимальну траєкторію польоту літака та завантаження повітряного судна (рюкзаку).

4.3 Основні категорії, ключові поняття та визначення теми

Алгоритм – набір інструкцій, які описують порядок дій виконавця, щоб досягти результату розв'язання задачі за скінченну кількість дій.

Варіаційна задача – пошук функцій, що оптимізують певні критерії.

Динамічне програмування – розділ математики, який присвячено теорії і методам розв'язання багатокрокових задач оптимального управління.

Змінна – математична величина, значення якої може змінюватись у межах певної задачі.

Ітерація (від лат. iteratio – повторювання) – багатозначний термін, який залежно від контенту може означати: 1) повторне застосування математичної операції (із зміненими даними) при розв'язанні обчислювальних задач, яке дає можливість поступово наблизитися до правильного результату; 2) результат багаторазового повторення якоїсь математичної операції.

Математична операція – відображення, що ставить у відповідність одному або декільком елементам множини (аргументам) інший елемент (значення).

Математична постановка задачі – точне формулювання умов і цілей рішення задачі.

Параметр – величина, що входить у математичну формулу і зберігає своє постійне значення лише за цих умов.

Принцип (лат. principium – начало, основа): 1. Основне вихідне положення якої-небудь наукової системи, теорії, ідеології; засада. 2. Особливість, покладена в основу створення або здійснення чогось.

Рекурентний – той, що дає можливість відшукувати значення якоїсь величини за знайденими раніше іншими значеннями тієї самої величини.

4.4 Текст лекції

Поняття динамічного програмування

Розвиток виробничих сил й ускладнення зв'язків у суспільстві призвели до необхідності наукового підходу до проблеми планування й управління. Виявилося, що багато завдань планування й управління можна трактувати як процес поетапного вибору (прийняття рішень).

Динамічне програмування (ДП) визначає оптимальне рішення n-мірної задачі шляхом її декомпозиції на n етапів, кожний з яких представляє собою підзадачу відносно однієї змінної. Обчислювальна перевага такого підходу полягає в тому, що ми займаємось рішенням одномірних оптимізаційних задач замість однієї великої n-мірної задачі. **Фундаментальним принципом ДП**, що складає основу декомпозиції задачі на етапи, є *оптимальність*. Так як природа кожного етапу рішення залежить від конкретної оптимізаційної задачі, ДП не пропонує обчислювальних алгоритмів безпосередньо для кожного етапу. Обчислювальні аспекти рішення оптимізаційних задач на кожному етапі проєктуються і реалізуються окремо (що не виключає застосування єдиного алгоритму для всіх етапів).

Рекурентна природа розрахунків в динамічному програмуванні

Рекурентна природа розрахунків в динамічному програмуванні полягає в тому, що оптимальне рішення однієї підзадачі використовується в якості вхідних даних для наступної. Розв'язавши останню підзадачу, ми знаходимо оптимальне рішення вихідної задачі.

Побудова моделі динамічного програмування зводиться до таких основних моментів:

1) обирають спосіб ділення процесу на кроки;

2) вводять параметри стану $q_k = (q_k^{(1)}, q_k^{(2)}, \ldots, q_k^{(s)})$та змінні управління $v_k =(v_k^{(1)}, v_k^{(2)}, \ldots, v_k^{(s)})$на кожному кроці процесу;

3) записують рівняння стану$q_k = F(q_{k-1}, v_k)$;

4) вводять показники ефективності на $к$-му кроці$f(q_{k-1}, v_k)$і сумарний показник – цільову функцію $Z = \Sigma f(q_{k-1}, v_k)$, $k = 1, \ldots, n$;

5) вводять для розгляду умовні максимуми$Z_k*(q_{k-1})$показника ефективності від $к$-го кроку (включно) до кінця процесу та умовні оптимальні управління на $к$-му кроці$v_k*(q_{k-1})$;

6) згідно з обмеженнями задачі, визначають для кожного кроку множину допустимих управлінь на цьому кроці;

7) записують основні для розрахунків функціональні рівняння Беллмана (4.1)-(4.2):

$$Z_k*(q_{k-1}) = max\{f(q_{k-1}, v_k) + Z_{k+1}*(q_k)\}; \qquad (4.1)$$

$$Z_n*(q_{n-1}) = max\{f_n(q_{n-1}, v_n)\}, q_n \epsilon D_n. \qquad (4.2)$$

Незважаючи на одноманітність у загальній побудові моделі динамічного програмування, шо наведена вище, схема розрахунків видозмінюється залежно від розмірності задачі, характеру моделі (дискретна чи неперервна), виду функцій та інших характеристик моделі.

Алгоритми прямої (від початкового етапу до останнього) та **зворотної** (від останнього етапу до початкового) **прогонки** приводять до одного і того ж рішення. В спеціальній літературі з ДП незмінно використовується алгоритм зворотної прогонки, так як він більш ефективний з обчислювальної точки зору, хоча алгоритм прямої прогонки і виглядає більш логічним.

Розподіл задач динамічного програмування на етапи

Специфіка методу ДП полягає в тому, що для відшукання оптимального керування досліджуваний процес розподіляється на етапи, причому керування оптимізується щоразу тільки на одному етапі.

Динамічне програмування (планування) – це планування далекоглядне, з урахуванням майбутнього, а не короткозоре, коли керуються принципом «аби тільки добре зараз, а там – що буде».

Загальне правило знаходження оптимального керування в багатокроковому процесі полягає в тому, що керування на кожному кроці треба вибирати з урахуванням майбутнього. Із цього правила є виключення – це останній крок, де можна діяти без оглядки на майбутнє: його на останньому кроці немає.

Динамічне програмування ґрунтується на принципі знаходження на кожному кроці умовно оптимального керування для кожного можливого результату попереднього кроку.

Керуючись цим принципом, ми розвертаємо процес знаходження оптимального керування з кінця, знаходячи спочатку умовно оптимальне керування для кожного можливого результату передостаннього кроку, а потім на основі його умовно оптимальне керування на передостанньому кроці й т.д., поки не дійдемо до першого кроку. На першому кроці нам треба робити гіпотези про стан системи – ми знаємо, із чого починається процес, і можемо з урахуванням знайдених умовно оптимальних керувань на наступних кроках знайти безумовно оптимальне керування на першому кроці, тобто таке керування, що з урахуванням всіх умовно оптимальних керувань на наступних кроках дає оптимальне керування для всього процесу.

Отже, методологія динамічного програмування складається в розподілі завдання на етапи та поетапній побудові оптимального керування на кожному кроці.

Оптимізація багатокрокового процесу здійснюється на основі **принципу оптимальності Р. Беллмана**: якім б не був початковий стан системи, наступні рішення повинні бути оптимальними тільки щодо стану, отриманого в результаті попереднього рішення. Зміст цього принципу полягає в тому, що при плануванні кожного кроку багатоетапного процесу необхідно враховувати не тільки вигоду, одержувану на даному кроці, але також загальну вигоду, одержувану по закінченні всього процесу.

Динамічне програмування –це метод оптимізації, який полягає в тому, що переміщення функції мети до оптимуму відбувається крок за кроком, причому кожний крок визначається лише поточним станом і не залежить від попередніх подій.

До особливостей задач ДП можна віднести:

1. Багатоетапність: t=1, 2, ..., T.

2. Побудова розв'язку задачі цілеспрямованим перебором у прямому та зворотному напрямках («лавиноподібна» стратегія).

3. Оптимум цільової функції на етапі t дійсний як відносно цього етапу, так і відносно наступних етапів t+1, t+2, ..., T.

4. Кожний етап – це окрема задача, яка може бути розв'язана різними методами, а забезпечення цільової функції етапу – це вектор: $Z = (z_1, z_2, \dots, z_n)$.

Переваги методу ДП:

1) ДП представляє собою, насамперед, засіб рішення задач, які можуть бути вирішені й іншими методами.

Цінність же методу полягає в іншому підході: багатокроковий процес прийняття рішень заміняється послідовністю однокрокових процесів прийняття рішення.

2) ДП дає математичний апарат для рішення задач, які раніше не вміли вирішувати. Зокрема, варіаційні задачі з обмеженнями типу нерівностей, рішення яких пов'язане зі значними труднощами, легко вирішуються методом ДП.

3) ДП має велику загальність і може застосовуватись для широкого кола задач.

Недоліки методу ДП:

1) У ДП відсутній загальний алгоритм, придатний для всіх задач. Кожна задача має свої власні труднощі й у кожному випадку треба знайти найбільш підходящу методику оптимізації.

2) Принциповим недоліком методу ДП є великий необхідний об’єм пам'яті ЕОМ при рішенні дискретних задач.

Застосування динамічного програмування при вирішенні задач оптимізації виробничого процесу на авіаційному транспорті.

До типових задач динамічного програмування відносяться задачі: розрахунку траєкторії літака (задача комівояжера), завантаження літака(задача про рюкзак), про інвестиції, про розклад літаків, прогнозування термінів ремонту будівельних конструкцій тощо.

Задача розрахунку траєкторії літака

Літак у деякій точці летить на висотіH_0 та зі швидкістюV_0. Йому необхідно піднятися на висоту H_kта набути швидкість V_kз мінімальною витратою паливаw. Дискретні витрати палива (у кг) при збільшенні висоти або швидкості наведені на рисунку лініями, а витрати w від поточної точки до кінцевої – кружечками. Перед розрахунком усі кружечки білі та пусті, а лінії – однакової структури (рис. 4.1).

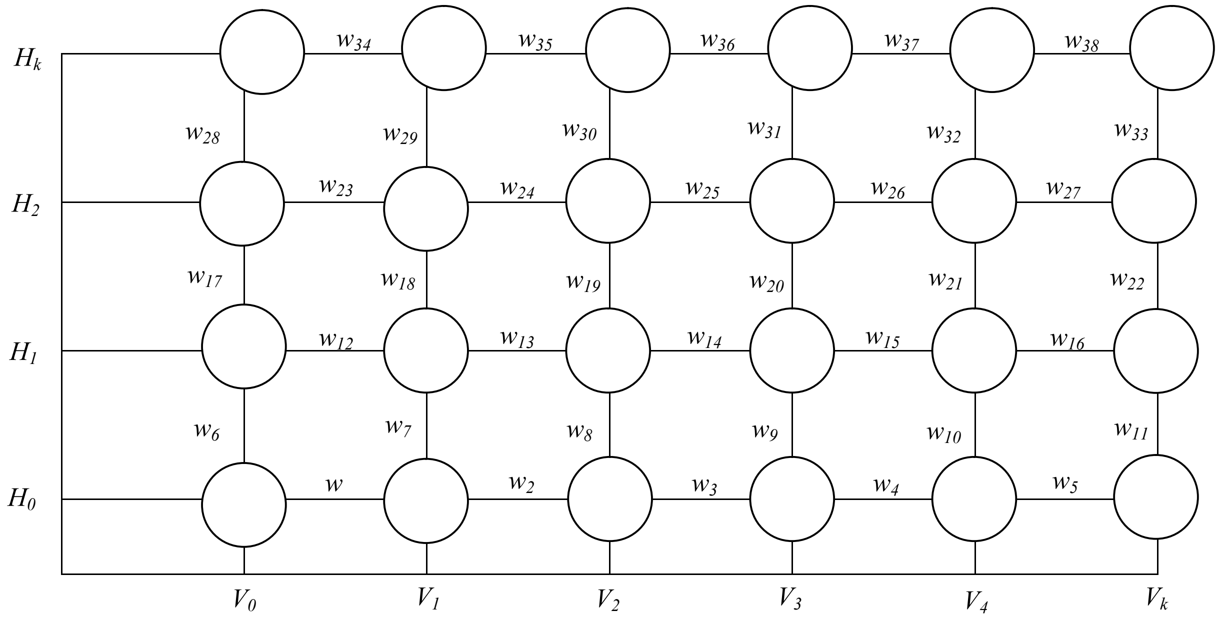


Рисунок 4.1 – Дискретні витрати палива при збільшенні висоти або швидкості, кг

Математична модель задачі про завантаження літака (4.3)-(4.4):

– цільова функція (4.3):

$$L = \sum_{i=0}^{k} \sum_{j=0}^{k} w_{ij} \to min; \qquad (4.3)$$

– обмеження (4.4):

$$w_{ij} \geq 0,\ i = \overline{0,k},\ j = \overline{0,k}. \qquad (4.4)$$

де w_{ij}– витрати палива від i-ї доj-ї точки.

Вважатимемо, що літак знаходиться в точці (H_k, V_k), де витрати пальногоw=0. Далі виконуються наступні дії:

– у точку(H_k, V_k) літак може потрапити з точки(H_k, V_4), деw= 3(0+3) або з точки(H_2, V_k), де такожw = 3(0+3). Шляхи можливого руху позначаються жирною лінією, а витрати паливаwна них – цифрами;

–у точки(H_k, V_4) та(H_2, V_k) можна потрапити лише з точки (H_2, V_4). Задача полягає в обранні найекономнішого переміщення на даному етапі. Отже, для точки(H_2, V_4) w = 7(min{3+4, 3+9});

–для точки(H_1, V_k)w=9(3+6), а для(H_1, V_4)w= 12(min{7+5, 9+8}).

Таким чином, необхідно охопити всі точки можливого переміщення літака, визначаючи для кожної з них w – оптимальну витрату палива для переміщення з даної точки до(H_k, V_k), при цьому не забуваючи позначати оптимальний на даному етапі шлях, аж доки розв'язок не дійде до точки(H_0, V_0).

Оптимальний шлях від точки (H_0, V_0) до точки (H_k, V_k) визначається шляхом переміщення між цими точками незабороненими шляхами(рис. 4.2).

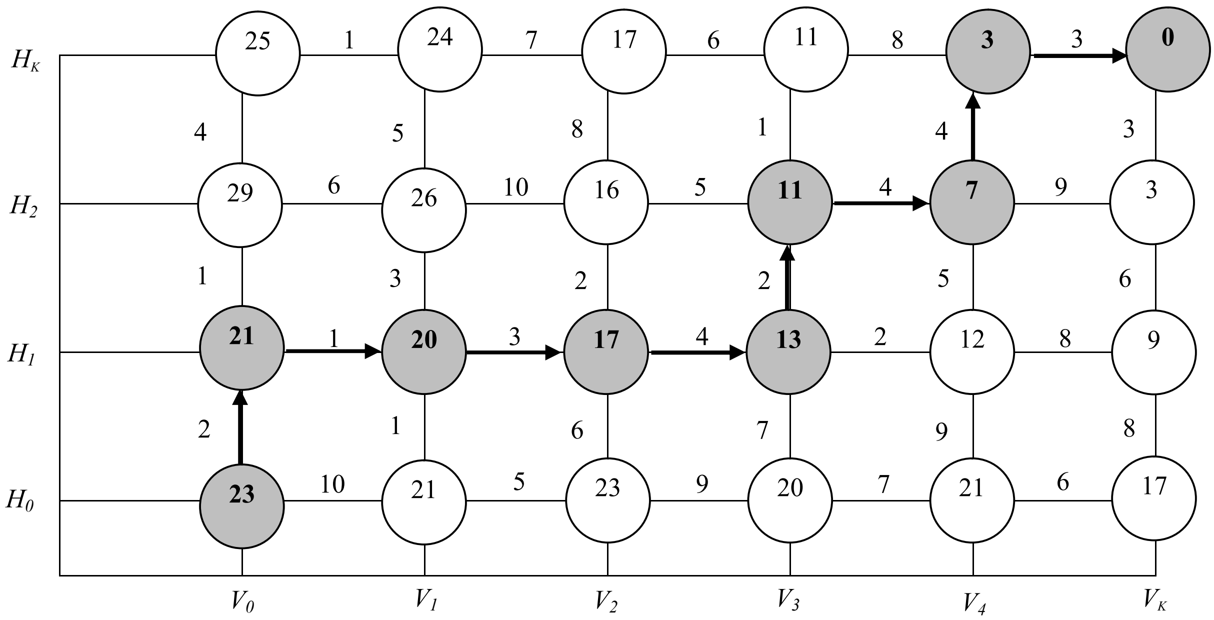


Рисунок 4.2 – Розв'язання задачі розрахунку оптимальної траєкторії польоту літака

Відповідь: (H_0, V_0) (H_1, V_0) (H_1, V_3) (H_2, V_3) (H_2, V_4) (H_k, V_4) (H_k, V_k)– оптимальний шлях з витратою палива 23 од.

Задача про рюкзак (завантаження транспортного засобу)

Необхідно завантажити транспортний засіб (літак) вантажоємністю, наприклад, G = 20 т наявними речами з вагою g_i та вартістю v_i таким чином, щоб сумарна вартість вантажу була максимальною. Дані про вагу g_i та вартість v_i речей наведені в табл. 4.1.

Таблиця 4.1 – Вихідні дані задачі про завантаження літака

Річ	Вага, т	Вартість, тис. грн.	Максимальна кількість, шт.
1	3	30	6
2	5	60	4
3	9	80	2

Математична модель задачі (4.5)-(4.6):

– цільова функція (4.5):

$$L = \sum_{j=1}^{n} v_j\, x_j \to max; \qquad (4.5)$$

– обмеження (4.6):

$$\sum_{j=1}^{n} g_j\, x_j \leq G; \qquad (4.6)$$

$$x_j \geq 0,\ j = \overline{1,n};\ v_j,\ g_j,\ x_j,\ G – \text{цілі числа},$$

де x_j– кількість завантажених речей j.

$$30x_1 + 60x_2 + 80x_3 \to \max;$$

$$3x_1 + 5x_2 + 9x_3 \leq 20.$$

Перша ітерація включає перебір всіх можливих варіантів x_1 та x_2 з метою визначення займаної ваги та вартості (табл. 4.2). Потім необхідно скласти підсумкову таблицю ітерації, впорядкувавши записи за зростанням ваги з вибором найкращих вартісних показників з першої таблиці (табл. 4.3). Ці показники знадобляться при доданні потрібної ваги з x_1 та x_2 в наступній ітерації.

Таблиця 4.2 – Перша ітерація (використання x_1, x_2)

Вага, т	Вартість, тис. грн.	x_1	x_2
0	0	0	0
3	30	1	0
6	60	2	0
9	90	3	0
12	120	4	0
15	150	5	0
18	180	6	0
5	60	0	1
8	90	1	1
11	120	2	1
14	150	3	1
17	180	4	1
20	210	5	1
10	120	0	2
13	150	1	2
16	180	2	2
19	210	3	2
15	180	0	3
18	210	1	3

Друга ітерація полягає в поступовому збільшенніx_3 від 0 до 2 та заповненні вагового резерву значеннями з підсумкової таблиці першої ітерації (адже вони є оптимальними для x_1 таx_2) (табл. 4.4). Складання підсумкової таблиці аналогічне (табл. 4.5).

Таблиця 4.3 – Підсумкова таблиця першої ітерації (використання x_1, x_2)

Вага, т	Вартість, тис. грн.	x_1	x_2
0-2	0	0	0
3-4	30	1	0
5-7	60	0	1
8-9	90	1	1
10-12	120	0	2
13-14	150	1	2
15-17	180	0	3
18-20	210	1	3

Таблиця 4.4 – Друга ітерація (використання x_1, x_2, x_3)

Вага, т	Вартість, тис. грн.	x_1	x_2	x_3
0	0	0	0	0
3	30	1	0	0
5	60	0	1	0
8	90	1	1	0
10	120	0	2	0
13	150	1	2	0
15	180	0	3	0
18	210	1	3	0
9	80	0	0	1
12	110	1	0	1
14	140	0	1	1
17	170	1	1	1
19	200	0	2	1
18	160	0	0	2

Таблиця 4.5 – Підсумкова таблиця другої ітерації (використання x_1, x_2, x_3)

Вага, т	Вартість, тис. грн.	x_1	x_2	x_3
0-2	0	0	0	0

3-4	30	1	0	0
5-7	60	0	1	0
8-9	90	1	1	0
10-12	120	0	2	0
13-14	150	1	2	0
15-17	180	0	3	0
18-20	210	1	3	0

Відповідь:x_1=1,x_2=3, x_3=0з вартістю 210 тис.грн.

4.5 Висновки по лекції.

ДП визначає оптимальне рішення *n*-мірної задачі шляхом її декомпозиції на *n* етапів, кожний з яких представляє собою підзадачу відносно однієї змінної. **Рекурентна природа** розрахунків в динамічному програмуванні полягає в тому, що оптимальне рішення однієї підзадачі використовується в якості вхідних даних для наступної. Розв'язавши останню підзадачу, ми знаходимо оптимальне рішення вихідної задачі. **Алгоритми прямої** (від початкового етапу до останнього) та **зворотної** (від останнього етапу до початкового) **прогонки** приводять до одного і того ж рішення.

До **типових задач** динамічного програмування відносяться задачі: розрахунку траєкторії літака, про рюкзак, про інвестиції, про розклад літаків, завантаження торгового судна, прогнозування термінів ремонту будівельних конструкцій тощо.

Змістовий модуль 2 *Ймовірнісні математичні моделі виробничих процесів на авіаційному транспорті*

Тема 5 *Моделювання виробничих процесів на авіаційному транспорті методом кореляційно-регресійного аналізу* (2 год.)

5.1 Мета та завдання лекції.

Метою лекції є ознайомлення з роллю кореляційно-регресійного аналізу в оптимізації виробничих процесів на авіаційному транспорті.

Завдання лекції:

- розкрити сутність кореляційно-регресійного аналізу;
- розглянути підхід до визначення виду залежності між часом та інтенсивністю виробничих процесів;
- ознайомити з методикою розрахунку коефіцієнтів регресії та визначення рівняння регресії;
- надати алгоритм обчислення коефіцієнту кореляції;
- розглянути методику прогнозування інтенсивності виробничих процесів та побудови лінії регресії.

5.2 План лекції

5.1 Сутність кореляційно-регресійного аналізу.

5.2 Визначення виду залежності між часом та інтенсивністю виробничих процесів на авіаційному транспорті (виду лінії регресії).

5.3 Методика розрахунку коефіцієнтів регресії та визначення рівняння регресії.

5.4 Обчислення коефіцієнту кореляції.

5.5 Методика прогнозування інтенсивності виробничих процесів на авіаційному транспорті, побудова лінії регресії.

5.3 Основні категорії, ключові поняття та визначення теми

Багатофакторний регресійний аналіз (на основі множинної регресії) – вид регресійного аналізу, що включає у себе розгляд двох і більше незалежних змінних величин.

Коваріація – міра спільної мінливості двох випадкових змінних.

Коефіцієнт кореляції – показник, який використовують для вимірювання щільності зв'язку між результативними і факторними ознаками у кореляційно-регресійній моделі за лінійної залежності.

Коефіцієнт регресії – це кутовий коефіцієнт у прямолінійному рівнянні кореляційного зв'язку; у лінійній функції рівняння регресії він показує на скільки одиниць в середньому зміниться результативна ознака (y) при зміні факторної ознаки (x) на одиницю свого натурального виміру.

Кореляційно-регресійний аналіз – це побудова та аналіз економіко-математичної моделі у вигляді рівняння регресії (рівняння кореляційного зв’язку), що виражає залежність результативної ознаки від однієї або кількох ознак-факторів і дає оцінку міри щільності зв’язку.

Метод найменших квадратів – один із методів регресійного аналізу, який використовується для статистичного оцінювання параметрів регресійної моделі за емпіричними даними. Згідно з цим методом параметри моделі повинні відповідати такому рівнянню регресії, що забезпечує найменше значення суми квадратів відхилень емпіричних даних від тих, що обчислені за рівнянням регресії.

Парний (простий) регресійний аналіз – вид регресійного аналізу, що включає у себе розгляд однієї незалежної змінної величини.

Регресія (англ. regression) – форма зв'язку між випадковими величинами, закон зміни математичного очікування однієї випадкової величини залежно від значень іншої.

Регресійний аналіз (англ. regression analysis) – це метод визначення відокремленого і спільного впливу факторів на результативну ознаку та кількісної оцінки цього впливу шляхом використання відповідних критеріїв.

Рівняння регресії (рівняння кореляційного зв’язку) – рівняння, що відображає зміну середньої величини однієї ознаки (y) в залежності від іншої (x).

