

СИСТЕМНИЙ АНАЛІЗ

Основні поняття системного аналізу

Поняття системи

Система – ціле, складене з частин. Категорія «система» означає цілісну сукупність взаємозв'язаних частин. В теорії систем існує чимало підходів до розуміння сутності системи. Наведемо деякі з них:

Система (від грец. Σύστημα – складене з частин) – сукупність сутностей (об'єктів) і зв'язків між ними, виділених з середовища на певний час і з певною метою.

Система – цілісна множина об'єктів (елементів), пов'язаних між собою взаємними відносинами.

Система – організована безліч структурних елементів, що взаємопов'язані і виконують певні функції.

Система – будь-яка сукупність елементів (підсистем), об'єднаних між собою в єдине ціле процесами взаємодії (матеріально-інформаційного обміну) для реалізації спільної функції (досягнення спільної мети).

Системою називається будь-який об'єкт, що складається з елементів та зв'язків між ними.

Таким чином, **система** – це сукупність елементів, які певним чином взаємопов'язані, утворюють цілісність, а також взаємодіють між собою для виконання заданих цільових функцій.

У свою чергу й елементи системи можна розглядати як системи, якщо визначити інший критерій декомпозиції.

Систему S будемо розглядати у вигляді

$$S = (M, X_S, X_{\bar{S}}, F)$$

де M – множина елементів системи,

X_S – множина зв'язків між елементами системи,

$X_{\bar{S}}$ – множина зв'язків між елементами системи та зовнішнім середовищем,

F – множина нових (системних) функцій, властивостей, призначень.

Системний аналіз займається питаннями вивчення і моделювання систем, їхньої структури, функцій та поведінки. **Методи** системного аналізу допомагають розуміти, як окремі компоненти системи взаємодіють один з одним, і як ці взаємодії впливають на

загальну ефективність системи. Системний аналіз застосовується в різних галузях, таких як інженерія, управління, економіка, охорона здоров'я та інформаційні технології.

Системний аналіз це сукупність формалізованих, слабо формалізованих і неформалізованих методів і процедур, що дозволяють реалізувати системний підхід до управління системами і функціонуванням складних систем на різних етапах життєвого циклу.

Теоретико-множинне визначення системи

Система – це множина об'єктів, між якими існують певні відношення, а також їх атрибути. Під об'єктами розуміють компоненти (елементи) системи. Це, наприклад, підсистеми (тобто може існувати ієрархія підсистем) або окремі об'єкти системи.

Атрибути – це властивості об'єктів.

Відношення задають певний закон, за яким визначається деяке відображення в одній і тій же множині об'єктів. Згідно з цим визначенням поняття множина і елемент є аксіоматичними. Таким чином, система S задається парою елементів:

Таким чином, систему S можна задати парою елементів:

$$S=(X_s, R_s),$$

де X_s і R_s – множини відповідно елементів (об'єктів) системи і відношень між ними. Відношення визначають взаємодію між об'єктами.

У загальному випадку n -відношення R в множинах $X_1, X_2, \dots, X_n$ є деякою підмножиною декартового добутку $X_1 \times X_2 \times \dots \times X_n$, який складений з n -вимірних наборів виду $(x_1, x_2, \dots, x_n)$, де $x_i \in X_i, i=1, 2, \dots, n$.

Якщо відношення R в окремому випадку задається, наприклад, деякою функцією, яка визначає зв'язок між певним елементом $x \in X$ і певним елементом y підмножини Y , то $f: X \rightarrow Y$, тобто вважаємо, що функція f перетворить значення з множини X у значення підмножини Y . Для функції f множина X – це область визначення, а підмножина Y – область значень функції. Функцію f можна розглядати як множину впорядкованих пар елементів (x, y) , де $y=f(x)$.

Що стосується **атрибутів системи**, то вони подібні до **функцій**, визначених на підмножині об'єктів. Відмінність атрибутів від функцій полягає в тому, що два різних атрибути з точки зору поняття функції можуть бути однаковими. **Атрибут** A задається парою елементів (i, f) , де i – ім'я атрибута, а f – функція, визначена на підмножині об'єктів. У динамічних об'єктів атрибут також може бути функцією від часу t .

Вивчаючи систему більш глибоко, усвідомлюємо, що вона може **складатися з підсистем** або бути одним з елементів більшої системи, тобто може існувати **ієрархія систем**. Наприклад, двигун є підсистемою автомобіля, який, у свою чергу, є підсистемою транспортного потоку магістралі.

Технології системного аналізу

Технології системного аналізу – це сукупність методів, моделей, інструментів і процедур, що застосовуються для дослідження, проектування, оптимізації та управління складними системами різної природи (технічними, інформаційними, організаційно-економічними, соціотехнічними) в умовах невизначеності та багатокритеріальності.

Застосування технологій системного аналізу дозволяє формалізувати складні проблеми; підвищити обґрунтованість управлінських рішень; зменшити ризики проектування; забезпечити цілісне бачення системи; підвищити ефективність функціонування складних систем.

Моделювання

Модель (лат. міра, зразок) – це якийсь об’єкт, який в певних умовах замінює об’єкт-оригінал, відтворюючи властивості і характеристики оригіналу, які цікавлять нас, маючи при цьому суттєві переваги використання (наочність, видимість, доступність випробувань та ін.)

Моделювання – це дослідження будь-яких процесів, явищ або систем (об’єктів) шляхом побудови і вивчення їх моделей; використання моделей для визначення або уточнення характеристик і раціоналізації способів побудови знову створюваних об’єктів.

Моделювання як метод наукового пізнання виникло у зв’язку з необхідністю розв’язування завдань, які з тих чи інших причин не можуть бути розв’язані безпосередньо. Вони виникають у випадках, коли об’єкт недосяжний за своєю природою, коли він ще не існує і потрібно обрати кращий варіант його створення, коли дослідження реального об’єкта вимагає багато часу, економічно невигідне та ін.

Модель – це заміслювач об’єкта дослідження, що знаходиться з ним в такій відповідності, яка дозволяє отримати нове знання про цей об’єкт.

Для оцінки ступеня відповідності **моделі і оригіналу** вводиться поняття «адекватність моделі». Адекватною називають таку модель, для якої вимоги повноти

точності і істинності моделі виконуються не взагалі в повній мірі, а тільки в тій мірі, яка призводить до досягнення мети.

Таким чином, **модель** – це реально існуюча або абстрактна система, яка, замінюючи і відображаючи в пізнавальних процесах іншу систему – оригінал, перебуває з нею у відношенні схожості.

Стани системи (моделі)

Станом динамічної системи (моделі) в деякий момент часу t називається множина значень всіх її параметрів (змінних), вимірених одночасно у цей момент. При зміні значення хоча б одного параметра системи в наступний момент часу говорять, що стан системи змінився. Стан системи зручно розглядати як точку в багатовимірному просторі. Множина всіх можливих станів системи називається простором станів системи.

Систему можна розділити на елементи різними способами залежно від формулювання завдання, мети і її уточнення в процесі системного аналізу.

Для дослідження системи із n взаємозв'язаних елементів, необхідно вивчити $n(n-1)$ зв'язків. Якщо характеризувати ці зв'язки найпростішим чином, тобто відмічати в будь-який момент часу t_i тільки наявність або відсутність впливу, то **загальна кількість станів системи** буде дорівнювати $2^{n(n-1)}$.

Структура системи

Структура системи – це організована сукупність зв'язків між її елементами. Під таким зв'язком розуміють можливість впливу одного елементу системи на інший. Середовище – це сукупність елементів зовнішнього світу, які не входять до складу системи, але впливають на її поведінку або властивості.

Система є **відкритою**, якщо існує зовнішнє середовище, яке впливає на систему, і **закритою**, якщо зовнішнє середовище відсутнє або не враховується, у зв'язку з поставленими цілями досліджень.

Структура системи – сукупність зв'язків і відношень між частинами цілого, необхідних для досягнення мети системи.

На рис. 1.1-1.4 наведено базові топології структур систем (представлені у вигляді орієнтованих графів).

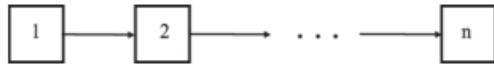


Рис. 1.1. Структура лінійного типу

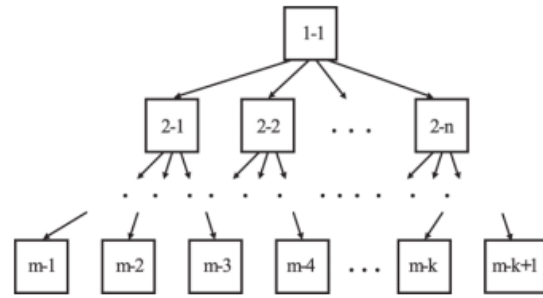


Рис. 1.2. Структура ієрархічного типу

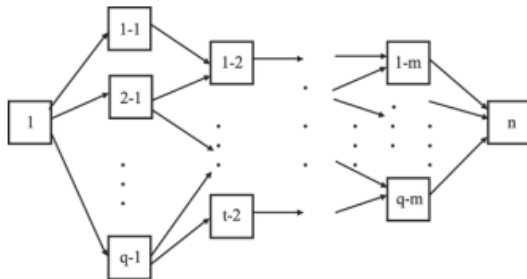


Рис. 1.3. Структура мережевого типу

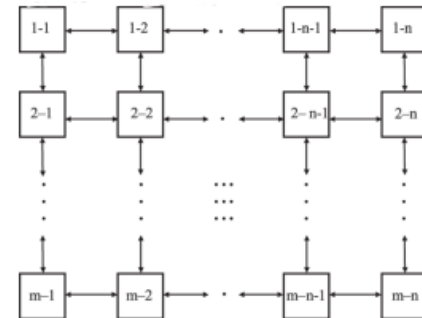
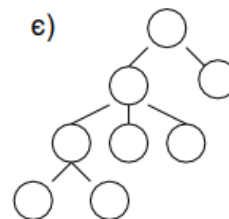
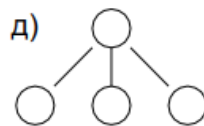
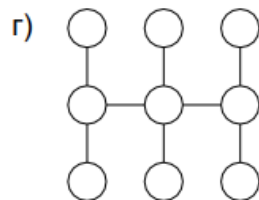
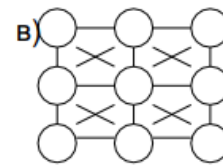
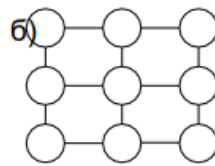
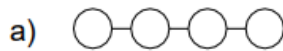


Рис. 1.4. Структура матричного типу

Крім згаданих структур використовуються і інші, які утворюються за допомогою комбінацій чотирьох основних топологій структур.

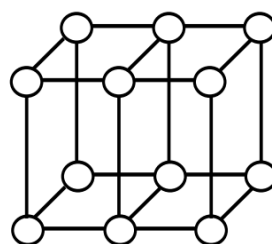
Приклади різних типів структур системи:



а) лінійна; б) матрична; в) мережева; г) кістякова;

д) ієрархічна; є) деревовидна.

Структура **кристалічного** (просторово-матричного) типу:



Структура системи є **зв'язною**, якщо можливий обмін ресурсами між **будь-якими двома підсистемами системи** (тобто граф, яким представляються зв'язки у системі, є зв'язним).

Якщо структура або елементи системи погано (частково) визначені, то така множина об'єктів називається погано або слабо структурованою.

Декомпозиція

Декомпозиція – це **процедура формального поділу цілого** на частини для подальшого аналізу цих частин.

Декомпозиція системи – це процес **розбиття складної системи** на менші, більш керовані частини або підсистеми. Це дозволяє краще розуміти, аналізувати та управляти системою. Основні кроки декомпозиції включають визначення меж системи, ідентифікацію основних компонентів, поділ на підсистеми, аналіз взаємодії та оцінку функціональності. Декомпозиція дозволяє спростити складні процеси, робить систему більш прозорою та легшою для управління.

Приклади:

1. Будівництво будинків. Процес будівництва будинку розбивається на окремі етапи: проектування, фундамент, стіни, дах, внутрішні роботи і т.д. Кожен з цих етапів є підсистемою, яка виконується окремо, але в кінцевому результаті утворює єдину систему – будинок.
2. Розробка програмного забезпечення. Створення складного програмного забезпечення розбивається на декілька модулів або компонентів, кожен з яких відповідає за певну функціональність. Наприклад, користувацький інтерфейс, база даних, серверна частина тощо.
3. Автомобільна промисловість. Виробництво автомобіля розбивається на створення окремих деталей і компонентів, таких як двигун, шасі, кузов, електроніка і т.д. Кожна з цих частин виготовляється окремо, але в кінцевому результаті збирається в єдине ціле.
4. Освітні програми. Навчальний процес у школі або університеті розбивається на окремі курси та предмети. Кожен курс є окремою підсистемою, що складається з модулів, уроків або лекцій.

5. Проектний менеджмент. Великі проекти часто розбиваються на менші завдання або підпроекти. Це допомагає краще керувати проектом, контролювати строки виконання та ресурси.

6. Бізнес-процеси. Організаційні процеси в компаніях розбиваються на окремі функціональні підрозділи, такі як маркетинг, продажі, виробництво, обслуговування клієнтів. Кожен з цих підрозділів є підсистемою, що вносить свій вклад у загальну роботу компанії.

Декомпозиція дозволяє спростити складні системи та ефективніше ними управляти. Декомпозиція системи застосовується в багатьох сферах нашого життя.

Аналіз і синтез в системних дослідженнях

Аналіз – це процес визначення властивостей, притаманних системі. Суть аналізу полягає в поділі цілого на частини, складного – на сукупність простіших компонент.

Синтез – об'єднання різних елементів об'єкта в єдине ціле.

Аналіз і синтез – це два ключових методи, що використовуються в **дослідженнях систем** для розуміння їх структури, функціонування та взаємодії компонентів. Аналіз полягає у розчленуванні складної системи на її складові частини, щоб краще зрозуміти, як кожна з цих частин функціонує окремо. Цей метод включає **декомпозицію**, вивчення **функцій** і ідентифікацію **зв'язків** між компонентами системи. Синтез, навпаки, полягає у об'єднанні окремих частин або підсистем у єдине ціле з метою **оцінки їх колективної поведінки та функціональності**. Цей метод включає інтеграцію, оцінку взаємодії та системне моделювання.

Використання аналізу та синтезу дозволяє отримати всебічне розуміння системи, оцінити її ефективність, визначити можливі проблеми та розробити шляхи їх вирішення. Ці методи широко використовуються у різних галузях, включаючи інженерію, економіку, біологію та соціальні науки.

Аналіз і синтез допомагають розуміти об'єкти та системи. Аналіз розпочинається з **поділу об'єкта на частини**, тоді як синтез передбачає **розгляд об'єкта як єдиного цілого або як частини макросистеми**. Далі, в аналізі пояснюється **поведінка кожної частини окремо**, а в синтезі – **поведінка об'єкта в межах макросистеми**. Знання про окремі частини в процесі аналізу перетворюється на розуміння цілого, тоді як у синтезі розуміння макросистеми використовується для пояснення її частин шляхом розкриття їхніх функцій. На завершальному етапі аналіз дозволяє розкрити **структуру системи та її принципи роботи**, забезпечуючи отримання знань. Водночас синтез демонструє, чому система має саме такий вигляд, і як вона **функціонує**, приводячи до розуміння. Таким чином, продуктом аналізу є знання, а продуктом синтезу – розуміння.

Властивості системи

Будь-яка система має такі властивості:

- відокремленість від навколишнього середовища;
- наявність кордонів;
- характер зв'язків з навколишнім середовищем;
- структурованість – наявність елементів і зв'язків між ними;
- динамічність – здатність змінюватися в часі;
- емерджентність – поява таких властивостей системи, які не були у її елементів;
- ієрархічність – така структура, коли кожна частина системи теж являє собою систему, що складається з елементів;
- підпорядкованість функціонування всіх елементів системи одній меті.

Для всіх визначень системи спільним є те, що **система** – цілісний комплекс взаємозв'язаних елементів, який має певну структуру і взаємодіє із зовнішнім середовищем.

Життєвий цикл системи

Життєвим циклом системи називають період часу від моменту появи потреби в створенні системи до моменту її утилізації.

Етапи життєвого циклу системи:

- 1) етап появи функціональної потреби в системі;
- 2) етап формування системи;
- 3) етап нормального функціонування, зростання і розвитку системи;
- 4) етап регресу (загибелі системи);
- 5) етап формування утилізованої системи для досягнення інших цілей.

Управління системою

Види узагальненої мети системи:

- досягнення певного кінцевого стану;
- реалізація необхідного порядку зміни станів (необхідний рух);

– забезпечення необхідного напрямку руху системи без конкретизації кінцевого стану системи.

Закон управління – це певне правило вироблення послідовності управляючих дій, що забезпечують досягнення керованою системою заданої мети.

Цільова функція управління - це показник якості управління.

Етапи програмно-цільового методу управління:

1) вироблення програмного управління для забезпечення програмного руху об'єкта управління до мети;

2) вироблення коригуючого управління, що зводить відхилення результату реального управління від програмного руху до допустимого значення.

Способи опису систем

Різноманітні описи системи відображають певні групи їх **властивостей** і дозволяють виявити **впорядкованість, структурованість і функціональну організованість** системи. Будь-яка система або об'єкт передусім цікаві своїм **призначенням, місцем**, яке вони займають серед інших систем і об'єктів в навколишньому світі, своєю **функцією**. Тому для характеристики системи передусім повинен бути одержаний **функціональний опис**, який дозволяє оцінити призначення системи, **її відношення до інших систем**, її контакти з навколишнім світом, напрями можливих функціональних змін. **Функціональний опис пов'язує зовнішні впливи на систему з її реакцією, відповіддю, поведінкою, дією на елементи системи.**

Функціональний опис **може задаватися** деяким оператором в алгебраїчній, логічній, диференціальній, інтегрально-диференціальній формі, який входить в скалярне, векторне або матричне рівняння. Оператор складається на основі вимірювання зовнішніх характеристик (принцип чорного ящика: вивчення зв'язку – вплив – реакція) або на основі знань про будову системи. Функціональний опис виходить із **цільових функцій системи.**

Послідовність дій при виконанні системою деякої функції відображає зміст закону поведінки, яка залежить від процесів, які протікають в системі, які протікають в середині системи (закони внутрішньої поведінки), і від процесів, в які залучена вся система в рамках метасистеми (закони зовнішньої поведінки). Кожна підсистема сама містить набір елементів, що виконують свої частки функції, тому закони внутрішнього функціонування системи одночасно є законами зовнішнього функціонування для будь-якої підсистеми цієї системи – закони зовнішнього функціонування першого нижнього рівня.

Уявлення про будову системи дає її **морфологічний опис (МО)**, що дозволяє виділити основні елементи, зв'язки, визначити тип структури. Можливими є три варіанти представлення морфологічного опису: з позицій теоретико-множинного підходу, у вигляді структури та у вигляді відповідної матриці суміжностей.

Опис (специфікація) системи – це ідентифікація елементів та підсистем, які визначають систему, їх взаємозв'язків, цілей, функцій та ресурсів, тобто це опис допустимих станів системи.

Для опису системи важливо знати, яка її структура, які її функції і зв'язки із зовнішнім середовищем.

Морфологічний (структурний або топологічний) опис системи – це опис будови або структури системи, або опис сукупності елементів цієї системи і необхідного для досягнення мети набору відношень між цими елементами.

Функціональний опис системи – опис законів функціонування, еволюції системи, алгоритмів її поведінки, «роботи».

Інформаційний опис системи – це опис інформаційних зв'язків, як системи із зовнішнім середовищем, так і підсистем із системою.

Класифікація систем

З погляду **характеру зв'язків** параметрів системи із навколишнім середовищем розрізняють:

- закриті системи – будь-який обмін енергією, речовиною і інформацією з навколишнім середовищем відсутній;
- відкриті системи – вільно обмінюються енергією, речовиною і інформацією з навколишнім середовищем. У відкритих системах можуть відбуватися явища ускладнення або спонтанного виникнення порядку.

За **рангом** системи розрізняють як:

- підсистему – система, що є частиною іншої системи і здатна виконувати відносно незалежні функції, має підцілі, спрямовані на досягнення загальної мети системи;
- надсистему – більш велика система, частиною якої є дана система.

За **формою подання** системи поділяють на абстрактні, логічні та символічні.

Абстрактні системи – це уможливлене подання образів або моделей матеріальних систем, які поділяються на описові (логічні) і символічні (математичні).

Логічні системи є результатом дедуктивного або індуктивного подання матеріальних систем. Їх можна розглядати як системи понять і визначень (сукупність уявлень) про структуру, про основні закономірності станів і про динаміку матеріальних систем.

Символічні системи становлять собою формалізацію логічних систем, вони поділяються на класи:

- матеріальні;
- статичні математичні системи або моделі, які можна розглядати як опис засобами математичного апарату стану матеріальних систем (рівняння стану);
- динамічні математичні системи або моделі, які можна розглядати як математичну формалізацію процесів матеріальних (або абстрактних) систем;
- квазістатичні (квазідинамічні) системи, що знаходяться в нестійкому положенні між статикою і динамікою, які за одних дій поведуться як статичні, а за інших – як динамічні.

За **походженням** розрізняють системи (елементів, зв'язків, підсистем):

- штучні (знаряддя, механізми, машини, автомати, роботи і т. ін.);
- природні (живі, неживі, екологічні, соціальні і т. ін.);
- віртуальні (уявні, хоча вони в дійсності реально не існують, але функціонують так само, як і в разі, якщо б реально існували);
- змішані (економічні, біотехнічні, організаційні тощо).

За **характером змінних** системи можуть бути:

- з якісними змінними (що мають тільки змістовний опис);
- з кількісними змінними (що мають дискретно або безперервно описувані кількісно змінні);
- системи змішаного (кількісно-якісного) опису.

За **типом опису закону** (законів) функціонування системи поділяють на:

- «чорний ящик» (невідомий повністю закон функціонування системи; відомі тільки вхідні і вихідні повідомлення системи);
- непараметризовані (закон не описаний, описуємо з допомогою хоча б невідомих параметрів, відомі лише деякі апріорні властивості закону);
- параметризовані (закон відомий з точністю до параметрів і його можна віднести до деякого класу залежностей);

– «білий (прозорий) ящик» (повністю відомий закон).

За **способом управління** системою (у системі) розрізняють:

- керовані ззовні системи (без зворотного зв'язку, регульовані, керовані структурно, інформаційно або функціонально);
- керовані зсередини (саморегульовані або самокеровані – програмно керовані, регульовані автоматично, адаптовані – пристосовані за допомогою керованих змін станів і саморганізовані – змінюють у часі й у просторі свою структуру найбільш оптимально, упорядковують свою структуру внаслідок дії внутрішніх і зовнішніх чинників);
- з комбінованим управлінням (автоматичні, напіваавтоматичні, автоматизовані, організаційні).

Великі системи. Прості та складні системи

Система називається **великою**, якщо її дослідження або моделювання ускладнюється через велику розмірність (наприклад через велику кількість станів системи). Велика система зазвичай зводиться до системи меншої розмірності за допомогою розбиття задачі на ряд задач меншої розмірності.

Система називається **складною**, якщо в ній не вистачає ресурсів (в першу чергу інформаційних) для ефективного опису системи та керування нею. Складність системи може бути як зовнішньою, так і внутрішньою.

Розрізняють так-звані «**жорсткі**» і «**м'які**» системи. Дослідження «жорстких» систем зазвичай спирається на такі категорії, як «проекування», «оптимізація», «реалізація», «цільова функція» та ін. Для «м'яких» систем частіше використовуються такі категорії, як «можливість», «бажаність», «адаптовність», «раціональність» тощо. Методи також відрізняються: для «жорстких» систем притаманні методи оптимізації, теорія ймовірностей та математична статистика, теорія ігор тощо, для «м'яких» систем – багатокритеріальна оптимізація і прийняття рішень (часто в умовах невизначеності), метод Дельфі, теорія катастроф, нечіткі множини та нечітка логіка, евристичне програмування тощо.

Прості системи зазвичай характеризуються малою кількістю компонентів, передбачуваною поведінкою та невеликою взаємодією. Вони складаються з невеликої кількості частин або елементів, мають зрозумілі правила та закономірності, які легко передбачити, та компоненти простої системи взаємодіють один з одним прямо й у невеликій кількості варіантів. Приклади простих систем: годинниковий механізм, світлофор, простий ланцюг електричного струму.

Складні системи зазвичай характеризуються багатьма компонентами, непередбачуваною поведінкою та взаємодією на різних рівнях. Вони складаються з великої кількості частин або елементів, мають складні закономірності й можуть мати випадкову або хаотичну поведінку, та компоненти складної системи можуть взаємодіяти один з одним у різних варіаціях і на різних рівнях. Приклади складних систем: екосистеми, економічні системи, мережі зв'язку.

Тест 1

1. Поняття системи
2. Теоретико-множинне визначення системи
3. Технології системного аналізу
4. Моделювання
5. Стани системи (моделі)
6. Структура системи
7. Декомпозиція
8. Аналіз і синтез в системних дослідженнях
9. Властивості системи
10. Життєвий цикл системи
11. Способи опису систем
12. Класифікація систем
13. Великі системи. Прості та складні системи

Складність системи. Обчислення складності системи

Складність системи – це багатогранне поняття, яке можна оцінювати різними способами, в залежності від контексту системи та її характеристики. Однак, у деяких випадках складність може бути **формалізована** за допомогою конкретних математичних підходів.

Система є складною, якщо вона утворена з великої кількості елементів, що взаємодіють між собою. Наприклад, система керування польотів літаків великого аеродрому, міський транспорт, виробничий процес, інформаційна система та ін. **Структура** кожної системи складається з підсистем, що є дуже важливим етапом при побудові формального опису складної системи, що дозволяє іноді спрощувати дослідження складних систем.

Стан системи – це набір значень її параметрів у певний момент часу, який повністю описує її поведінку та розвиток у майбутньому (за детермінованими або стохастичними правилами).

Якщо система має два стани, то можна позначити їх, наприклад, як S_1 та S_2 , а відповідні ймовірності перебування системи в цих станах – $P(S_1)$ та $P(S_2)$. Для правильної моделі виконується умова нормування:

$$P(S_1) + P(S_2) = 1.$$

Приклад. Розглянемо систему, що моделює сервер, який може бути або в **робочому стані** (S_1) або в **стані збою** (S_2). Якщо ймовірність нормальної роботи сервера становить 0.95, то ймовірність збою дорівнює:

$$P(S_2) = 1 - P(S_1) = 1 - 0.95 = 0.05.$$

Такі моделі часто використовують у теорії надійності, черг та марковських процесах.

Приклади формул для оцінки складності системи

Розглянемо n типів елементів системи. Для кожного типу інтуїтивно встановлюється величина **складності елемента**, що вимірюється деяким числом.

Складність елемента i -того типу позначимо через s_i .

Складністю системи, що містить елементи зі складністю s_i ($i = \overline{1, n}$), назовемо величину:

$$S = \sum_{i=1}^n s_i k_i, \quad (1.1)$$

де k_i – кількість елементів i -того типу, що входить у систему (без урахування зв'язків між елементами).

Кількість елементів системи визначається за формулою:

$$N = \sum_{i=1}^n k_i \quad (1.2)$$

Нехай m – **фактична** (реальна) кількість зв'язків у системі, тоді **відносну кількість реалізованих зв'язків** у системі α можна визначити за формулою:

$$\alpha = \frac{m}{N(N-1)} \quad (1.3)$$

Отже, складність системи можна оцінити величиною:

$$S = (1 + \alpha\gamma) \sum_{i=1}^n s_i k_i, \quad (1.4)$$

де γ – **відносна складність зв'язків** порівняно зі складністю елементів системи (визначається експертами).

Інформаційна складність (ентропія Шеннона)

Одна з поширених метрик складності пов'язана з ентропією Шеннона, яка вимірює **кількість інформації або невизначеності** в системі.

Формула для обчислення ентропії системи:

$$H(X) = - \sum_{i=0}^n P(x_i) \log_2 P(x_i)$$

де $H(X)$ – ентропія випадкової змінної X , $P(x_i)$ – ймовірність появи стану x_i , n – кількість можливих станів системи.

Ентропія – це важливе поняття в термодинаміці та теорії інформації, що відображає **рівень хаосу або невизначеності** в системі.

У контексті термодинаміки: Ентропія вимірює ступінь неупорядкованості молекул у системі. Чим вища ентропія, тим більше хаосу й неупорядкованості в системі.

Формула для визначення зміни ентропії:

$$\Delta S = \frac{Q}{T}$$

де ΔS – зміна ентропії, Q – кількість тепла, що додається або відводиться від системи, T – температура.

У контексті **інформаційної теорії** (Шеннона): Ентропія вимірює **середню кількість інформації**, яку несе одне повідомлення з джерела інформації. Вона відображає **невизначеність або непередбачуваність** даних.

Формула для ентропії Шеннона:

$$H(X) = - \sum_{i=1}^n P(x_i) \log_2 P(x_i)$$

де $H(X)$ – ентропія випадкової величини X , $P(x_i)$ – ймовірність виникнення конкретного результату x_i .

Найменше можливе значення ентропії Шеннона дорівнює нулю. Це трапляється тоді, коли інформаційне джерело повністю детерміноване, тобто коли ми точно знаємо, яке значення прийме змінна X . В такому випадку немає ніякої невизначеності, і ентропія дорівнює нулю:

$$H(X) = 0$$

Іншими словами, якщо повідомлення несе абсолютно передбачувану інформацію, то кількість нової інформації дорівнює нулю, і, відповідно, ентропія теж дорівнює нулю.

Найбільше можливе значення ентропії Шеннона досягається тоді, коли всі можливі події рівноймовірні. У такому випадку випадкова змінна X має рівномірний розподіл ймовірностей.

Коли всі можливі події рівноймовірні, ймовірність кожної події $P(x_i) = \frac{1}{n}$. У цьому випадку формула ентропії:

$$H(X) = - \sum_{i=1}^n \frac{1}{n} \log_2 \frac{1}{n} = - \sum_{i=1}^n \frac{1}{n} (-\log_2 n) = \sum_{i=1}^n \frac{1}{n} \log_2 n.$$

Це максимальне значення ентропії, яке вказує на максимальну невизначеність або непередбачуваність системи.

Цикломатична складність (McCabe's Cyclomatic Complexity)

Використовується в програмній інженерії для вимірювання логічної складності програми. Формула для обчислення цикломатичної складності:

$$V(G)=E-N+2P$$

де $V(G)$ – цикломатична складність графа програми, E – кількість ребер у графі, N – кількість вузлів, P – кількість компонентів зв'язності (зазвичай $P=1$ для однієї програми).

Приклад. Обчислити цикломатичну складність програми для фрагменту коду, який має умовні оператори та цикли. Для виконання завдання представити код у вигляді графа потоку управління.

```
def example_function(x):
```

```
    if x > 0:
```

```
        print("Positive")
```

```
    elif x == 0:
```

```
        print("Zero")
```

```
    else:
```

```
        print("Negative")
```

```
    for i in range(3):
```

```
        print(i)
```

Чим більше точок прийняття рішень, тим більше складність програми. Якщо програма має високий номер складності, то ймовірність помилки висока із збільшенням часу на обслуговування та усунення несправностей.

Функціональна складність системи

Функціональна складність оцінює кількість і різноманітність функцій підсистем.

Приклад. Розглянемо програмну систему, яка складається з трьох модулів. Кожен модуль виконує певну кількість функцій, а також взаємодіє з іншими модулями. Для оцінки функціональної складності використовується наступний підхід:

1. Визначається кількість функцій у кожному модулі:

Модуль 1: 5 функцій.

Модуль 2: 8 функцій.

Модуль 3: 3 функції.

2. Враховується складність взаємодій між модулями (в залежності від кількості зв'язків між ними):

Зв'язок між Модулем 1 і Модулем 2: 2 зв'язки.

Зв'язок між Модулем 2 і Модулем 3: 1 зв'язок.

Зв'язок між Модулем 1 і Модулем 3: 1 зв'язок.

Оцінка функціональної складності обчислюється за формулою:

$$C = \sum_{i=1}^n F_i + \sum_{j=1}^m L_j,$$

де F_i – кількість функцій в i -му модулі, L_j – кількість зв'язків між модулями, n – кількість модулів, m – кількість зв'язків між модулями.

Розв'язання:

1. Сума кількості функцій у модулях:

$$\sum_{i=1}^n F_i = 5 + 8 + 3 = 16.$$

2. Сума кількості зв'язків між модулями:

$$\sum_{j=1}^m L_j = 2 + 1 + 1 = 4.$$

3. Загальна функціональна складність системи:

$$C = \sum_{i=1}^n F_i + \sum_{j=1}^m L_j = 16 + 4 = 20.$$

Відповідь: Функціональна складність системи становить $C=20$.

Інтерпретація результату:

1. **Складність модулів.** Основна частина складності пов'язана з функціональністю окремих модулів (16 функцій).

2. **Складність взаємодій.** Зв'язки між модулями додають до загальної складності лише 4 одиниці. Це свідчить про те, що система має помірну кількість взаємодій між модулями.
3. **Прогноз.** Для зменшення складності системи можна оптимізувати кількість функцій у модулях або зменшити кількість зв'язків між ними.

Складність взаємодій (в загальному випадку)

Складність системи, що враховує кількість елементів і зв'язків між ними, може бути визначена як:

$$C = k \cdot n \cdot (n - 1)$$

де C – складність системи, k – коефіцієнт, що враховує інтенсивність взаємодій, n – кількість елементів у системі.

Когнітивна складність (Модель Карді)

Когнітивну складність інтерфейсу можна оцінити за допомогою кількості завдань та рівня їх складності. Вона вимірюється як:

$$C = \sum_{i=1}^n t_i \cdot d_i$$

де C – когнітивна складність, t_i – час виконання i -го завдання, d_i – рівень складності завдання.

Ці формули лише **частково** відображають складність системи в різних **контекстах**. Для конкретної системи або **ситуації** можуть застосовуватися інші підходи або формули. **Складність** – це багатоаспектне поняття, тому обчислення її завжди залежить від **конкретних** характеристик системи та того, що саме треба **оцінити** або **оптимізувати**.

Надійність системи

Для сприйняття системи **важливою характеристикою** її, як і будь-якого об'єкта, є надійність. Сприймається нами об'єкт, коли він функціонує і ми бачимо його як такий. Якщо ж він не працює, не функціонує, він втрачає для нас сенс як об'єкт. Тобто надійність визначає існування об'єкта.

Надійністю називається здатність об'єкта зберігати певну властивість у певному проміжку простору і в певному проміжку часу. Альтернативою надійності є ризик. Ризиком називається настання події, яке є небажаним.

При побудові нових систем, а також при реструктуризації наявних важливо вміти розв'язувати задачі з оптимізації надійності цих систем. Це пояснюється тим, що рівень надійності має закладатися в тактико-технічні характеристики і вимоги на побудову систем. Рівень вимог, тобто рівень показників надійності, має визначатись практикою і процесом споживання об'єкту.

Характеристики (показники) надійності

Оскільки **надійність** будь-якого об'єкта пов'язується із **випадком** (випадок, коли об'єкт функціонує, і випадок, коли він не функціонує), то основними характеристиками надійності мають бути такі, що мають імовірнісний характер. Базовою характеристикою є **ймовірність**.

Позначимо ймовірність безвідмовної роботи:

$$P\{\xi \geq t_0\} = 1 - P\{\xi < t_0\} = 1 - F(t_0), \quad (2.1)$$

де ξ – випадкова величина, що описує момент відмови.

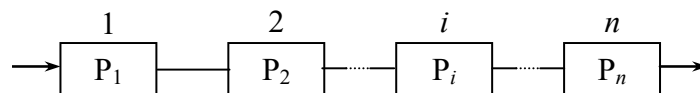
З формули (2.1) випливає, що **функція надійності** $R(t)$ – це різниця між одиницею і функцією розподілу:

$$R(t) = 1 - F(t).$$

Надійність складних систем з детермінованою зв'язністю елементів

1) Системи з послідовним зв'язком елементів.

Система з послідовним з'єднанням елементів – це така, при якій вихід з ладу хоча б одного елемента призводить до виходу з ладу всієї системи.



Позначимо через n кількість послідовно зв'язаних елементів системи.

Нехай p_i – це **надійність** i -го елемента. Визначимо надійність системи P_c .

Подія (1-й і 2-й елементи працюють) рівнозначна перетину подій (і 1-й елемент працює, і 2-й елемент працює). Вважаємо, що виходи з ладу елементів незалежні, тобто вихід (поломка) одного елемента не впливає на надійність інших елементів.

$$P(A_1 \cdot A_2) = P(A_1) \cdot P(A_2) = p_1 p_2$$

$$P_c = p_1 p_2 \cdot \dots p_i \cdot \dots p_n$$

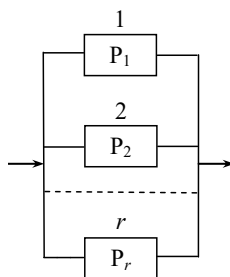
$$P_c = \prod_{i=1}^n p_i \quad (2.2)$$

2) Система з паралельним зв'язком елементів.