1.4 Текст лекції

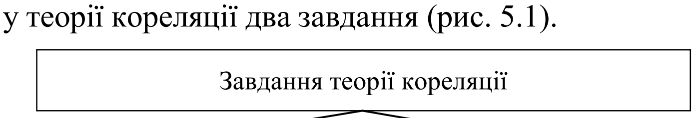
Сутність кореляційно-регресійного аналізу

Кореляційно-регресійний аналіз – це побудова та аналіз економіко-математичної моделі у вигляді рівняння регресії (рівняння кореляційного зв’язку), що виражає залежність результативної ознаки від однієї або кількох ознак-факторів і дає оцінку міри щільності зв’язку.

Основні ідеї теорії кореляції вперше висунув англійський учений Ф. Гальтон наприкінці 70-х рр. XIX ст. Досліджуючи закономірності спадковості, він виявив, що кількісні ознаки батьків у дітей пом’якшувалися, «повертали» до середніх величин за сукупністю. Такий зв’язок учений назвав «регресією». Цей термін закріпився за рівнянням, яке дає змогу за величиною однієї кореляційно пов’язаної ознаки розраховувати середні величини іншої.

Розвинув теорію кореляції учень Ф. Гальтона К. Пірсон, який використовував коефіцієнт кореляції як вимірник щільності зв’язку. Він розробив методи аналізу взаємозв’язку двох змінних, теорію часткових і чистих коефіцієнтів кореляції, теорію багатофакторної кореляції. Свій внесок у розвиток математичної статистики зробили Р. Фішер та учень Пірсона В. Госсен (псевдонім Стьюдент). Логічні й математичні питання теорії кореляції вивчав український математик-економіст Є. Слуцький.

Правильне застосування кореляційних методів дає змогу зрозуміти глибинну сутність процесів взаємозв’язків. Кореляційні зв’язки виявляються не в кожному окремому випадку, а в середньому для багатьох випадків. У цих зв’язках між причиною і наслідком немає повної відповідності, а спостерігається лише певне співвідношення. Особливості кореляційних зв’язків породжують у теорії кореляції два завдання (рис. 5.1).



Визначити теоретичну форму зв’язку (регресійний аналіз) | Виміряти щільність зв’язку (кореляційний аналіз)

Рисунок 5.1 – Завдання кореляційної теорії

Перше полягає в тому, щоб знайти форму функціонального зв’язку, яка найбільшою мірою відповідає суті кореляційної залежності. Друге – виміряти за допомогою спеціальних показників, якою мірою кореляційний зв’язок наближається до зв’язку функціонального.

Кореляційно-регресійний аналіз складається з 4 етапів (рис. 5.2).

Попередній (апріорний) аналіз → Збирання інформації та її первинна обробка → Побудова моделі (рівняння регресії) → Оцінка та аналіз моделі

Рисунок 5.2 – Етапи кореляційно-регресійного аналізу

Такий поділ досить умовний, оскільки окремі етапи тісно пов’язані між собою, а результат, отриманий на одному етапі, дає змогу скоригувати висновки попередніх етапів кореляційно-регресійного аналізу. Під попереднім аналізом розуміють весь процес дослідження явища, що розглядається, до збору вихідної інформації. На цьому етапі формуються основні напрями всього кореляційно-регресійного аналізу, формулюється завдання дослідження, обирається методика вимірювання результативного показника, тобто вимірник, який найкраще характеризує цей показник, визначається кількість факторів, що найсуттєвіше впливають на результативну ознаку. Факторні ознаки повинні відповідати таким вимогам:

- бути кількісними, найкраще – неперервними,
- розраховуватися з відношення до однієї бази,
- не дублювати одна одну, тобто не відображати ту саму сторону досліджуваного явища.

До інформаційної бази кореляційно-регресійного аналізу ставляться відповідні вимоги:

- сукупність має бути досить великою за обсягом (за кількістю одиниць або спостережень), щоб визначені у процесі кореляційно-регресійного аналізу статистичні характеристики були достатньо типовими й надійними;
- вихідні дані мають бути якісно та кількісно однорідними;
- якісна однорідність передбачає наближеність умов формування результативних і факторних ознак, кількісна – відсутність одиниць спостереження, які за своїми числовими характеристиками суттєво відрізняються від основної маси даних.

Необхідно підкреслити дві особливості, властиві кореляційному аналізу:

1) при використанні кореляційного методу вирішальне значення має всебічний, економічно усвідомлений попередній аналіз даних господарської діяльності. Слід пам'ятати, що зв'язок між ознаками і властивостями не є результатом математичних розрахунків, а лежить в природі самих економічних явищ і за допомогою методів математичної статистики можна лише виразити об'єктивно існуючі закономірності економічних процесів;

2) кореляцію можна виявити, лише досліджуючи достатньо велику сукупність спостережень, оскільки кореляційні зв'язки виявляються в формі спряженого варіювання двох або кількох зіставлених ознак.

Прикладом використання кореляційної залежності для прогнозування та прийняття управлінських рішень можуть служити криві попиту та пропозиції, на основі яких будуються моделі, які описують наслідки зміни цін.

Наприкінці XIX в. німецький статистик Е. Енгель сформулював закони і побудував криві, згідно з якими з ростом доходу частка витрат на харчування скорочується, на одяг і житло залишається незмінною, а на освіту та лікування - збільшується. Ці криві послужили вихідним пунктом побудови різних моделей, що описують поведінку покупців при зміні їхніх доходів і відповідно використовуваних при прогнозуванні попиту на товари і послуги.

Німецький дослідник Г. Госсен сформулював твердження про залежність споживчої оцінки корисності від кількості благ і дав їм математичну інтерпретацію.

Визначення виду залежності між часом та інтенсивністю виробничих процесів на авіаційному транспорті (виду лінії регресії)

Авіаційна діяльність – це складний багатогранний процес, на який впливає велика кількість різноманітних чинників. За допомогою теоретичного аналізу рівня інтенсивності виробничого процесу визначено, що його дослідження можна здійснити з допомогою кореляційно-регресійного аналізу.

Щоб оцінити рівень інтенсивності виробничих процесів, недостатньо окремих статистичних даних. Показники статистичної звітності відображають лише окремі кількісні дані. Статистичні показники як одиничні дані мають обмежену цінність, тому що для управління процесом підвищення інтенсивності виробничих процесів має значення оцінка загальних тенденцій. Для того щоб охарактеризувати процес з врахуванням взаємозв'язку, взаємозалежності та взаємозумовленості його показників, необхідно скористатися принципом інтеграції. Такий принцип вимагає розробки та використання інтегрального показника, який формується шляхом згортання багатьох одиничних показників.

Інтегральний показник інтенсивності процесу виробництва перебуває під впливом багатьох факторних ознак. До того ж слід враховувати, що кожна окрема факторна ознака не справляє вирішального впливу на кінцеву висхідну ознаку, але їх сукупний вплив є відчутним. Для встановлення зв'язку між цими факторними ознаками і вихідною ознакою використовують множинний кореляційно-регресійний аналіз. Послідовність здійснення процедури кореляційно-регресійного аналізу зв'язку між показниками зображена на рис. 5.3.

Вибираючи показники для оцінювання чинників, які впливають на інтенсивність процесу виробництва, враховується їх функціональне призначення та важливість кожного чинника, а також питома вагу у загальній оцінці.

Для визначення індивідуального внеску кожної ознаки у загальну оцінку явища необхідно встановити вид та форму функції зв'язку. Для цього можна скористатися множинною кореляцією – побудувати багатофакторну регресійну модель, яка дасть можливість виявити спільний вплив різноманітних факторів, які формують інтенсивність виробничого процесу, адже кожен такий чинник може не справляти вагомого впливу на результуючу ознаку, а також визначити найвагоміші чинники і відсіяти неістотні.

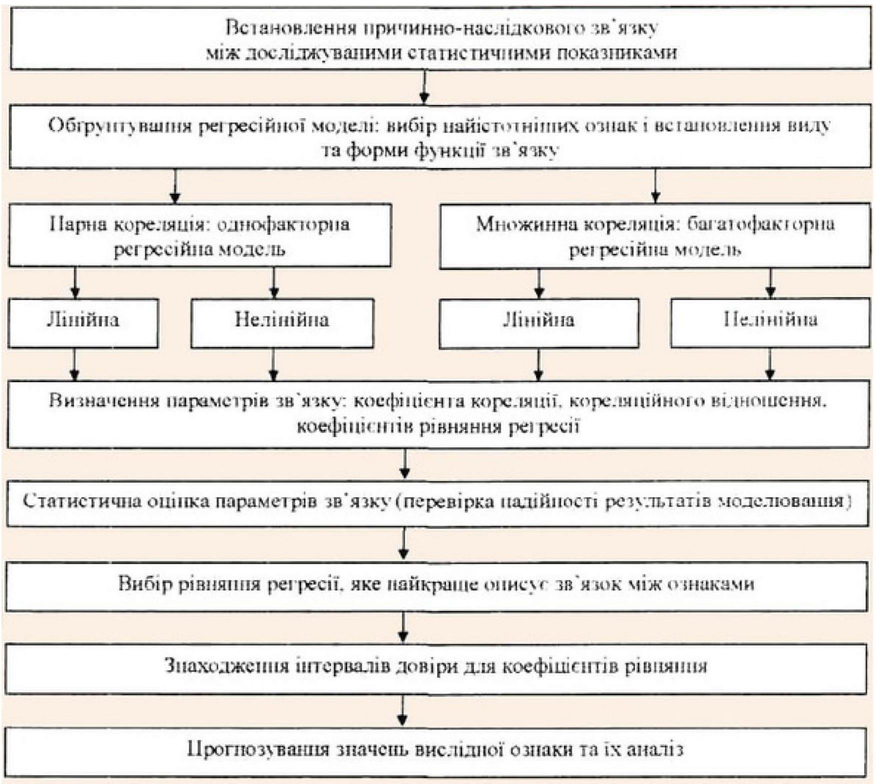


Рисунок 5.3 – Етапи побудови і реалізації моделі кореляційно-регресійного аналізу

Для отримання достовірних результатів, слід створити інформаційні умови для виявлення основних проблем і чинників, які стимулюють чи стримують інтенсивність виробничого процесу. Тобто, методика оцінювання повинна не тільки відображати існуючий стан, але й надавати інформацію для своєчасного прийняття рішень щодо покращення управління цим процесом.

Методика розрахунку коефіцієнтів регресії та визначення рівняння регресії

Регресія (англ. regression) – форма зв'язку між випадковими величинами, закон зміни математичного очікування однієї випадкової величини залежно від значень іншої. Розрізняють прямолінійну, криволінійну, ортогональну, параболічну та інші регресії, а також лінію і площину регресії.

Регресійний аналіз (англ. regression analysis) – це метод визначення відокремленого і спільного впливу факторів на результативну ознаку та кількісної оцінки цього впливу шляхом використання відповідних критеріїв.

Регресійний аналіз проводиться на основі побудованого **рівняння регресії** і визначає внесок кожної незалежної змінної у варіацію досліджуваної (прогнозованої) залежної змінної величини.

Основним завданням регресійного аналізу є визначення впливу факторів на результативний показник (в абсолютних показниках). Передусім для цього необхідно підібрати та обґрунтувати рівняння зв'язку, що відповідає характеру аналітичної стохастичної залежності між досліджуваними ознаками. Рівняння регресії показує як в середньому змінюється результативна ознака Y_x під впливом зміни факторних ознак x_i.

У загальному вигляді рівняння регресії можна представити так (5.1):

$$Y_x = f(x_1, x_2, \ldots, x_n), \qquad (5.1)$$

де Y_x – залежна змінна величина;

x – незалежні змінні величини (фактори).

Залежно від кількості змінних величин виділяють різні види регресійного аналізу. Якщо змінна величина завжди одна, то змінних може бути як одна, так і декілька. Виходячи з цього, виділяють **два види регресійного аналізу**: парний (простий) регресійний аналіз і регресійний аналіз на основі множинної регресії, або багатофакторний.

Парний регресійний аналіз – вид регресійного аналізу, що включає у себе розгляд однієї незалежної змінної величини, а **багатофакторний** – відповідно дві величини і більше.

Зважаючи на характер зв'язку, в регресійному аналізі можуть використовуватися лінійні та нелінійні функції. Для визначення характеру залежності та, відповідно, побудови рівняння регресії доцільно застосувати графічний метод, порівняння рівнобіжних рядів вихідних даних, табличний метод.

Так, графічний метод дає найбільш наочну картину розміщення крапок на графіку, завдяки чому можна виявити напрям і вид залежності між досліджуваними показниками: прямолінійна чи криволінійна.

За допомогою порівняння рівнобіжних рядів ознак можна спостерігати за рівномірністю їх взаємних змін. Якщо зміна факторної ознаки x призводить до відносно рівномірної зміни результативноїY_x, тоді використовується лінійна функція (наприклад, залежність між урожайністю культур і кількістю внесених добрив).

Найпростішим рівнянням парної регресії, що описує лінійну залежність між факторною і результативною ознаками, є рівняння прямої, яке має такий вигляд (5.2):

$$Y_x = a_0 + a_1 x, \qquad (5.2)$$

деY_x – залежна змінна, яка оцінюється або прогнозується (результативна ознака);

a_0 – вільний член рівняння;

a_1 – коефіцієнт регресії;

x – незалежна змінна (факторна ознака), яка використовується для визначення залежної змінної.

Параметри рівняння обчислюються на основі системи нормальних рівнянь **методом найменших квадратів** (5.3):

$$\Sigma y = na_0 + a_1\Sigma x; \qquad (5.3)$$
$$\Sigma xy = a_0\Sigma x + a_1\Sigma x^2.$$

Звідси коефіцієнти регресії (5.4)-(5.5):

$$a_1 = (n\Sigma xy - \Sigma x\Sigma y)/(n\Sigma x^2 - (\Sigma x)^2); \qquad (5.4)$$
$$a_0 = (\Sigma y\Sigma x^2 - \Sigma xy\Sigma x)/(n\Sigma x^2 - (\Sigma x)^2), \qquad (5.5)$$

або$a_0 = \bar{y} - a_1\bar{x}$.

Для зручності розрахунків регресійного та кореляційного аналізу доцільно використати таку форму таблиці (табл. 5.1).

Таблиця 5.1 – Вихідні та розрахункові дані для обчислення кореляційно-регресійних характеристик (парна прямолінійна кореляція)

№ з/п	Вихідні дані		Розрахункові величини			Y_x
	Факторна ознака(х)	Результативна ознака(у)	x^2	y^2	xy	
1						
2						
...						
X (в середньому)						

Основне змістовне навантаження в рівнянні регресії несе коефіцієнт регресії. Найчастіше застосовуються лінійні рівняння або приведені до лінійного вигляду. ***Коефіцієнт регресії*** – це кутовий коефіцієнт у прямолінійному рівнянні кореляційного зв'язку. У лінійній функції рівняння регресії він показує на скільки одиниць в середньому зміниться результативна ознака y при зміні факторної ознаки x на одиницю свого натурального виміру. Тобто, коефіцієнт регресії – це варіація y, яка припадає на одиницю варіації x. Коефіцієнт регресії має одиницю виміру результативної ознаки. За наявності прямого зв'язку коефіцієнт регресії є додатною величиною, а за зворотного зв'язку – від'ємною.

Параметр a_0– вільний член рівняння регресії, тобто це значення y при x=0. Цей показник має тільки розрахункове значення у випадках, коли х не має нульових значень.

У разі, коли зі зміною факторної ознаки результативна змінюється нерівномірно, використовуються нелінійні функції. Так, якщо зміна факторного показника сприяла прискореній динаміці результативного показника (наприклад, вплив обсягу грошової маси на рівень інфляції), доцільно використати степеневу функцію (5.6):

$$Y_x = ax^b, \qquad (5.6)$$

У випадку, коли під впливом факторної ознаки результативна змінюється нерівномірно, причому з уповільненням, використовується рівняння гіперболи (5.7):

$$Y_x = a + b / x, \qquad (5.7)$$

Прикладом такої залежності є залежність рівня продуктивності праці робітників від рівня їх заробітної плати.

Якщо зміна факторної ознаки супроводжується нерівномірною варіацією результативної ознаки із зміною напряму зв'язку, нелінійна регресія описується рівнянням параболи (5.8):

$$Y_x = a + bx + cx^2, \qquad (5.8)$$

Регресійні моделі передбачають значення змінної Y на основі заданих значень змінних X. Процедура підбору параметрів моделі з використанням передбачення на основі вибірки даних в межах діапазону її значень відома як ***інтерполяція***. Передбачення за межами діапазону значень даних відомо як ***екстраполяція***. Виконання екстраполяції тісно залежить від регресійних припущень. Чим далі від даних поширюється екстраполяція, тим більшим буде відхилення моделі від реальних значень.

Обчислення коефіцієнту кореляції

Коефіцієнт кореляції – показник, який використовують для вимірювання щільності зв'язку між результативними і факторними ознаками у кореляційно-регресійній моделі при лінійній залежності. За абсолютною величиною коефіцієнт кореляції коливається в межах від -1 до +1. Чим ближчий цей показник до 0, тим слабший зв'язок, чим ближчий він до ±1 – тим зв'язок тісніший. Знак «плюс» при коефіцієнті кореляції означає прямий зв'язок між ознаками x і y, знак «мінус» – обернений.

Коефіцієнт кореляції Пірсона між двома змінними дорівнює коваріації двох змінних, або сумі добутків відхилень, поділеній на добуток їх стандартних відхилень. Нехай, є дві вибірки $x^n=(x_1, \ldots, x_n)$та $y^n=(y_1, \ldots, y_n)$. Коефіцієнт кореляції Пірсона розраховують за формулою (5.9):

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} = \frac{cov(x, y)}{\sqrt{s_x^2 s_y^2}}, \qquad (5.9)$$

де $\overline{x}, \overline{y}$ – вибіркові середні x^nтаy^n;

s_x^2, s_y^2 – вибіркові дисперсії;

$r_{xy} \in [-1, 1]$.

Різні автори пропонують різні підходи до інтерпретації значення коефіцієнта кореляції. В той же час, всі критерії є певною мірою умовними, і не повинні трактуватися надто прискіпливо. Інтерпретація кореляції залежить від контексту та мети. Наприклад, показник кореляції 0.9 може бути дуже низьким у випадку дослідження законів фізики з використанням високоякісного обладнання, проте може трактуватися як дуже високий в гуманітарних науках, де існує вплив багатьох інших факторів (табл. 5.2).

Таблиця 5.2 – Значущість кореляції

Кореляція	Негативна	Позитивна
Відсутня	−0.09 до 0.0	0.0 до 0.09
Низька	−0.3 до −0.1	0.1 до 0.3
Середня	−0.5 до −0.3	0.3 до 0.5
Висока	−1.0 до −0.5	0.5 до 1.0

Метод найменших квадратів (***МНК***) є одним із методів регресійного аналізу, який використовується для статистичного оцінювання параметрів регресійної моделі за емпіричними даними. Згідно з цим методом параметри моделі повинні відповідати такому рівнянню регресії, що забезпечує найменше значення суми квадратів відхилень емпіричних даних від тих, що обчислені за рівнянням регресії. Так, з двох різних наближень тієї ж самої емпіричної функції, що задана у вигляді таблиці, кращим вважається те, для якого сума квадратів відхилення має найменше значення. Для ілюстрації суті методу найменших квадратів можна розглянути рис. 5.4.

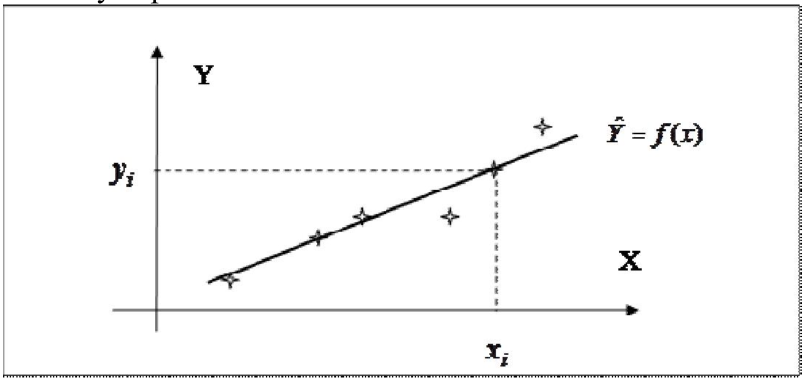


Рисунок 5.4 – Ілюстрація методу найменших квадратів

Графік функції проходитиме таким чином, щоб різниця між значеннями функції $\hat{Y} = f(x)$та ординатами емпіричних точок була б якомога менше.

Основи метода найменших квадратів були розроблені Карлом Фрідріхом Гауссом у зв’язку з задачами, що вирішуються теорією помилок, тобто математичною теорією, яка досліджує точність результатів вимірювань. Гаусс настільки ретельно провів дослідження нормального розподілу випадкових похибок, що їх крива щільності ймовірностей має назву гауссіани.

Узагальнення умов застосування методу найменших квадратів сформульовано у теоремі Гаусса – Маркова.

Методика прогнозування інтенсивності виробничих процесів на авіаційному транспорті, побудова лінії регресії

Приклад. Авіакомпанія повинна прийняти рішення щодо скорочення або розширення наявного парку повітряних суден. З цією метою були зібрані статистичні дані про попит на послуги з повітряних перевезень, які надавалися авіакомпанією впродовж 6 місяців поточного року (табл. 5.3). Необхідно за результатами статистичних даних спрогнозувати показники попиту на авіаційні послуги до кінця року за допомогою методу кореляційно-регресивного аналізу.

Таблиця 5.3 – Статистичні дані про кількість перевезених пасажирів за 6 місяців поточного року

t, міс.	1	2	3	4	5	6
Кількість пасажирів, $K*10^3$, чол.	1,2	1,1	1,5	1,3	1,1	1,0

Алгоритм розв’язання задачі

1 Для візуального визначення виду лінії регресії в **кореляційному полі** наносимо точки, що відповідають вихідним даним, одержуємо частково-ламану криву і спостерігаємо, що отримані точки можна апроксимувати прямою лінією (рис. 5.5). Вісь абсцис x відповідає часу t, ордината y – кількості обслугованих пасажирів (клієнтів)$K*10^3$. Таким чином, для опису отриманих точок можна використовувати лінійну регресію виду $\boldsymbol{y = a_0 + a_1x}$, де коефіцієнти регресії a_0 і a_1 знаходяться за допомогою **методу найменших квадратів,** який дозволяє оцінити теоретичну залежність між змінними.

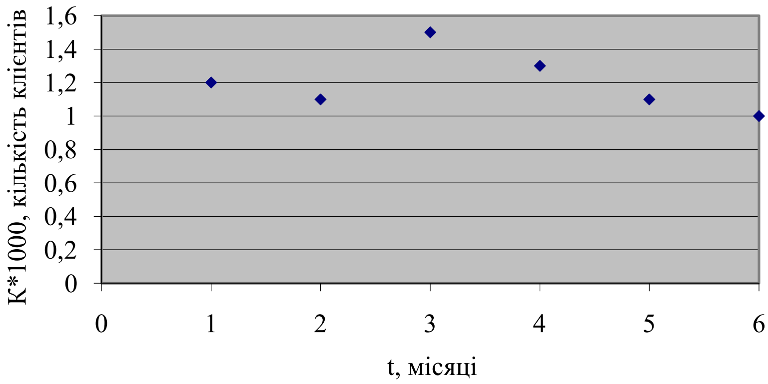


Рисунок 5.5 – Залежність зміни кількості клієнтів від часу

2 Для визначення коефіцієнтів регресії a_0 і a_1 дані обчислень заносимо в табл. 5.4.

Таблиця 5.4 – Методика обчислення коефіцієнтів регресії

	x, t	y, $K*10^3$	x^2	y^2	xy	$x+y$	$(x+y)^2$
	1	1,2	1	1,44	1,2	2,2	4,84
	2	1,1	4	1,21	2,2	3,1	9,61
	3	1,5	9	2,25	4,5	4,5	20,25
	4	1,3	16	1,69	5,2	5,3	28,09
	5	1,1	25	1,21	5,5	6,1	37,21
	6	1	36	1,0	6,0	7,0	49,0
Σ	21	7,2	91	8,8	24,6	28,2	149

Виконаємо перевірку за формулою (5.10):

$$\Sigma(x+y)^2 = \Sigma x^2+2\Sigma xy+\Sigma y^2. \qquad (5.10)$$

149 = 91+2·24,6+8,8.