Системи з паралельним зв'язком елементів, або паралельні системи, є важливими для забезпечення високої надійності та відмовостійкості. Така система передбачає одночасну роботу кількох елементів, що виконують однакові завдання, тобто якщо один з елементів відмовляється, інші можуть продовжити виконання задачі без значної втрати продуктивності.

Ключові переваги паралельних систем:

1. Резервні копії елементів мінімізують ризики відмови. Підвищена надійність.
2. Відмова одного з компонентів не призводить до загальної зупинки системи. Висока відмовостійкість.
3. Одночасна робота кількох компонентів підвищує загальну продуктивність системи. Покращена продуктивність.



Виведемо формулу для надійності такої системи.

$$1 - \overline{p_2} = 1 - (1 - p_1) \cdot (1 - p_2)$$

$$P_c = 1 - (1 - p_1) \cdot (1 - p_2) \cdot (1 - p_3) \cdot \dots \cdot (1 - p_r)$$

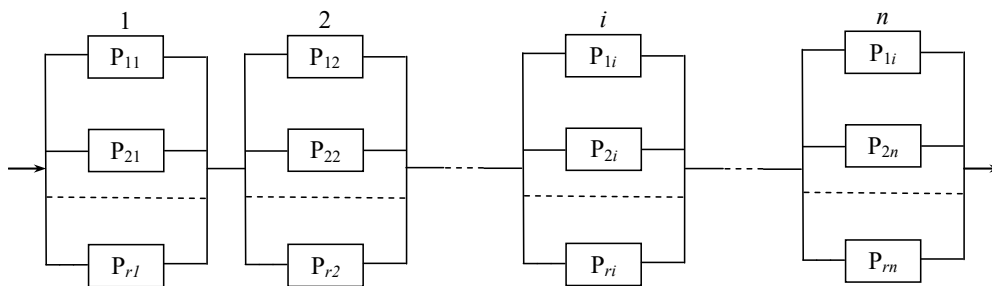
$$P_c = 1 - \prod_{j=1}^r (1 - p_j) \quad (2.3)$$

Система з паралельним з'єднанням елементів виражає функціональну ідентичність цих елементів. Це дає право говорити про функціональне резервування, наприклад, 1-го елемента. У тому випадку, якщо резервують ідентичними елементами, то говорять про дублювання і формула (2.3) набуває вигляду:

$$P_c = 1 - (1 - p_1)^r.$$

3) Система послідовно з'єднаних зарезервованих елементів.

Розглянемо систему типу (1), де кожен елемент має свій резерв.



$$P_c = P_I \cdot P_{II} \cdot \dots \cdot P_i \cdot \dots \cdot P_n$$

$$P_{ci} = 1 - \prod_{j=1}^{r_i} (1 - p_{ji})$$

$$P_c = \prod_{i=1}^n \left[1 - \prod_{j=1}^{r_i} (1 - p_{ij}) \right]. \quad (2.4)$$

Формула (2.4) описує надійність системи n послідовно зв'язаних елементів, які неідентично зарезервовані кожен своєю кількістю.

У частковому випадку, коли кількість резерву одна й та ж і дорівнює r , формула набуває вигляду:

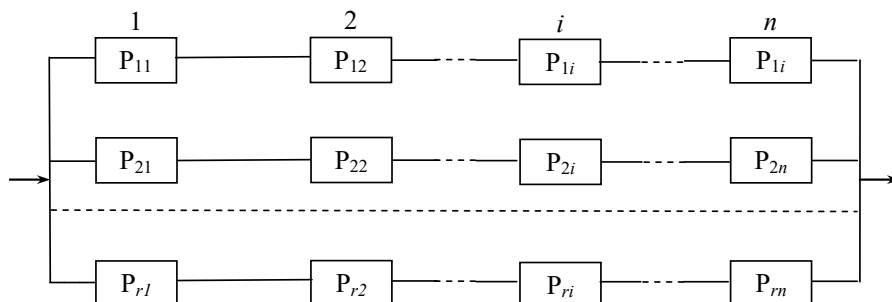
$$P_c = \prod_{i=1}^n \left[1 - \prod_{j=1}^r (1 - p_{ij}) \right]$$

У частковому випадку, коли маємо дублювання елементів (резервування ідентичними), формула набуває вигляду:

$$P_c = \prod_{i=1}^n \left[1 - (1 - p_i)^{r_i} \right]$$

4) Перше узагальнення систем 1-3.

Розглянемо систему паралельно-послідовного з'єднання елементів.



$$P_c = 1 - \prod_{i=1}^r \left(1 - \prod_{j=1}^{n_i} p_{ij} \right) \quad (2.5)$$

Частковий випадок, коли кількість елементів в кожному ланцюжку однакова n , формула (2.5) набуває вигляду:

$$P_c = 1 - \prod_{i=1}^r \left(1 - \prod_{j=1}^n p_{ij} \right) /$$

У частковому випадку, коли маємо дублювання (резервування ідентичними елементами), формула (2.5) набуває вигляду:

$$P_c = 1 - \left(1 - \prod_{j=1}^n p_j \right)^r .$$

Тест 2

1. Складність системи. Обчислення складності системи
2. Приклади формул для оцінки складності системи
3. Інформаційна складність (ентропія Шеннона)
4. Цикломатична складність. Побудова графа потоку управління
5. Функціональна складність системи
6. Складність взаємодій
7. Когнітивна складність (Модель Карді)
8. Надійність системи
9. Надійність складних систем з детермінованою зв'язністю елементів

Імітаційне моделювання як метод дослідження систем великої складності

Основні методи імітаційного моделювання

Аналітичний метод застосовується для імітації процесів в основному для малих і простих систем, де **відсутній фактор випадковості**. Наприклад, коли процес їх функціонування описаний **диференціальними** або інтегродиференціальними рівняннями. Метод названий умовно, оскільки він об'єднує можливості імітації процесу, модель якого отримана у вигляді аналітично замкнутого розв'язання або розв'язання, отриманого методами обчислювальної математики.

Метод статистичного моделювання спочатку розвивався як метод статистичних випробувань (Монте-Карло). Це числовий метод, що полягає в отриманні оцінок характеристик імовірності, що збігаються з розв'язанням аналітичних задач (наприклад, із розв'язанням рівнянь і обчисленням певного інтеграла). Надалі цей метод став застосовуватися для імітації процесів, що відбуваються в системах, **усередині яких є джерело випадковості** або які схильні до випадкових дій. Він отримав назву методу статистичного моделювання.

Комбінований метод (аналітико-статистичний) дозволяє об'єднати переваги аналітичного і статистичного методів моделювання. Він застосовується у випадку побудови моделі, яка складається з різних модулів, що являють собою набір як статистичних, так і аналітичних моделей, які взаємодіють як єдине ціле. Причому в набір модулів можуть входити не тільки модулі, що відповідають динамічним моделям, але й модулі, які відповідають статистичним математичним моделям.

Імітаційне моделювання – це потужний інструмент для аналізу й розуміння складних систем. Основні задачі імітаційного моделювання включають:

1. Розробка моделей для кращого розуміння, як функціонують реальні системи, наприклад, виробничі лінії, транспортні системи чи бізнес-процеси.
2. Вивчення можливих результатів і наслідків різних сценаріїв для прийняття рішень, таких як зміна виробничих планів чи впровадження нових технологій.
3. Прогнозування поведінки системи.
4. Визначення найефективніших стратегій і шляхів виконання завдань з мінімальними затратами ресурсів і часу. Задачі **оптимізації**.
5. Використання імітаційних моделей для навчання персоналу та підготовки до різних ситуацій без реального ризику.
6. Надання інструментів для **оцінки варіантів рішень і вибору найкращого**.

Імітаційне моделювання дозволяє досліджувати та тестувати різні гіпотези, не витрачаючи реальні ресурси та час. Це корисно в багатьох галузях, таких як виробництво, логістика, охорона здоров'я, економіка і навіть військова справа.

Принципи побудови імітаційних моделей

Процес функціонування складної системи можна розглядати як зміну її станів, що описуються її фазовими змінними $Z_1(t), Z_2(t), \dots, Z_n(t)$ в n -вимірному просторі.

Задача імітаційного моделювання полягає в отриманні траєкторії руху даної системи в n -мірному просторі $(Z_1, Z_2, \dots, Z_n)$, а також обчислення деяких показників, залежних від вихідних сигналів системи і їх властивостей. У цьому випадку “рух” системи розуміється в загальному значенні – як будь-яка зміна, що відбувається в ній.

Відомі два принципи побудови моделі процесу функціонування систем.

Принцип Δt . Розглянемо цей принцип спочатку для детермінованих систем. Припустимо, що початковий стан системи відповідає значенням $Z_1(t_0), Z_2(t_0), \dots, Z_n(t_0)$. Принцип Δt припускає перетворення моделі системи до такого вигляду, щоб значення $Z_1, Z_2, \dots, Z_n$ в момент часу $t_1 = t_0 + \Delta t$ можна було обчислити через початкові значення, а у момент $t_2 = t_1 + \Delta t$ через значення на попередньому кроці і так для кожного i -го кроку ($\Delta t = \text{const}, i = \overline{1, n}$) за рекурсивним методом.

Для систем, де випадковість є визначальним фактором, принцип Δt полягає в такому:

1. Визначається умовний розподіл імовірності на першому кроці ($t_1 = t_0 + \Delta t$) для випадкового вектора, позначимо його $(Z_1, Z_2, \dots, Z_n)$. Умова полягає в тому, що початковий стан системи відповідає точці траєкторії $(Z_1^0, Z_2^0, \dots, Z_n^0)$.

2. Обчислюються значення координат точки траєкторії руху системи ($t_1 = t_0 + \Delta t$), як значення координат випадкового вектора $(Z_1^1, Z_2^1, \dots, Z_n^1)$, заданого розподілом, знайденим на попередньому кроці.

3. Відшукуються умовний розподіл вектора $(Z_1^2, Z_2^2, \dots, Z_n^2)$ на другому кроці ($t_2 = t_1 + \Delta t$), за умови отримання відповідних значень $Z_i^1, i = \overline{1, n}$ на першому кроці і т.д., поки $t_i = t_0 + i\Delta t$ не набуде значення $t_n = t_0 + n\Delta t$.

Принцип Δt є універсальним, може бути застосованим для широкого класу систем. Його недоліком є неекономічність з погляду витрат машинного часу.

Принцип особливих станів (принцип δz)

При розгляді деяких видів систем можна виділити два види станів:

- *звичайний*, в якому система знаходиться більшість часу, при цьому $Z_i(t), (i = \overline{1, n})$ змінюються плавно.
- *особливий*, характерний для системи в деякі моменти часу, причому стан системи змінюється в ці моменти стрибком.

Принцип особливих станів відрізняється від принципу Δt тим, що крок за часом у цьому випадку не постійний, а є величиною випадковою і визначається відповідно до інформації про попередній особливий стан.

Прикладами систем, що мають особливі стани, є системи масового обслуговування. Особливі стани з'являються в моменти надходження заявок, у моменти звільнення каналів тощо. Для таких систем застосування принципу Δt є нераціональним, оскільки при цьому можливі пропуски особливих станів і необхідні методи їх виявлення.

Приклад побудови імітаційної моделі

Імітувати процес функціонування системи в часі при заданому Δt . Побудувати графік процесу функціонування системи.

Розглянемо перетворення диференціального рівняння до вигляду, що дозволяє застосувати принцип Δt на прикладі.

Задача про охолодження тіла. Швидкість охолодження тіла в повітрі пропорційна різниці між температурою тіла і температурою повітря. Температура повітря дорівнює 20°C . Відомо, що протягом 20 хвилин тіло охолоджується від 100 до 60°C . Визначити закон зміни температури тіла θ залежно від часу t .

Диференціальне рівняння має вигляд: $\theta'(t)=k(\theta-20)$, де k – коефіцієнт пропорційності. Імітувати процес охолодження тіла в часі із кроком $\Delta t=25$ (хвилин), якщо при $t=0$ температура тіла $\theta=100^{\circ}\text{C}$. Визначити, через скільки годин тіло повністю охолоне (тобто температура тіла θ буде дорівнювати температурі повітря).

Розв'язання. Відповідно до умови задачі маємо:

$$\frac{d\theta}{dt} = k(\theta - 20) \quad (5.1)$$

Для розв'язання рівняння (5.1) розділимо змінні:

$$\frac{d\theta}{(\theta - 20)} = kdt,$$

Проінтегруємо ліву та праву частини рівняння:

$$\int \frac{d\theta}{(\theta - 20)} = \int kdt$$

У результаті отримаємо:

$$\ln(\theta - 20) = kt + \ln C,$$

що після потенціювання дає:

$$\theta - 20 = Ce^{kt},$$

а, отже, після відповідних перетворень отримаємо загальний розв'язок рівняння (5.1):

$$\theta = 20 + Ce^{kt} \quad (5.2)$$

Використовуючи вираз (5.2) та початкові умови (при $t=0$ $\theta=100$), знаходимо значення константи C . Звідси $100 = 20 + Ce^{k \cdot 0}$, тобто $C=80$.

Підставляючи $C=80$ у рівняння (5.2), отримаємо:

$$\theta = 20 + 80e^{kt}$$

Коефіцієнт пропорційності k визначаємо з додаткової умови: $t=20$, $\theta=60$. Звідси

$$60 = 20 + 80e^{20k} \text{ або } e^{20k} = \frac{1}{2}, \text{ а, отже, } e^k = \left(\frac{1}{2}\right)^{\frac{1}{20}}.$$

Таким чином, знайдено закон, за яким визначається температура тіла в будь-який момент часу t , тобто шукана функція має вигляд:

$$\theta = 20 + 80\left(\frac{1}{2}\right)^{\frac{t}{20}}. \quad (5.3)$$

Для того щоб імітувати процес функціонування системи в часі при заданому Δt , необхідно скористатися такою залежністю:

$$y_i = y(t_0 + i\Delta t).$$

За умови, що $t_0 = 0$ і $\Delta t = 25$, отримаємо таку залежність:

$$y_i = y(25i). \quad (5.4)$$

А, отже, рівняння (5.4), враховуючи залежність (5.6) для $i = \overline{1, n}$, можна представити у вигляді:

$$\theta_i = 20 + 80\left(\frac{1}{2}\right)^{\frac{25i}{20}}$$

Розглянемо всі значення θ_i в момент часу t_i (табл. 5.1) та побудуємо траєкторію процесу охолодження тіла (рис. 5.1).

Таблиця 5.1

t	θ
0	100
5	87,27171
10	76,56854

t	θ
15	67,56828
20	60
25	53,63586
30	48,28427
35	43,78414
40	40
45	36,81793
50	34,14214
55	31,89207
60	30
65	28,40896
70	27,07107
75	25,94604
80	25
85	24,20448
90	23,53553
95	22,97302
100	22,5
105	22,10224
110	21,76777
115	21,48651
120	21,25
...	...
500	20

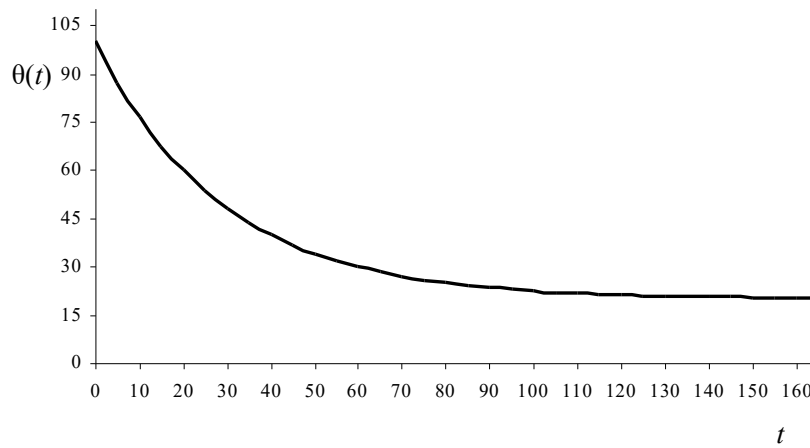


Рис. 5.1. Графік процесу охолодження тіла

Отже, з отриманих даних у таблиці 5.1 та графіка (рис.5.1), можна зробити висновок, що тіло майже набуде температури повітря через 130 хвилин або $2\frac{1}{2}$ години, а через 500 хвилин або $8\frac{1}{3}$ години температура тіла повністю дорівнюватиме температурі повітря.

Так, задача про охолодження тіла є класичним прикладом імітаційного моделювання фізичних процесів. Наведемо ще кілька прикладів, що демонструють застосування імітаційного моделювання в різних галузях:

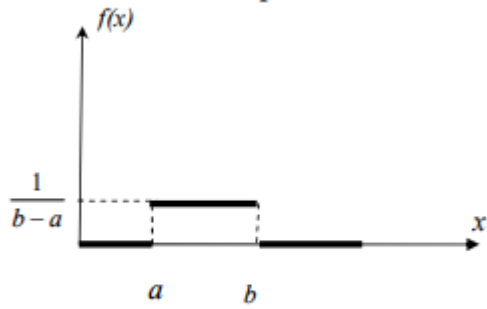
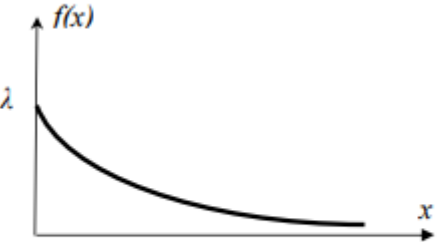
1. **Розповсюдження епідемій.** Моделювання поширення вірусу в популяції з урахуванням рівня імунізації, карантинних заходів, швидкості передачі тощо.
2. **Черги в сервісних системах.** Наприклад, моделювання роботи кас у супермаркетах або обслуговування заявок у дата-центрах для визначення оптимальної кількості обслуговуючих пристроїв.
3. **Динаміка транспорту.** Моделювання потоків машин у міському середовищі для оптимізації роботи світлофорів або аналізу ефективності зміни організації руху.
4. **Розпад радіоактивних речовин.** Використання стохастичних моделей для прогнозування кількості ядер, що розпадуться за певний проміжок часу.
5. **Моделювання погодних процесів.** Прогнозування кліматичних змін на основі багатофакторного аналізу змін температури, вологості, атмосферного тиску.
6. **Процеси дифузії.** Наприклад, моделювання дифузії газів у закритому середовищі або поширення фарби у воді.
7. **Робота комп'ютерних мереж.** Імітація завантаженості мережі та оцінка ефективності нових протоколів маршрутизації.
8. **Фінансові ринки.** Аналіз ризиків інвестування за допомогою імітацій Монте-Карло, що дозволяють передбачити можливі сценарії змін курсів валют і акцій.
9. **Виробничі процеси.** Моделювання роботи виробничих ліній, оптимізація часу обробки деталей та пошук "вузьких місць" у виробництві.
10. **Автоматизовані системи управління.** Наприклад, тестування роботи штучного інтелекту в безпілотних автомобілях або роботизованих маніпуляторах.

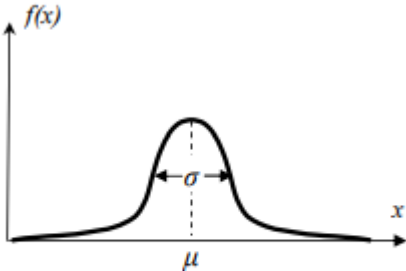
Наведені приклади демонструють широкий спектр застосування імітаційного моделювання у фізиці, інженерії, економіці та інших сферах.

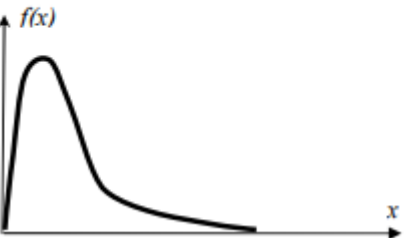
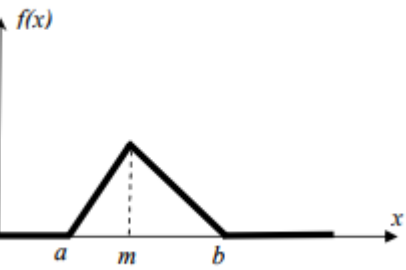
ПОБУДОВА МОДЕЛІ ІМІТАЦІЇ НЕПЕРЕРВНИХ ВИПАДКОВИХ ВЕЛИЧИН

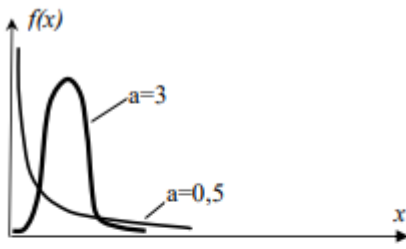
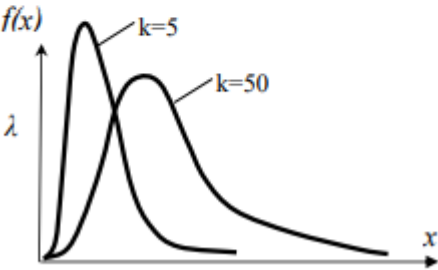
Моделювання неперервних випадкових величин є важливим інструментом для аналізу складних систем, де **змінні** мають **безперервний характер** (наприклад, температура, швидкість, час). У випадку таких змінних часто використовуються різні види **неперервних розподілів** ймовірностей, наприклад, нормальний, експоненційний або рівномірний розподіли.

Закони розподілу неперервних випадкових величин

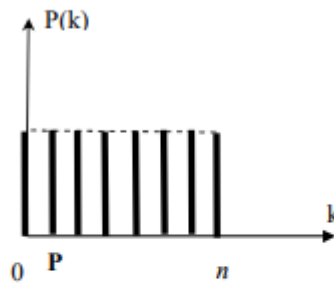
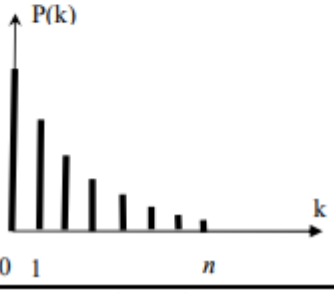
Графічне представлення щільності закону розподілу	Формульне представлення щільності закону розподілу
Рівномірний 	$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ 0, & x \leq a, x \geq b \end{cases}$ $\mu = \frac{a+b}{2}, \quad \sigma = \frac{b-a}{2\sqrt{3}}$
Експоненціальний (показниковий) 	$f(x) = \begin{cases} \lambda e^{-\lambda x}, & 0 \leq x < \infty \\ 0, & x < 0 \end{cases}$ $\mu = \frac{1}{\lambda}, \quad \sigma = \frac{1}{\lambda}$

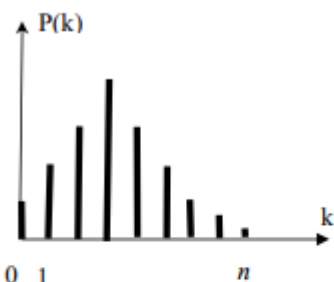
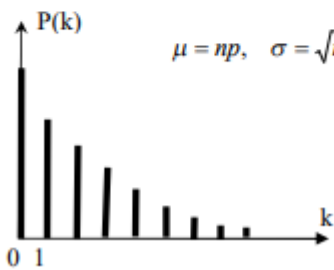
Нормальний (Гауса) 	$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, -\infty \leq x \leq \infty$ $\mu = \mu, \quad \sigma = \sigma$
--	--

Графічне представлення щільності закону розподілу	Формульне представлення щільності закону розподілу
Логнормальний 	$f(x) = \begin{cases} \frac{1}{xb\sqrt{2\pi}} e^{-\frac{(\ln x - a)^2}{2b^2}}, & x > 0 \\ 0, & x \leq 0 \end{cases}$ $\mu = e^{a + \frac{b^2}{2}}$ $\sigma = \sqrt{e^{2a+2b^2}(e^{b^2} - 1)}$
Трикутний 	$f(x) = \begin{cases} \frac{2(x-a)}{(m-a)(b-a)}, & a \leq x \leq m \\ \frac{2(b-x)}{(b-m)(b-a)}, & m \leq x \leq b \end{cases}$ $\mu = \frac{a+m+b}{3}$ $\sigma = \frac{\sqrt{a(a-m) + b(b-a) + m(m-b)}}{3\sqrt{2}}$

Вейбулла 	$f(x) = \begin{cases} ab^{-a} x^{a-1} e^{-\left(\frac{x}{b}\right)^a}, & x > 0 \\ 0, & x \leq 0 \end{cases}$ $\mu = \frac{b}{a} \Gamma\left(\frac{1}{a}\right)$ $\sigma = b \sqrt{\frac{1}{a} \left(2\Gamma\left(\frac{2}{a}\right) - \frac{1}{a} \left(\Gamma\left(\frac{1}{a}\right) \right)^2 \right)}$
Ерланга 	$f(x) = \begin{cases} \lambda e^{-\lambda x}, & 0 \leq x \leq \infty \\ 0, & x < 0 \end{cases}$ $\mu = k \cdot \frac{1}{\lambda}, \quad \sigma = \sqrt{k} \cdot \frac{1}{\lambda}$

Закони розподілу дискретних випадкових величин

Графічне представлення закону розподілу	Формульне представлення закону розподілу
Рівномірний 	$P(k) = \frac{1}{n}, \quad k = 0, 1, \dots, n$ $\mu = \frac{n}{2}, \quad \sigma = \frac{n}{2\sqrt{3}}$
Пуассона 	$P(k) = \frac{\lambda^k}{k!} e^{-\lambda}, \quad k = 0, 1, \dots, n$ $\mu = \lambda, \quad \sigma = \sqrt{\lambda}$

Біноміальний 	$P(k) = C_n^k p^k (1-p)^{n-k}, \quad k = 0, 1, \dots, n$ $\mu = np, \quad \sigma = \sqrt{np(1-p)}$
Геометричний 	$P(k) = p(1-p)^k, \quad k = 0, 1, \dots$ $\mu = (1-p)p, \quad \sigma = \frac{\sqrt{1-p}}{p}$

Побудова імітаційної моделі випадкової події

Імітаційне моделювання результатів випробувань випадкових подій – це потужний інструмент для аналізу складних систем.

При побудові алгоритмів, які моделюють будь-який складний процес для розв’язання задачі методом статистичних випробувань, доводиться працювати з випадковими об’єктами різноманітної природи. Зокрема, у багатьох випадках виникає необхідність моделювання результатів незалежних або залежних випробувань у схемі випадкових подій.

Нехай у результаті випробувань подія A настає з імовірністю p , протилежна подія $\bar{A}$ – з імовірністю $1-p$. Для того щоб імітувати результат випробувань, вибираючи випадкове число із сукупності з рівномірним законом розподілу в інтервалі $[0,1]$, необхідно подію A замінити деякою іншою подією B , яка може відбутися з тією ж імовірністю. Отже, виконаємо випробування відносно події B . Якщо при імітаційному випробуванні відбудеться подія B , то будемо вважати, що в імітаційній схемі відбулася подія A ; здійснення події $\bar{B}$ поставимо у відповідність здійсненню події $\bar{A}$ імітаційної схеми.

Далі, нехай X – випадкова величина з рівномірним законом розподілу в інтервалі $[0,1]$. Визначимо подію B як подію, що полягає в тому, що вибране значення ξ випадкової величини X виявиться меншим або рівним p :

$$B = \{\xi \leq p\} \quad (6.1)$$

Ймовірність події B визначаємо за формулою:

$$P(B) = \int_0^p dx = p.$$

Таким чином, отримаємо, що, оскільки $P(B) = P(A)$, то подію B може імітувати подія A . Протилежна подія $\bar{B}$ визначається як

$$\bar{B} = \{\xi > p\}, \quad (6.2)$$

тому

$$P(\bar{B}) = \int_p^1 dx = 1 - p = P(\bar{A})$$

У загальному випадку маємо $A_1, A_2, \dots, A_s$ – повну групу несумісних подій, що відбуваються з імовірностями $p_1, p_2, \dots, p_s$ відповідно. У цьому випадку

$$p_1 + p_2 + \dots + p_s = 1.$$

Визначимо імітаційну повну групу несумісних подій $B_1, B_2, \dots, B_s$ таким чином. Подія B_m ($m = 1, 2, \dots, s$) полягає в тому, що вибрані значення ξ випадкової величини X задовольняють нерівності

$$l_{m-1} < \xi \leq l_m, \quad (6.4)$$

де

$$l_r = \sum_{i=1}^r p_i. \quad (6.5)$$

У такому випадку

$$P(B_m) = \int_{l_{m-1}}^{l_m} dx = p_m = P(A_m).$$

Складемо алгоритм моделювання результату випробування у вказаній схемі. Вибираємо випадкове число ξ із сукупності з рівномірним законом розподілу в інтервалі $[0, 1]$ та порівнюємо його із p .

Якщо виконується умова

$$\xi < p \quad (6.3)$$

(тобто відбулася подія B , що визначається за (6.1)), то результатом випробування вважаємо A . Якщо умова (6.3) не виконується, тобто відбулася подія $\overline{B}$, яка визначається за умовою (6.2), то вважаємо, що результатом випробування є подія $\overline{A}$.

У загальному випадку маємо $A_1, A_2, \dots, A_s$ – повну групу несумісних подій, що відбуваються з імовірностями $p_1, p_2, \dots, p_s$ відповідно. У цьому випадку $p_1 + p_2 + \dots + p_s = 1$.

Визначимо імітаційну повну групу несумісних подій $B_1, B_2, \dots, B_s$ таким чином. Подія B_m ($m = 1, 2, \dots, s$) полягає в тому, що вибрані значення ξ випадкової величини X задовольняють нерівності

$$l_{m-1} < \xi \leq l_m, \quad (6.4)$$

де

$$l_r = \sum_{i=1}^r p_i. \quad (6.5)$$

У такому випадку

$$P(B_m) = \int_{l_{m-1}}^{l_m} dx = p_m = P(A_m).$$

Цю процедуру іноді називають «вибором за жеребкуванням». Моделюючий алгоритм для імітації результату випробувань у цій схемі полягає в такому.

Формуємо довільно випадкове число ξ вихідної рівномірної сукупності та порівнюємо його послідовно з усіма величинами l_r , що визначаються рівностями (6.5), а оскільки група подій $B_1, B_2, \dots, B_s$ повна, то обов'язково відбудеться будь-яка подія B_m , яка полягає у виконанні нерівностей (6.4); тоді вважаємо, що результатом імітаційного випробування є подія A_m .

Під час розв'язування прикладних задач часто доводиться моделювати *сумісні випробування*. Розберемо принципи такого моделювання на простому прикладі.

Для незалежних подій. Нехай *незалежні події* A_1 і A_2 мають імовірності p_1 і p_2 відповідно. Для моделювання результату сумісних випробувань необхідно довільно вибрати два випадкових числа та здійснити два порівняння, послідовно перевіряючи умову, аналогічну (6.3), відносно подій A_1 і A_2 .

В іншому випадку сумісні випробування мають повну групу можливих результатів з відповідними ймовірностями:

$$\begin{aligned} P(A_1 A_2) &= p_1 p_2 \\ P(\bar{A}_1 A_2) &= (1 - p_1) p_2 \\ P(A_1 \bar{A}_2) &= p_1 (1 - p_2) \\ P(\bar{A}_1 \bar{A}_2) &= (1 - p_1) (1 - p_2) \end{aligned}$$

Для моделювання результату сумісних випробувань необхідно визначити один із можливих станів. Для цього необхідно вибрати одне випадкове число ξ , але порівнянь може бути набагато більше.

Використання одного або іншого варіанта полягає в зручності побудови алгоритму, економії машинного часу та оперативної пам'яті. Звичайно перший варіант виявляється більш економічним, ніж другий.

Для залежних подій. При сумісних випробуваннях може статися, що події A_1 і A_2 – *залежні*. У такому випадку нехай задані p_1 і p_2 – ймовірності безумовного настання відповідно подій A_1 і A_2 та $P(A_2 | A_1)$ – умовна ймовірність події A_2 за умови, що відбулася подія A_1 . Так само, як і в разі моделювання незалежних сумісних випробувань, тут можна проводити моделювання за двома алгоритмами.

Перший алгоритм можна побудувати так. Вибираємо випадкове число ξ_1 із сукупності з рівномірним законом розподілу і перевіряємо умову:

$$\xi_1 \leq p_1 \quad (6.6)$$

Виконання умови (6.6) означає здійснення події A_1 , а, отже, при моделюванні випробування щодо події A_2 потрібно скористатися ймовірністю $P(A_2 | A_1)$. Для цього вибираємо наступне випадкове число ξ_2 і перевіряємо умову

$$\xi_2 \leq P(A_2 | A_1) \quad (6.7)$$

Якщо умова (6.7) виконана, то ми отримаємо кінцевий результат: відбулася подія $A_1 A_2$. Якщо ж умова (6.7) не виконана, то кінцевий результат сумісних випробувань буде $A_1 \overline{A_2}$.

Ми розглянули ланцюжок алгоритму за умови виконання нерівності (6.6); якщо ж вона виявилася невиконаною, то відбулася подія $\overline{A_1}$ і надалі нам потрібна ймовірність $P(A_2 | \overline{A_1})$, яку легко визначити, користуючись теоремою про повну ймовірність:

$$P(A_2) = P(A_1) \cdot P(A_2 | A_1) + P(\overline{A_1}) \cdot P(A_2 | \overline{A_1}) \quad (6.8)$$

Визначаючи зі співвідношення (6.8) ймовірність $P(A_2 | \overline{A_1})$ та замінюючи $P(A_1) = p_1$, $P(A_2) = p_2$, отримаємо

$$P(A_2 | \overline{A_1}) = \frac{p_2 - p_1 \cdot P(A_2 | A_1)}{1 - p_1} \quad (6.9)$$

Потім знову витягуємо випадкове число ξ_2 , перевіряємо виконання нерівності

$$\xi_2 \leq P(A_2 | \overline{A_1}) \quad (6.10)$$

і одержуємо кінцевий результат випробувань: $\overline{A_1} A_2$ або $\overline{A_1} \overline{A_2}$, залежно від того, виконується нерівність (6.10) чи ні.

Другий алгоритм моделює здійснення однієї події з повної групи $A_1 A_2$, $A_1 \overline{A_2}$, $\overline{A_1} A_2$ і $\overline{A_1} \overline{A_2}$ з відповідними ймовірностями:

$$\begin{aligned} P(A_1 A_2) &= p_1 P(A_2 | A_1) \\ P(A_1 \overline{A_2}) &= p_1 [1 - P(A_2 | A_1)] \\ P(\overline{A_1} A_2) &= (1 - p_1) P(A_2 | \overline{A_1}) \\ P(\overline{A_1} \overline{A_2}) &= (1 - p_1) [1 - P(A_2 | \overline{A_1})] \end{aligned}$$

де $P(A_2 | \overline{A_1})$ визначається співвідношенням (6.9).

Необхідно відзначити, що алгоритми в розглянутих прикладах виявляються справедливими лише в тому випадку, коли при моделюванні використовуються

випадкові числа, вибрані з сукупності з рівномірним розподілом ймовірності в проміжку $[0,1]$.

Приклад моделювання інформаційної системи за допомогою нормального розподілу

Нормальний розподіл часто використовують для апроксимації різноманітних природних явищ та економічних процесів, оскільки він є універсальним для багатьох реальних ситуацій.

Умова. Інформаційна система обробляє запити користувачів, а час обробки одного запиту є нормально розподіленою випадковою величиною зі середнім значенням $\mu=5$ секунд і стандартним відхиленням $\sigma=1$ секунда.

1. Визначити ймовірність того, що обробка одного запиту триватиме

а) Менше ніж 4 секунд.

б) Між 4 і 6 секундами.

2. Побудувати графік нормального розподілу, що моделює процес обробки запитів.

Розв'язання:

1. Використання функції нормального розподілу.

Функція нормального розподілу визначається як:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

2. Визначення ймовірностей:

а) Ймовірність того, що $X < 6$:

$$P(X < 6) = F\left(\frac{6-\mu}{\sigma}\right),$$

б) Ймовірність того, що $4 < X < 6$:

$$P(4 < X < 6) = F\left(\frac{6-\mu}{\sigma}\right) - F\left(\frac{4-\mu}{\sigma}\right).$$

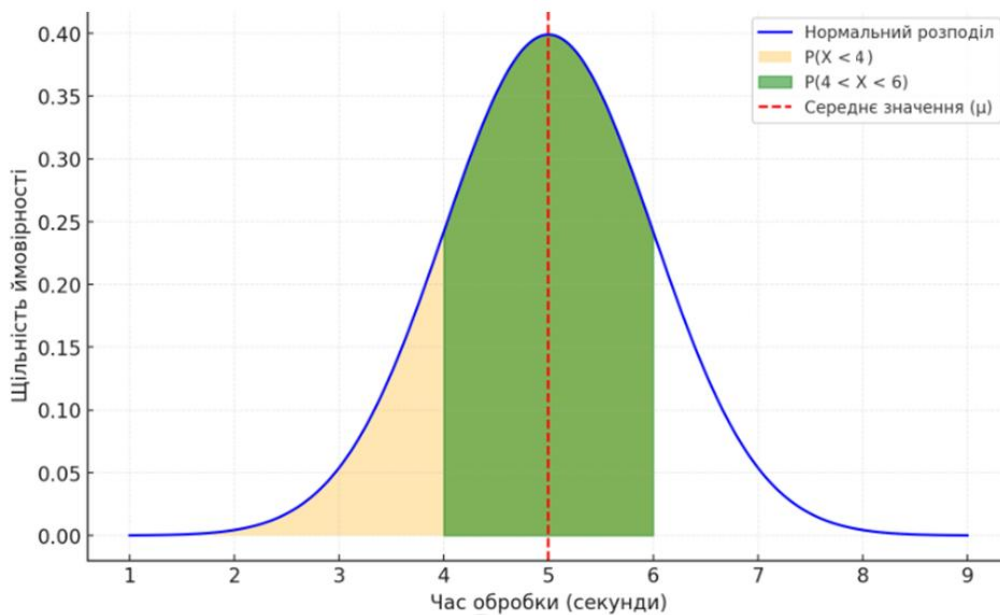
Ймовірність, що час обробки запиту менше ніж 6 секунд:

$$P(X < 6) \approx 0.8413 \text{ (або 84.13\%).}$$

Ймовірність, що час обробки запиту знаходиться між 4 і 6 секундами:

$$P(4 < X < 6) \approx 0.6827 \text{ (або 68.27\%).}$$

Нормальний розподіл обробки запиту



На графіку:

Синя крива представляє функцію щільності нормального розподілу.