Таким чином, табличні розрахунки зроблені вірно.

3 Обчислимо коефіцієнти регресії за формулами (5.11)-(5.12):

$$a_0= (\Sigma y\Sigma x^2 -\Sigma xy\Sigma x)/(n\Sigma x^2 - (\Sigma x)^2); \qquad (5.11)$$

$$a_1=(n\Sigma xy - \Sigma x\Sigma y)/(n\Sigma x^2 - (\Sigma x)^2), \qquad (5.12)$$

де n – кількість місяців.

a_0=(7,2*91 – 24,6*21)/(6*91-21^2)=(655,2-516,6)/(546-441)=138,6/105=1,32;

a_1=(6*24,6-21*7,2)/(6*91-21^2)=(147,6-151,2)/(546-441)=-3,6/105=-0,03.

4 Рівняння регресії, що визначає апроксимуючу лінійну функцію для даних задачі, визначається як y = 1,32 - 0,03x.

Аналіз рівняння показує, що кожний місяць кількість обслугованих клієнтів зменшується на 0,03*1000=30 чоловік.

5 Використовуючи дані табл. 5.3, обчислимо коефіцієнт кореляції -1 ≤ r ≤ 1 за формулою (5.13):

$$r = \frac{\Sigma xy-(\frac{1}{n})(\Sigma x)(\Sigma y)}{\sqrt{\left[x^2-(\frac{1}{n})(\Sigma x)^2\right]\left[y^2-(\frac{1}{n})(\Sigma y)^2\right]}} \qquad (5.13)$$

$$r = \frac{24{,}6-\left(\frac{1}{6}\right)\cdot 21\cdot 7{,}2}{\sqrt{(91-(\frac{1}{6})\cdot 21^2)(8{,}8-(\frac{1}{6})\cdot 7{,}2^2)}} = \frac{-0{,}6}{\sqrt{17{,}5\cdot 0{,}16}} = \frac{-0{,}6}{1{,}67} = -0{,}36.$$

Коефіцієнт кореляції r = -0,36; r< 0.

Значення коефіцієнта кореляції показує, що змінні x та y мають зворотній (так як значення r від'ємне) слабкий (так як значення r ближче до 0, чим до -1) зв'язок. Тобто, з часом кількість обслугованих клієнтів зменшується.

6 На основі отриманого рівняння регресії ***у*=1,32 - 0,03*х*** в табл. 5.5 оцінимо кількість обслугованих клієнтів за 1-12 місяці поточного року і за цими даними побудуємо лінію регресії (рис. 5.6).

Таблиця 5.5 – Розрахунок кількості клієнтів

х, міс.	*у*=1,32 - 0,03*х*, кількість клієнтів, $K*10^3$
1	1,29
2	1,26
3	1,23
4	1,20
5	1,17
6	1,14
7	1,11
8	1,08
9	1,05
10	1,02
11	0,99
12	0,96

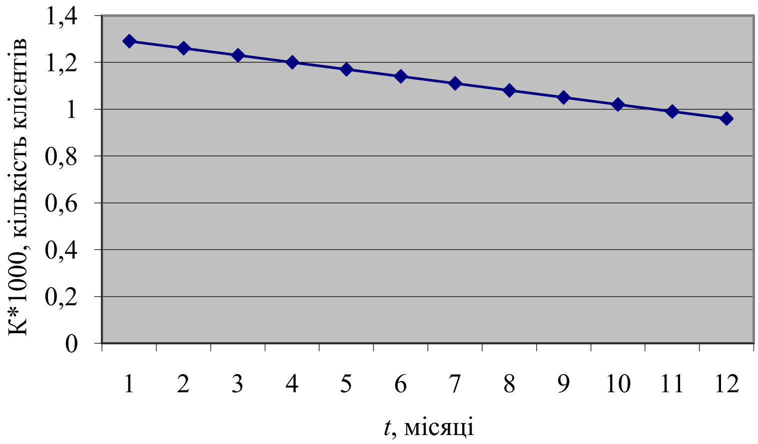


Рисунок 5.6 – Лінія регресії

7 Робимо висновок. Значення коефіцієнта кореляції -0,36 показує, що зв'язок між часом і кількістю обслугованих пасажирів (клієнтів) зворотній (з часом кількість клієнтів зменшується), слабкий (плин часу несуттєво впливає на зміну кількості клієнтів). Відповідно до отриманого рівняння регресії *у*=1,32 – 0,03*х* прогнозується зменшення кількості обслугованих клієнтів близько 30 чоловік кожного місяця, що свідчить про недоцільність розширення виробничих потужностей підприємства ближчим часом.

5.5 Висновки по лекції

Кореляційно-регресійний аналіз – це побудова та аналіз економіко-математичної моделі у вигляді рівняння регресії (рівняння кореляційного зв’язку), що виражає залежність результативної ознаки від однієї або кількох ознак-факторів і дає оцінку міри щільності зв’язку.

Виділяють **два види регресійного аналізу**: парний (простий) регресійний аналіз і регресійний аналіз на основі множинної регресії, або багатофакторний. Зважаючи на характер зв'язку, в регресійному аналізі можуть використовуватися лінійні та нелінійні функції.

Для вимірювання щільності зв'язку між результативними і факторними ознаками у кореляційно-регресійній моделі за лінійної залежності використовують **коефіцієнт кореляції.**

Одним із методів регресійного аналізу є **метод найменших квадратів**, який використовується для статистичного оцінювання параметрів регресійної моделі за емпіричними даними. Згідно з цим методом параметри моделі повинні відповідати такому рівнянню регресії, що забезпечує найменше значення суми квадратів відхилень емпіричних даних від тих, що обчислені за рівнянням регресії.

Тема 6*Моделювання управління запасами на авіаційному транспорті* (2 год.)

6.1 Мета та завдання лекції.

Метою лекції є ознайомлення з роллю управління запасами на авіаційному транспорті.

Завдання лекції:

- охарактеризувати поняття та класифікацію запасів;
- представити узагальнену модель управління запасами;
- розкрити типи моделей управління запасами;
- ознайомити з рішенням класичної задачі економічного розміру заказу;
- представити графічну інтерпретацію моделі управління запасами.

6.2 План лекції

6.1 Поняття та класифікація запасів.

6.2 Узагальнена модель управління запасами.

6.3 Типи моделей управління запасами.

6.4 Рішення класичної задачі економічного розміру заказу для забезпечення запасами авіаційного транспорту.

6.5 Графічна інтерпретація моделі управління запасами.

6.3 Основні категорії, ключові поняття та визначення теми

Витрати – зменшення обсягу матеріальних цінностей, коштів тощо, які відбуваються в процесі свідомої людської діяльності.

Дефіцит – недостатність засобів, ресурсів порівняно із заздалегідь запланованим або необхідним рівнем.

Замовлення – доручення виготовити, виконати, підготувати або доставити що-небудь у певний, наперед визначений строк.

Запас – будь-який ресурс, що зберігається для задоволення майбутніх потреб.

Матеріально-технічне постачання- це процес постачання на склади підприємства чи відразу на робочі місця необхідних відповідно до планів виробництва матеріально-технічних ресурсів.

Методика – сукупність взаємозв'язаних способів та прийомів доцільного проведення будь-якої роботи.

Попит – кількість продукції, яку споживачі бажають і спроможні купити на ринку за певну ціну за певний проміжок часу.

Склад – це обладнане місце, приміщення або споруда, інфраструктура, різноманітне обладнання та внутрішня транспортна система, яка застосовується для прийому, розміщення та зберігання матеріальних цінностей, підготовки їх до споживання та видачі споживачу.

Точка поновлення – рівень запасу, при якому робиться нове замовлення.

Управління запасами – складний комплекс заходів, спрямований на забезпечення максимально високого рівня обслуговування покупців при мінімізації поточних витрат, пов'язаних із утримуванням запасів.

6.4 Текст лекції

Поняття та класифікація запасів

Як у бізнесі, так і у виробництві прийнято підтримувати розумний запас матеріальних ресурсів або комплектуючих для забезпечення безперервності виробничого процесу. Традиційно запас розглядається як неминучі витрати, коли дуже низький рівень запасу призводить до високовартісних зупинок виробництва, а дуже високий – до «омертвіння» капіталу. Задача управління запасами визначає рівень запасу, який урівноважує два згадані крайні випадки.

Виникнення теорії управління запасами можна зв'язати з роботами Ф. Еджуорта й Ф. Харріса, що з'явилися наприкінці XIX – початку XX століття, у яких досліджувалася проста оптимізаційна модель для визначення економічного розміру партії поставки для складської системи з постійною рівномірною витратою й періодичним надходженням збереженого продукту.

Запасом називається будь-який ресурс, що зберігається для задоволення майбутніх потреб. Прикладами запасів можуть стати напівфабрикати, готові вироби, матеріали, різні товари, а також готівка, що перебуває в сховищі.

Існують причини, що спонукують підприємства створювати запаси:

1) дискретність поставок при безперервному споживанні;

2) упущений прибуток у випадку відсутності запасу;

3) випадкові коливання: попиту за період між поставками; обсягу поставок; тривалості інтервалу між поставками; передбачувані зміни кон'юнктури: сезонність попиту; сезонність виробництва.

Існують також причини, що спонукують підприємства прагнути до мінімізації запасів на складах: плата за зберігання запасу; фізичні втрати при зберіганні; моральне зношування продукту.

Існує проблема **класифікації** наявних запасів. Для рішення цього завдання використовується методика адміністративного спостереження. Ціль її полягає у визначенні тієї частини запасів підприємства, що вимагає найбільшої уваги з боку відділу постачання. Для цього кожний компонент запасів розглядається за двома параметрами:

1)частка в загальній кількості запасів підприємства;

2)частка в загальній вартості запасів.

Методика 20/80.Відповідно до цієї методики компоненти запасу, що становлять 20% його загальної кількості та 80% його загальної вартості, повинні відслідковуватися відділом постачання більш уважно.

Методика *ABC*-аналізу. В основі класифікації лежить принцип Парето. У рамках цієї методики запаси, наявні в розпорядженні підприємства, розділяються на три групи: *A, B* и *C*. Група *A*: 10% загальної кількості запасів та 65% їхньої вартості; *B*: 25% загальної кількості запасів та 25% їхньої вартості; *C*: 65% загальної кількості запасів та близько 10% їхньої вартості.

Саме *найменша по обсягу та найцінніша* частина запасів може стати предметом особливого контролю та математичного моделювання.

Необхідно відзначити, що класифікація запасів може бути заснована не тільки на показниках долі в загальній вартості та у загальній кількості. Деякі види запасів можуть бути віднесені до більше високого класу на підставі таких характеристик, як специфіка поставок, якість і т.д. Перевага методики розподілу запасів на класи полягає в тому, що для кожного з них можна вибрати свій порядок контролю й управління.

Відзначимо деякі моменти політики управління запасами, класифікація яких проведена на основі ABC-аналізу.

1.Запаси групи А вимагають більше уважного та частого проведення інвентаризації; правильність обліку запасів цієї групи повинна підтверджуватися частіше.

2.Планування і прогнозування запасів групи А повинне характеризуватися більшим ступенем точності, ніж груп В і С.

3.Для групи А потрібно намагатися створити страховий запас, щоб уникнути більших витрат, пов'язаних з відсутністю запасів цієї групи.

4.Методи та прийоми управління запасами, розглянуті далі, повинні застосовуватися насамперед до груп А і В. Що стосується запасів групи 3, звичайно момент поновлення замовлення по них визначають виходячи з конкретних умов, а не на основі кількісного методу, щоб звести до мінімуму витрати на їхній контроль.

Узагальнена модель управління запасами

Будь-яка модель повинна відповідати на два питання:

1. Яку кількість продукції замовити? – Розмір замовлення.
2. Коли замовити? – Залежить від типу системи управління.

Якщо система передбачає періодичний контроль – стан запасів контролюється через рівні проміжки часу (щодобово, щомісячно).

Якщо система передбачає неперервний контроль стану запасу, то точка замовлення визначається рівнем запасу, при якому необхідно розміщувати нове замовлення.

Таким чином, розв’язок узагальненої задачі управління визначається так:

1. У випадку періодичного контролю стану запасу слід забезпечувати постачання нової кількості ресурсів в розмірі обсягу замовлення через рівні проміжки часу.

2. У випадку неперервного контролю стану запасу необхідно розміщувати нове замовлення у розмірі обсягу замовлення, коли його рівень досягає точки замовлення.

Розмір і точка замовлення визначаються з умов мінімізації сумарних витрат системи управління запасами, що можна визначити у вигляді функції цих двох змінних (6.1):

$$A = B + C + D + E, \tag{6.1}$$

де A - сумарні (сукупні) витрати системи управління запасами;
B - витрати на придбання;
C - витрати на оформлення замовлення;
D - витрати на зберігання замовлення;
E - втрати від дефіциту.

Витрати на придбання визначаються вартістю одиниці продукції, що закупається (запасу, який зберігається). Ця вартість може бути постійною або залежати від розміру замовлення, що виражається у вигляді оптових знижок (ціна одиниці продукції знижується із зростанням розміру замовлення).

Витрати на оформлення замовлення – постійні витрати, що пов’язані з його розміщенням (для виготовлення продукції) на інших виробництвах. Не залежать від обсягу замовлення. Таким чином, при задоволенні попиту на протязі заданого проміжку часу шляхом розміщення більш дрібних замовлень (частіше) витрати зростають у порівнянні з тим, коли попит задовольняється шляхом розміщення великих замовлень.

Витрати на зберігання замовлення представляють собою витрати на зберігання запасу на складі (відсоток на інвестований капітал, витрати на зберігання, утримання і нагляд), вони збільшуються із збільшенням обсягу запасу.

Втрати від дефіциту – втрати, пов’язані із відсутністю запасу необхідної продукції. Включають як потенційні втрати прибутку, так і втрату довіри клієнтів.

Рис. 6.1 ілюструє залежність чотирьох компонентів витрат узагальненої моделі управління запасами від рівня запасу.

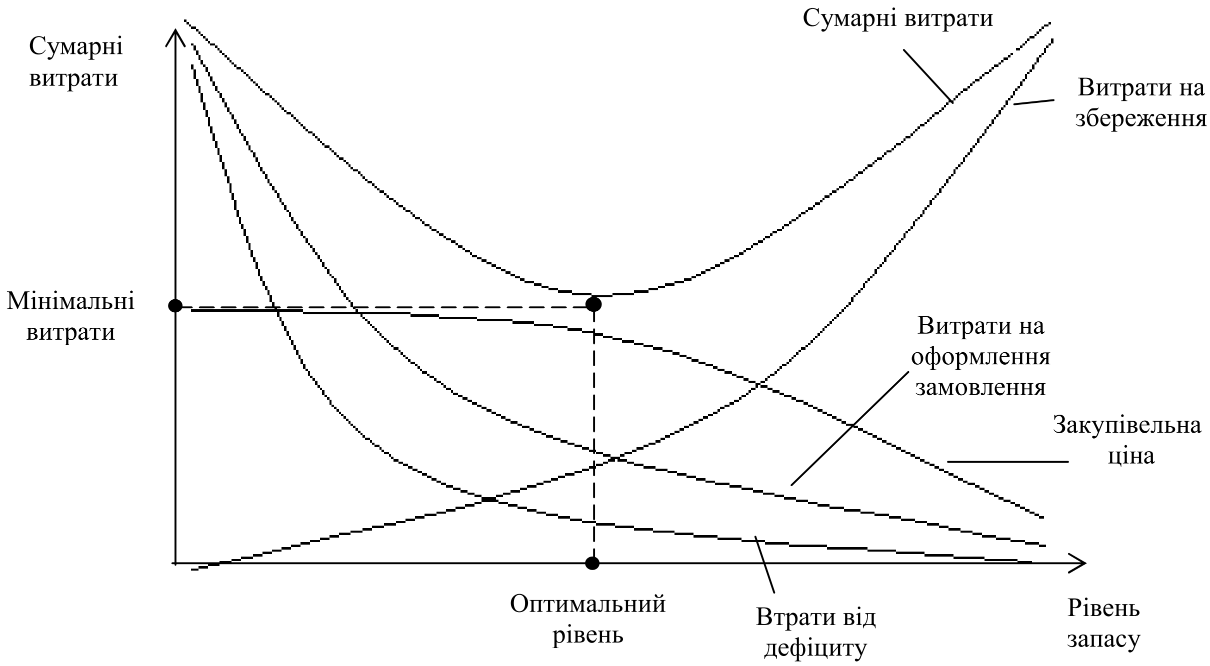


Рисунок 6.1 – Узагальнена модель управління запасами

Оптимальний рівень запасу відповідає мінімуму сумарних витрат.

Зауважимо, що модель управління запасами не обов’язково повинна містити усі чотири види витрат, оскільки деякі з них можуть бути незначними. На практиці деякі з них можна не враховувати при умові, що вони не становлять суттєвої частини загальних витрат.

У керуванні матеріальними запасами існує ряд протиріч.

Критерієм оптимізації матеріальних запасів є мінімізація усіх витрат, зв'язаних з величиною запасів, що залежить від процесу матеріально-технічного постачання.

Основні умови, яким повинні задовольняти системи управління матеріальними запасами:

1) обсяг запасів повинний забезпечувати безперервність виробничого процесу,

2) розмір запасів повинний бути мінімальним з метою скорочення витрат на збереження матеріального запасу, на будівництво складських приміщень та іммобілізацію матеріальних ресурсів.

Типи моделей управління запасами

Велика різноманітність моделей управління запасами пояснюється характером попиту.

Детермінований попит (достовірно відомий):

- статичний – інтенсивність попиту відома і незмінна в часі (зустрічається рідко);
- динамічний – інтенсивність попиту відома, але змінюється в часі.

Ймовірнісний попит (описаний ймовірнісним розподілом):

- стаціонарний – функція щільності ймовірності попиту незмінна в часі;
- нестаціонарний – функція щільності ймовірності попиту змінюється в часі, найбільш точно описує характер попиту, але математично складна.

Характер попиту – один з основних факторів при побудові моделі, але є ще й інші фактори, що впливають на вибір моделі:

<u>Запізнення поставок або терміни виконання замовлень</u>. Після розміщення замовлення воно може бути виконане терміново або через деякий час. Інтервал часу між моментом розміщення замовлення і його постачанням називається <u>запізненням постачання</u>. Ця величина може бути детермінованою або випадковою.

<u>Поповнення запасу</u>. Хоча система управління запасами може функціонувати при запізненні постачання, процес поповнення запасу може здійснюватися миттєво або рівномірно в часі. <u>Миттєве поповнення</u> запасу відбувається при умові, коли замовлення надходять від зовнішнього джерела. <u>Рівномірне поповнення</u> – коли продукція, що запасається, виробляється самою організацією.

У загальному випадку система може функціонувати при додатному запізненні постачання та рівномірному поповненні запасу.

<u>Період часу</u> визначає інтервал, на протязі якого здійснюється регулювання рівня запасу. Він буває кінцевий та нескінченний в залежності від проміжку часу, коли можна надійно прогнозувати.

<u>Число пунктів накопичення запасу</u>. В системи управління запасами може включатися декілька пунктів зберігання запасу. У деяких випадках ці пункти організовані таким чином, що один виступає як постачальник для іншого – <u>система управління запасами із розгалуженою структурою</u>.

<u>Число видів продукції (однопродуктова або багатопродуктова модель)</u>. В системі управління запасами може бути більше одного виду продукції. Цей фактор враховується, коли є декілька залежностей між видами продукції. Наприклад, для деяких із них може використовуватись одне складське приміщення.

Рішення класичної задачі економічного розміру заказу для забезпечення запасами авіаційного транспорту

Модель управління запасами найпростішого типу – **однопродуктова статична модель**, яка характеризується постійним у часі попитом, миттєвим поповненням запасу і відсутністю дефіциту.

Економічний об’єм замовлення витратних матеріалів складає (6.1):

$$y = \sqrt{\frac{2DK}{h}}, \qquad (6.1)$$

де D – інтенсивність попиту;

K – витрати на оформлення, пов’язані з розміщенням замовлення;

h – витрати на зберігання одиниці витратних матеріалів.

Довжина циклу замовлення витратних матеріалів складає (6.2):

$$t_0 = \frac{y}{D}. \qquad (6.2)$$

Якщо термін виконання замовлення (час із моменту замовлення до моменту його виконання) L перевищує тривалість циклу замовленняt_0, то необхідно визначити ефективний термін виконання замовлення L_e. Число цілих циклів, що міститься в L, дорівнює (6.3):

$$n = (\text{найбільше ціле} \le L / t_0). \qquad (6.3)$$

Ефективний термін виконання замовлення розраховується за формулою (6.4):

$$L_e = L - nt_0. \qquad (6.4)$$

Точка поновлення (рівень запасу, при якому робиться нове замовлення) замовлення має місце при рівні запасу, який визначається за формулою (6.5):

$$T_з = L_e D. \qquad (6.5)$$

Щоденні витрати, пов’язані з утриманням запасу у відповідності з оптимальною стратегією, визначаються за формулою (6.6):

$$U = \frac{K}{t_0} + h\left(\frac{y}{2}\right). \qquad (6.6)$$

Графічна інтерпретація моделі управління запасами

Графічна інтерпретація моделі управління запасами витратних матеріалів наводиться на рис. 6.2.

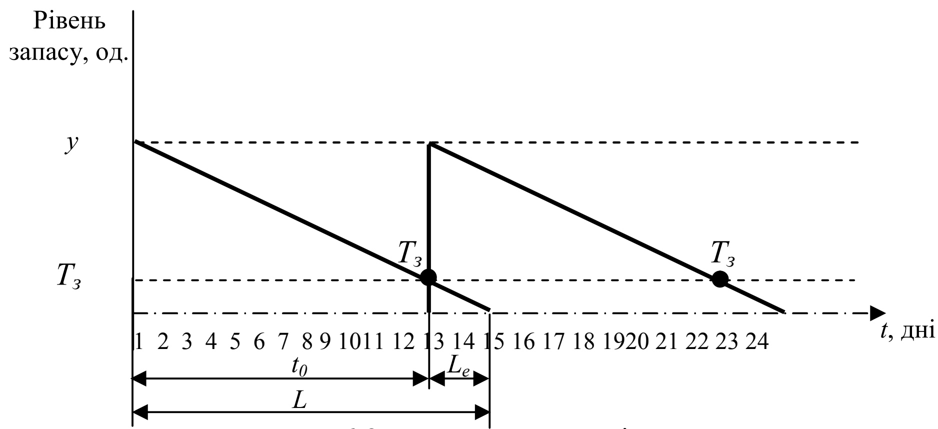


Рисунок 6.2 – Модель управління запасами

Оптимальну стратегію замовлення витратних матеріалів можна сформулювати наступним чином: замовляти y одиниць кожний цикл замовлення витратних матеріалівt_0, коли їх кількість на складі знижується до $T_з$ одиниць. Витрати, пов'язані з оптимальним управлінням запасом, складають U у.о./день.

6.5 Висновки по лекції.

Запасом називається будь-який ресурс, що зберігається для задоволення майбутніх потреб. Існує проблема класифікації наявних запасів. Для рішення цього завдання використовуються **методики 20/80** та **ABC**.

Оптимальний рівень запасу відповідає мінімуму сумарних витрат: на придбання, на оформлення замовлення, на зберігання замовлення, від дефіциту запасів.

Велика різноманітність моделей управління запасами пояснюється **характером попиту**. Модель управління запасами найпростішого типу – **однопродуктова статична модель**, яка характеризується постійним у часі попитом, миттєвим поповненням запасу і відсутністю дефіциту.

Рішення **класичної задачі економічного розміру заказу** дозволяє знаходити економічний об'єм замовлення, довжину циклу замовлення, ефективний термін виконання замовлення, точку поновлення замовлення, а також щоденні витрати, пов'язані з утриманням запасу.

Тема 7 *Моделювання виробничих процесів на авіаційному транспорті методами теорії масового обслуговування*(2 год.)

7.1 Мета та завдання лекції

Метою лекції є ознайомлення з роллю теорії масового обслуговування в оптимізації виробничих процесів на авіаційному транспорті.