Область під кривою до $X=6$ відповідає ймовірності $P(X < 6)$.

Зелена область між $X=4$ і $X=6$ відповідає ймовірності $P(4 < X < 6)$.

Цей підхід ілюструє, як нормальний розподіл використовується для моделювання часу обробки запитів у інформаційній системі.

Моделювання неперервних випадкових величин за допомогою експоненційного розподілу

Експоненціальний розподіл часто стосується **кількості часу**, поки **не відбудеться якась конкретна подія**. Наприклад, кількість часу (починається зараз) до землетрусу має експоненціальний розподіл. Інші приклади включають тривалість часу, у хвилинух, міжміських ділових телефонних дзвінків та кількість часу, у місяцях, триває автомобільний акумулятор. Також можна показати, що значення зміни, яке у вас є в кишені або гаманці, приблизно слідує за експоненціальним розподілом.

Значення для експоненціальної випадкової величини відбуваються наступним чином. Тут менше великих значень і більше малих значень. Наприклад, маркетингові дослідження показали, що **кількість грошей**, які клієнти витрачають за одну поїздку в супермаркет, слідує **експоненціальному розподілу**. Є **більше людей**, які витрачають **невеликі суми грошей** і **менше людей**, які витрачають **великі суми грошей**.

Експоненціальні розподіли зазвичай використовуються при розрахунках надійності продукту або тривалості часу, який триває виріб.

Випадкова величина для експоненціального розподілу є безперервною і часто вимірює плин часу, хоча вона може бути використана в інших додатках. Типовими питаннями можуть бути: «яка ймовірність того, що якась подія відбудеться протягом наступних x годин або днів, або яка ймовірність того, що якась подія відбудеться між x_1 годинами та x_2 годинами, або яка ймовірність того, що подія займе більше x_1 годин для виконання?»

Функція щільності ймовірності задається:

$$f(x) = \frac{1}{\mu} e^{-\frac{1}{\mu}x}$$

де μ – історичний середній час очікування.

Випадкова величина для експоненціального розподілу має середнє і стандартне відхилення $1/\mu$.

Приклад. Моделювання часу до наступного обслуговування в черзі за допомогою експоненційного розподілу.

Завдання. Розглянемо сервісну точку (наприклад, банкомат або сервісний центр), де обслуговування клієнтів відбувається через випадкові проміжки часу. Час до наступного обслуговування клієнта розподілений за експоненційним законом з параметром $\lambda = 0,2$ (це означає, що середній час до обслуговування — 5 хвилин).

Необхідно побудувати модель, яка буде імітувати час до наступного обслуговування клієнта за допомогою неперервної випадкової величини.

Кроки для побудови моделі:

1. **Визначення розподілу.** Оскільки ми використовуємо експоненційний розподіл, то $f(t) = \lambda e^{-\lambda t}$, $t \geq 0$, де $f(t)$ – функція щільності ймовірності; λ – параметр розподілу (у нашому випадку $\lambda = 0,2$); t – час до наступного обслуговування.
2. **Генерація випадкових чисел.** Для генерації випадкових величин із експоненційного розподілу використовуємо наступну формулу:

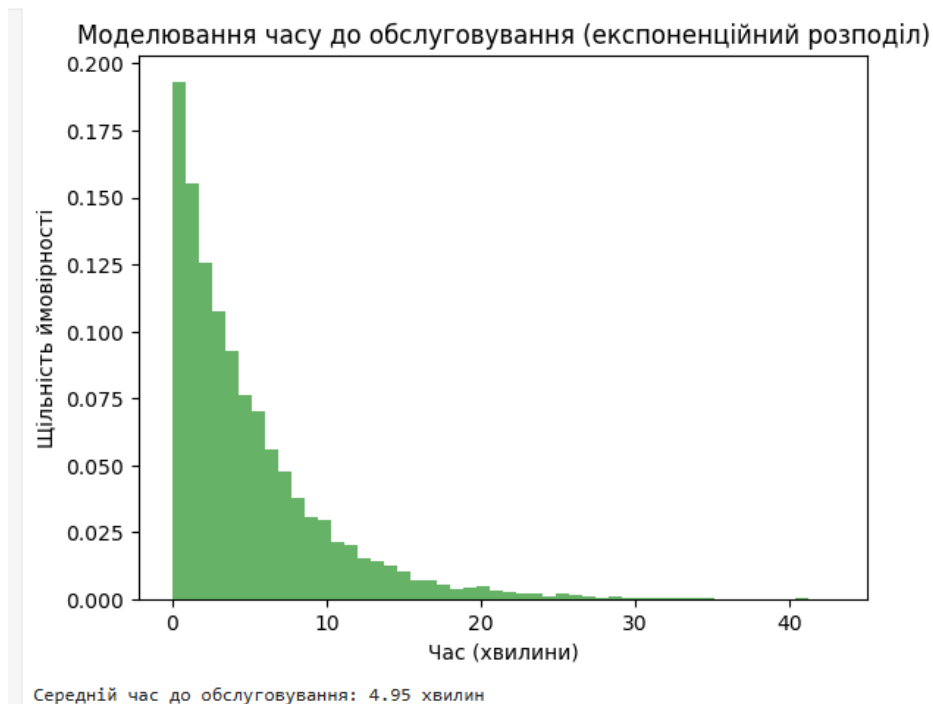
$$t = \frac{-\ln(1 - U)}{\lambda}$$

де U – випадкова величина, що генерується рівномірно на відрізку $[0, 1]$;

t – згенерований час до наступного обслуговування.

Для кожного симульованого часу U обчислюємо відповідне значення t .

3. Створення моделі та проведення симуляцій



Аналіз результатів:

Гістограма. Побудуємо гістограму для 10 000 симуляцій, щоб побачити, як розподіляється час до обслуговування за експоненційним розподілом.

Середнє значення. Обчислимо середній час до обслуговування, щоб переконатися, що він наближається до теоретичного середнього, яке дорівнює $1/\lambda$. Для нашого випадку $1/\lambda=5$ хвилин.

Результат. Якщо ми проведемо симуляції, то побачимо, що більшість випадків має час обслуговування, який буде невеликим, але з часом кілька довших періодів також з'являються, що характерно для експоненційного розподілу. Середнє значення буде близьким до 5 хвилин, що підтверджує правильність вибору параметрів моделі.

Висновки. Моделювання неперервних випадкових величин за допомогою експоненційного розподілу дозволяє врахувати випадкові затримки в системах обслуговування. У нашому прикладі ми оцінюємо час до наступного обслуговування, але подібний підхід можна застосувати і для інших задач, де потрібна оцінка неперервних випадкових величин.

Побудова моделі імітації дискретних випадкових величин

Випадкова величина називається *дискретною*, якщо її часткові (можливі) значення можна занумерувати скінченною кількістю натуральних чисел від 1 до k або всіма натуральними числами.

Дискретна випадкова величина X може бути задана: 1) рядом розподілу, 2) функцією розподілу.

Рядом розподілу називається сукупність усіх можливих значень x_i та відповідні їм ймовірності $p_i = P(X = x_i)$. Ряд розподілу може бути задано у вигляді таблиці (табл. 8.1) або формули.

Таблиця 8.1

X	x_1	x_2	$\dots$	x_n
$P(X = x_i)$	P_1	P_2	$\dots$	P_n

Ймовірності P_i задовольняють умову $\sum_{i=1}^n P_i = 1$, де число можливих значень n може бути скінченним або нескінченним.

Графічне зображення ряду розподілу називається *многокутником розподілу*. Для його побудови можливі значення випадкової величини x_i відкладаються по осі Ox , а ймовірності p_i – по осі Oy ; точки A_i з координатами (x_i, p_i) з'єднуються лініями (рис. 8.1).

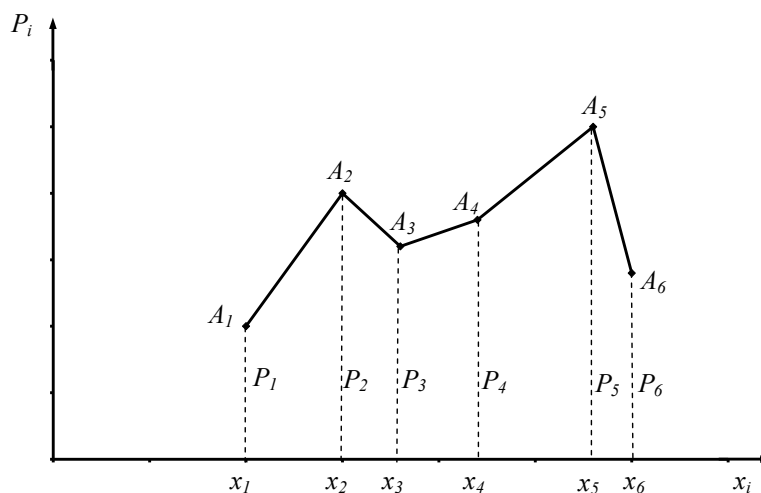


Рис. 8.1. Многокутник розподілу

Функцією розподілу дискретної випадкової величини X називається функція $F(x)$, яка дорівнює ймовірності $P(X < x)$ того, що випадкова величина X буде менша від довільно вибраного значення x . Функція $F(x)$ обчислюється за формулою:

$$F(x) = P(X < x) = \sum_{x_i < x} P(X = x_i),$$

де нерівність $x_i < x$ під знаком суми вказує, що сумування поширюється на всі ті значення x_i , які менші за x .

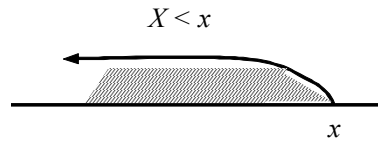


Рис.8.2.

Геометрично функція розподілу виражає ймовірність попадання випадкової величини в півінтервал, що лежить на числовій осі лівіше від точки x (рис. 8.2).

Побудуємо графік функції розподілу для дискретної випадкової величини, що задана рядом розподілу у вигляді таблиці (табл. 8.1).

1. Нехай $x \leq x_1$. Оскільки випадкова величина X не набуває можливих значень, які знаходяться лівіше від точки x , то подія X у цьому випадку неможлива та її ймовірність дорівнює нулю.

$$F(x) = P(X < x) = 0$$

2. Нехай тепер $x_1 < x \leq x_2$. При цьому випадкова величина X набуває єдиного можливого значення x_1 , що знаходиться лівіше від x , з імовірністю P_1 . Тому: $F(x) = P(X < x_2) = P(X = x_1) = P_1$

3. Нехай далі $x_2 < x \leq x_3$. При цьому випадкова величина може набути або значення x_1 , або x_2 , що знаходиться лівіше від x . Використовуючи теорему додавання ймовірностей несумісних подій, отримаємо: $F(x) = P(X = x_1) + P(X = x_2) = P_1 + P_2$

4. Для випадку $x_{n-1} < x \leq x_n$ аналогічно отримаємо:

$$F(x) = P(X = x_1) + P(X = x_2) + \dots + P(X = x_{n-1}) = P_1 + P_2 + \dots + P_{n-1}$$

5. Нехай $x_n > x$. Тоді випадкова величина X набуде одне з усіх можливих значень $x_1, x_2, \dots, x_n$. Ця подія достовірна, а, отже, її ймовірність дорівнює одиниці. Тому $F(x) = 1$.

Зобразимо отримані результати графічно (рис. 8.3).

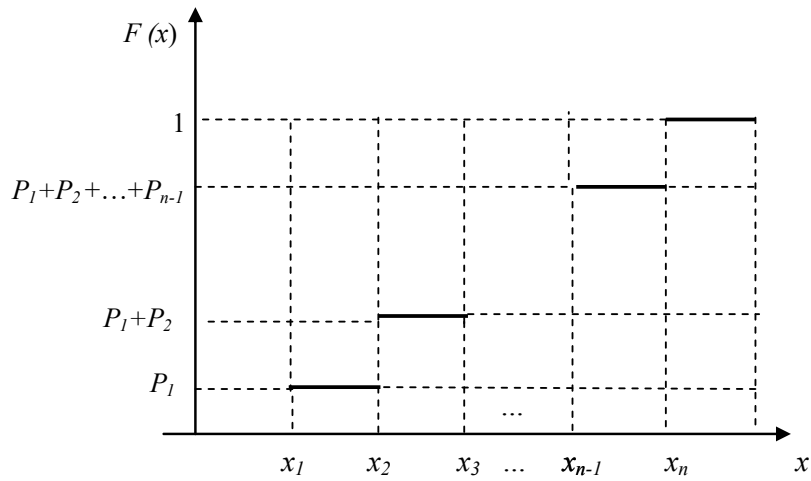


Рис. 8.3. Функція розподілу $F(x)$ дискретної випадкової величини X

Отже, функція розподілу дискретної випадкової величини – це розривна ступінчаста лінія, стала між можливими значеннями випадкової величини. З побудови функції розподілу випливає, що розмір стрибка функції дорівнює ймовірності відповідного значення дискретної випадкової величини.

Тест 3

1. Імітаційне моделювання як метод дослідження систем великої складності
2. Основні методи імітаційного моделювання
3. Приклад побудови імітаційної моделі
4. Побудова моделі імітації неперервних випадкових величин
5. Приклад моделювання інформаційної системи за допомогою нормального розподілу
6. Моделювання неперервних випадкових величин за допомогою експоненційного розподілу
7. Побудова моделі імітації дискретних випадкових величин

Математичне моделювання систем за допомогою випадкових процесів. Марківський процес

Випадковим процесом $\xi(t)$ називають функцію не випадкового аргументу t , яка при кожному фіксованому значенні аргументу є випадковою величиною. Випадкові процеси (ВП) широко застосовують як математичні моделі систем і явищ, які з'являються, змінюються випадковим (непередбачуваним) чином. ВП знаходять своє застосування в багатьох дисциплінах, включаючи такі науки, такі як біологія, хімія, екологія, неврологія і фізика, а також у технічних галузях, таких як обробка зображень, обробка сигналів, у теорії інформації, інформатиці, криптографії і телекомунікації, у фінансах.

Розглянемо випадковий процес як функцію двох аргументів $\xi(t)=X(t,\omega)$, де $t \in T$, $\omega \in \Omega$. Змінну t будемо інтерпретувати як час, ω – це випадкова величина. Тоді зафіксувавши момент часу $t = t_0$ дістанемо випадкову величину $X(t_0, \omega)$, яку називають **перерізом випадкового процесу у момент часу t_0** . При фіксованому $\omega = \omega_0$ отримаємо не випадкову функцію $x(t)=X(t, \omega_0)$, яку називають **реалізацією (траєкторією) випадкового процесу**.

ВП $\xi(t)$ називають процесом із дискретним часом, якщо система S , у якій він відбувається, може змінювати свої стани тільки у визначені, наперед відомі моменти часу $t_1, t_2, \dots, t_n$, які називають кроками (або етапами) цього процесу.

Область визначення (існування) ВП – множина T може бути скінченною або зліченною. Вважається, що в проміжках часу між сусідніми кроками t_i, t_{i+1} система зберігає свій стан. Не виключається можливість, що система не змінить свого стану впродовж кількості кроків, більшої ніж два. ВП $\xi(t)$ із дискретним часом називають також випадковою послідовністю: $\xi(t_1), \xi(t_2), \dots, \xi(t_n)$, або випадковим ланцюгом.

ВП $\xi(t)$ називають процесом з неперервним часом, якщо система S , у якій він відбувається, може змінювати свої стани в будь-які, випадкові моменти часу, що неперервно заповнюють числову пряму або її підмножину. Множина T такого процесу є нескінченною та незліченною.

ВП $\xi(t)$, який відбувається в системі S , називають **процесом з дискретними станами**, якщо в будь-який момент часу $t \in T$ множина його станів є скінченною або зліченною; іншими словами, якщо його переріз у будь-який момент t описується однією дискретною випадковою величиною $\xi(t)$ – в одновимірному випадку та n -вимірною випадковою величиною $\xi_1(t), \xi_2(t), \dots, \xi_n(t)$ – в багатовимірному випадку.

ВП $\xi(t)$, який відбувається в системі S , називають процесом з неперервними станами, якщо множина можливих станів системи S незліченна; іншими словами, якщо

переріз процесу в будь-який момент часу t описується неперервною (або мішаною) випадковою величиною – в одновимірному випадку та векторною $ВВ$ – у багатовимірному випадку (неперервною або мішаною).

За вказаними ознаками ВП поділяють на чотири класи – ВП із:

дискретним часом і дискретними станами (ДД);

дискретним часом і неперервними станами (ДН);

неперервним часом і дискретними станами (НД);

неперервним часом і неперервними станами (НН).

ВП із дискретним часом є легшими для вивчення, бо неперервні за часом процеси вимагають більш складних математичних методів і знань, зокрема через те, що множина T незліченна.

Приклад 1. ВП типу ДД. Особа купила m білетів виграшного займу, які можуть вигравати та погашатися в заздалегідь відомі моменти часу (тиражі) $t_1, t_2, \dots, t_n$. Випадковий процес $\xi(t)$ – число білетів, на які припав виграш до моменту t .

Приклад 2. ВП типу ДН. У певні моменти часу t_1 і t_2 , реєструється температура повітря $\xi(t)$ у заданій точці простору. Послідовність значень цієї величини – випадковий процес $\xi(t)$ з дискретним часом і неперервними станами.