Завдання лекції:

- ознайомити з поняттям випадкових процесів та їх класифікацією;
- розкрити предмет теорії масового обслуговування та області її застосування при вирішенні задач оптимізації виробничих процесів;
- розглянути основні компоненти систем масового обслуговування;
- надати класифікацію систем масового обслуговування;
- розглянути підходи до моделювання функціонування систем масового обслуговування, розкрити сутність процесів розмноження і загибелі.

7.2 План лекції

7.1 Поняття випадкових процесів. Класифікація випадкових процесів.

7.2 Предмет теорії масового обслуговування та області її застосування при вирішенні задач оптимізації виробничих процесів на авіаційному транспорті.

7.3 Основні компоненти теорії масового обслуговування.

7.4 Класифікація систем масового обслуговування.

7.5 Моделювання функціонування систем масового обслуговування. Процеси розмноження і загибелі.

7.3 Основні категорії, ключові поняття та визначення теми

Випадковий (стохастичний) процес – випадкова функція аргументу часу t.

Ефективність обслуговуючої системи – характеристика рівня виконання цією системою функцій, для яких вона призначена.

Інтенсивність потоку – середнє число подій, які появляються за одиницю часу.

Найпростіший (пуасонівський) потік – потік подій, який має наступні властивості: стаціонарність, відсутність наслідків, ординарність.

Нестаціонарний випадковий процес – випадковий процес, у якому всі його характеристики змінюються з плином часу.

Переріз випадкової функції – значення випадкової функції при фіксованому значенні її аргументу (це деяка випадкова величина).

Потік подій – послідовність подій, які наступають у випадкові моменти часу.

Реалізація (траєкторія) випадкової функції – невипадкові значення випадкової функції, які вона приймає при проведенні конкретного випробування.

Система масового обслуговування – це система, яка обслуговує вимоги, що надходять до неї (заявки). Основними елементами (компонентами) системи є: вхідний потік вимог; канали обслуговування; черга вимог; вихідний потік вимог.

Стаціонарний випадковий процес – випадковий процес, характеристики якого не залежать від вибору початку відліку, тобто є однорідними щодо часу.

7.4 Текст лекції

Поняття випадкових процесів. Класифікація випадкових процесів

Поняття випадкового процесу введено в XX столітті і пов'язано з іменами А.М. Колмогорова (1903-1987), О.Я. Хінчіна (1894-1959), Є.Є. Слуцького (1880-1948), Н. Вінера (1894-1965).

Це поняття в наші дні є одним з центральних не тільки в теорії ймовірностей, але також в природознавстві, інженерній справі, економіці, організації виробництва, теорії зв'язку. Теорія випадкових процесів належить до категорії математичних дисциплін, які найбільш швидко розвиваються. Безсумнівно, що ця обставина значною мірою визначається її глибокими зв'язками з практикою. XX століття не могло задовольнитися тою ідейною спадщиною, яку було отримано від минулого. Дійсно, в той час, як фізика, біолога, інженера цікавив процес, тобто зміна досліджуваного явища в часі, теорія ймовірностей пропонувала їм як математичного апарату лише засоби, які вивчали стаціонарні стани.

Для дослідження зміни в часі теорія ймовірностей кінця XIX - початку XX століття не мала ні розроблених приватних схем, ні тим більш загальних прийомів. А необхідність їх створення буквально стукала у вікна та двері математичної науки. Вивчення броунівського руху в фізиці підвело математику до порога створення теорії випадкових процесів.

Необхідно згадати ще про дві важливі групи досліджень, розпочатих в різний час і з різних приводів.

По-перше, це роботи А.А. Маркова (1856-1922) з вивчення ланцюгових залежностей. По-друге, роботи Є.Є. Слуцького (1880-1948) з теорії випадкових функцій.

Обидва ці напрями грали дуже істотну роль у формуванні загальної теорії випадкових процесів.

Для цієї мети вже був накопичений значний вихідний матеріал, і необхідність побудови теорії носилися в повітрі.

Залишалося здійснити глибокий аналіз наявних робіт, висловлених в них ідей і результатів і на його базі здійснити необхідний синтез.

Нехай G_t – деяка множина дійсних чисел. Якщо кожному значенню $t\epsilon G_t$ поставлена у відповідність випадкова величина $X(t)$, то кажуть, що на множині G_t задана випадкова функція $X(t)$. Множину G_t при цьому називають областю визначення випадкової функції.

Наприклад, якщо U – випадкова величина, то функція $X(t)=t^2U$ буде випадковою. При t=2 одержимо випадкову величину X_1=4U, при t=1,5 – випадкову величину X_2=2,25Uі т.д.

Значення випадкової функції при фіксованому значенні аргументу називається її **перерізом** (це деяка випадкова величина). При проведенні конкретного випробування випадкова функція буде приймати значення функції невипадкової, яку називають її **реалізацією або траєкторією**. Реалізації випадкової функції $X(t)$ позначають $x_1(t), x_2(t), \dots$, де індекс вказує номер випробування.

Наприклад, якщо $X(t) = U\sin t$, деU– випадкова величина, яка в першому випробуванні прийняла можливе значення $u_1=3$, а в другому випробуванні $u_2 = 4{,}6$, то реалізаціями $X(t)$ будуть відповідно невипадкові функції $x_1(t) = 3\sin t$, $x_2(t) = 4{,}6\sin t$.

Випадковим (стохастичним) процесом називають випадкову функцію аргументу t, який відіграє роль часу.

Наприклад, якщо літак повинен летіти із заданою сталою швидкістю, то в дійсності внаслідок дії випадкових факторів (коливання температури, зміни сили вітру та ін.), врахувати вплив яких наперед неможливо, швидкість буде змінюватись. В цьому прикладі швидкість літака є випадковою функцією від неперервно змінного аргументу – часу, тобто швидкість є випадковим процесом.

Якщо аргумент випадкової функції змінюється дискретно, то відповідні йому значення випадкової функції (випадкові величини) утворюють випадкову послідовність.

Випадкові процеси мають характеристики, аналогічні характеристикам випадкових величин, але вони є не деякими числами, а невипадковими функціями.

Математичним очікуванням випадкового процесу $X(t)$ називають невипадкову функцію $m_x(t)$, значення якої при кожному фіксованому значенні аргументу t дорівнює математичному очікуванню випадкової величини відповідного перерізу (7.1):

$$m_x(t) = M(X(t)). \qquad (7.1)$$

Дисперсією випадкового процесу $X(t)$називають невипадкову функцію $D_x(t)$, значення якої при кожному фіксованому значенні аргументу t дорівнює дисперсії випадкової величини відповідного перерізу (7.2):

$$D_x(t) = D(X(t)). \qquad (7.2)$$

Середнім квадратичним відхиленням випадкового процесу $X(t)$ називають квадратний корінь із його дисперсії (7.3):

$$\sigma_x(t) = \sqrt{D_x(t)}. \qquad (7.3)$$

Властивості математичного очікування:

1) математичне очікування невипадкової функції $\varphi(t)$ дорівнює самій невипадковій функції:

$M(\varphi(t))=\varphi(t)$;

2) невипадковий множник $\varphi(t)$можна виносити за знак математичного очікування:

$M(\varphi(t)\cdot X(t))=\varphi(t) \cdot M(X(t)) =\varphi(t)m_x(t)$;

3) математичне очікування суми двох випадкових процесів дорівнює сумі математичних очікувань доданків:

$M(X(t) + Y(t)) = m_x(t) + m_y(t)$.

Властивості дисперсії:

1) дисперсія невипадкової функції $\varphi(t)$дорівнює нулю:

$D(\varphi(t)) = 0$;

2) за знак дисперсії можна виносити квадрат невипадкового множника $\varphi(t)$:

$D(\varphi(t) \cdot X(t)) =\varphi^2(t) \cdot D_x(t)$;

3) дисперсія суми випадкового процесу і невипадкової функції дорівнює дисперсії випадкового процесу:

$D(X(t) + \varphi(t)) =D_x(t)$.

Математичне очікування, дисперсія і середнє квадратичне відхилення характеризують випадковий процес далеко не повністю. Знаючи тільки ці три характеристики, зокрема, нічого не можна сказати про степінь залежності двох перерізів. Для оцінки цієї залежності вводять нову характеристику – кореляційну функцію.

Кореляційною функцією випадкового процесу $X(t)$ називають функцію $K_x(t_1, t_2)$ двох незалежних аргументів t_1 і t_2, значення якої при кожній парі фіксованих значень аргументів дорівнює кореляційному моменту відповідних їм перерізів (7.4):

$$K_x(t_1, t_2)= M((X(t_1) - m_x(t_1))((X(t_2) - m_x(t_2)). \qquad (7.4)$$

При рівних між собою значеннях аргументів $t_1=t_2 =t$ кореляційна функція випадкового процесу дорівнює його дисперсії:

$K_x(t, t)= M((X(t) - m_x(t))^2) = D_x(t)$.

Властивості кореляційної функції:

1) кореляційна функція симетрична відносно своїх аргументів:

$K_y(t_1, t_2)= K_x(t_2, t_1)$;

2) якщо $\varphi(t)$ – невипадкова функція, а процес$Y(t) = X(t) +\varphi(t)$, то $m_y(t) = m_x(t) + \varphi(t)$, $K_y(t_1, t_2)= K_x(t_1, t_2)$;

3) якщо $\varphi(t)$ – невипадкова функція, а процес $Y(t) = X(t)\varphi(t)$, то $m_y(t) = m_x(t)\varphi(t)$, $K_y(t_1, t_2)= \varphi(t_1)\varphi(t_2)K_x(t_1, t_2)$.

Випадкові процеси (ВП) класифікують за такими ознаками:

1) за залежністю характеристик ВП від початку відліку t: стаціонарні; нестаціонарні;

2) за типом аргументу t і реалізацій: дискретні ВП з дискретним часом; неперервні ВП з дискретним часом; дискретні ВП з неперервним часом; неперервні ВП з неперервним часом;

3) за складністю математичного апарату: загального вигляду ВП; напівмарковські ВП; марковські ВП.

Для **нестаціонарного** ВП усі його характеристики змінюються з плином часу. ВП, характеристики якого не залежать від вибору початку відліку, тобто є однорідними щодо часу, називається **стаціонарним**. Математичне очікування і дисперсія стаціонарного ВП сталі, тобто $m_x(t) = m_x = \text{const}$, $D_x(t) = D_x = \text{const}$, а кореляційна функція $K_x(t, \theta)$залежить тільки від величини різниці між аргументами t іθ: $K_x(t, \theta) = K(\theta - t) = K_x(\tau)$.

Для будь-якого ВП функція $K_x(t, \theta)$симетрична, тобто $K_x(t, \theta) = K_x(\theta, t)$. Тоді для стаціонарного процесу одержимо: $K_x(\tau) = K_x(-\tau)$.

Припустимо, що у випадкові моменти часу відбувається деяка подія. Нас цікавить число появ цієї події в проміжках часу від 0 доt. Позначимо це число через $\xi(t)$. *Відносно процесу появи події припустимо*, що він: 1) стаціонарний; 2) без наслідків; 3) ординарний. В ці припущення вкладається наступний зміст:

1) стаціонарність означає, що ймовірність появи k подій в будь-який проміжок часу залежить тільки від числа k і від тривалості t проміжку часу і не залежить від початку його відліку, тобто ймовірність появи k подій за проміжок часу тривалістю t є функцією, залежною тільки від k іt;

2) відсутність наслідків означає, що ймовірність появи k подій в будь-який проміжок часу не залежить від того, появились чи не появились події в моменти часу, які передували цьому проміжку, тобто передісторія потоку не впливає на ймовірності появи подій в близькому майбутньому;

3) ординарність полягає в тому, що поява двох і більше подій за малий проміжок часу практично неможлива, тобто ймовірність появи більше однієї події за малий проміжок часу значно менша ймовірності появи тільки однієї події.

Послідовність подій, які наступають у випадкові моменти часу, називають **потоком подій**. Потік подій, який має властивості 1) – 3), називають **найпростішим або пуасонівським**.

Інтенсивністю потоку λ називають середнє число подій, які появляються за одиницю часу. Якщо стала інтенсивності потоку λ відома, то ймовірність появи k подій простішого потоку за час t визначається за формулою Пуассона (7.5):

$$P_t(k) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}. \qquad (7.5)$$

Ланцюгом Маркова називається послідовність випробувань, в кожному з яких може відбутися одна і тільки одна із k несумісних подій A_4, A_2,..., A_k повної групи, причому умовна ймовірність $p_{ij}(s)$того, що в s-му випробуванні наступить подія $A_j (j = \overline{1, k})$ за умови, що в (s-1)-му випробуванні появилась подія $A_i (i = \overline{1, k})$, не залежить від результатів попередніх випробувань.

Наприклад, якщо послідовність випробувань утворює ланцюг Маркова, а повна група складається із чотирьох несумісних подій A_4, A_2,A_3, A_4, причому відомо, що в 6-му випробуванні появилась подія A_2, то умовна ймовірність того, що в 7-му випробуванні появиться подія A_4 (тобто ймовірність $p_{24}(7)$), не залежить від того, які події появились в 1-, 2-, …, 5-му випробуваннях.

Часто в теорії ланцюгів Маркова дотримуються іншої термінології і говорять про деяку фізичну системуS, яка в кожний момент часу перебуває в одному із k станів: S_1, S_2, ..., S_k і в окремі моменти часу змінює свій стан, тобто переходить із одного стану, наприкладi, в інший, наприкладj. Зокрема, після випробування система може залишитись в тому ж стані (“перейти” із стану i в станj). Таким чином, в новій термінології події, які утворюють повну групу, називають станами системи, а випробування – змінами її станів. Якщо зміна станів відбувається в певні фіксовані моменти часу, то кажуть, що розглядається ланцюг Маркова з дискретним часом (**дискретний марковський процес** (МП)), якщо перехід із стану в стан можливий в будь-який випадковий момент часу, то кажуть, що розглядається ланцюг Маркова з неперервним часом (неперервний МП). За введеною термінологією ланцюгом Маркова називають послідовність випробувань, в кожному з яких система приймає один із k станів повної групи, причому умовна ймовірність $p_{ij}(s)$ того, що в s-му випробуванні система буде знаходитись в стані j за умови, що в (s-1)-му випробувані вона знаходилась в стані i не залежить від результатів решти, раніше проведених випробувань.

Ланцюг Маркова називається **однорідним**, якщо умовна ймовірність $p_{ij}(s)$не залежить відs, в цьому випадку пишуть $p_{ij}(s)= p_{ij}$.

Предмет теорії масового обслуговування та області її застосування при вирішенні задач оптимізації виробничих процесів на авіаційному транспорті

Теорія масового обслуговування вийшла з теорії ймовірностей. Начало розробки практичних задач масового обслуговування поклав співробітник Копенгагенської телефонної компанії датський математик А.К. Ерланг (1878-1929 р.) у період між 1908-1922 рр. В 1909 р. з'явилася його робота "Теорія ймовірностей і телефонні переговори" й інші публікації, у яких були сформульовані перші прикладні задачі телефонії. Ці задачі були пов'язані з необхідністю впорядкувати роботу телефонної мережі й розробити методи оцінки якості обслуговування споживачів залежно від числа використовуваних пристроїв.

Узагальнення методів рішення різноманітних задач і розробка загальної теорії масового обслуговування пов'язана з ім'ям радянського математика О.Я. Хінчіна. У його книзі "Математичні методи теорії масового обслуговування" вперше були сформульовані загальні ідеї й методи теорії. Подальший розвиток теорії масового обслуговування пов'язане з ім'ям радянського математика Б.В. Гнеденко і його учнів А.М. Колмогорова, Н.П. Бусленко й ін. Із закордонних авторів відомі Д. Кендалл, Ф. Паллачек, Л. Токач й ін.

Загальною особливістю задач із застосуванням теорії масового обслуговування є випадковий характер досліджуваних процесів. Однієї з типових життєвих ситуацій варто вважати утворення черг при задоволенні яких-небудь потреб, що приводить до втрат робочого часу й непродуктивній витраті різних ресурсів.

У багатьох галузях людської діяльності мають місце процеси, що носять характер масового обслуговування:

- *побутове обслуговування* – обслуговування продавцями покупців у магазинах, ремонт різних побутових предметів у майстерень, розмови по телефоні, надання медичної допомоги, бібліотечне обслуговування, готельне обслуговування, пожежне обслуговування й т.п.;

- *у військовій справі* – обстріл літаків, катерів й інших видів техніки супротивника, так само як і бомбування з літака;

- *у виробництві* – транспортне й ремонтне обслуговування, організація постачання.

Типовим прикладом задачі масового обслуговування в авіації є обслуговування одним авіаційним техніком групи літаків. Якщо за техніком закріплено недостатньо літаків, то в моменти їхньої справності він простоює, якщо багато – він не може їх вчасно обслужити. Аналогічна ситуація виникає, якщо декілька (n) літаків обслуговується декількома (r)техніками.

Основи знань про черги, іноді називані **теорією черг або теорією масового обслуговування**, становлять важливу частину теорії управління виробництвом. Черги – звичайне явище. Вони можуть носити форму очікування ремонту літака в інжиніринговому центрі або очікування студентами консультації у професора. У табл. 7.1 перераховані деякі приклади виникнення черг у системах масового обслуговування.

Таблиця 7.1 – Приклади виникнення черг у системах масового обслуговування

Ситуація	Очікують у черзі	Процес обслуговування
Супермаркет	Покупці	Прийом касиром платні за покупки
Приймальня лікаря	Пацієнти	Прийом лікарем
Комп’ютер	Комп’ютерні програми	Виконання процесором

Телефонна компанія	Абоненти	Виконання заказів на міжміські переговори
Авіаційна технічна база	Літаки	Передпольотне обслуговування
Аеродром	Літаки	Дозвіл на зліт, посадку

Моделі черг (як і лінійне програмування, моделі управління запасами, методи мережевого аналізу проєктів) використовуються й у сфері управління матеріальним виробництвом, і в сфері обслуговування. Аналіз черг у термінах довжини черги, середнього часу очікування, середнього часу обслуговування й інших факторів допомагає нам краще зрозуміти принципи організації системи обслуговування. Очікування пацієнта в приймальні лікаря й очікування лагодження зламаного дриля в ремонтній майстерні мають багато загального з погляду управління процесом обслуговування. Обидва процеси використовують людські ресурси й ресурси обладнання для задоволення потреб клієнтів.

Фахівець, ухвалюючи рішення щодо удосконалюванні системи масового обслуговування, оцінює зміни, що виникають у витратах на функціонування системи й у витратах, пов'язаних з очікуванням клієнтів. Можна найняти велику кількість співробітників, які будуть швидко обслуговувати клієнтів. Так, адміністратор супермаркету може зменшити черги в каси, збільшуючи у години пік кількість продавців і касирів. Для роботи в касах банків або аеропортів у години пік можуть бути притягнуті додаткові співробітники. Однак зниження часу очікування звичайно пов’язане з витратами на створення й оснащення робочих місць, з оплатою праці додаткового персоналу. Ці витрати можуть бути досить значні.

Можна заощадити на трудовитратах. Але тоді клієнт може не дочекатися обслуговування або втратити бажання повернутися ще раз. В останньому випадку система масового обслуговування буде зазнавати втрат, які можна назвати витратами очікування. У деяких системах обслуговування, наприклад у швидкій допомозі, витрати, пов'язані із тривалим очікуванням, можуть виявитися надзвичайно високими. Основний економічний принцип удосконалювання систем масового обслуговування складається в оцінці загальних очікуваних витрат, що включають витрати на обслуговування й втрати, які несе система в результаті очікування клієнта.

Теорія масового обслуговування вивчає закономірності протікання процесів, пов'язаних з масовим обслуговуванням, розробкою кількісних методів пошуку таких об'єктивних характеристик, які забезпечують своєчасне задоволення вимог на обслуговування.

У вітчизняній і закордонній літературі теорію масового обслуговування називають по-різному: теорією лінії очікування, теорією черг, теорією групоутворення, проблемами скупченості й ін. Ця наукова дисципліна займається описом, аналізом і дослідженням різних за своїм змістом явищ із метою виявлення й створення необхідних передумов для їхнього якісного функціонування. При цьому під якістю обслуговування розуміється не те, як добре виконана робота, а наскільки вона вчасно виконана, чи не утвориться черга на обслуговування вимог або чи не відбувається втрата вимог на обслуговування через одночасну зайнятість обслуговуючого персоналу.

У самому загальному виді черга на обслуговування може виникати з наступних причин:

- недостатня кількість або недостатня продуктивність обслуговуючих апаратів (обслуговуючого персоналу);
- нерегулярне надходження вимог;
- зміна (варіювання) тривалості обслуговування.

При організації виробництв, коли вимоги на обслуговування надходять рівномірно, через рівні проміжки часу, і коли вони вчасно обслуговуються, ніякої задачі масового обслуговування не виникає.

Основні компоненти теорії масового обслуговування

Система масового обслуговування (СМО) – це система, яка обслуговує вимоги, що надходять до неї (заявки). Основними елементами (компонентами) системи є:

- вхідний потік вимог;
- канали обслуговування;
- черга вимог;
- вихідний потік вимог.

Вимоги (заявки) на обслуговування надходять через дискретні (постійні або випадкові) інтервали часу.

Важливо знати закон розподілу вхідного потоку.

Канали (прилади) необхідні для обслуговування цих заявок. Обслуговування триває деякий час, постійний або випадковий.

Випадковий характер потоку заявок та часу обслуговування призводить до того, що в деякі моменти часу на вході СМО може виникнути черга, в інші моменти – канали можуть бути недозавантаженими або взагалі простоювати. Якщо у момент надходження заявки всі прилади зайняті, заявка копіюється у комірку буфера і чекає там початку обслуговування. Заявки, що знаходяться в буфері, складають чергу на обслуговування.

Якщо всі комірки буфера зайняті, заявка отримує відмову в обслуговуванні і втрачається.

Процес роботи СМО представляє собою випадковий процес з дискретними станами та неперервним часом. Стан СМО змінюється стрибком в моменти реалізації подій (надходження нової або закінчення обслуговування вимоги, моменту, коли вимога, виходить з черги).

З вимог, які вже обслуговані, формується *вихідний потік*.

Кожна СМО, залежно від кількості каналів, їх продуктивності, характеру потоку заявок, має деяку *пропускну здатність*, яка дозволяє їй більш чи менш успішно справлятися з потоком вимог.

Задача теорії масового обслуговування полягає в побудові моделей, які пов‘язують задані умови роботи СМО з показниками ефективності системи, що описують її спроможність впоратися з потоком вимог. Під *ефективністю* обслуговуючої системи розуміють характеристику рівня виконання цією системою функцій, для яких вона призначена.

У системах масового обслуговування розрізняють три основних етапи, які проходить кожна заявка:

1) поява заявки на вході в систему;
2) проходження черги;
3) процес обслуговування, після якого заявка залишає систему.

На кожному етапі використовуються певні характеристики, які варто обговорити перш, ніж будувати математичні моделі.

Характеристики входу:

1) число заявок на вході (розмір популяції);
2) режим надходження заявок у систему обслуговування;
3) поводження клієнтів.

Число заявок на вході. Число потенційно можливих заявок (розмір популяції) може вважатися або нескінченним (необмежена популяція), або кінцевим (обмежена популяція). Якщо число заявок, що надійшли на вхід системи з моменту початку процесу обслуговування добудь-якого заданого моменту часу, є лише малою частиною потенційно можливого числа клієнтів, популяція на вході розглядається як *необмежена.* Приклади необмежених популяцій: автомобілі, що проходять через пропускні пункти на швидкісних дорогах, покупці в супермаркеті й т.п. У більшості моделей черг на вході розглядаються саме необмежені популяції.

Якщо кількість заявок, які можуть надійти в систему, порівняно із числом заявок, що вже перебувають у системі масового обслуговування, популяція вважається *обмеженою.* Приклад обмеженої популяції: комп'ютери, що належать конкретної організації й надходять на обслуговування в ремонтну майстерню.

Режим надходження заявок у систему обслуговування. Заявки можуть надходити в систему обслуговування відповідно до певного графіка (наприклад, один пацієнт на прийом до стоматолога кожні 15 хв., один автомобіль на конвеєрі кожні 20 хв.) або випадковим образом. Появи клієнтів вважаються випадковими,якщо вони незалежні друг від друга й точно непередбачені. Часто в задачах масового обслуговування число появ в одиницю часу може бути оцінене за допомогою пуассонівського розподілу ймовірностей. При заданому темпі надходження (наприклад, два клієнти в годину або чотири вантажівки у хвилину) дискретний розподіл Пуассона описується наступною формулою:

$$p(x) = \frac{e^{-\lambda} \cdot \lambda^{x}}{x!} \quad \text{для } x = 0,\ 1,\ \ldots,$$

де $p\,(x)$ – ймовірність надходження x заявок в одиницю часу;

x – число заявок в одиницю часу;

λ – середнє число заявок в одиницю часу (темп надходження заявок);

e = 2,7182 – основа натурального логарифма.

Відповідні значення ймовірностей $p(x)$ неважко визначити за допомогою таблиці пуассонівського розподілу. Якщо, наприклад, середній темп надходження заявок - два клієнти в годину, то ймовірність того, що протягом години в систему не надійде ні однієї заявки, дорівнює 0,135, ймовірність появи однієї заявки - близько 0,27, двох - також близько 0,27, три заявки можуть з'явитися з ймовірністю 0,18, чотири - з ймовірністю близько 0,09 і т.д. Ймовірність того, що за годину в систему надійдуть 9 заявок або більше, близька нулю.

На практиці ймовірності появи заявок, зрозуміло, не завжди підкоряються пуассонівському розподілу (вони можуть мати якийсь інший розподіл). Тому потрібно проводити попередні дослідження для того, щоб перевірити, що пуассонівський розподіл може служити гарною апроксимацією.

Поводження клієнтів. Більшість моделей черг ґрунтується на припущенні, що поводження клієнтів є стандартним, тобто кожна вступник у систему заявка встає в чергу, чекає обслуговування й не залишає систему доти, поки її не обслужать. Інакше кажучи, клієнт (людина або машина), що встала в чергу, чекає доти, поки він не буде обслужений, не залишає черга й не переходить із однієї черги в іншу.

Життя значно складніше. На практиці клієнти можуть покинути чергу тому, що вона виявилася занадто довгою. Може виникнути й інша ситуація: клієнти чекають своєї черги, але з якихось причин ідуть необслугованими. Ці випадки також є предметом теорії масового обслуговування, однак тут не розглядаються.

Характеристики черги:

1) Довжина черги. Довжина може бути обмежена або не обмежена. Довжина черги (черга) *обмежена,* якщо вона за якимись причинами (наприклад, через фізичні обмеження) не може збільшуватися нескінченно. Якщо черга досягає свого максимального розміру, то наступна заявка в систему не допускається й відбувається відмова. Довжина черги *не обмежена,* якщо в черзі може перебувати будь-яке число заявок. Наприклад, черга автомобілів на бензозаправці.

2) Правило обслуговування. Більшість реальних систем використовує правило «першим прийшов – першим пішов» *(FIFO* – first in, first out). У деяких випадках, наприклад у прийомному спокої лікарні, на додаток до цього правила можуть установлюватися різні *пріоритети.* Пацієнт із інфарктом у критичному стані, очевидно, буде мати пріоритет в обслуговуванні в порівнянні з пацієнтом, що зламав палець. Порядок запуску комп'ютерних програм – інший приклад установлення пріоритетів в обслуговуванні.

Характеристики процесу обслуговування:

1) Конфігурація системи обслуговування (число каналів і число фаз обслуговування). Системи обслуговування різняться по числу каналів обслуговування. Звичайна кількість каналів можна визначити як число клієнтів, обслуговування яких може бути почате одночасно, наприклад: число майстрів у перукарні. Приклади одноканальної системи обслуговування: банк, у якому відкрите єдине віконце для обслуговування клієнтів, або ресторан, що обслуговує клієнтів в автомобілях. Якщо ж у банку відкрито кілька віконець для обслуговування, клієнт очікує в загальній черзі й підходить до першого вікна, що звільнилося, то ми маємо справу із багатоканальною однофазовою системою обслуговування. Більшість банків, також, як поштові відділення й авіапарки, є багатоканальними системами обслуговування.

2) Режим обслуговування. Як і режим надходження заявок, режим обслуговування може характеризуватися або постійним, або випадковим часом обслуговування. При *постійному* часі на обслуговування будь-якого клієнта затрачається однаковий час. Така ситуація може спостерігатися на автоматичній мийці автомобілів. Однак більш часто зустрічаються ситуації, коли час обслуговування має *випадковий* розподіл. У багатьох випадках можна припустити, що час обслуговування підкоряється *експоненційному розподілу* з функцією розподілу $F(\tau) = p(t < \tau) = 1 - e - \tau\mu$, де $p(t < \tau)$ – ймовірність того, що фактичний час t обслуговування заявки не перевищить заданої величини τ - середнє число заявок, що обслуговуються в одиницю часу; e = 2,7182 - підстава натурального логарифма.

Параметри моделей черг. При аналізі систем масового обслуговування використовуються технічні й економічні характеристики.

Найбільше часто використовуються наступні ***технічні характеристики:***

1) середній час, що клієнт проводить у черзі;

2) середня довжина черги;

3) середній час, що клієнт проводить у системі обслуговування (час очікування плюс час обслуговування);

4) середнє число клієнтів у системі обслуговування;

5) ймовірність того, що система обслуговування виявиться незайнятою;

6) ймовірність певного числа клієнтів у системі.

Серед ***економічних характеристик*** найбільший інтерес представляють наступні:

1) витрати очікування в черзі;

2) витрати очікування в системі;

3) витрати обслуговування.

Класифікація систем масового обслуговування

Системи масового обслуговування класифікують за різними ознаками.

Перша класифікація за наявністю черг:

- системи ***з відмовами*** (без черги) - заявка, яка надійшла в момент, коли всі канали зайняті, отримує відмову, покидає СМО і надалі не обслуговується;

- системи ***з очікуванням*** (з чергою) - заявка, що прийшла в момент, коли всі канали зайняті, не відкидається, а стає в чергу і чекає можливості бути обслугованою. В свою чергу класифікують:

а) по ***довжині черги*** - з обмеженою довжиною черги, які допускають чергу, але з обмеженим числом місць в ній;

b) за ***часом очікування*** - з обмеженим часом очікування, що допускають чергу, але з обмеженим терміном перебування кожної вимоги в ній;

c) по ***дисципліні обслуговування*** – з обслуговуванням по пріоритету, що допускають чергу, але деякі заявки обслуговуються поза чергою (тобто по пріоритету):

FIFO (*First Input - First Output*): першим прийшов - першим обслужений;

LIFO (*Last Input - First Output*): останнім прийшов - першим обслужений;

FIRO (F*irst Input - Random Output*): першим прийшов - обслужений у випадковому порядку;

обслуговування з пріоритетами.

Друга класифікація за числом каналів обслуговування:

-***одноканальні;***

-***багатоканальні.***

Третя класифікація за місцем знаходження джерела вимог:

-***розімкнені***, коли джерело знаходиться поза системою. Характеристики потоку заявок в такій системі не залежать від того, в якому стані сама СМО (скільки каналів зайнято);

-***замкнені***, коли джерело знаходиться в самій системі. У такому разі - залежать.

Четверта класифікація за числом фаз (або послідовних етапів) обслуговування одного клієнта:

- ***однофазовими*** є такі системи, у яких клієнт обслуговується в одному пункті (на одному робочому місці), потім залишає систему. Ресторан для обслуговування автомобілів, у якому офіціант одержує гроші й приносить замовлення в автомобіль, являє приклад однофазової системи;

- якщо ж у ресторані потрібно зробити замовлення в одному місці, оплатити його в іншому й одержати їжу в третьому, то ми маємо справу із ***багатофазовою*** (три фази) системою обслуговування.

На рис. 7.1 наведені системи обслуговування різної конфігурації.

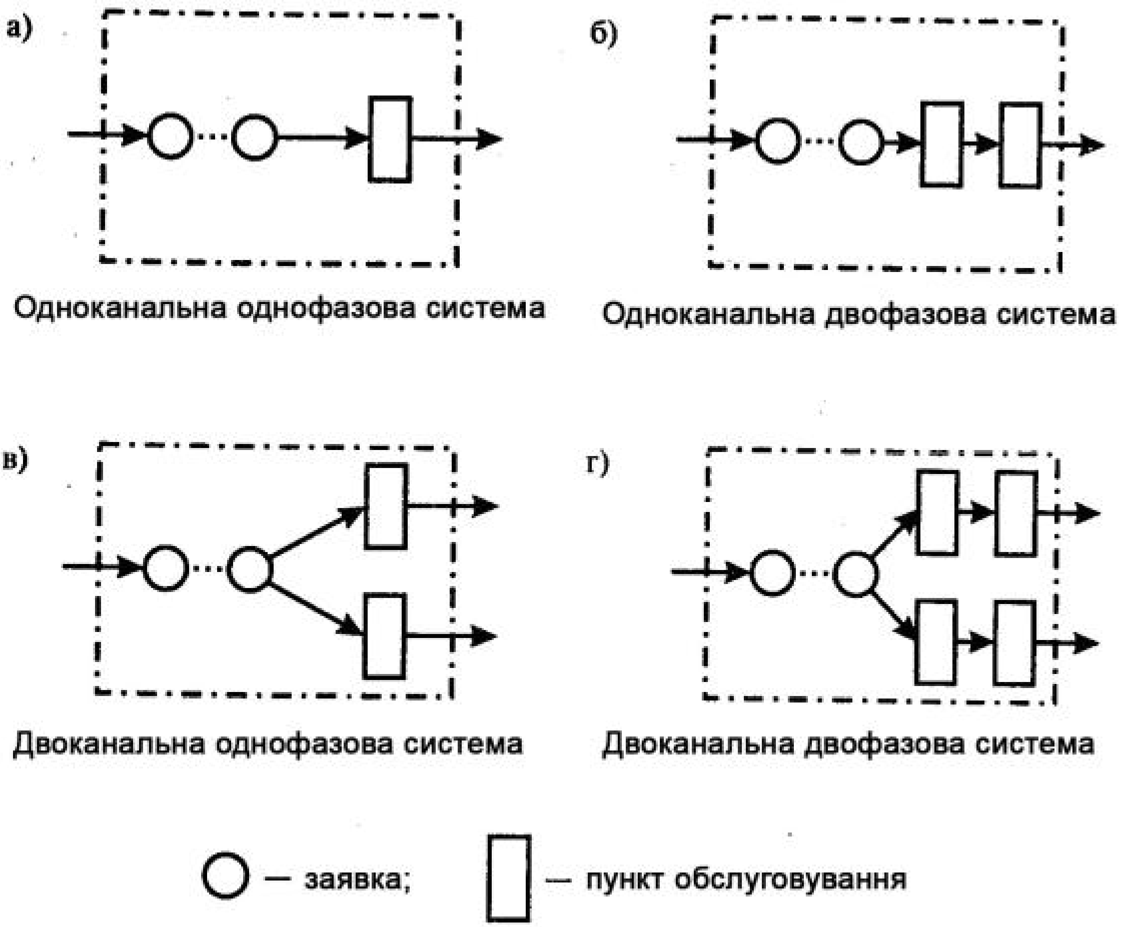


Рисунок 7.1 – Системи масового обслуговування різної конфігурації

Однією з форм класифікації систем масового обслуговування є **кодова (символьна) класифікація Д. Кендалла**. За цією класифікацією для позначення типу СМО використовуються позначення вигляду $X/Y/N/n$. Де X – закон розподілу інтервалів надходження заявок, Y – закон розподілу часу обслуговування, N – число каналів обслуговування, n – число місць в черзі.

Позначення законів розподілу в позиціях X і Y виконується зазвичай буквами з наступного списку:

M – експоненційне,

E_k – ерлангівське порядку k,

R – рівномірне,

D – детерміноване (постійна величина)

G – довільне (будь-якого вигляду) і т.д.

Якщо число місць в черзі не обмежене, то позиція *n* не вказується. Наприклад, $M \mid M \mid 1$ означає просту СМО (обидва розподіли експоненційні, канал обслуговування один, черга не обмежена), а позначення $R \mid D \mid 2 \mid 100$ відповідає СМО з рівномірним розподілом інтервалів вступу вимог, фіксованим часом їх обслуговування, двома каналами і 100 місцями в черзі. У цій СМО заявки, що приходять в моменти, коли всі місця в черзі зайняті, покидають систему (тобто втрачаються).

Моделювання функціонування систем масового обслуговування.

Залежно від сполучення наведених вище характеристик можуть розглядатися різні моделі систем масового обслуговування.

Ознайомимося з декількома найбільш відомими моделями. Всі вони мають наступні загальні характеристики:

а) пуассонівський розподіл ймовірностей надходження заявок;

б) стандартне поводження клієнтів;

в) правило обслуговування *FIFO* (першим прийшов – першим обслужений);

г) єдина фаза обслуговування.

I *Модель А* – модель одноканальної системи масового обслуговування *M/M/*1 з *пуассонівським* вхідним потоком заявок і *експоненційним* часом обслуговування.

Найбільше часто зустрічаються задачі масового обслуговування з єдиним каналом. У цьому випадку клієнти формують одну чергу до єдиного пункту обслуговування. Припустимо, що для систем цього типу виконуються наступні умови:

1.Заявки обслуговуються за принципом «першим прийшов — першим обслужений» *(FIFO),* причому кожний клієнт очікує своєї черги до кінця незалежно від довжини черги.

2.Появи заявок є незалежними подіями, однак середнє число заявок, що надходять в одиницю часу, незмінно.

3.Процес надходження заявок описується пуассонівським розподілом, причому заявки надходять із необмеженої множини.

4.Час обслуговування описується експоненційним розподілом ймовірностей.

5.Темп обслуговування вище темпу надходження заявок.

Нехай λ - число заявок в одиницю часу; μ - число клієнтів, що обслуговуються в одиницю часу;*n* -число заявок у системі.

Тоді система масового обслуговування описується рівняннями, наведеними нижче.

Формули для опису системи M/M/1:

$L_s = \frac{\lambda}{\mu-\lambda}$-середнє число клієнтів у системі;

$W_s = \frac{1}{\mu-\lambda}$- середній час обслуговування одного клієнта в системі (час очікування плюс час обслуговування);

$L_q = \frac{\lambda^2}{\mu(\mu-\lambda)}$- середнє число клієнтів у черзі;

$W_q = \frac{\lambda}{\mu(\mu-\lambda)}$- середній час очікування клієнта в черзі;

$r = \frac{\lambda}{\mu}$- характеристика завантаженості системи (доля часу, протягом якого система зайнята обслуговуванням);

$P_0 = 1 - \frac{\lambda}{\mu}$- ймовірність відсутності заявок у системі;

$P_{n>k} = (\frac{\lambda}{\mu})^{k+1}$ - ймовірність того, що в системі перебуває більш ніж *k* заявок.

II *Модель В* – багатоканальна система обслуговування *M/M/S.*У багатоканальній системі для обслуговування відкриті два канали або більш. Передбачається, що клієнти очікують у загальній черзі й звертаються в перший канал, що звільнився, обслуговування.

Приклад такої багатоканальної однофазової системи можна побачити в багатьох банках: із загальної черги клієнти звертаються в перше віконце, що звільнилося, для обслуговування.

У багатоканальної системі потік заявок підкоряється *пуассонівському* закону, а час обслуговування – *експоненційному.* Прихожий першим обслуговується першим, і всі канали обслуговування працюють в однаковому темпі. Формули, що описують модель *B,* досить складні для використання. Для розрахунку параметрів багатоканальної системи обслуговування зручно використовувати відповідне програмне забезпечення.

Для багатоканальної системи з необмеженою чергою повинне *r* виконуватися умова *n* < 1, де *r* – параметр завантаження системи (середнє число зайнятих каналів), *n* – мінімальна кількість каналів, при якому черга не буде рости нескінченно. У противному випадку граничні ймовірності існувати не можуть.

Формули для опису системи M/M/S:

$P_0 = \left(1 + \frac{r}{1!} + \frac{r^2}{2!} + \cdots + \frac{r^n}{n!} + \frac{r^{n+1}}{n!(n-r)}\right)^{-1}$ -ймовірність того, що система вільна;

$P_n = \frac{r^n}{n!} P_0$ - ймовірність того, що в системі перебуває *n* заявок;

$P_q = \frac{r^{n+1}}{n!(n-r)} P_0$ - ймовірність того, що заявка виявиться в черзі;

$r = \frac{\lambda}{\mu}$ - середнє число зайнятих каналів;

$L_q = \frac{r^{n+1} P_0}{n \cdot n! (1 - \frac{r}{n})^2}$ - середнє число заявок у черзі;

$L_s = L_q + r$ - середнє число заявок у системі;

$W_q = \frac{1}{\lambda} L_q$ - час знаходження заявки в черзі;

$W_s = \frac{1}{\lambda} L_s$ - час знаходження заявки в системі.

III *Модель C* – модель із постійним часом обслуговування *M/D/*1. Деякі системи мають *постійний,* а не експоненційно розподілений час обслуговування. У таких системах клієнти обслуговуються протягом фіксованого періоду часу, як, наприклад, на автоматичній мийці автомобілів. Для моделі *C* з постійним темпом обслуговування значення величин L_q і W_q удвічі менше, ніж відповідні значення в моделі *A, що* має змінний темп обслуговування.

Формули, що описують модель B:

$L_q = \frac{\lambda^2}{2\mu(\mu-\lambda)}$ - середня довжина черги;

$W_q = \frac{\lambda}{2\mu(\mu-\lambda)}$ - середній час очікування в черзі;

$L_s = L_q + \frac{\lambda}{\mu}$ - середнє число клієнтів у системі;

$W_s = W_q + \frac{1}{\mu}$ - середній час очікування в системі.

IV *Модель D* – модель із обмеженою популяцією.

Якщо число потенційних клієнтів системи обслуговування *обмежено,* ми маємо справу зі спеціальною моделлю. Таке завдання може виникнути, наприклад, якщо мова йде про обслуговування устаткування технічної служби, що має п'ять верстатів.

Особливість цієї моделі в порівнянні із трьома розглянутими раніше в тім, що існує *взаємозалежність* між довжиною черги й темпом надходження заявок.

V. *Модель E* – модель із обмеженою чергою. Модель відрізняється від попередніх тем, що число місць у черзі *обмежено.* У цьому випадку заявка, що прибула в систему, коли всі канали й місця в черзі зайняті, залишає систему необслугованою, тобто одержує відмову.

Як окремий випадок моделі з обмеженою чергою можна розглядати *модель із відмовами,* якщо кількість місць у черзі скоротити до нуля.

Порівняльна характеристика різних моделей систем масового обслуговування наведена в табл. 7.2.

Таблиця 7.2 – Порівняльна характеристика різних моделей СМО

Мод	Назва	Приклад	Числока	Чис	Розподіленнячасун	Розподіленнячасуоб	Кількістьк	Порядокпрох

ель	(технічненайменування)		налів	ло фаз	адходження	слуговування	лієнтів	одження черги
A	Проста система*(M/M/I)*	Довідковебюро ваеропорту	Один	Одна	Пуассонівське	Експоненційне	Необмежене	*FIFO*
B	Багатоканальна система*(M/M/S)*	Каси аеропорту	Декілька	Одна	Пуассонівське	Експоненційне	Необмежене	*FIFO*
C	Рівномірнеобслуговування*(M/D/I)*	Автоматична мийка	Один	Одна	Пуассонівське	Постійне	Необмежене	*FIFO*
D	Обмежена популяція	Літаки невеликої авіакомпанії	Один	Одна	Пуассонівське	Експоненційне	Обмежене	*FIFO*
E	Обмежена довжина черги	Злітно-посадкові смуги	Декілька	Одна	Пуассонівське	Експоненційне	Обмежене	*FIFO*

Процеси загибелі та розмноження

У теорії масового обслуговування широке поширення має спеціальний клас випадкових процесів – так званий *процес загибелі і розмноження (Марковський процес з дискретними станами і безперервним часом, Марковський ланцюг).* Назва цього процесу пов'язана з рядом біологічних задач, де він є математичною моделлю зміни чисельності біологічних популяцій.

Граф станів процесу загибелі і розмноження має вигляд, показаний на рис. 7.2.

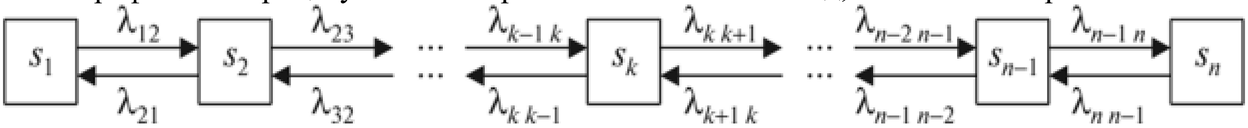


Рисунок 7.2 – Граф станів процесу загибелі і розмноження

Розглянемо впорядковану множину станів системиS_0, S_1, S_2, ..., S_k. Переходи можуть здійснюватися з будь-якого стану тільки в стани з сусідніми номерами, тобто зі стану S_kможливі переходи тільки або в станS_{k-1}, або в стан S_{k+1}.

Припустимо, що всі потоки подій, що переводять систему по стрілках графа, найпростіші з відповідними інтенсивностями$\lambda_{k,\,k+1}$ або $\lambda_{k+1,\,k}$.

З графу, представленого на рис. 7.2, складемо і вирішимо рівняння алгебри для **граничних ймовірностей станів** (їх існування випливає з можливості переходу з кожного стану в кожний інший і кінцевості числа станів).

Відповідно до правила складання таких рівнянь отримаємо:

для стану S_0:

$\lambda_{01}p_0 = \lambda_{10}p_1$,

для стану S_1:

$(\lambda_{12}+ \lambda_{10})p_1 = \lambda_{01}p_0+ \lambda_{21}p_2$, яке приводиться до вигляду$\lambda_{12}p_1 = \lambda_{21}p_2$.

Аналогічно, записуючи рівняння для граничних ймовірностей інших станів, можна отримати наступну систему рівнянь (рівняння Колмогорова для стаціонарного режиму) (7.6):

$$\begin{aligned} &\lambda_{01}p_0 = \lambda_{10}p_1; \\ &\lambda_{12}p_1 = \lambda_{21}p_2; \\ &\ldots\ldots\ldots\ldots \\ &\lambda_{k-1,k}p_{k-1} = \lambda_{k,k-1}p_k; \\ &\ldots\ldots\ldots\ldots \\ &\lambda_{n-1,n}p_{n-1} = \lambda_{n,n-1}p_n, \end{aligned} \qquad (7.6)$$

до якої додається нормувальна умова (7.7):

$$p_0 + p_1 + p_2 + \ldots + p_n = 1 \qquad (7.7)$$

Вирішуючи наведену систему рівнянь (7.6)-(7.7), можна отримати (7.8)-(7.9):

$$p_0 = \left(1 + \frac{\lambda_{01}}{\lambda_{10}} + \frac{\lambda_{12}\lambda_{01}}{\lambda_{21}\lambda_{10}} + \dots + \frac{\lambda_{n-1,n\dots}\lambda_{12}\lambda_{01}}{\lambda_{n,n-1\dots}\lambda_{21}\lambda_{10}}\right)^{-1}, \quad (7.8)$$

$$p_1 = \frac{\lambda_{01}}{\lambda_{10}} p_0, p_2 = \frac{\lambda_{12}\lambda_{01}}{\lambda_{21}\lambda_{10}} p_0, \dots, p_n = \frac{\lambda_{n-1,n\dots}\lambda_{12}\lambda_{01}}{\lambda_{n,n-1\dots}\lambda_{21}\lambda_{10}} p_0. \quad (7.9)$$

Легко помітити, що в формулах (7.9) для $p_1, p_2, \dots, p_n$ коефіцієнти p_0 є складовими, які стоять після одиниці у формулі (7.8). Чисельники цих коефіцієнтів представляють добуток всіх інтенсивностей, які стоять біля стрілок, що ведуть зліва направо до даного стану S_k(k = 1, 2, …, n), а знаменники – добуток всіх інтенсивностей, що стоять біля стрілок, що ведуть справа наліво зі стану S_k до S_0.