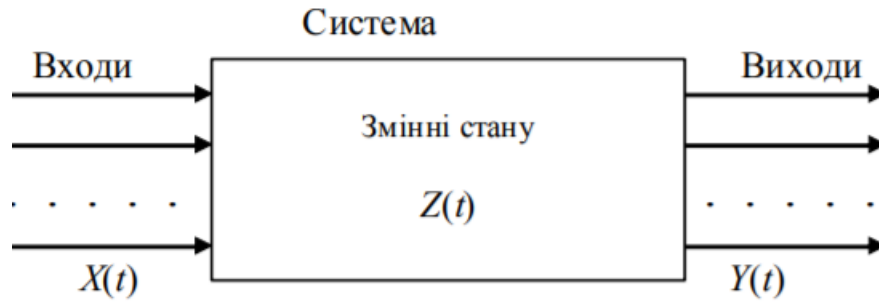
Приклад 3. ВП типу НД. Технічний прилад складається з n вузлів, які можуть у ході роботи приладу відмовляти (виходити з ладу). Випадковий процес $\xi(t)$ – кількість вузлів, що відмовили до моменту t .

Приклад 4. ВП типу НН. У мікроскоп протягом проміжку часу T спостерігається переміщення малої частинки в рідині, яка здійснює броунівський рух. Координати частинки в момент t – випадковий процес з неперервними станами та неперервним часом.

Випадковий процес як зміна стану системи

Функціонування системи в часі може бути описане різними способами.

Стан системи є фундаментальним поняттям системного аналізу. У будьякий момент часу він описується фіксованим набором значень змінних, які називають змінними стану. У загальному випадку їх можна подати у вигляді вектора: $Z = (z_1, z_2, \dots, z_n)$, і у кожній окремій задачі ці змінні мають певний фізичний зміст.



Під дією вхідних впливів $X(t)$ стан системи $Z(t)$ та її реакція (вихід) $Y(t)$ змінюються в часі. Змінні стану при цьому є проміжними і характеризують внутрішній стан системи. Математична модель системи, описана через змінні стану, в детермінованому випадку має таку структуру:

$$\dot{Z}(t) = \varphi[z(t), X(t)]. \quad (3.1)$$

$$Y(t) = f[z(t), X(t)]. \quad (3.2)$$

Ця модель описує ситуацію, коли стан системи в наступний момент часу [рівняння (3.1)] та її реакція в даний момент часу [рівняння (3.2)] цілком визначено станом системи $Z(t)$ і вхідним впливом $X(t)$ в даний момент часу. За таких умов рівняння (3.1) називається рівнянням стану; а рівняння (3.2) – рівнянням виходу системи.

Ми будемо розглядатимемо процеси, що відбуваються в системах третього і четвертого типу, тобто ті, що характеризуються дискретною множиною станів, зокрема марковські процеси. Зауважимо, що математичний апарат розроблений для опису згаданих процесів та ланцюгів Маркова широко використовується у теорії систем, дослідженні операцій та в інших прикладних дисциплінах. Це зумовлено багатьма причинами, серед яких можна визначити такі:

1. Багато реальних технічних систем можна подати у вигляді скінченної множини станів, при цьому їхня поведінка в процесі функціонування адекватно моделюється саме марковськими процесами.

2. Високий рівень розробки теорії марковських процесів із дискретною множиною станів дозволяє використовувати її для розв'язування багатьох прикладних задач.

Марківський процес – це випадковий процес, що базується на *принципі Маркова*. Згідно з цим принципом ймовірність значень випадкової величини у наступні моменти $t + \tau$ не залежать від того, які значення вона приймала у попередні моменти часу.

Простий ланцюг Маркова. Матриця переходу

Нехай система знаходиться в одному із станів $E_1, E_2, \dots, E_k$. У певні фіксовані моменти часу $t_1, t_2, \dots, t_n, \dots$ система під впливом випадкових факторів може переходити із одного стану в інший, при цьому у будь-який момент часу t_n ймовірність для системи з'явитись у наперед заданому стані E_j цілком визначається тим станом, у якому вони знаходились безпосередньо перед стрибком і не залежить від усіх попередніх станів, у яких система знаходилась до моменту часу t_{n-1} . Поведінка цієї системи описується **простим ланцюгом Маркова**. Отже, ланцюг Маркова є випадковим процесом з дискретним часом.

Ймовірність переходу із стану E_i (у момент t_{n-1}) до стану E_j у момент t_n зветься **перехідною ймовірністю** цього ланцюга і позначається таким чином: $p_{ij}^{(n)} = P(E_j, t_n | E_i, t_{n-1})$

Ланцюг Маркова задається **матрицею переходу** P , елементами якої є перехідні ймовірності $p_{ij}^{(n)}$, а також ймовірності усіх станів системи у початковий момент часу $t_0 = 0$. Ці ймовірності $p_{01}, p_{02}, \dots, p_{0k}, \dots$ зовуться **початковими ймовірностями** станів системи. Якщо всі перехідні ймовірності $p_{ij}^{(n)}$ не залежать від часу (тобто $p_{ij}^{(n)} = p_{ij}, n = 1, 2, \dots$), тоді ланцюг Маркова називається **однорідним**. В цьому випадку матриця P має такий вигляд:

$$P = \begin{pmatrix} p_{11} & p_{12} & \dots & p_{1k} & \dots \\ p_{21} & p_{22} & \dots & p_{2k} & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

за умовою, що $0 \leq p_{ij} \leq 1$ та $\sum_j p_{ij} = 1$ при будь-якому фіксованому i , тобто сума елементів кожного рядка дорівнює одиниці. Загальний елемент p_{ij} цієї матриці має два індекси: перший визначає номер стану E_i в даний момент, а другий – номер майбутнього стану E_j , в який перейде система. Якщо $p_{ij} = 0$, то це означає, що випадкова подія, яка полягає у безпосередньому переході системи із i -го стану у j -ий неможлива. Випадок $p_{ij} = 1$ відповідає твердженню, що із i -го стану система після наступного випробування завжди переходить в j -ий стан.

Ймовірність переходу системи із стану в стан за m кроків

Позначимо ймовірність того, що система, яка керується однорідним ланцюгом Маркова, перейде із стану E_i в стан E_j через $p_{ij}(m)$ за m кроків. Наприклад, $p_{24}(3)$

визначає ймовірність переходу за три кроки із другого стану в четвертий. Матриця переходів за m кроків має вигляд:

$$P_m = \begin{pmatrix} p_{11}(m) & p_{12}(m) & \dots & p_{1k}(m) & \dots \\ p_{21}(m) & p_{22}(m) & \dots & p_{2k}(m) & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix}.$$

Кожне значення ймовірності $p_{ij}(m)$ задовольняє **рівнянню Маркова**:

$$p_{ij}(m) = \sum_l p_{il}(s)p_{lj}(m-s); \quad (9.4)$$

Де s може приймати будь-яке ціле значення на відрізку $[0, m]$.

Безумовна ймовірність $p_j(n) = \sum_i p_{0i} p_{ij}(n)$ називається **абсолютною ймовірністю** системі з'явитись у момент t_n у стані E_j . Тоді $P_m = P^m$. Так, при $s = 1$ маємо:

$$\text{Для } m = 2 \quad P(2) = P(1)P(2-1) = P(1)P(1) = P^2;$$

$$\text{для } m = 3 \quad P(3) = P(1)P(3-1) = P(1)P(2) = P(1)P^2(1) = P^3;$$

.....
 для m $P_m = P^m$.

Класифікація станів

Стани $s_1, s_2, \dots$ системи ланцюга Маркова мають різне функціональне навантаження, що і визначає їх класифікацію.

Означення. Стан s_j називається **досяжним зі стану** s_i якщо існує таке ціле невід'ємне число n , що з ймовірністю, відмінною від нуля, система може перейти зі стану s_i у стан s_j за n кроків, тобто $p_{ij}^{(n)} > 0$.

Означення не виключає і випадок $s_j = s_i$.

Означення. Множина станів C називається **замкненою**, якщо ніякий стан зовні C не є досяжним з ніякого стану, що входить у C . Множина станів C буде замкненою тоді і тільки тоді, коли $p_{ij} = 0$ при всіх значеннях i та j таких, що s_i належить C , а s_j не належить C .

Таким чином, якщо в матриці ймовірностей переходу за n кроків $P_n(n=1,2,\dots)$ викреслити всі рядки і стовпці, що відповідають станам, які не входять у замкнену множину C , то дістанемо нову стохастичну матрицю. Ланцюг Маркова з цією стохастичною матрицею можна вивчати незалежно від інших станів, що не входять у замкнену множину C . Отже, на кожній замкненій множині станів однозначно визначається деякий ланцюг Маркова.

Означення. Ланцюг Маркова називається *незвідним*, якщо в ньому немає ніяких замкнених множин станів, крім множини всіх станів. Підсумовуючи отримані результати, можна одержати *критерій незвідності*: ланцюг Маркова є незвідним тоді і тільки тоді, коли довільний його стан досяжний з довільного іншого стану.

Означення. Стан ланцюга Маркова називається *поглинальним*, якщо замкнена множина станів у ньому складається лише з одного цього стану.

Іншими словами, поглинальним станом ланцюга Маркова є стан, потрапляючи в який процес у ньому залишається. Стан s_i є поглинальним тоді і тільки тоді, коли $p_{ii}=1$. З поглинального стану недосяжний ніякий інший стан.

Означення. Стан s_i ланцюга Маркова називається *зворотним*, якщо система, яка в початковий момент часу перебувала в цьому стані, обов'язково повернеться коли-небудь у цей же стан s_i .

Таким чином, якщо стан s_i зворотний, то

$$u_{ii} = \sum_{n=1}^{\infty} u_{ii}^{(n)} = 1, \quad (9)$$

де $u_{ii}^{(n)}$ – ймовірність першого повернення системи в стан s_i у n -й момент часу, якщо і в початковий момент система там перебувала.

Нехай $t_1^{(ii)}$ – час першого повернення системи в стан s_i якщо і в початковий момент часу вона перебувала в ньому. Узагальнена випадкова змінна $t_1^{(ii)}$ може набувати значень $1, 2, \dots, \infty$, відповідно з ймовірностями $u_{ii}^{(1)}, u_{ii}^{(2)}, \dots, 1-u_{ii}$.

Означення. Стан s_i ланцюга Маркова називається *періодичним*, якщо перше повернення в цей стан може відбутися лише за кількістю моментів часу, яка кратна деякому фіксованому натуральному числу T , яке більше за одиницю.

Отже, число T набуває значень $1, 2, \dots$. Найменше число T , що задовольняє умову $u_{ii}^{(\neq nT)}=0$, називається *періодом стану* s_i .

Ймовірність перебування системи в заданому стані на n –му кроці

Нехай ланцюг Маркова заданий стохастичним вектором p та стохастичною матрицею P . Позначимо через $p_k^{(n)}$ ймовірність того, що ланцюг Маркова через n кроків буде знаходитись у стані s_k . Всі такі ймовірності запишемо у вигляді вектора

$$p(n) = (p_1^{(n)}, p_2^{(n)}, \dots, p_n^{(n)}). \quad (10)$$

Зрозуміло, що де $p_i(0)$ – ймовірність перебування системи в стані s_i при $t_0=0$. Таким чином, $p(0) = p$.

Знайдемо ймовірність перебування системи в заданому стані на 1-му кроці. Зрозуміло, що перейти у стан s_k за один крок система може з будь-якого початкового стану s_i , в якому вона перебувала з ймовірністю $p_i(0)$:

$$\begin{aligned} s_1 &\rightarrow s_k \\ s_2 &\rightarrow s_k \\ &\dots \\ s_i &\rightarrow s_k \\ &\dots \\ s_n &\rightarrow s_k \end{aligned}$$

Тоді за формулою повної ймовірності маємо

$$p_i^{(1)} = \sum_{i=1}^n p_i(0) \cdot p_{ik}.$$

Сума одержана за правилом скалярного добутку стохастичного вектора p на i -й стовпець стохастичної матриці P . Для всіх можливих значень i отримаємо вектор $p(1)$. Правило побудови елементів $p_i^{(1)}$ вказує, що вектор ймовірностей перебування системи в заданому стані на 1-му кроці дорівнює добуткові стохастичного вектора на стохастичну матрицю, тобто

$$p(1) = p P. \quad (11)$$

Зауважимо, що вектор $p(1)$ є стохастичним.

Аналогічно можна показати, що

$$p(2) = p(1) P = p P^2. \quad (12)$$

Методом математичної індукції можна довести, що

$$p(n+1) = p(n) P = p P^{n+1}. \quad (13)$$

Граничні ймовірності станів. Ймовірність перебування системи в заданому стані в далекому майбутньому

Якщо ланцюг Маркова функціонує нескінченно довго, то за певних умов у системі може встановитися стаціонарний режим, при якому система продовжує блукання за станами, але ймовірності цих станів не змінюються (не залежать від номера кроку). Такі ймовірності називаються граничними (фінальними) ймовірностями ланцюга Маркова. Граничні ймовірності можна тлумачити як середній час перебування системи в даному стані.

Нехай ланцюг Маркова ергодичний та існують границі $\lim_{n \rightarrow \infty} p_k^{(n)} = p_{\bullet k}$, які не залежать від початкових умов. Позначимо вектор граничних ймовірностей $p_{\bullet} = (p_{\bullet 1}, p_{\bullet 2}, \dots, p_{\bullet k}, \dots)$. Необхідно знайти компоненти цього вектора.

Розглянемо співвідношення $p(n+1) = p(n) P$. Перейдемо до границі за умови: $n \rightarrow \infty$. Враховуючи, що для стаціонарного режиму ланцюга Маркова ймовірність перебування системи в заданому стані на $n+1$ -му кроці дорівнює ймовірності перебування системи в заданому стані на n -му кроці, тобто

$$p_j(n+1) = \sum_{i=1}^n p_i(n) \cdot p_{ij} = p_j, \text{ де } p_j \text{ не залежить від } n, \text{ отримаємо:}$$

$$\begin{aligned} p_{\bullet} &= p_{\bullet} \cdot P, \\ p_{\bullet} \cdot I &= p_{\bullet} \cdot P, \end{aligned}$$

де I – одинична матриця.

Отже,

$$p_{\bullet} \cdot (P - I) = 0$$

Матриця $D = P - I$ називається *динамічною матрицею*.

Таким чином, розв'язок поставленої задачі, якщо він існує, є розв'язком *динамічної системи рівнянь*:

$$\begin{cases} p_{\bullet} \cdot D = 0 \\ \sum_{i=1} p_{\bullet i} = 1 \end{cases} \quad (14)$$

Він називається *стаціонарним режимом* ланцюга Маркова.

Запишемо систему (14) в координатній формі:

$$(p_{\bullet 1}, p_{\bullet 2}, \dots, p_{\bullet k}, \dots) \cdot \begin{pmatrix} p_{11} - 1 & p_{12} & \dots \\ p_{21} & p_{22} - 1 & \dots \\ \dots & \dots & \dots \end{pmatrix} = (0, 0, \dots).$$

тобто

$$\begin{cases} (p_{11} - 1) \cdot p_{\bullet 1} + p_{21} \cdot p_{\bullet 2} + p_{31} \cdot p_{\bullet 3} + \dots = 0 \\ p_{21} \cdot p_{\bullet 1} + (p_{22} - 1) \cdot p_{\bullet 2} + p_{32} \cdot p_{\bullet 3} + \dots = 0 \\ \dots \\ p_{\bullet 1} + p_{\bullet 2} + p_{\bullet 3} + \dots = 1 \end{cases} \quad (15)$$

Зауважимо, що обернені значення розв'язків системи рівнянь (15) визначають середні значення часів перших повернень системи у відповідні стани, тобто дорівнюють величинам $M(\tau_1^{(11)}), M(\tau_1^{(22)}), \dots$

Тест 4

1. Математичне моделювання систем за допомогою випадкових процесів. Марковський процес
2. Простий ланцюг Маркова. Матриця переходу
3. Ймовірність переходу системи із стану в стан за n кроків
4. Ймовірність перебування системи в заданому стані на n –му кроці
5. Граничні ймовірності станів. Ймовірність перебування системи в заданому стані в далекому майбутньому

Методи системного аналізу

Методи системного аналізу – це сукупність підходів, що використовуються для дослідження складних систем з метою прийняття обґрунтованих рішень. Системний аналіз дозволяє розглядати об'єкти як частину цілісної структури, враховуючи взаємозв'язки елементів, цілі, ресурси та зовнішнє середовище. Основними етапами системного аналізу є: постановка задачі, формалізація, побудова моделі, аналіз альтернатив, вибір оптимального рішення та його впровадження.

Ключові методи системного аналізу включають структурний аналіз, функціонально-вартісний аналіз, морфологічний аналіз, ієрархічний аналіз, методи теорії графів, аналіз надійності, SWOT-аналіз, PEST-аналіз, метод Делфі, сценарний аналіз, а також методи прийняття рішень в умовах невизначеності та ризику.

Структурний аналіз дозволяє виявити внутрішню будову системи, її підсистеми та елементи, а також встановити зв'язки між ними. **Функціонально-вартісний аналіз (ФВА)** використовується для оптимізації функцій системи відносно витрат. **Морфологічний аналіз** застосовується при розв'язанні слабо формалізованих задач шляхом побудови морфологічних таблиць та виявлення можливих комбінацій рішень. **Метод аналізу ієрархій (АНР)** дозволяє формалізувати процес прийняття рішень, розкладаючи складну задачу на простіші рівні.

У системному аналізі також широко використовуються **математичне моделювання, імітаційне моделювання, оптимізаційні методи**, включаючи лінійне, нелінійне та динамічне програмування. За допомогою **теорії графів** здійснюється аналіз структурних характеристик систем, наприклад, у задачах маршрутизації або управління проектами (методи CPM та PERT). **Методи оцінювання ефективності** дозволяють зіставити альтернативні варіанти за набором критеріїв.

Системний підхід вимагає урахування не лише технічних, але й соціально-економічних, екологічних та політичних аспектів функціонування систем. Системний підхід базується на принципах цілісності, багатоальтернативності, пріоритетності цілей, зворотного зв'язку та ітеративності.

Загалом, системний аналіз є міждисциплінарним підходом, що забезпечує глибоке розуміння складних проблем та сприяє ефективному управлінню змінами, прогнозуванню та стратегічному плануванню.

Метод дерева цілей

Метод дерева цілей є одним із ключових методів системного аналізу, що використовується для планування, моделювання та оцінки складних систем. Основною ідеєю методу є ієрархічне представлення цілей системи у вигляді "дерева", де кожна загальна ціль розбивається на підцілі, що забезпечують її досягнення. Такий підхід

дозволяє структуровано і послідовно аналізувати задачі та визначати шляхи їх вирішення.

Метод дерева цілей широко використовується в інженерії, управлінні проєктами, економіці, розробці програмного забезпечення та багатьох інших сферах, де важливим є структурований підхід до вирішення задач. Це дозволяє підвищити ефективність процесу прийняття рішень і забезпечити досягнення бажаних результатів.