7.5 Висновки по лекції

Випадковим (стохастичним) процесом називають випадкову функцію аргументу t, який відіграє роль часу. Випадкові процеси мають **характеристики**: математичне очікування, дисперсію, середнє квадратичне відхилення, кореляційну функцію.

Для **нестаціонарного** ВП усі його характеристики змінюються з плином часу. ВП, характеристики якого не залежать від вибору початку відліку, тобто є однорідними щодо часу, називається **стаціонарним**.

Система масового обслуговування – це система, яка обслуговує вимоги, що надходять до неї (заявки). Основними елементами (компонентами) системи є: вхідний потік вимог; канали обслуговування; черга вимог; вихідний потік вимог.

СМО класифікують за різними ознаками: за наявністю черг, за числом каналів обслуговування, за місцем знаходження джерела вимог, за числом фаз (або послідовних етапів) обслуговування одного клієнта.

Найбільш відомі моделі СМО мають наступні **загальні характеристики**: пуассонівський розподіл ймовірностей надходження заявок; стандартне поводження клієнтів; правило обслуговування *FIFO* (першим прийшов – першим обслужений); єдина фаза обслуговування.

У теорії масового обслуговування широке поширення має спеціальний клас випадкових процесів – так званий **процес загибелі і розмноження** (Марковський процес з дискретними станами і безперервним часом, Марковський ланцюг).

Тема 8 *Імітаційне моделювання виробничих процесів на авіаційному транспорті* (2 год.)

8.1 Мета та завдання лекції

Метою лекції є ознайомлення з роллю імітаційного моделювання в оптимізації виробничих процесів на авіаційному транспорті.

Завдання лекції:

- ознайомитись з предметом та областями застосування імітаційного моделювання;
- розглянути метод Монте-Карло;
- розкрити підходи до визначення необхідного числа випробувань;
- розглянути способи моделювання випадкових величин з заданим законом розподілу;
- розглянути етапи розвитку обчислювальної техніки і застосування сучасних технологій для моделювання виробничих процесів.

8.2 План лекції

8.1 Предмет і області застосування імітаційного моделювання при вирішенні задач оптимізації виробничих процесів на авіаційному транспорті.

8.2 Загальні відомості про статистичне моделювання (метод Монте-Карло).

8.3 Визначення необхідного числа випробувань.

8.4 Моделювання випадкових величин з заданим законом розподілу.

8.5 Розвиток обчислювальної техніки і застосування сучасних технічних засобів для моделювання виробничих процесів на авіаційному транспорті.

8.3 Основні категорії, ключові поняття та визначення теми

Генератор випадкових чисел – обчислювальний або фізичний пристрій, спроєктований для генерації послідовності номерів чи символів, які не відповідають будь-якому шаблону, тобто, є випадковими.

Імітаційна модель (у вузькому значенні) – логіко-математичний опис об'єкта, який може бути використаний для експериментування на комп'ютері в цілях проєктування, аналізу і оцінки функціонування об'єкта.

Імітаційне моделювання – це метод дослідження, заснований на тому, що система, яка вивчається, замінюється імітатором і з ним проводяться експерименти з метою отримання інформації про цю систему.

Комп'ютер (ЕОМ)(від англ. computer - обчислювач) – електронно-обчислювальний пристрій, який виконує операції введення інформації, її зберігання, обробки за певною програмою, а також виведення отриманих результатів у формі, придатної для сприйняття людиною.

Метод статистичного моделювання (метод Монте-Карло) – це спосіб дослідження невизначених (стохастичних) економічних об’єктів і процесів, коли не повністю (до певної міри) відомими є внутрішні взаємодії в цих системах.

Обчислювальна машина – це технічний пристрій, призначений для введення, збереження, обробки і виводу інформації.

Обчислювальні оператори – описують будь-яку, як завгодно складну і громіздку групу операцій.

Оператори формування реалізацій випадків процесів – вводяться для імітації дії різних випадкових факторів, які супроводжують досліджуваний процес. За допомогою цих операторів виконується реалізація випадкових подій, випадкових величин, випадкових факторів і т. ін.

Оператори формування невипадкових величин - формують різні константи і невипадкові функції часу.

Псевдовипадкові (випадкові) числа – це числа, отримані за деяким правилом (формулою), що імітує значення випадкової величини.

8.4 Текст лекції

Предмет і області застосування імітаційного моделювання при вирішенні задач оптимізації виробничих процесів на авіаційному транспорті

Імітаційне моделювання охоплює методологію створення моделей систем, методи алгоритмізації та засоби програмних реалізацій імітаторів, планування, організацію і виконання на ЕОМ експериментів з імітаційними моделями, машинну обробку даних та аналіз результатів.

Імітаційне моделювання– це метод, який дозволяє будувати моделі процесів, що описують, як ці процеси проходили б насправді. Таку модель можна «програти» в часі як для одного випробування, так і заданої їх кількості. При цьому результати визначатимуться випадковим характером процесів. За цими даними можна отримати достатньо стійку статистику.

Імітаційне моделювання – це метод дослідження, заснований на тому, що система, яка вивчається, замінюється імітатором і з ним проводяться експерименти з метою отримання інформації про цю систему. Експериментування з імітатором називають **імітацією** (імітація – це дослідження сутності явища, не вдаючись до експериментів на реальному об'єкті).

Імітаційне моделювання – це окремий випадок математичного моделювання. Існує клас об'єктів, для яких з різних причин не розроблені аналітичні моделі або не розроблені методи розв'язування задач про такі моделі. В цьому випадку математична модель замінюється імітатором або імітаційною моделлю.

Імітаційна модель (у вузькому значенні) – логіко-математичний опис об'єкта, який може бути використаний для експериментування на комп'ютері в цілях проєктування, аналізу і оцінки функціонування об'єкта.

Імітація як метод розв'язування нетривіальних задач отримала початковий розвиток у зв'язку із створенням ЕОМ в 1950х-1960х роках.

Імітаційне моделювання, як інструмент експериментального дослідження складних систем, охоплює методологію створення моделей систем, методи алгоритмізації та засоби програмних реалізацій імітаторів, планування, організацію і виконання на ЕОМ експериментів з імітаційними моделями, машинну обробку даних та аналіз результатів. При цьому динамічні й стохастичні характеристики реальних процесів відображаються в моделі за допомогою спеціально сконструйованих процедур.

Слід зазначити, що діапазон застосування імітації на ЕОМ надзвичайно широкий — від конкретних форм діяльності авіаційних підприємств до імітації авіаційної галузі в цілому.

Серед напрямів використання імітаційного моделювання необхідно розглянути такі:

1) прогнозування розвитку авіаційної транспортної системи;

2) розробка і впровадження інформаційних систем в авіації;

3) створення системи протиповітряної оборони країни і планування військових операцій;

4) охорона навколишнього середовища від впливу авіації;

5) навчання та підготовка авіаційного персоналу;

6) розрахунок оптимальних схем заходу на посадку та виходу з району аеродрому.

Машинна імітація представляє собою цілий науковий напрям. Активне впровадження машинної імітації у сферу розв’язання різноманітних завдань організації і управління виробництвом, інтенсивна експлуатація імітаційних методів у всіх галузях інженерно-економічної діяльності, широке залучення ідей і методів машинного моделювання до підготовки наукових і виробничих кадрів – важливі народногосподарські завдання, успішне виконання яких багато в чому визначить ефективність суспільного виробництва в цілому.

Надзвичайно важливу роль методи машинної імітації мають відігравати при розв’язанні проблем комп’ютеризації інформаційних процесів на підприємствах і в установах, при створенні інформаційних систем економіко-організаційного управління. Стратегія розвитку сучасних інформаційних систем, зокрема систем підтримки прийняття рішень, має забезпечити аналітику формулювання і розв’язання такого класу задач:

1. Аналітичні – обчислення необхідних показників і статистичних характеристик авіаційної діяльності на основі ретроспективної (зверненої у минуле) інформації з баз даних.

2. Візуалізація даних – наглядне графічне та табличне відображення наявної інформації.

3. Здобуття знань – визначення взаємозв’язків і взаємозалежностей бізнес-процесів на базі існуючої інформації.

4. Імітаційні – проведення на ЕОМ експериментів з математичними моделями, які описують поведінку складних систем. Задачі цього класу застосовуються для аналізу можливих наслідків прийняття того чи іншого рішення (аналіз типу «Що, якщо?..»).

5. Синтез управління – визначення допустимих керуючих дій, які забезпечують досягнення поставлених цілей.

6. Оптимізаційні – засновані на інтеграції імітаційних, управлінських, оптимізаційних та статистичних методів моделювання і прогнозування. Машинна імітація процесів управління виробництвом, зокрема для оптимізації планування виробництва в цехах і на дільницях машинобудівних підприємств, для оптимального керування страховими заділами деталей тощо, застосовується порівняно давно і досить успішно. Важлива роль відводиться машинній імітації в процесі автоматизації підприємства в цілому. Зазначається, що автоматизація роботи підприємства – дуже відповідальний крок. Ця робота передбачає три етапи: етап інжинірингу – побудова моделі діяльності компанії; етап реінжинірингу –

здійснення аналізу й удосконалення моделі; етап управління – моніторинг роботи фірми в рамках створеної моделі. При цьому засобами аналізу виступають різні методики, зокрема такі, як функціонально-вартісний аналіз, імітаційне моделювання.

Основною перевагою імітаційних моделей (simulation model) в порівнянні з аналітичними є можливість вирішувати завдання виняткової складності з урахуванням випадкових чинників.

Метод імітаційного моделювання вдається успішно реалізувати за допомогою ЕОМ. Проте використання ЕОМ для цілей імітаційного моделювання вимагає вміння розробки моделюючого алгоритму, який повинен відтворити формальний процес складної системи. Моделюючий алгоритм дозволяє за вихідними даними отримати відомості про стани виробничого процесу в довільний момент часу.

Розробка моделюючого алгоритму неможливо без глибокого знання модельованого об'єкта і його функціонування, причому, в процесі розробки відбувається поглиблення та уточнення розуміння об'єкта. Тому процес розробки моделі має і самостійне значення. Оскільки дозволяє виявити недоліки, розкрити резерви, відкрити нові можливості об'єкта ще до моделювання і дати важливі практичні рекомендації щодо вдосконалення об'єкту і підвищенню ефективності його функціонування.

Однією з переваг моделювання виробничих процесів є можливість розгляду змінних факторів у всьому діапазоні їх значень.

Загальні відомості про статистичне моделювання (метод Монте-Карло)

Метод статистичного моделювання (метод Монте-Карло) – це спосіб дослідження невизначених (стохастичних) об’єктів і процесів, коли не повністю (до певної міри) відомі внутрішні взаємодії в цих системах.

Цей метод полягає у модельному відтворенні процесу за допомогою стохастичної математичної моделі та обчисленні характеристик цього процесу. Одне таке відтворення можливого (випадкового) стану функціонування модельованої системи називають реалізацією (чи імітаційним прогоном; далі – прогоном).

Після кожного прогону реєструють сукупність параметрів, що характеризують випадкову подію (її реалізацію). Метод ґрунтується на багатократних прогонах (випадкових реалізаціях) на підставі побудованої моделі з подальшим статистичним опрацюванням отриманих даних з метою визначення числових характеристик досліджуваного об’єкта (процесу) у вигляді статистичних оцінок його параметрів. Процес моделювання економічної системи зводиться до машинної імітації досліджуваного процесу, котрий моделюється на ЕОМ з усіма суттєвими невизначеностями, випадковостями і породженим ними ризиком. Імітаційне моделювання нерідко має назву симулятивного моделювання. Перші відомості про метод Монте-Карло були опубліковані в кінці 40-х pp. XX століття. Авторами методу є американські математики – економісти Дж. Нейман і С. Улам.

Метод Монте-Карло (за назвою міста, яке відоме своїми гральними домами)– загальна назва групи числових методів, заснованих на одержанні великої кількості реалізацій стохастичного (випадкового) процесу, який формується у той спосіб, щоб його імовірнісні характеристики збігалися з аналогічними величинами задачі, яка вирішується. Використовується для вирішення задач у фізиці, математиці, економіці, оптимізації, теорії управління тощо.

Метод Монте-Карло, який називають також методом статистичного моделювання, дає змогу вирішувати ймовірнісні проблеми статистичними методами. Теорія цього методу вказує, як найдоцільніше вибрати випадкову величину *X*, як знайти її можливі значення.

Метод Монте-Карло тісно пов'язаний із завданнями теорії ймовірностей, математичної статистики й обчислювальної математики. У зв'язку із завданням моделювання випадкових величин (особливо рівномірно розподілених) істотну роль відіграють також методи теорії чисел.

Метод Монте-Карло придатний для вирішення багатьох завдань, пов'язаних із моделюванням надзвичайних ситуацій, тому його застосування виправдане насамперед у тих завданнях, які допускають теоретико-ймовірнісний опис.

Оскільки метод Монте-Карло потребує **проведення великої кількості випробувань**, його часто називають **методом статистичних випробувань**. Він має деякі очевидні **переваги**:

• не потребує жодних припущень щодо регулярності за винятком квадратичної інтегрованості; це може бути корисним, тому що в багатьох випадках трапляються дуже складні функції, властивості регулярності яких непросто встановити;

• забезпечує здійснення процедури навіть у багатовимірному випадку, коли чисельне інтегрування непридатне, наприклад за числа вимірів >**10**.

• легко застосовується за малих обмежень або без попереднього аналізу завдання.

Метод Монте-Карло має й деякі **недоліки**, а саме:

• межі похибки точно не визначені, а включають деяку випадковість;

• статистична похибка спадає дуже повільно;

• необхідні таблиці випадкових чисел.

Повільна збіжність є істотним недоліком методу, однак можна скористатися його модифікаціями, які забезпечують високий порядок збіжності за певних припущень. Щоправда, обчислювальна процедура при цьому ускладнюється і за складністю наближається до інших процедур обчислювальної математики. Збіжність методу Монте-Карло визначають як збіжність за ймовірністю. Цю обставину навряд чи варто вважати його недоліком, оскільки ймовірнісні методи достатньою мірою виправдовують себе в практичних застосуваннях.

Імітаційне моделювання за методом Монте-Карло (Monte-Carlo Simulation) дає змогу побудувати математичну модель з невизначеними параметрами, і, знаючи їх ймовірнісні розподіли, а також зв'язок між змінами параметрів (кореляцію), отримати розподіл досліджуваної функції.

Метод Монте-Карло часто застосовують для обчислення надійності складних систем із великою кількістю елементів. Аналіз ризиків із використанням методу імітаційного моделювання Монте-Карло є поєднанням методів аналізу чутливості та аналізу сценаріїв на базі теорії ймовірностей. **Результат такого комплексного аналізу – розподіл ймовірностей можливих результатів.**

Суть методу ілюструє такий приклад. Нехай відома невипадкова функція, скажімо збиток$Y=\varphi(X)$, аргументом якої є випадкова величина X із заданим законом розподілу $H_X(x)$. Треба оцінити закон розподілу випадкової величини Y або параметри закону розподілу (якщо його вигляд заздалегідь відомий). Для цього за відомою функцією розподілу $H_X(x)$проводять n випробувань, тобто розігрують (моделюють) n можливих значень X:$x_1, x_2, \ldots, x_N$, і з цими числами виконують дії відповідно до функціональної залежності $\varphi(X)$:

$y_1=\varphi(x_1), y_2=\varphi(x_2), \ldots, y_n=\varphi(x_n)$.

Отриману вибірку $y_1, y_2, \ldots, y_n$ обробляють відомими методами математичної статистики й отримують параметри і закон розподілу випадкової величини Y.

Для функції кількох аргументів алгоритм методу не зазнає істотних змін. У цьому разі розігрують вибірки для кожного аргументу X_i згідно із законом його розподілу $H_X(x)$ (припускають стохастичну незалежність аргументів):

$x_{i1}, x_{i2}, \ldots, x_{iN}, i = 1, 2, \ldots, m$.

У подальшому отримують вибірку $\{y_i\}$, виконавши дії відповідно до функціональної залежності$Y=\varphi(X_1, X_2, \ldots, X_m)$:

$y_1=\varphi(x_{11}, x_{21}, \ldots, x_{m1})$;

$y_2=\varphi(x_{21}, x_{22}, \ldots, x_{m2})$;

…………………………

$y_n=\varphi(x_{1n}, x_{2n}, \ldots, x_{mn})$.

Оцінку числа необхідних випробувань, які забезпечать необхідну похибку обчислень не вищу за задану, можна отримати на основі співвідношень граничних теорем.

Класичне завдання оцінювання математичного сподівання випадкової величини методом Монте-Карло полягає в такому.

Нехай вдалося деяким способом обчислити значення незалежних реалізацій $y_i(i = 1, 2, \ldots, n)$ випадкової величини Y, для якої існують і скінченні її математичне очікування MY і дисперсія DY. Тоді середнє арифметичне значення $\overline{Y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ за достатньо великого n має нормальний розподіл, і при заданому рівні довіри γ справедлива нерівність (8.1):

$$\left| MY - \frac{1}{n}\sum_{i=1}^{n} y_i \right| \leq \alpha \frac{\sqrt{DY}}{\sqrt{n}}, \tag{8.1}$$

де α – константа, яка визначається вибором величин γ.

Із цієї нерівності випливає, що кількість обчислень для досягнення заданої точності при заданому рівні довіри пропорційна величині $\sqrt{DY} / \sqrt{n}$.

Загалом підвищення точності обчислення на один порядок потребує збільшення кількості обчислень на два порядки. Багато фахівців вважає, що це є основним недоліком методу Монте-Карло.

Метод Монте-Карло спирається на використання випадкових чисел. Розподілені за заданим законом випадкові числа $x_к$ зазвичай отримують у два етапи:

• знаходження випадкового числа R, рівномірно розподіленого в інтервалі (0, 1);

• перетворення рівномірно розподілених випадкових чисел R_i на ті, які шукають x_k.

Генератори випадкових чисел є у всіх статистичних пакетах. За умови їх використання алгоритм методу імітації Монте-Карло складається з наведених нижче кроків.

Крок 1. Спираючись на використання статистичного пакета, випадковим чином вибирають, ґрунтуючись на ймовірнісній функції розподілу, значення змінної, котра є одним з параметрів визначення функції.

Крок 2. Обране значення випадкової величини поряд зі значеннями змінних, які є екзогенними, використовують під час підрахунку значення функції.

Кроки 1 і 2 повторюють багато разів, наприклад 1000, й отримані 1000 значень використовують для побудови щільності розподілу досліджуваної величини зі своїми власними математичним сподіванням і стандартним відхиленням.

Ймовірнісний розподіл регулює ймовірність вибору значень із певного інтервалу. В рамках моделі ймовірнісного аналізу ризиків проводять велику кількість ітерацій, що дають змогу встановити, як поводиться результативний показник (у яких межах коливається, як розподілений) у разі підстановки в модель різних значень змінної відповідно до заданого розподілу.

Отже, розглянуті методи уможливлюють отримання оцінки ризиків виникнення аварій та надзвичайних ситуацій на потенційно небезпечних об'єктах, об'єктах підвищеної небезпеки, що має важливе значення для оцінювання ступеня небезпеки і складання декларації безпеки об'єкта.

Розробка і вдосконалення теоретичних і методологічних засобів імовірнісного аналізу безпеки складних систем залишається актуальною науковою проблемою і важливим практичним завданням.

Імітаційне моделювання, при якому відтворюються випадкові явища, називається статистичним імітаційним моделюванням. Статистичне імітаційне моделювання базується на чисельному статистичному методі вирішення математичних завдань, який називається методом Монте-Карло.

В основі імітаційного моделювання лежить **метод статистичних випробувань(метод Монте-Карло)**, який базується на використанні випадкових чисел, тобто можливих значень деякої випадкової величини з заданим розподілом ймовірностей. Кожного разу, коли на хід модельованого процесу впливає випадковий фактор, його вплив імітується за допомогою спеціально організованого розіграшу (“жеребу”). Таким чином, будується одна реалізація випадкового явища, котра відтворює якби результат одного досліду. В процесі моделювання формується велике число реалізацій.

Метод статистичного моделювання зазвичай включає наступні етапи:

1. Спочатку дається опис функціонування системи, тобто опис завдань, що стоять перед системою, уточнюються вихідні (відправні) положення; розглядаються обмеження; виділяються підпроцеси; намічаються характеристики, які потрібно отримати на виході, і вибирається цільова функція або критерій, за допомогою якого буде проводитися оцінка ефективності функціонування системи.

2. Проводиться збір та обробка інформації, що характеризує роботу підпроцесів системи і всього процесу в цілому.

3. Виконується формалізація роботи системи, тобто виділяються головні фактори і виключаються другорядні, якими можна знехтувати. На основі цього складається відповідальна система адекватна математична модель процесу.

4. Складається алгоритм прийнятої математичної моделі у вигляді операторної блок-схеми.

5. Складається програма для багаторазового відтворення на ЕОМ процесі при числі реалізацій, що забезпечують задану точність.

6. Виконується моделювання роботи системи на ЕОМ і видача на друк основних результатів моделювання. **Зазвичай при цьому отримують**:

а) точкові оцінки, тобто математичне сподівання, дисперсію для кожного з підпроцесів і по всьому процесу в цілому;

б) інтервальні оцінки, тобто довірчі інтервали та довірчі смуги розкиду середнього результату для кожного з підпроцесів і всього процесу в цілому;

в) будуються криві рівнянь регресії, що характеризують залежність досліджуваних параметрів від різних аргументів.

Крім перерахованого, можуть обчислюватися спеціальні характеристики, властиві розглянутого явища. Все це дозволяє прогнозувати перебіг процесу і введенням відповідних поправок оптимізувати його перебіг.

Узагальнена структура дослідження системи методом статистичного моделювання показана на рис. 8.1.

Із вище зазначеного виходить, що **суть методу статистичного моделювання** зводиться до побудови для процесу функціонування досліджуваної системи S деякого моделюючого алгоритму, який імітує поведінку і взаємодію елементів системи з урахуванням випадкових вхідних дій і дій зовнішнього середовища E, і реалізації цього алгоритму з використанням програмно–технічних засобів ЕОМ.

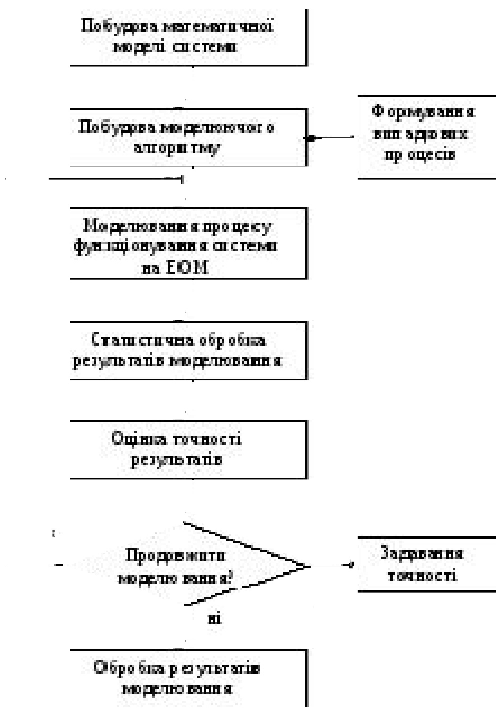


Рисунок 8.1 – Схема алгоритму статистичного моделювання процесів функціонування системи

За математичні схеми, які використовуються для формування дії випадкових факторів, приймаються *випадкові події, випадкові величини і випадкові функції.*

Реалізацію випадкових величин і функцій здійснюють з допомогою спеціального механізму розіграшу в ході статистичного моделювання. В результаті статистичного моделювання системи *S* отримується серія часткових значень шуканих величин або функцій, статистична обробка котрих дозволяє отримати відомості про поведінку реального об'єкту або процесу в довільний момент часу.

Якщо кількість реалізацій достатньо велика, то отримані результати моделювання системи набувають статистичної стійкості і з достатньою точністю можуть бути прийняті як оцінки шуканих характеристик процесу функціонування системи *S*.

Для моделювання процесу на ЕОМ необхідно перетворити його математичну модель в моделюючий алгоритм. Методика побудови моделюючих алгоритмів передбачає використання наступних **основних типів операторів**:

1 **Обчислювальні оператори**. В операторних схемах моделюючих алгоритмів кожний обчислювальний оператор може описувати будь-яку, як завгодно складну і громіздку групу операцій.

2 **Оператори формування реалізацій випадків процесів**. Їх вводять для імітації дії різних випадкових факторів, які супроводжують досліджуваний процес. З допомогою цих операторів виконується реалізація випадкових подій, випадкових величин, випадкових факторів і т. ін.

3 **Оператори формування невипадкових величин**. З їх допомогою формуються різні константи і невипадкові функції часу.

4 **Лічильники**.

Метод статистичних випробувань – це числовий метод математичного моделювання випадкових величин, який передбачає безпосереднє включення випадкового фактору в процес моделювання і є його істотним елементом. Вплив випадкових факторів на систему моделюється за допомогою випадкових чисел. Результатом моделювання є випадкові процеси або величини, які характеризують систему, що моделюється. Щоб їх імовірнісні характеристики (імовірність деяких подій, математичне очікування, дисперсія випадкових величин, імовірності попадання випадкової величини в задану область та ін.) співпадали з аналогічними параметрами реальної системи або процесу під час моделювання потрібно отримати велику кількість реалізацій випадкових величин або процесів. Таким чином, метод полягає в багатократному проведенні випробувань побудованої імовірнісної моделі і подальшій статистичній обробці результатів моделювання з метою визначення шуканих характеристик розглядуваного процесу у вигляді оцінок його параметрів. Точність оцінок цих параметрів визначає ступінь наближення розв'язку задачі до ймовірнісних характеристик.

На практиці метод статистичних випробувань доцільно використовувати в таких випадках, коли:

♦ розв'язувати задачу цим методом простіше, ніж будь-яким іншим;

♦ досліджується система, функціонування якої визначається багатьма ймовірнісними параметрами елементарних явищ;

♦ важко або неможливо побудувати аналітичну імовірнісну модель системи.

Важливою властивістю цього методу є те, що для звичайних числових методів обсяг обчислень зростає в разі збільшення розмірності задачі приблизно як показникова функція розмірності задачі, а для методу статистичних випробувань — лише як лінійна функція розмірності.

Визначення необхідного числа випробувань

Теоретичною основою методу статистичного моделювання є **закон великих чисел**. У теорії ймовірностей закон великих чисел ґрунтується на доведенні низки теорем для різних умов збіжності за ймовірністю середніх значень результатів (на підставі великої кількості спостережень) до деяких величин.

У загальному вигляді закон великих чисел (теорема П.Л.Чебишева) записується так (8.2):

$$lim\, P\left(\left|\frac{\sum x_i}{N} - M(x)\right| < \varepsilon\right) \to 1, \qquad (8.2)$$

де P – ймовірність складної події;

$M(x) = \overline{X}$ – математичне очікування випадкової величини;

$M(x) = \frac{\sum x_i}{N}$ – середнє арифметичне спостережуваних значень;

N – число випробувань (число реалізацій);

ε – як завгодно мале позитивне число.

Теорема Чебишева формулюється так: «При великому числі випробувань середнє арифметичне спостережуваних значень випадкової величини сходиться по ймовірності до її математичного очікування».

При переході до відносних (без розмірності) параметрам, маємо окремий випадок закону великих чисел (теорема Я. Бернуллі), який аналітично записується так (8.3):

$$lim\, P\left(\left|\frac{m_i^*}{N} - P\right| < \varepsilon\right) \to 1, \qquad (8.3)$$

деm_i* – число появи події (частота);

$P_i^* = \frac{m_i^*}{N}$ – частість події;

P– ймовірність події.

Теорема Я. Бернуллі формулюється так: «При великому числі випробувань частота події сходиться по ймовірності до ймовірності події».

Графічно закон великих чисел (і його окремий випадок) представлений на рис. 8.2.

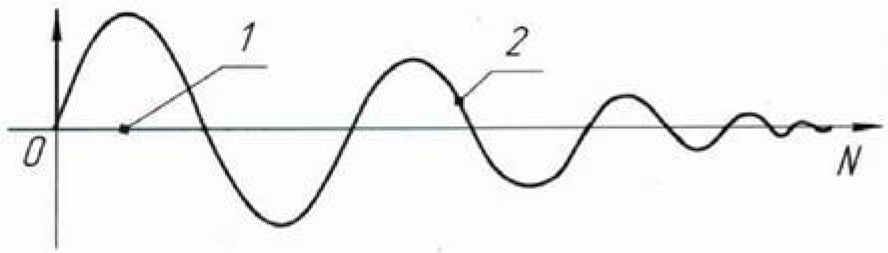


Рисунок 8.2 – Графічне представлення закону великих чисел: 1 – математичне очікування випадкової величини*M(x)* (ймовірність події *P*), 2 – середнє арифметичне значень спостережень*M*(x)*(частота подіїP_i*)

З рис. 8.2 випливає, що в міру збільшення числа випробувань середнє арифметичне значень спостережень випадкової величини*M*(x)* і частота події P_i* асимптотично і необмежено наближається до математичного очікування *M(x)*і ймовірності події *P*.

Це означає, що якщо справити велике число випробувань, то одержувані статистичні характеристики (середні значення) можуть розглядатися як істинні. Зазначене положення і становить математичну основу методу статистичного моделювання, тобто методу Монте-Карло.

Отже, ідея методу Монте-Карло проста і полягає в наступному: виробляється «розіграш» процесу (явища) за допомогою спеціально організованої процедури, який дає випадковий результат. Кожен «розіграш» дає нову, відмінну від інших, реалізацію досліджуваного процесу. Якщо таких реалізацій проведено багато, то ці безліч реалізацій можна використовувати як статистичний матеріал, обробивши який методами математичної статистики, отримуємо характеристики які нас цікавлять: ймовірності станів, математичне очікування і т.д.

Для отримання достовірних результатів, кількість проведених експериментів, вимірювань, анкетних опитувань тощо повинна бути не меншою за 30 реалізацій.

Моделювання випадкових величин з заданим законом розподілу

Кожного разу, коли на хід модельованого процесу впливає випадковий чинник, його вплив імітується за допомогою спеціально організованого розіграшу (жеребкування). Таким способом будується випадкова реалізація модельованого явища, яка є одним із результатів дослідження. За результатами окремого досліду, звичайно, не можна робити висновок щодо закономірностей досліджуваного процесу. Але за великої кількості реалізацій середні характеристики (математичне очікування, мода, медіана), що їх виробляє (генерує) модель, набувають стійких властивостей, котрі посилюються зі зростанням кількості реалізацій (прогонів). Звісно, залишається певний ризик, який характеризується тим, що модель є гомогенною, існує неповнота даних тощо.

Для моделювання випадкової величини потрібно знати закон її розподілу. *Найзагальнішим способом* отримання послідовності випадкових чисел, що є послідовністю реалізацій випадкової величини, котра розподілена за довільним законом, є спосіб, в основі якого – процес формування їх з вихідної послідовності випадкових чисел. Ця послідовність є послідовністю реалізацій випадкової величини, що розподілена в інтервалі [0; 1] згідно з рівномірним законом розподілу.

Згадану послідовність випадкових чисел з рівномірним законом розподілу можна отримати двома способами:

- використанням таблиць випадкових чисел (вручну);
- застосуванням генераторів випадкових чисел (спеціальних програм, що входять до складу програмного забезпечення комп’ютера).

Нині використовують псевдовипадкові числа, що відповідають рівномірному закону розподілу. **Псевдовипадкові (випадкові) числа** – це числа, отримані за деяким правилом (формулою), що імітує значення випадкової величини. Розроблено низку алгоритмів для отримання псевдовипадкових чисел. Датчики псевдовипадкових чисел є складовими більшості програмних комплексів.

У складі трансляторів майже всіх алгоритмічних мов є стандартні процедури (чи функції), котрі генерують випадкові (точніше, псевдовипадкові) числа, що є реалізаціями послідовності випадкових чисел із рівномірним законом розподілу. Наприклад, у складі транслятора мови Visual Basic – стандартна **функція RND**, що видає випадкові дійсні числа одинарної точності в інтервалі (0; 1). Звернення до цієї функції може мати вигляд x = RND, де x – можливе значення (реалізація) випадкової величини, яка рівномірно розподілена на інтервалі (0; 1).

Для перетворення послідовності випадкових чисел, що є реалізаціями випадкової величини з рівномірним законом розподілу в інтервалі (0; 1), у послідовність випадкових чисел, що є реалізаціями випадкової величини із заданою інтегральною функцією розподілу $F(x)$, треба із сукупності випадкових чисел з рівномірним законом розподілу в інтервалі [0; 1]вибрати випадкове число ξ і розв’язати рівняння (8.4):

$$F(x \text{ відносно} \xi) = \chi. \qquad (8.4)$$

У випадку, коли задана функція щільності ймовірності $f(x)$, співвідношення (8.4) набирає вигляду (8.5):

$$\int_{-\infty}^{x} f(x)dx = \xi. \qquad (8.5)$$

Для низки законів розподілу отримано аналітичний розв’язок рівняння (8.4), результат якого наведено в табл. 8.1.

Таблиця 8.1 – Формули для моделювання випадкових величин

Закони розподілу випадкової величини	Щільність розподілу	Формули для моделювання випадкових величин

Експоненційний (показовий)	$f(x)=\lambda e^{-\lambda x}$	$x_i=-\frac{1}{\lambda}\ln\xi_i$
Вейбула	$f(x)=\frac{a}{b}\left(\frac{x}{a}\right)^{a-1}\exp\left[-\left(\frac{x}{b}\right)^{a}\right]$	$x_i=-b(\ln\xi_i)^{1/a}$
Гамма-розподіл (η – цілі числа)	$f(x)=\frac{\lambda^{\eta}}{\Gamma(\eta)}e^{-\lambda x}x^{\eta-1}$	$x_i=-\frac{1}{\lambda}\sum_{j=1}^{\eta}\ln(1-\xi_{ij})$
Нормальний	$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{\frac{-(x-m)^2}{2\sigma^2}}$	$x_i=m+\sigma\left(\sum_{j=1}^{12}\xi_{ij}-6\right)$

Моделювання дискретної випадкової величини. Розподіл дискретної випадкової величини X може бути поданий у вигляді табл. 8.2.

Таблиця 8.2 – Розподіл дискретної випадкової величини X

x_i	x_1	x_2	…	x_n
p_i	p_1	p_2	…	p_n

Тут p_j – імовірність того, що випадкова величина X набуває значення x_j, тобто $p_j = P\{X=x_j\}, j = 1, \ldots, n$. При цьому виконується умова $\sum_{j=1}^{n} p_j = 1$.

Поділимо інтервал (0; 1) на n відрізків, довжини котрих дорівнюють заданим ймовірностям p_j, $j = 1, \ldots, n$, що формується генератором випадкових чисел, які відповідають рівномірному закону розподілу на інтервалі (0; 1), попадає до інтервалу ξ. Якщо випадкове число p_k, то випадкова величина X набуває значення x_k. Отже, під час моделювання дискретної випадкової величини фактично використовується та сама процедура, що й за моделювання повної групи попарно несумісних подій (рис. 8.3).

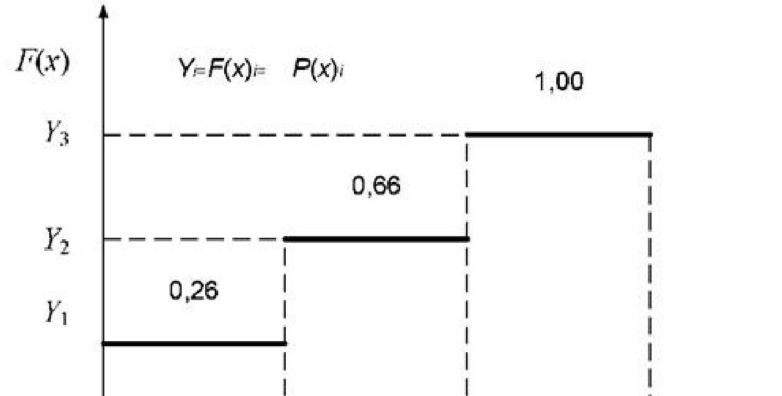


Рисунок 8.3 – Графік функції розподілу ймовірностей дискретної випадкової величини

Моделювання випадкових величин з рівномірним розподілом. Генератор випадкових чисел генерує послідовність реалізацій випадкової величини ξ з рівномірною функцією розподілу на інтервалі [0; 1]. Для того, щоб отримати реалізацію випадкової величини з рівномірним розподілом на інтервалі $[a; b]$ (рис. 8.4), необхідно розв'язати відносно x рівняння (8.6):

$$\xi = F(x) = \frac{x-a}{b-a}, \tag{8.6}$$

звідси $x = a + \xi(b-a)$.

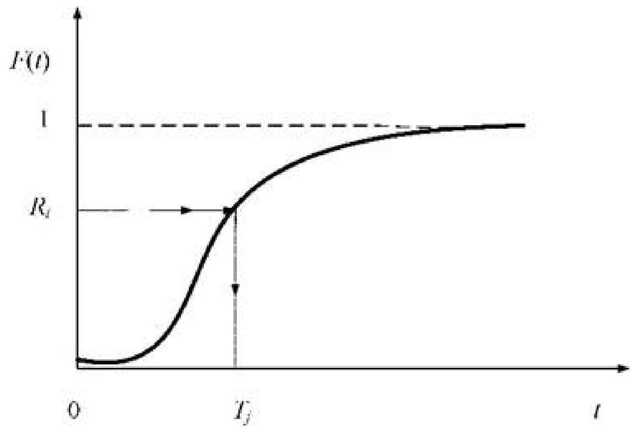


Рисунок 8.4 – Графік функції розподілу ймовірностей неперервної випадкової величини

Моделювання випадкових величин з інтервально-постійною функцією розподілу. Нехай є підстави наближено подати функцію розподілу випадкової величини *X*, яка задана на відрізку [a_0; a_n], інтервально-постійною функцією щільності розподілу *f(x)*. Це означає, що відрізок [a_0; a_n] поділено на *n* відрізків так, що відомі ймовірності попадання на кожен з них (p_k, $k = 0, 1, \dots, n$) (рис. 8.5).

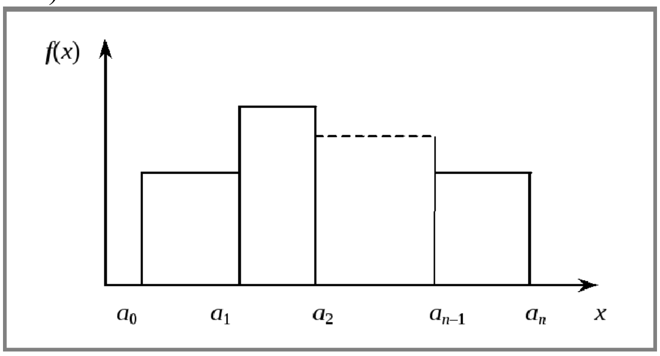


Рисунок 8.5 – Інтервально-постійна функція щільності розподілу випадкової величини

Тобто,$\int_{a_{k-1}}^{a_k} f(x)dx = p_k,\ k = 1, \dots,\ n.$

З умови $f(x) = const = c_k$ на кожному частковому інтервалі випливає, що реалізація випадкової величини *X* може бути визначена за формулою(8.7):

$$x_k = a_{k-1} + \xi(a_k - a_{k-1}), k = 1, \dots,\ n, \qquad (8.7)$$

де ξ– реалізація випадкової величини, рівномірно розподіленої на інтервалі (0; 1);

a_{k-1}– ліва межа часткового інтервалу;

a_k– права межа часткового інтервалу.

Попадання у будь-який частковий інтервал можна розглядати як подію, що входить до складу повної групи попарно несумісних подій, а номер відповідного інтервалу – як дискретну випадкову величину ηз розподілом (табл. 8.3).

Таблиця 8.3 – Розподіл дискретної випадкової величини η

η_i	1	2	…	n
p_i	p_1	p_2	…	p_n

Тому процедура моделювання загалом полягає у такому:

1. За допомогою генератора випадкових чисел (ГВЧ) моделюємо дискретну випадкову величину η– номер інтервалу.

2. За допомогою ГВЧ розігруємо випадкову величину ξ (з рівномірним розподілом на інтервалі (0; 1) і визначаємо реалізацію випадкової величини *X*. Блок-схему алгоритму наведено на рис. 8.6.

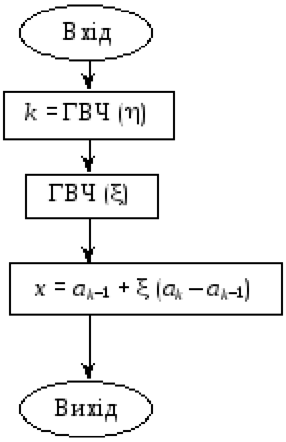


Рисунок 8.6 – Блок-схема алгоритму моделювання випадкової величини з інтервально-постійною функцією щільності розподілу

Розвиток обчислювальної техніки і застосування сучасних технічних засобів для моделювання виробничих процесів на авіаційному транспорті

Для інформатизації всіх сфер суспільства потрібні програмно-апаратні засоби, обчислювальна техніка і пристрої зв'язку. Різні технічні засоби забезпечують прийом і передачу основних видів інформації – мови, даних, зображення у статиці і динаміці – на фізичному рівні з максимальним використанням слуху та зору. Безпосередньо з людиною пов'язані відносно громіздкі пристрої, що забезпечують проходження різноманітних людино-машинних вхідних і вихідних потоків інформації (дисплеї, клавіатури, друку вальні пристрої, сканери, миші, джойстики тощо). Технічні засоби зв'язку забезпечують передачу потоків інформації до навколишнього середовища. На підприємстві залежно від масштабу і специфіки виробництва для збереження й обробки інформації можуть використовувати від одного до тисячі комп'ютерів.

Обчислювальна машина – це технічний пристрій, призначений для введення, збереження, обробки і виводу інформації.

Загальні принципи роботи обчислювальних пристроїв сформульовані фон Нейманом.

Коротка історія розвитку обчислювальної техніки

Сучасним комп'ютерам передували механічні й електромеханічні пристрої. Так, **В. Шикард** створив годинник зі спеціальними обчисленнями. Це була шестирозрядна машина, що могла складати, віднімати числа та інформувати користувача про переповнення даних дзвінком. У 1642 р. французький математик **Б. Паскаль** сконструював десяткове колесо для рахунків, яке могло підсумовувати числа до восьми знаків.

Обчислювальні пристрої розповсюдилися у 1820 р., коли француз Ч. Кальмар винайшов **арифмометр** - машину, що могла робити чотири основні арифметичні дії. Завдяки своїй універсальності арифмометри використовувалися досить довго. Багато вчених і винахідників удосконалили ці пристрої. Так, у 1880 р. швед В. Однер, що жив у Росії, створив арифмометр із перемінним числом зубців, аналог якого "Фелікс" випускався у Радянському Союзі до 70-х років XX ст.

Початок ери комп'ютерів пов'язаний з ім'ям **Ч. Беббіджа** – англійського математика, який звернув увагу на те, що машина здатна виконувати складні математичні обчислення шляхом багаторазового повторення простих кроків. Для повторення операцій у машині Ч. Беббіджа, призначеній для вирішення диференціальних рівнянь, використовувалася енергія пари. Таким чином, процес обчислень був автоматизований, тобто проходив без участі людини. Аналітична машина Ч. Беббіджа, що складалася більше ніж з 50 000 деталей, мала всі основні компоненти сучасного комп'ютера: пристрій вводу інформації, блок управління, запам'ятовуючий пристрій і пристрій виводу результатів. Ч. Беббідж запропонував загальну структуру обчислювальної машини, де була реалізована дія умовного переходу. Аналітична машина могла виконувати певний набір інструкцій, що записувалися на перфокарти - кожній інструкції аналітичної машини відповідала певна послідовність отворів на перфокарті. У 1843 р. для машини Ч. Беббіджа **А. Лавлейс** (перший програміст у світі) написала першу складну програму (розрахунок чисел Бернуллі), використовуючи бінарну систему числення. Ідеї Ч. Беббіджа та А. Лавлейс випередили свій час більш як на століття і не могли бути повноцінно реалізовані у той час, проте вони вплинули на розвиток обчислювальної техніки.

Передумовою електромеханічного етапу розвитку обчислювальної техніки була необхідність великої кількості розрахунків у промисловості, економіці, військовій справі. У 1889 р. Д. Фелт винайшов перший настільний калькулятор, а американський винахідник Г. Холлерит сконструював пристрій для розв'язування статистичних задач, що застосовувався з метою обробки результатів перепису населення США. У ньому перфокарти використовувалися для збереження даних. Спеціальний електричний прилад розпізнавав отвори на перфокартах і посилав сигнали у пристрій обробки. У 1896 р. Г. Холлерит заснував компанію з виробництва перфораторів Tabulating Machine Company, яка у 1924 р. після серії об'єднань перетворилася у відому International Business Machines (IBM).

На початку 30-х років лічильно-перфораційна техніка використовувалась у Радянському Союзі для астрономічних розрахунків. У 40-х роках були створені складні релейні системи з програмним управлінням, що стали прямими попередниками електронно-обчислювальних машин (К. Цузе, Д. Атанасов, Г. Айткен). У машині К. Цузе (1938 р.) програма записувалась на перфоровану кінострічку (подібно тому, як це було реалізовано в перших вітчизняних машинах "Урал"). Машина Z1 могла працювати з раціональними числами. Ідеї К. Цузе (наприклад, паралельне програмування) випередили час майже на півстоліття. Д. Атапасов створив і запатентував перші електронні схеми вузлів електронно-обчислювальної машини (ЕОМ) (тригер Бонч-Бруєвича – у 1913 р.) та в 1942 р. побудував ЕОМ, що виконувала додавання і віднімання. Електронна машина Colossus була створена А. Тьюрингом у 1943 р. з функцією дешифрування. У1936 р. англійський математик А. Тьюринг та (незалежно від нього) американський математик і логік Е. Пост висунули й розробили концепцію абстрактної обчислювальної машини. Машина А. Тьюринга – гіпотетичний універсальний перетворювач дискретної інформації, що моделює обчислювальні системи. А. Тьюринг та Е. Пост показали принципову можливість вирішення будь-якої проблеми, що піддається алгоритмізації, за допомогою скінченних автоматів.

Розвиток сучасної обчислювальної техніки прийнято розглядати з погляду зміни поколінь комп'ютерів. Крім елементної бази і часу використання, враховують такі параметри, як швидкодія, архітектура, програмне забезпечення, рівень розвитку зовнішніх пристроїв. Ще одним якісним показником є галузь застосування комп'ютерів.

У розвитку ЕОМ вирізняють **п'ять поколінь комп'ютерів**. В основу класифікації можна покласти елементну базу, за якою будують ЕОМ.

І покоління. 1945-1955 рр. – на електронних лампах. У 1946 р. Дж. Моучлі і Дж.П. Еккерт сконструювали першу електронну обчислювальну машину ENIAC (Електронний обчислювальний інтегратор і калькулятор). В ENIAC електромеханічні реле були замінені на електронні вакуумні лампи (він містив 18 000 вакуумних ламп і 70 000 резисторів). ENIAC виявився універсальною обчислювальною машиною. Він використовувався для розрахунків у галузі атомної енергетики, прогнозів погоди, аеродинаміки тощо.

Класична архітектура ЕОМ і принципи Дж. фон Неймана. У 1945 р. американський математик Дж. фон Нейман, приєднавшись до групи розробників ENIAC, описав архітектуру майбутнього комп'ютера EDV AC. Він запропонував увести до складу комп'ютера спеціальний пристрій для збереження команд і даних – пам'ять – і реалізувати можливість передачі керування від однієї програми до іншої.

У процесі роботи зі створення першої у світі лампової ЕОМ у 1946 р. були сформульовані базові принципи побудови ЕОМ, що актуальні й нині, а саме: обґрунтування переваги використання бінарної системи для подання інформації та збереження програми як набору бітів у тій самій пам'яті, що й інформації, яку ця програма обробляє.