Опис методу. На верхньому рівні дерева розташовується загальна ціль системи або проєкту. Від неї відгалужуються більш конкретні підцілі, які необхідно виконати для досягнення основної мети. Кожна підціль може, у свою чергу, розкладатися на ще дрібніші завдання. Така ієрархія триває доти, поки всі елементи не будуть чітко визначені і придатні для реалізації.

Етапи побудови дерева цілей

1. Формулювання основної цілі.
2. Визначення основних підцілей, що забезпечують досягнення головної мети.
3. Подальший поділ кожної підцілі на ще конкретніші задачі (до рівня, де задачі можна безпосередньо реалізувати).
4. Перевірка логічної узгодженості між рівнями ієрархії.
5. Оцінка і встановлення пріоритетів серед підцілей.

Приклад застосування. Розглянемо приклад розробки інформаційної системи для управління підприємством.

Головна ціль: Розробити ефективну інформаційну систему.

Підцілі першого рівня:

- Розробка архітектури системи.
- Забезпечення захисту даних.
- Організація зручного інтерфейсу користувача.

Для "Розробки архітектури системи" можна визначити такі підцілі другого рівня:

- Аналіз вимог користувачів.
- Вибір технологічної платформи.
- Проєктування структури бази даних.

Для "Забезпечення захисту даних":

- Реалізація шифрування.
- Організація системи резервного копіювання.
- Впровадження механізмів аутентифікації.

Переваги методу дерева цілей: Забезпечує чітку структуру і логіку при плануванні. Дає змогу виявити всі важливі складові частини системи. Сприяє виявленню пріоритетів і ключових етапів роботи.

Недоліки: Можливі труднощі при побудові великого й деталізованого дерева для дуже складних систем. Потребує значних ресурсів для аналізу і деталізації.

Метод Делфі

Метод Делфі є одним із найефективніших методів колективного експертного прогнозування, який заснований на опитуванні групи експертів з метою отримання узгодженої думки щодо певної проблеми. Цей метод широко застосовується у сфері управління, планування, економіки, технічних наук та соціальних досліджень, особливо коли потрібно оцінити майбутні події, ризики чи ймовірність досягнення цілей.

Опис методу. Метод Делфі базується на багатоетапному анонімному опитуванні експертів. Після кожного раунду опитування результати аналізуються, узагальнюються, і експертам надається інформація про середні оцінки або загальне узгодження, щоб вони могли переглянути свої відповіді у наступному раунді. Завдяки анонімності знижується вплив авторитету або групового тиску, а повторність раундів дозволяє досягти більшої об'єктивності та точності.

Етапи методу

1. Формулювання проблеми та визначення цілей прогнозу.
2. Підбір групи експертів, які мають відповідні знання та досвід.
3. Розробка опитувальника, що містить питання про основну проблему.
4. Проведення першого раунду опитування з отриманням індивідуальних оцінок від кожного експерта.
5. Узагальнення результатів першого раунду та надання експертам інформації для перегляду оцінок у наступному раунді.
6. Повторення раундів до досягнення стабільності оцінок або узгодженої думки.

Метод Делфі є універсальним інструментом, який допомагає приймати обґрунтовані рішення у складних ситуаціях, особливо коли необхідно врахувати кілька точок зору і змодельовати можливі сценарії майбутнього розвитку.

Приклад застосування. Розглянемо приклад використання методу Делфі для прогнозування розвитку транспортної системи міста.

Проблема. Який транспортний засіб буде домінувати у міському середовищі через 20 років?

Етапи:

1. Експертам пропонується оцінити ймовірність домінування електромобілів, громадського транспорту, велосипедів тощо.
2. Після першого раунду результати узагальнюються, наприклад, більшість експертів прогнозує домінування електромобілів.
3. Експертам надаються узагальнені дані, і вони переглядають свої відповіді з урахуванням аргументів колег.
4. Після кількох раундів формується узгоджена думка, наприклад, що електромобілі та велосипеди будуть основними видами транспорту.

Переваги методу Делфі: Забезпечує високу точність прогнозів завдяки використанню знань експертів. Мінімізує вплив суб'єктивності та групового тиску. Дозволяє враховувати думки фахівців з різних галузей.

Недоліки: Потребує значних ресурсів для організації та проведення опитувань. Результати залежать від компетентності обраних експертів.

Метод аналізу ризиків

Метод аналізу ризиків є важливим інструментом системного аналізу, який дозволяє ідентифікувати, оцінювати та мінімізувати вплив потенційних ризиків на систему. Цей метод застосовується у багатьох сферах, включаючи управління проєктами, фінанси, інженерію, охорону здоров'я та інформаційні технології. Основна мета аналізу ризиків – забезпечення стабільності та ефективності системи, запобігання аваріям і зниження витрат, пов'язаних із наслідками ризиків.

Опис методу. Аналіз ризиків складається з кількох ключових етапів:

1. **Ідентифікація ризиків.** Визначення потенційних загроз для системи, включаючи внутрішні та зовнішні фактори.
2. **Оцінка ризиків.** Кількісна або якісна оцінка ймовірності виникнення кожного ризику та його потенційних наслідків для системи.
3. **Пріоритезація ризиків.** Встановлення пріоритетів на основі ступеня впливу ризиків, що дозволяє зосередитися на найбільш критичних.
4. **Розробка стратегій мінімізації ризиків.** Визначення заходів, спрямованих на усунення або зниження впливу ризиків (наприклад, резервування, покращення процесів або впровадження технологій).
5. **Моніторинг ризиків.** Постійний контроль і оновлення ризиків, оскільки система і зовнішні умови можуть змінюватися.

Метод аналізу ризиків є універсальним інструментом, який підходить для управління складними системами в умовах невизначеності. Його використання сприяє забезпеченню стійкості систем, підвищує їхню надійність та ефективність.

Приклади застосування методу аналізу ризиків

1. **Фінансовий аналіз ризиків.** У банківській сфері аналіз ризиків використовується для оцінки кредитного ризику клієнтів. Оцінюється ймовірність неповернення кредиту та потенційні втрати банку. На основі результатів розробляються політики резервування та страхування.
2. **Інженерія.** У будівництві аналіз ризиків дозволяє оцінити можливі аварії, наприклад, зсув ґрунту або недостатню міцність матеріалів. Вживаються заходи, такі як, вибір нових матеріалів або зміна проєктної документації.
3. **Інформаційні технології.** Аналіз кіберризиків для захисту даних компанії від хакерських атак. Впровадження заходів, таких як шифрування даних і регулярне оновлення програмного забезпечення.

Переваги методу аналізу ризиків: Дозволяє виявляти слабкі місця системи до їхнього прояву. Знижує витрати на ліквідацію наслідків, завдяки превентивним заходам. Покращує ефективність прийняття рішень через пріоритезацію ризиків.

Недоліки: Точність залежить від якості доступних даних і компетенції аналізаторів. Може бути ресурсозатратним для складних систем.

Метод моделювання

Метод моделювання є потужним інструментом системного аналізу, що використовується для дослідження складних систем через створення їхніх моделей. Моделі представляють спрощену версію реальної системи, яка зберігає основні характеристики та взаємозв'язки. Цей метод дозволяє вивчати поведінку системи в різних умовах, прогнозувати її функціонування та приймати обґрунтовані рішення.

Опис методу. Моделювання передбачає створення математичних, фізичних або комп'ютерних моделей системи, які використовуються для аналізу її властивостей. Метод дозволяє імітувати процеси, які відбуваються в системі, враховуючи зовнішні фактори, зміни параметрів та взаємодію елементів.

Етапи моделювання

1. Визначення мети моделювання (наприклад, аналіз роботи або оптимізація системи).

2. Вибір типу моделі (математична, фізична, комп'ютерна тощо).
3. Формулювання ключових характеристик системи та визначення параметрів моделі.
4. Побудова моделі та проведення симуляції.
5. Аналіз результатів моделювання та внесення коригувань у модель або систему.

Метод моделювання є універсальним інструментом, що використовується у багатьох сферах, включаючи інженерію, економіку, екологію, управління та планування. Завдяки цьому методу можна проводити обґрунтований аналіз, прогнозування та оптимізацію систем.

Приклади застосування методу моделювання

1. Моделювання водопостачальної системи

Мета: Оптимізація розташування насосних станцій для забезпечення рівномірного розподілу води в місті.

Процес: Створюється математична модель гідравлічної системи, враховуючи рівень споживання води та топографію території.

Результати: Визначаються оптимальні місця для розташування насосів, що мінімізують витрати енергії та покращують ефективність системи.

2. Моделювання транспортного потоку

Мета: Зниження заторів на дорогах.

Процес: Створюється комп'ютерна модель дорожньої мережі, що враховує рух транспорту, кількість автомобілів та години пік.

Результати: Розробляються рекомендації для оптимізації розташування світлофорів та введення нових маршрутів.

3. Моделювання теплового режиму будівлі

Мета: Покращення енергоефективності будівлі.

Процес: Математична модель теплових потоків у будівлі, яка враховує ізоляційні матеріали, температуру зовнішнього середовища та внутрішнє тепло.

Результати: Визначається необхідність удосконалення теплоізоляції або заміни систем опалення.

Переваги методу моделювання: Дозволяє аналізувати поведінку системи у різних сценаріях без реальних експериментів. Знижує витрати, пов'язані з тестуванням систем. Забезпечує глибоке розуміння процесів та взаємозв'язків у системі.

Недоліки: Точність моделі залежить від якості даних та побудови моделі. Вимагає значних ресурсів для створення складних моделей.

Метод системного графа

Метод системного графа є одним із інструментів системного аналізу, який дозволяє візуалізувати складну систему у вигляді графа. Граф складається з вершин (елементів системи) та ребер (зв'язків між елементами), що забезпечує наочність і структурованість аналізу. Цей метод широко використовується для дослідження взаємозв'язків у технічних, соціальних та інформаційних системах.

Опис методу. Системний граф є математичною моделлю, яка відображає структуру системи та взаємодії її компонентів. Кожна вершина графа представляє окремий елемент системи, а ребра описують взаємозв'язки між цими елементами. Залежно від специфіки системи граф може бути орієнтованим (з напрямками зв'язків) або неорієнтованим.

Етапи використання методу

1. Ідентифікація елементів системи (вершин графа).
2. Визначення зв'язків між елементами (ребер графа).
3. Побудова графа системи, де вершини і ребра відображають структуру системи.
4. Аналіз графа для визначення ключових компонентів та їх взаємозалежностей.
5. Розробка рекомендацій для оптимізації або реорганізації системи.

Метод системного графа є універсальним та ефективним для аналізу взаємозв'язків між компонентами систем, що дозволяє покращити її функціонування і забезпечити стабільність.

Приклади застосування методу системного графа

1. Аналіз виробничого процесу

Вершини: окремі етапи виробництва (наприклад, закупівля матеріалів, виготовлення деталей, збирання продукції).

Ребра: взаємозв'язки між етапами (наприклад, постачання матеріалів до складу чи передача деталей до цеху).

Аналіз графа дозволяє визначити "вузькі місця", які уповільнюють виробництво, і пропонує способи їх усунення.

2. Моделювання транспортної мережі

Вершини: транспортні вузли (станції, аеропорти).

Ребра: маршрути між вузлами.

Граф допомагає оптимізувати маршрути для зменшення часу доставки або витрат на перевезення.

3. Аналіз соціальних мереж

Вершини: користувачі соціальної мережі.

Ребра: зв'язки між користувачами (дружба, підписки).

Метод дозволяє визначити найбільш впливових користувачів (ключові вершини) або групи з високим рівнем взаємодії.

Переваги методу системного графа: Забезпечує наочне представлення структури системи. Дозволяє виявити ключові елементи та їх роль у системі. Сприяє розробці ефективних стратегій оптимізації.

Недоліки: Складність побудови графа для великомасштабних систем із багатьма компонентами. Потребує спеціалізованих інструментів для аналізу складних графів.

Метод аналізу за критеріями

Метод аналізу за критеріями є одним із основних методів системного аналізу, який використовується для оцінки альтернатив і прийняття оптимальних рішень. Цей метод дозволяє порівнювати варіанти за низкою визначених критеріїв, враховуючи їхню важливість та вплив на систему. Він широко застосовується у менеджменті, економіці, техніці, інженерії та інших сферах, де необхідно вибирати найкращий варіант із кількох можливих.

Опис методу. Метод полягає у визначенні критеріїв оцінки, за якими будуть порівнюватися альтернативи. Кожному критерію може бути присвоєна вага, що відображає його відносну важливість. Далі проводиться оцінка кожної альтернативи за цими критеріями, яка може бути якісною або кількісною. Після оцінки здійснюється агрегація результатів, що дозволяє вибрати оптимальний варіант.

Етапи аналізу за критеріями

1. Визначення цілі аналізу та постановка задачі.
2. Формулювання критеріїв, які мають значення для оцінки варіантів.

3. Присвоєння ваг кожному критерію (у разі необхідності).
4. Оцінка альтернатив за кожним із критеріїв.
5. Розрахунок загальної оцінки для кожної альтернативи (наприклад, використовуючи метод зваженої суми).
6. Вибір оптимального варіанту.

Метод аналізу за критеріями є ефективним інструментом для прийняття рішень у складних ситуаціях, де необхідно враховувати багато факторів.

Приклади застосування методу аналізу за критеріями

1. Вибір постачальника обладнання

Критерії: ціна, якість продукції, термін поставки, гарантійне обслуговування.

Вага: якість – найвища вага, ціна – середня вага, термін поставки – нижча вага.

Результат: постачальник з оптимальним співвідношенням критеріїв буде обраний.

2. Оцінка інвестиційних проєктів

Критерії: прибутковість, ризик, строк реалізації, соціальний вплив.

Вага: ризик – найважливіший критерій, строк реалізації – менш важливий.

Результат: вибір найбільш прибуткового проєкту з мінімальними ризиками.

3. Розподіл бюджетних коштів між проєктами

Критерії: значимість проєкту, ефективність використання коштів, тривалість виконання.

Вага: значимість – ключовий фактор.

Результат: фінансування найбільш важливих проєктів із високою ефективністю.

Переваги методу: Забезпечує системний та структурований підхід до оцінки альтернатив. Дозволяє враховувати множинні критерії одночасно. Підвищує обґрунтованість і прозорість процесу прийняття рішень.

Недоліки: Точність аналізу залежить від обґрунтованості критеріїв та ваг. Може вимагати значного обсягу даних для детальної оцінки.

Таким чином, системний аналіз дозволяє забезпечити структурний і послідовний підхід до вирішення складних задач, враховуючи багатовимірні взаємозв'язки. Системний аналіз – це комплексний науковий підхід до вирішення складних задач, що

передбачає дослідження, моделювання і оптимізацію систем різної природи. Методи системного аналізу застосовуються для аналізу складних взаємозалежностей і пошуку оптимальних рішень. Кожен із методів має свої переваги і недоліки, які варто враховувати під час їх вибору для конкретного проєкту.

Метод дерева цілей полягає у структуризації цілей системи шляхом їх ієрархічного поділу на підцілі. Основна мета знаходиться на вершині дерева, а всі проміжні завдання розташовуються нижче, забезпечуючи досягнення головної цілі. *Приклад:* Розробка програмного забезпечення — головна ціль. Підцілі: розробка інтерфейсу, проєктування архітектури, забезпечення безпеки даних.

Метод Делфі базується на анонімному опитуванні експертів для отримання узгодженого прогнозу. Результати кожного етапу узагальнюються, і експерти переглядають свої оцінки на основі спільної думки. *Приклад:* Прогнозування розвитку транспортної системи міста через 10 років.

Метод моделювання дозволяє створювати моделі реальних систем для аналізу їхньої поведінки в різних сценаріях. Використовуються математичні, комп'ютерні або фізичні моделі. *Приклад:* Моделювання водопостачальної системи для визначення оптимального розташування насосів.

Метод аналізу ризиків оцінює ймовірність виникнення ризиків і їхні наслідки для системи. Результати використовуються для розробки стратегій мінімізації ризиків. *Приклад:* Аналіз фінансових ризиків у проєкті будівництва.

Метод системного графа забезпечує представлення системи у вигляді графа, де вершини — це елементи системи, а ребра — зв'язки між ними. Дозволяє візуалізувати структуру і виявляти ключові елементи. *Приклад:* Побудова графа взаємозв'язків компонентів виробничої лінії.

Метод аналізу за критеріями використовує множину критеріїв для оцінки альтернатив і прийняття оптимальних рішень. *Приклад:* Вибір постачальника обладнання на основі критеріїв ціни, якості та термінів доставки.



Моделі Систем Масового Обслуговування: Вступ

Системи масового обслуговування (СМО) є важливим інструментом для оптимізації процесів у різних сферах, включаючи телефонію, банківську справу та логістику. СМО дозволяють ефективно управляти потоком вимог, мінімізувати час очікування та підвищувати задоволеність клієнтів.

Основні елементи СМО включають джерела вимог, чергу та обслуговуючі пристрої. Класифікація СМО здійснюється за різними ознаками, такими як тип черги, кількість каналів та дисципліна обслуговування. Наприклад, call-центр банку є типовою СМО, де кількість операторів, час очікування та інтенсивність потоку дзвінків визначають ефективність її роботи.

Мета презентації - надати всебічний огляд моделей СМО, показників їх ефективності та практичних аспектів застосування для оптимізації бізнес-процесів.

Показники Ефективності Систем Масового Обслуговування

Ефективність СМО оцінюється за допомогою ряду ключових показників. Серед них інтенсивність потоку вимог (λ) та інтенсивність обслуговування (μ), які визначають здатність системи обробляти вхідні запити. Важливими є також середня кількість вимог у системі (L) та в черзі (L_q), а також середній час перебування вимоги в системі (W) та в черзі (W_q).

Ймовірність відмови в обслуговуванні ($P_{відм}$) показує, наскільки часто система не може задовольнити запити клієнтів, а коефіцієнт використання каналів обслуговування (ρ) відображає ефективність використання ресурсів. Для розрахунку цих показників використовуються формули, специфічні для різних моделей СМО (наприклад, $M/M/1$, $M/M/c$).

Наприклад, для СМО $M/M/1$ з $\lambda = 10$ вимог/годину та $\mu = 12$ вимог/годину, можна розрахувати L , L_q , W та W_q , що дозволить оцінити поточну продуктивність та виявити потенційні "вузькі місця".



Середній час очікування



Кількість клієнтів в черзі



Відсоток відмов

Система Масового Обслуговування з Обмеженою Довжиною Черги

СМО з обмеженою довжиною черги характеризуються тим, що кількість місць для очікування обмежена. Це означає, що вимоги, які прибувають, коли черга вже заповнена, не приймаються в систему. Такі системи часто зустрічаються там, де фізичний простір обмежений, наприклад, на парковках або в приймальнях лікарів.

Обмеження довжини черги впливає на показники ефективності СМО. Зокрема, ймовірність відмови в обслуговуванні зростає, оскільки частина вимог відкидається. Однак, це може призвести до зменшення часу очікування для тих вимог, які потрапляють у чергу.

Розглянемо приклад автомийки з обмеженою чергою, де є 5 місць для очікування. Важливо розрахувати ймовірність того, що клієнт не зможе скористатися послугами через відсутність місць у черзі, щоб збалансувати завантаженість та якість сервісу.

Переваги

- Зменшення часу очікування
- Контроль завантаженості

Недоліки

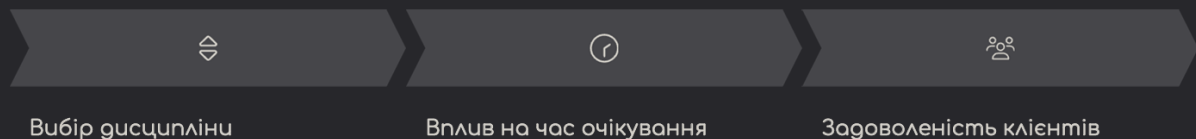
- Відмова клієнтам
- Втрата потенційного прибутку

Системи Масового Обслуговування з Очікуванням

У СМО з очікуванням вимоги, які не можуть бути негайно обслуговані, стають у чергу та очікують своєї черги. Важливою характеристикою таких систем є дисципліна черги, яка визначає порядок обслуговування вимог. Найпоширеніші дисципліни черги - FIFO (First In, First Out), LIFO (Last In, First Out) та пріоритетне обслуговування.

Дисципліна черги суттєво впливає на час очікування та інші показники ефективності. Наприклад, пріоритетне обслуговування може зменшити час очікування для важливих вимог, але збільшити його для інших. Прикладами СМО з очікуванням є лікарні швидкої допомоги та служби технічної підтримки.

Розглянемо вплив різних дисциплін черги (FIFO, пріоритет) на середній час очікування в лікарні швидкої допомоги. Пріоритетне обслуговування пацієнтів з невідкладними станами може врятувати життя, але збільшити час очікування для менш критичних випадків.



Система Масового Обслуговування з Обмеженим Часом Очікування

СМО з обмеженим часом очікування характеризуються тим, що клієнти покидають чергу, якщо час очікування перевищує певний ліміт. Це типово для сфер, де час є критичним фактором, таких як доставка їжі та онлайн-консультації. Обмеження часу очікування впливає на показники ефективності, зокрема, на ймовірність втрати клієнтів.

Важливо розрахувати ймовірність того, що клієнт покине чергу, не дочекавшись обслуговування, щоб оцінити втрати та розробити стратегії для зменшення часу очікування. Прикладом є СМО з обмеженим часом очікування для кур'єрської доставки.

Потрібно розрахувати кількість замовлень, що будуть скасовані через перевищення часу очікування, щоб оптимізувати логістику та забезпечити своєчасну доставку. Це дозволить підвищити задоволеність клієнтів та уникнути фінансових втрат.



Обмежений час очікування

Клієнти покидають чергу



Кур'єрська доставка

Своєчасна доставка



Скасовані замовлення

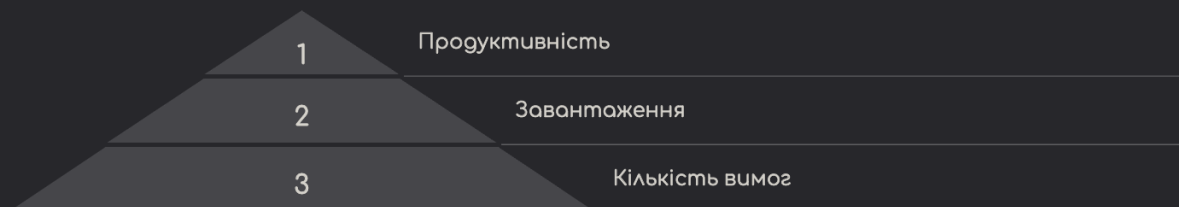
Оптимізація логістики

Замкнені Системи Масового Обслуговування

Замкнені СМО відрізняються від інших тим, що в них відсутній зовнішній потік вимог. Вимоги циркулюють всередині системи, переходячи від одного етапу обслуговування до іншого. Прикладами таких систем є комп'ютерні системи та виробничі лінії.

Ефективність замкнених СМО аналізується за допомогою показників продуктивності та завантаження. Важливо визначити оптимальну кількість вимог у системі, щоб уникнути перевантаження та забезпечити максимальну продуктивність. Час обробки запитів та завантаження процесора є ключовими показниками ефективності.

Розглянемо замкнену СМО в комп'ютерній системі, де кількість користувачів, час обробки запитів та завантаження процесора визначають продуктивність системи. Важливо збалансувати ці параметри, щоб забезпечити швидку та ефективну роботу системи для всіх користувачів.



Порівняльний Аналіз Моделей СМО

Різні моделі СМО мають свої унікальні характеристики та показники ефективності. Важливо провести порівняльний аналіз, щоб визначити оптимальну модель для конкретного застосування. Критерії вибору моделі включають тип черги, кількість каналів обслуговування, дисципліну черги та обмеження на час очікування.

Кожна модель має свої переваги та недоліки, які слід враховувати при виборі. Наприклад, модель $M/M/1$ є простою та легкою в аналізі, але може бути недостатньо точною для складних систем. Модель $M/M/c$ дозволяє враховувати наявність кількох каналів обслуговування, що підвищує її реалістичність.

Для зручності порівняння можна створити таблицю, в якій будуть вказані основні характеристики та показники ефективності різних моделей СМО. Це дозволить швидко оцінити переваги та недоліки кожної моделі та зробити обґрунтований вибір.

Модель	Тип черги	Кількість каналів	Дисципліна черги
$M/M/1$	Необмежена	1	FIFO
$M/M/c$	Необмежена	c	FIFO

Висновки та Практичне Застосування

Моделювання СМО є потужним інструментом для оптимізації процесів обслуговування та підвищення ефективності роботи підприємств. Вибір оптимальної моделі СМО дозволяє мінімізувати час очікування, збільшити пропускну здатність та підвищити задоволеність клієнтів.

Практичні рекомендації щодо застосування моделей СМО включають аналіз поточних процесів обслуговування, визначення ключових показників ефективності, вибір відповідної моделі СМО та впровадження змін на основі результатів моделювання. Подальші дослідження у сфері моделювання СМО можуть бути спрямовані на розробку більш складних та реалістичних моделей, які враховують специфічні особливості різних галузей.

У цій презентації ми розглянули основні аспекти моделювання СМО, показники їх ефективності та практичні аспекти застосування. Сподіваємося, що ця інформація буде корисною для вас у вашій роботі. Дякую за увагу!

Аналіз процесів

Оптимізація процесів обслуговування

Вибір моделі

Впровадження змін

Майбутні дослідження

Складніші моделі

Системи масового обслуговування (СМО) – це математичні моделі, які описують процеси обслуговування великої кількості заявок (клієнтів, повідомлень, викликів) у різних сферах: телекомунікації, транспорт, виробництво, ІТ. СМО дозволяють досліджувати ефективність роботи сервісних систем та знаходити оптимальні параметри їх функціонування. Основними елементами системи масового обслуговування є потік заявок, прилади або канали обслуговування та черга. Потік заявок описує надходження клієнтів чи повідомлень у систему, найчастіше моделюється за допомогою пуассонівського процесу. Канали обслуговування – це ресурси, які виконують роботу з обслуговування заявок, наприклад оператори, сервери чи каси. Якщо всі канали зайняті, заявки накопичуються у черзі, де вони очікують на обслуговування відповідно до певної дисципліни, найпоширенішою з яких є принцип FIFO – перший прийшов, перший обслужений.

У системах масового обслуговування важливим елементом є дисципліна черги, яка визначає порядок, у якому заявки потрапляють на обслуговування. Найпоширенішими дисциплінами є FIFO та LIFO. Принцип FIFO (*First In – First Out*) означає, що перший клієнт, який став у чергу, буде обслужений першим. Це найбільш справедливий і зрозумілий підхід, який широко застосовується у банківських касах, магазинах, транспортних вузлах та серверних системах. Він забезпечує рівність усіх заявок, але може бути неефективним у випадках, коли деякі з них потребують термінового обслуговування. Принцип LIFO (*Last In – First Out*) передбачає, що останній клієнт, який став у чергу, буде обслужений першим. Такий підхід використовується рідше, але має переваги у ситуаціях, коли нові заявки мають пріоритет над старими. Прикладом є організація роботи стеків у програмуванні, де останній доданий елемент обробляється першим. Недоліком LIFO є те, що заявки, які надійшли раніше, можуть залишатися необробленими тривалий час. Таким чином, вибір дисципліни черги залежить від характеру задачі: FIFO забезпечує справедливість і послідовність, тоді як LIFO дозволяє швидко реагувати на нові події, але може призвести до затримки старих заявок.

Системи масового обслуговування класифікують за кількістю каналів (одноканальні та багатоканальні), за наявністю черги (з чергою або без неї), за дисципліною обслуговування (FIFO, LIFO, пріоритетні системи) та за типом потоку заявок (пуассонівський, детермінований, марковський). Основними показниками ефективності роботи таких систем є середня довжина черги, середній час очікування заявки, ймовірність відмови, завантаженість каналів та пропускна здатність системи.

Для дослідження систем масового обслуговування застосовують аналітичні методи, зокрема формули Ерланга та моделі Маркова, чисельні методи наближених розрахунків, а також імітаційне моделювання, що дозволяє відтворювати роботу системи на комп'ютері. Практичне значення СМО полягає у можливості оптимізації роботи телефонних станцій, мереж передачі даних, транспортних вузлів, серверних систем та виробничих процесів. Таким чином, системи масового обслуговування є універсальним інструментом для аналізу та підвищення ефективності роботи складних багатоканальних систем у сучасних умовах.

Ключовим завданням теорії масового обслуговування є визначення показників ефективності, які дозволяють оцінити якість функціонування системи. До основних показників ефективності належать: середня інтенсивність вхідного потоку заявок, середня інтенсивність обслуговування, коефіцієнт завантаження системи, ймовірність простою обслуговуючих пристроїв, ймовірність відмови в обслуговуванні, середня довжина черги, середній час очікування заявки в черзі, середній час перебування заявки в системі, а також пропускна здатність системи. Ці характеристики дозволяють аналізувати баланс між витратами на ресурси та якістю обслуговування.

Система масового обслуговування з обмеженою довжиною черги характеризується тим, що кількість заявок, які можуть одночасно перебувати в системі, є скінченною. Якщо черга досягає максимально допустимої довжини, нові заявки не

приймаються і вважаються втраченими. Такі моделі є характерними для буферів комп'ютерних мереж, систем пакетної передачі даних та телефонних станцій. Основним показником ефективності для таких систем є ймовірність відмови, яка визначає частку заявок, що не отримали обслуговування.

Системи масового обслуговування з очікуванням припускають, що всі заявки, які надходять у систему, стають у чергу та очікують обслуговування незалежно від її довжини. У таких моделях, за умови стабільності системи, відмови відсутні, але зростає середній час очікування. Прикладом є класична модель $M/M/1$ або $M/M/c$ з нескінченною чергою. Основна увага в аналізі таких СМО приділяється середній довжині черги та часу очікування, що є критичними показниками якості сервісу.

Система масового обслуговування з обмеженим часом очікування є більш реалістичною моделлю, у якій заявка залишає систему, якщо її час перебування в черзі перевищує певне граничне значення. Такі системи широко застосовуються для моделювання поведінки клієнтів у сервісних службах, коли користувачі можуть відмовитися від послуги через надмірне очікування. Основними характеристиками тут є ймовірність втрати заявки через середній фактичний час очікування та ефективна пропускна здатність системи.

Замкнені системи масового обслуговування відрізняються тим, що кількість заявок у системі є сталою. Після завершення обслуговування заявка не залишає систему остаточно, а повертається у вхідний потік після певного часу. Такі моделі застосовуються для аналізу комп'ютерних систем з обмеженою кількістю процесів, транспортних систем із фіксованим парком транспортних засобів, а також виробничих систем замкненого циклу. Для замкнених СМО характерними показниками ефективності є середній час обігу заявки, продуктивність системи та ступінь завантаження обслуговуючих пристроїв.

Таким чином, теорія масового обслуговування надає потужний математичний апарат для аналізу та оптимізації реальних систем, дозволяючи знаходити компроміс між ресурсними витратами та якістю обслуговування залежно від обраної моделі СМО та її параметрів.

Функціональне моделювання та оцінювання ризиків динамічних бізнес-процесів складних систем

Аналіз, функціональне моделювання та оцінювання ризиків бізнес-процесів організаційно-економічних і виробничо-технічних систем є важливими складовими системного аналізу, спрямованими на підвищення ефективності функціонування складних соціотехнічних об'єктів. Бізнес-процеси таких систем характеризуються багаторівневою структурою, динамічністю, наявністю людського фактора та впливом зовнішнього середовища, що зумовлює необхідність використання формалізованих методів дослідження.

Аналіз бізнес-процесів передбачає ідентифікацію основних функцій системи, визначення вхідних і вихідних потоків ресурсів, інформації та фінансів, а також встановлення взаємозв'язків між підрозділами та виконавцями. У процесі аналізу виявляються «вузькі місця», надлишкові операції, дублювання функцій та потенційні джерела ризиків. Для організаційно-економічних систем особлива увага приділяється узгодженості управлінських рішень, а для виробничо-технічних систем – надійності технологічних процесів і обладнання.

Функціональне моделювання є основним інструментом формалізації бізнес-процесів, який дозволяє подати систему у вигляді сукупності функцій та операцій із чітко визначеними входами, виходами, керуючими впливами та механізмами виконання. Найбільш поширеними є методи структурного аналізу, зокрема SADT та IDEF0, які забезпечують ієрархічну декомпозицію системи від загального рівня до детальних операцій. Функціональні моделі сприяють кращому розумінню логіки роботи системи та є основою для її оптимізації й реінжинірингу.

Важливою частиною дослідження бізнес-процесів є оцінювання ризиків, яке полягає у виявленні можливих небажаних подій, оцінюванні ймовірності їх виникнення та потенційних наслідків. У системному аналізі ризики розглядаються як результат взаємодії внутрішніх і зовнішніх факторів, включаючи організаційні, технічні, економічні та інформаційні чинники. Для кількісного та якісного аналізу ризиків використовуються методи експертних оцінок, сценарного аналізу, аналізу «відмов і наслідків», а також імітаційне та стохастичне моделювання.

Методологія системного аналізу відіграє ключову роль у розв'язанні задач прогнозування, управління, аналізу та проєктування динамічних процесів. Вона ґрунтується на принципах цілісності, ієрархічності, множинності моделей та урахування невизначеності. У задачах прогнозування системний аналіз дозволяє враховувати динаміку станів системи, вплив випадкових факторів і можливі сценарії розвитку. В управлінні забезпечується формування обґрунтованих рішень з урахуванням багатьох критеріїв ефективності та ризиків.

Важливе місце в сучасному системному аналізі займає розробка та дослідження функціональних моделей із використанням CASE-засобів проєктування складних систем. CASE-технології забезпечують автоматизацію процесів аналізу, проєктування та документування систем, підтримують побудову діаграм, моделей даних і процесів, а також їх узгодження між різними рівнями абстракції. Використання CASE-засобів підвищує якість проєктних рішень і зменшує ймовірність помилок на ранніх етапах життєвого циклу системи.

CASE-засоби проєктування складних систем (Computer-Aided Software/System Engineering) – це програмні інструменти, призначені для підтримки системного аналізу, моделювання, проєктування, документування та супроводу складних інформаційних, організаційно-економічних і виробничо-технічних систем протягом усього їх життєвого циклу. CASE-засоби забезпечують формалізований опис системи на різних рівнях

абстракції та підтримують узгодженість між вимогами, моделями, проектними рішеннями і реалізацією.

Основні функції CASE-засобів:

- аналіз і формалізація вимог до системи;
- побудова функціональних, структурних та інформаційних моделей;
- ієрархічна декомпозиція системи та бізнес-процесів;
- підтримка різних нотацій моделювання;
- автоматизоване документування;
- перевірка узгодженості моделей.

Методи структурного аналізу систем орієнтовані на функціональний опис і декомпозицію процесів, що є особливо ефективним на початкових етапах проектування. Водночас об'єктно-орієнтована методологія проектування дозволяє описувати систему з позиції взаємодіючих об'єктів, які поєднують дані та методи їх обробки. Такий підхід забезпечує кращу масштабованість, повторне використання компонентів і відповідність сучасним вимогам до розроблення програмних та інформаційних систем.

Отже, поєднання системного аналізу, функціонального та об'єктно-орієнтованого моделювання, оцінювання ризиків і використання CASE-засобів створює методологічну основу для ефективного дослідження, проектування та управління складними організаційно-економічними і виробничо-технічними системами в умовах динамічних змін і невизначеності.

Тест 5

1. Метод дерева цілей. Реалізація методу. Приклад застосування.
2. Метод Делфі. Реалізація методу. Приклад застосування.
3. Метод моделювання. Реалізація методу. Приклад застосування.
4. Метод аналізу ризиків. Реалізація методу. Приклад застосування.
5. Метод системного графа. Реалізація методу. Приклад застосування.
6. Метод аналізу за критеріями. Реалізація методу. Приклад застосування.
7. Показники ефективності систем масового обслуговування.
8. Система масового обслуговування з обмеженою довжиною черги.
9. Системи масового обслуговування з очікуванням.
10. Система масового обслуговування з обмеженим часом очікування.
11. Замкнені системи масового обслуговування.
12. Аналіз, функціональне моделювання та оцінювання ризиків бізнес процесів організаційно-економічних і виробничо-технічних систем
13. Методологія системного аналізу в задачах прогнозування, управління, аналізу та проектування динамічних процесів.

14.Розробка та дослідження функціональних моделей CASE-засобами проектування складних систем, методами структурного аналізу систем, об'єктно-орієнтованої методології проектування

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Стрелковська І.В. Теорія ймовірностей та випадкові процеси (для фахівців у галузі ІТ-технологій) / І.В. Стрелковська, В.М. Паскаленко. – Одеса: ОНАЗ, 2018. – 384 с.
2. Катренко А.В. Системний аналіз: підручник – Львів: «Новий Світ – 2000», 2024. – 396 с.
3. Методичні вказівки до лабораторних робіт з курсу “Системний аналіз” / М.Є.Фриз., Б.Б.Млинко. – Тернопіль: Вид-во ТНТУ імені І.Пулюя, 2020. – 37 с.
4. Литвинов А. Л. Л64 Теорія систем масового обслуговування : навч. посібник / А. Л. Литвинов ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2018. – 141 с.