Загальна схема роботи обчислювальної машини може бути подана таким чином: робота виконується за заздалегідь розробленою програмою – послідовністю команд; ці команди виконуються одна за одною, поки не буде виконана остання з них; послідовність виконання команд може бути змінена за допомогою команд передачі управління.

Дж. фон Нейман також запропонував структуру ЕОМ, до основних блоків якої входять пристрій управління і арифметико-логічний пристрій, внутрішня пам'ять, зовнішня пам'ять, пристрої вводу-виводу. Пристрій управління й арифметико-логічний пристрій у сучасних комп'ютерах об'єднані в один блок – процесор.

Інформація зберігається у пам'яті, яка поділяється на оперативну і зовнішню. Оперативна містить проміжні результати і дані, що безпосередньо використовуються, а зовнішня - інформацію, потрібну для подальшого використання. Зовнішні запам'ятовуючі пристрої мають набагато більшу місткість порівняно з оперативними, але з повільнішим доступом.

У 1945 р. фон Нейман підготував звіт, у якому визначив архітектуру й основні принципи роботи комп'ютера:

1. Комп'ютер складається з процесора, пам'яті і зовнішніх пристроїв.

2. Єдиним джерелом активності у комп'ютері є процесор, яким керує програма, що знаходиться в пам'яті.

3. Пам'ять комп'ютера складається з комірок, кожна з яких має свою унікальну адресу. Кожна з комірок зберігає команду програми або елемент даних.

4. У будь-який момент процесор виконує одну команду програми, адреса якої знаходиться в лічильнику команд.

5. Інформація надходить у процесор з пам'яті або від зовнішніх пристроїв. Перетворення інформації відбувається тільки в регістрах процесора.

6. Кожна команда програми містить такі розпорядження:

o з яких комірок взяти інформацію;

o які операції виконати з цією інформацією;

o в які комірки пам'яті помістити отриманий результат;

o як змінити вміст лічильника команд.

Процесор виконує команди програми відповідно до зміни вмісту лічильника команд, поки не дістане команду зупинитися.

Розроблені Дж. фон Нейманом основи архітектури ЕОМ виявилися настільки фундаментальними, що дістали в літературі назву "фон-нейманівська архітектура". У 1951 р. Д. Еккертта Дж. Моучлі побудували UNIVAC – перший комп'ютер, у якому були реалізовані всі принципи фон-нейманівської архітектури, продавши перший екземпляр цієї машини департаменту переписів населення США. UNIVAC став першим американським комерційним комп'ютером. Він коштував мільйони доларів і міг виконувати близько 5000 операцій за одну секунду.

Переважну більшість обчислювальних машин сьогодні становлять фон-нейманівські машини. Очевидно, значне відхилення від фон-нейманівської архітектури відбудеться в результаті розвитку ідеї машин п'ятого покоління, в основі обробки інформації в яких лежать не обчислення, а логічні виведення.

Роботи зі створення обчислювальних машин виконувалися також у Радянському Союзі. Група науковців-кібернетиків під керівництвом академіка **С.О. Лебедєва** (Л.Н . Дашевський, К.О. Шкабара та ін.) розпочала розробку першої вітчизняної ЕОМ МЕЛМ (Мала електронна лічильна машина) у Київському Інституті електротехніки АН УРСР у 1947-1948 рр. У листопаді 1950 р. запрацював макет МЕЛМ, на той час найшвидшої в Європі, а 25 грудня 1951 р. машину було прийнято в експлуатацію. Машина оперувала з 20-цифровими бінарними кодами зі швидкістю 50 операцій на секунду, мала оперативну пам'ять у 100 комірок на електронних лампах.

На початку 50-х років на МЕЛМ розв'язували задачі відомі радянські математики і механіки А.О. Дородніцин, О.А. Ляпунов, О.Ю. Ішлінський, М.В. Келдиш, М.О. Лаврентьєв, Б.В. Гнеденко та ін. На ній працювали перші радянські програмісти М.Р. Шура-Бура, В.С. Королюк, К.Л. Ющенко та ін.

У грудні 1957 р. на базі лабораторії обчислювальної техніки Інституту математики було створено Обчислювальний центр АН УРСР, який очолив **В.М. Глушков**. Від самого початку діяльність центру спрямовувалася на розвиток широкого комплексу фундаментальних і прикладних досліджень у галузі електронної обчислювальної техніки та її застосувань, на розв'язання проблем теоретичної і прикладної кібернетики, прикладної математики. Час від створення Обчислювального центру до його перетворення на Інститут кібернетики (1962 р.) був початковим періодом розвитку кібернетики та інформатики в Україні, а період створення й освоєння МЕЛМ (1948-1953 рр.) - початковим етапом розвитку електронно-обчислювальної техніки в СРСР.

II покоління. 1956-1965 рр. – на напівпровідникових транзисторах. Транзистор був винайдений у 1947 р. співробітниками американської компанії "Белл" У. Шоклі, Дж. Бардіним і У. Браттейном. Порівняно з електронними вакуумними лампами транзистори, що використовували електричні властивості напівпровідників, займали у 200 разів менше місця і споживали в 100 разів менше електроенергії. Тоді з'явилися нові технології збереження інформації на основі феритових сердечників, що давало змогу значно збільшити місткість пам'яті комп'ютера при одночасному зменшенні її розмірів. У 1956 р. у Массачусетському технологічному інституті було створено перший комп'ютер, повністю побудований на транзисторах.

Машинна мова, що застосовувалася в першому поколінні комп'ютерів, була незручною для сприйняття людиною. Щоб подолати це, був розроблений асемблер - мова, в якій для кодування команд застосовувалися мнемонічні позначення, а для даних, що зберігалися в пам'яті, - імена, що відповідали їхньому змісту.

Розширення сфери застосування комп'ютерів потребувало створення мов високого рівня. Одними з перших таких мов стали Фортран (FORmula TRANslation), призначена для складних формульних обчислень, і Кобол (COmmonBussiness Oriented Language), орієнтована на обробку фінансово-економічних даних.

III покоління. 1965-1971 рр. – на інтегральних мікросхемах. Ідея інтегральної мікросхеми - кремнієвого кристала, на якому монтуються мініатюрні транзистори та інші елементи, - була запропонована в 1958 р. інженером компанії Texas Instruments Дж. Кілбі. Він виготовив перший зразок мікросхеми, в якій на кристалі германія знаходилися п'ять транзисторів. У 1959 р. незалежно від нього Р. Нойс, який згодом заснував компанію Інтел, розробив аналогічну інтегральну мікросхему на кристалі кремнію. Мікросхеми працювали значно швидше ніж транзистори, були компактнішими і споживали набагато менше електроенергії. На основі цієї технології стали розроблятися мікросхеми, що містять сотні і тисячі елементів.

У 1964 р. компанія IBM випустила комп'ютер IBM System 360, побудований на основі інтегральних мікросхем. З випуску комп'ютерів цієї серії почалось масове виробництво обчислювальної техніки.

Паралельно з розвитком обчислювальної техніки вдосконалювалося і програмне забезпечення. У 1964 р. з'явилася мова програмування Бейсік (BASIC - Beginner's All-Purpose Symbolic Instruction Code), що дало змогу людям легко опанувати навички програмування. У 1970 р. швейцарець Н. Вірт розробив мову програмування Паскаль, спеціально призначену для навчання структурному програмуванню.

IV покоління. З 1971 р.– на мікропроцесорах. У 1969 р. компанія Intel випустила мікропроцесор - інтегральну мікросхему, на якій знаходився пристрій обробки інформації з власною системою команд. У 1971 р. був створений чотирирозрядний мікропроцесор I4004 - основа калькуляторів, які можна програмувати. Використання мікропроцесорів значно спростило конструкцію комп'ютерів. Практично відразу на їх основі з'явилися персональні комп'ютери, характерною рисою яких стали низька ціна і невеликі розміри.

29 жовтня 1969 р. відбулося випробування першої глобальної мережі ARPANet, що об'єднувала 4 американські університети, а в 1974 р. В. Серф запропонував змінити назву ARPANet на Internet.

Засновники фірми Apple С. Джобс і С. Возняк зібрали першу модель персонального комп'ютера в 1976 р. А в 1981 р. найбільша комп'ютерна компанія IBM представила свій перший персональний комп'ютер - IBM PC. Упродовж двох років було продано більше п'яти мільйонів цих комп'ютерів. Водночас компанія Microsoft почала випуск програмного забезпечення для IBM PC.

У 1995 р. Intel випустив новий процесор Pentium Pro, що містив 5,5 млн. транзисторів. Персональні комп'ютери з процесорами Pentium здатні виконувати 400 мільйонів операцій за секунду.

V покоління. Швидкодія комп'ютерів з фон-нейманівською архітектурою обмежена швидкістю світла, з якою електрони рухаються усередині схем ЕОМ. Тому дослідники піддають ревізії ці принципи структури ЕОМ. Однак підходи, засновані на заміні одного чи двох принципів фон Неймана, не дали бажаних результатів. Причина невдач крилася в органічній єдності всіх трьох принципів. Найуспішнішу спробу ревізії традиційних принципів архітектури ЕОМ здійснив у середині 70-х років радянський учений-академік В.М. Глушков.

Список програмного забезпечення для комп'ютерного моделювання:

1 Безкоштовні або з відкритим вихідним кодом:

- Advanced Simulation Library - програмне забезпечення з відкритим вихідним кодом для апаратного прискорення фізичної симуляції;
- Algodoo - 2D симулятор фізики;
- ASCEND - середовище моделювання з відкритим вихідним кодом на основі рівнянь;
- Cantera - пакет хімічної кінетики;
- Celestia - тривимірна астрономічна програма;
- CP2K - програма ab-initio молекулярної динаміки з відкритим вихідним кодом;
- DWSIM - симулятор хімічних процесів з відкритим вихідним кодом, сумісний з CAPE-OPEN;
- Elmer - програмне забезпечення для фізичного моделювання з відкритим вихідним кодом для Windows/Mac/Linux;
- Facsimile - безкоштовна бібліотека імітації дискретних подій з відкритим вихідним кодом;
- FreeFem ++ - безкоштовне програмне забезпечення для аналізу кінцевих елементів (FEA) з відкритим вихідним кодом;
- Freemat - безкоштовне середовище для швидкого проєктування, наукового прототипування і обробки даних однією мовою з MATLAB і GNU Octave;
- Galatea - багатоагентна багатомовна платформа для симуляції;
- GNU Octave - програмне забезпечення для математичного моделювання і симуляції з відкритим вихідним кодом, дуже схоже на MATLAB та Freemat;
- OpenModelica - середовище моделювання з відкритим вихідним кодом, засноване на Modelica - відкритому стандарті програмного забезпечення для моделювання;
- OpenFermion Cirq - перша платформа з відкритим кодом для перетворенняпитань хімії та матеріалознавства в квантові схеми;
- JModelica.org - безкоштовна програмна платформа з відкритим вихідним кодом, заснована на мові моделювання Modelica;
- Mobility Testbed - мультиагентний випробувальний стенд з відкритим вихідним кодом для алгоритмів координації транспорту;
- NetLogo - програмне забезпечення для багатоагентного моделювання з відкритим вихідним кодом;
- ns-3 - симулятор мережі з відкритим вихідним кодом;
- OpenFOAM - програмне забезпечення з відкритим вихідним кодом, що використовується для обчислювальної гідродинаміки (або CFD);
- OpenEaagles - багатоплатформенне середовище моделювання для створення прототипів та додатків моделювання;

- Open Source Physics - програмний проєкт на Java з відкритим вихідним кодом для викладання і вивчення фізики;
- OpenSim - програмна система з відкритим вихідним кодом для біомеханічного моделювання;
- Physics Abstraction Layer - фізичний пакет з відкритим вихідним кодом;
- Project Chrono - фреймворк для фізичного моделювання з відкритим вихідним кодом;
- SageMath - система для експериментів з алгебри та геометрії через Python;
- Scilab - безкоштовне програмне забезпечення з відкритим вихідним кодом для чисельних розрахунків і моделювання, схоже на MATLAB/Simulink;
- Simantics System Dynamics -система для моделювання великих ієрархічних моделей з багатовимірними змінними, створеними традиційним способом з використанням графіків запасів, потоків і діаграм причинно-наслідкових зв'язків;
- SimPy - пакет імітації дискретних подій з відкритим вихідним кодом, заснований на Python;
- Simulation of Urban MOility - пакет моделювання трафіку з відкритим вихідним кодом;
- SOFA - фреймворк з відкритим вихідним кодом для фізичного моделювання з акцентом на медичне моделювання;
- Код SU2 - платформа з відкритим вихідним кодом для моделювання динаміки рідини і оптимального проєктування форми;
- Step - програма симуляції двовимірної фізики з відкритим вихідним кодом (KDE);
- Tortuga - програмне середовище з відкритим вихідним кодом для моделювання дискретних подій в Java;
- UrbanSim - програмне забезпечення з відкритим вихідним кодом для моделювання землекористування, транспорту та екологічного планування.

2 Власне та комерційне програмне забезпечення:

- Adaptive Simulations - хмарні і повністю автоматизовані симуляції CFD;
- AGX Dynamics -програма для моделювання багатотільних та фізичних явищ в реальному часі;
- 20-sim - програмне забезпечення для багатодоменного моделювання на основі графа зв'язків;
- Actran - програмне забезпечення для моделювання методом кінцевих елементів для аналізу акустичної поведінки механічних систем і деталей;
- ADINA - програмне забезпечення для інженерного моделювання для структурних, рідинних, теплообмінних і мультифізичних завдань;
- ACSL і acslX - просунута мова безперервного моделювання;
- Simcenter Amesim - платформа для аналізу багатодоменних інтелектуальних систем, а також для прогнозування і оптимізації міждисциплінарної продуктивності,розроблено Siemens PLM Software;
- ANSYS -програма інженерного моделювання;
- AnyLogic - багатофункціональний інструмент імітаційного моделювання для бізнесу і науки,розроблено компанією AnyLogic;
- APMonitor - інструмент для динамічного моделювання, валідації та оптимізації багатодоменних систем з інтерфейсами до Python і MATLAB;
- Arena - програмне забезпечення для моделювання дискретних подій на основі блок-схеми, розроблене компанією Rockwell Automation;
- Automation Studio - програмне забезпечення для моделювання систем електроживлення, електричних систем і систем управління, розроблено Famic Technologies Inc.;
- CENOS Platform - засноване на FEA програмне забезпечення для настільного моделювання процесів індукційного нагріву різних сталей і сплавів: гарт, відпал, пайка, зварювання, термоусадка, з'єднання, попередній нагрів, кування;

- Chemical WorkBench - програмний інструмент для моделювання хімічної кінетики, розроблений Kintech Lab;
- CircuitLogix - програмне забезпечення для моделювання електроніки, розроблене компанією Logic Design Inc.;
- COMSOL Multiphysics - пакет програмного забезпечення для аналізу, розрахунку та моделювання, заснований на кінцевих елементах, для різних фізичних і технічних додатків, особливо пов'язаних явищ або фізики;
- CONSELF - заснована на браузері платформа моделювання CFD і FEA;
- DX Studio - набір інструментів для симуляції та візуалізації;
- Dymola - програмне забезпечення для моделювання та симуляції на основі мови Modelica;
- Ecolego - програмний інструмент для моделювання динамічних, детермінованих і ймовірнісних моделей;
- EcosimPro - програмне забезпечення для безперервного та дискретного моделювання і симуляції;
- Enterprise Architect - інструмент для поведінкового моделювання UML в поєднанні з призначеним для користувача інтерфейсом Win32;
- Enterprise Dynamics - програмна платформа для моделювання, розроблена компанією INCONTROL Simulation Solutions;
- ExtendSim - програмне забезпечення для моделювання дискретних подій, безперервного, дискретного і агентного моделювання;
- FEATool Multiphysics - набір інструментів для фізики кінцевих елементів і моделювання PDE для MATLAB;
- Flexsim - програма для моделювання дискретних подій;
- Fluent Inc. - програмне забезпечення для моделювання потоку рідини, турбулентності, теплообміну і реакцій для промислового застосування;
- GoldSim - програмне забезпечення для моделювання динаміки систем і дискретних подій, вбудоване в платформу Монте-Карло;
- HyperWorks - мультидисциплінарне програмне забезпечення для моделювання;
- IDA ICE - програмне забезпечення на основі рівнянь (DAE) для моделювання продуктивності будівлі;
- IES Virtual Environment (IESVE) - комплексне програмне забезпечення для аналізу і моделювання продуктивності будівлі;
- Isaac Dynamics - програмне забезпечення для динамічного моделювання процесів на звичайних і поновлюваних електростанціях;
- iThink - програмне забезпечення для системної динаміки і моделювання дискретних подій для бізнес-стратегії, державної політики та освіти,розроблено компанією ISEE Systems;
- JMAG - програмне забезпечення для моделювання та проєктування електричних пристроїв;
- Khimera - програмний інструмент для моделювання хімічної кінетики, розроблений Kintech Lab;
- Lanner WITNESS - платформа для моделювання окремих подій для моделювання процесів і експериментів;
- Lanner L-SIM Server -програма на основі Java для імітації моделей процесів на основі BPMN2.0;
- MADYMO - програмне забезпечення для автомобільної і транспортної безпеки, розроблене Нідерландської організацією прикладних наукових досліджень;
- Maple - універсальна система комп'ютерної алгебри, розроблена компанією Waterloo Maple Inc.;
- MapleSim - багатодоменний інструмент для моделювання і симуляції, розроблений Waterloo Maple Inc.;

- MATLAB - інструмент для програмування, моделювання і симуляції, розроблений MathWorks;
- Mathematica - обчислювальна програма, заснована на символічній математиці, розроблена Wolfram Research;
- Micro Saint Sharp - універсальний програмний інструмент для моделювання дискретних подій, що використовує метод графічної блок-схеми на мові C #, розроблений Alion Science and Technology;
- ModelCenter - платформа для інтеграції сторонніх інструментів/скриптів моделювання і симуляції, автоматизації робочих процесів, а також аналізу і оптимізації міждисциплінарного проєктування від Phoenix Integration;
- NEi Nastran - програмне забезпечення для інженерного моделювання напружень, динаміки і теплообміну в конструкціях;
- NI Multisim - електронна схема захоплення і симуляції програми;
- Plant Simulation - програмне забезпечення для моделювання, оптимізації ліній і процесів, розроблене Siemens PLM Software;
- PLECS - інструмент для системного моделювання електричних ланцюгів,розроблено Plexim;
- PRO/II - програмне забезпечення для стаціонарного моделювання хімічних процесів, широко використовується на нафтопереробних і газопереробних заводах;
- roject Team Builder - симулятор управління проєктами, який використовується для навчання;
- ProLB - програмне забезпечення для комп'ютерного моделювання гідродинаміки;
- PSF Lab - розраховує функцію розсіювання точок оптичного мікроскопа при різних умовах візуалізації на основі суворої векторної моделі;
- RoboLogix - програмне забезпечення для моделювання робототехніки, розроблене компанією Logic Design Inc.;
- Ship Simulator - комп'ютерна гра з моделювання транспортних засобів VSTEP, яка імітує маневрування різних суден в різних умовах;
- Simcad Pro - програмне забезпечення для симуляції процесу з моделлю On-The-Fly, яка змінюється під час симуляції,розроблено CreateASoft, Inc. Чикаго, США;
- SimEvents - частина MathWorks, яка додає моделювання дискретних подій в середовище MATLAB/Simulink;
- SimScale - вебплатформа для моделювання з можливостями CFD, FEA і термодинаміки;
- SIMUL8 - програмне забезпечення для моделювання на основі дискретних подій або процесів;
- SimulationX - програмне забезпечення для моделювання та симуляції на основі мови Modelica;
- Simulink - інструмент для моделювання блок-схем, електромеханічних систем і машин від MathWorks;
- SRM Engine Suite - інженерний інструмент, який використовується для моделювання викидів палива, згоряння і вихлопних газів в двигунах з мікросхемою;
- STELLA - програмне забезпечення для системної динаміки і моделювання дискретних подій для бізнес-стратегії, державної політики та освіти,розроблено компанієюISEE Systems;
- TRNSYS - програмне забезпечення для динамічного моделювання систем відновлюваної енергії, систем опалення, вентиляції та кондиціонування повітря, використання енергії в будівлях, а також пасивних і активних сонячних систем;
- Vensim - програмне забезпечення для системної динаміки і безперервного моделювання для додатків бізнесу і державної політики;
- VisSim - система для моделювання і опціональної генерації C-коду електричних, технологічних, керуючих, біомедичних, механічних і UML-систем карт станів;

- Vortex (програмне забезпечення) - повна платформа моделювання, що включає фізичний двигун реального часу для динаміки твердого тіла, генератор зображень, інструменти робочого столу (редактор і плеєр) і багато іншого. Також існує Vortex Studio Essentials, обмежена безкоштовна версія;
- Wolfram SystemModeler - програмне забезпечення для моделювання та симуляції на основі мови Modelica;
- Working Model - 2D динамічний симулятор з підключеннями до SolidWorks (демоверсія з відключеним SAVE безкоштовна);
- VisualSim Architect - електронне програмне забезпечення системного рівня для моделювання та симуляції електронних систем, програмно-апаратних засобів і напівпровідників;
- zSpace -програма для створення додатки для природничих наук.

8.5 Висновки по лекції

Імітаційне моделювання– це метод дослідження, заснований на тому, що система, яка вивчається, замінюється імітатором і з ним проводяться експерименти з метою отримання інформації про цю систему. Експериментування з імітатором називають **імітацією** (імітація – це дослідження сутності явища, не вдаючись до експериментів на реальному об'єкті).

Основною перевагою імітаційних моделей в порівнянні з аналітичними є можливість вирішувати завдання виняткової складності з урахуванням випадкових чинників.

Метод імітаційного моделювання вдається успішно реалізувати за допомогою ЕОМ. В основі імітаційного моделювання лежить **метод статистичних випробувань** (метод Монте–Карло), який базується на використанні випадкових чисел, тобто можливих значень деякої випадкової величини з заданим розподілом ймовірностей. **Метод статистичних випробувань** – це числовий метод математичного моделювання випадкових величин, який передбачає безпосереднє включення випадкового фактору в процес моделювання і є його істотним елементом. Вплив випадкових факторів на систему моделюється за допомогою випадкових чисел.

У розвитку ЕОМ вирізняють **п'ять поколінь комп'ютерів**. В основу класифікації можна покласти елементну базу, за якою будують ЕОМ.

СПИСОК ДЖЕРЕЛ ІНФОРМАЦІЇ

Література основна

1 Вітлінський В.В., Терещенко Т.О., Савіна С.С. Економіко-математичні методи та моделі: оптимізація : навч. пос. Київ : КНЕУ, 2016. 303 с.

2 Зеленський К.Х., Кіт Г.В., Чумаченко О.І. Комп'ютерне моделювання систем: навч. пос. Київ: Університет «Україна», 2014. 315 с.

3 Моделювання та оптимізація систем : підручник / В.М.Дубовой, Р.Н.Квєтний, О.І.Михальов, А.В. Усов. Вінниця : ПП «ТД«Еднльвейс», 2017. 804 с.

4 Павленко П.М., Філоненко С.Ф., Чередніков О.М., Трейтяк В.В. Математичне моделювання систем і процесів: навч. пос. Київ: НАУ, 2017. 392 с.

Література додаткова

5 Барабаш М.С., Кір’язєв П.М., Лапенко О.І., Ромашкіна М.А. Основи комп’ютерного моделювання : навч. посіб. Київ: НАУ, 2019. 500 с.

6 Бредюк В.І., Джоші О.І. Економіко-математичне моделювання в середовищі табличного процесора MS Excel : навч. посіб. Рівне : НУВГП, 2015. 242 с.

7 Харченко В.П., Шмельова Т.Ф., СікірдаЮ.В. Прийняття рішень в соціотехнічних системах : монографія. К. : Національний авіаційний університет, 2016. 308 с.

8 Socio-Technical Decision Support in Air Navigation Systems : Emerging Research and Opportunities : manuscript / T. Shmelova, Yu. Sikirda, N. Rizun, A.-B. M. Salem, Yu. Kovalyov. IGI Global book series Advances in Mechatronics and Mechanical Engineering (AMME). USA, Hershey : IGI Global, 2018. 305 p. DOI: 10.4018/978-1-5225-3108-1