

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

ІНСТИТУТ ІНФОРМАТИКИ ТА РАДІОЕЛЕКТРОНІКИ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК І ТЕХНОЛОГІЙ
(повне найменування інституту, факультету)

КАФЕДРА СИСТЕМНОГО АНАЛІЗУ ТА ОБЧИСЛЮВАЛЬНОЇ
МАТЕМАТИКИ
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістра

(ступінь вищої освіти)

на тему Аналіз оптимального розв'язку задачі про надійність електронного пристрою

Виконав: студентка 2 курсу, групи КНТ – 810м

Спеціальності 124 – Системний аналіз

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інтелектуальні технології та прийняття рішень

в складних системах

Залата Ю.О.

(прізвище та ініціали)

Керівник

Подковаліхіна О.О.

(прізвище та ініціали)

Рецензент

Стеганцев Є.В.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут Інформатики та радіоелектроніки
 Факультет Комп'ютерних наук і технологій
 Кафедра Системного аналізу та обчислювальної математики
 Ступінь вищої освіти магістр
 Спеціальність 124 – Системний аналіз
 (код і найменування)
 Освітня програма (спеціалізація) Інтелектуальні технології та прийняття рішень
в складних системах
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри проф. Г.В. Корніч
 « 21 » грудня 2021 року

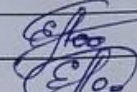
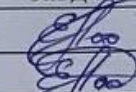
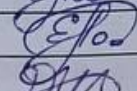
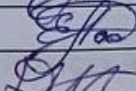
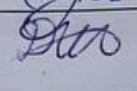
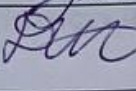
ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Залати Юлії Олександрівни

(прізвище, ім'я, по батькові)

- Тема проєкту (роботи) Аналіз оптимального розв'язку задачі про надійність електронного пристрою
 керівник проєкту (роботи) доцент Подковаліхіна Олена Олександрівна,
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
 затверджені наказом закладу вищої освіти від «10» грудня 2021 року № 490
- Строк подання студентом проєкту (роботи) 21 грудня 2021 року
- Вихідні дані до проєкту (роботи) умова задачі про надійність, перелік літературних джерел згідно теми дослідження.
- Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Проаналізовано публікації та дослідження останнього часу за заданою темою. Створено програмний код для рішення задачі про надійність з нормально розподіленими параметрами методом перебору даних, що розраховує оптимальний план та значення надійності, і заносить дані до Excel. Отримано та проаналізовано розв'язок задачі з вхідними параметрами з 10%, 20%, 30% відхиленнями для різних критеріїв оптимальності при 1000,4000,8000 та 12000 запусках.
- Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1	доцент Подковаліхіна О.О		
2	доцент Подковаліхіна О.О		
Нормоконтроль	ст. викл. Широкоград Д.В.		

7. Дата видачі завдання « 1 » вересня 2021 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Сформулювати мету та основні завдання дипломної роботи.	01.09.2021-13.09.2021	
2	Опрацювати літературу та існуючі дослідження за темою роботи.	14.09.2021-04.10.2021	
3	Оптимізація програми для вирішення задачі.	05.10.2021-25.10.2021	
4	Розрахунки та аналіз даних.	26.10.2021-22.11.2020	
5	Оформлення пояснювальної записки.	23.11.2021-06.12.2021	
6	Попередній захист дипломного проекту та отримання рецензій.	14.12.2021	
7	Захист дипломної роботи.	21.12.2021	

Студентка


(підпис)

Залата Ю.О.
(прізвище та ініціали)

Керівник проекту (роботи)


(підпис)

Подковаліхіна О.О.
(прізвище та ініціали)

РЕФЕРАТ

Дипломна робота: 42 с., 15 рис., 15 табл., 1 дод., 24 літературних джерел.

Об'єкт дослідження – задача про рюкзак в умовах статистичної невизначеності.

Предмет дослідження – задача про надійність з нормальними вхідними параметрами.

Мета роботи – аналіз оптимального розв'язку задачі про надійність з нормально розподіленими параметрами.

Методи дослідження – метод повного перебору.

В дипломній роботі було створено програмний код для розрахунку задачі про надійність з нормально розподіленими параметрами методом перебору даних, який автоматично заносить дані до Excel. Було розглянуто задачу про надійність з нормальними вхідними параметрами, отримано оптимальні розв'язки задачі для різної кількості експериментів, проаналізовано отримані плани за декількома критеріями оптимальності.

ЗАДАЧА ПРО НАДІЙНІСТЬ, СТАТИСТИЧНА НЕВИЗНАЧЕНІСТЬ, КРИТЕРІЇ ПТИМАЛЬНОСТІ, MATLAB, EXCEL.

ЗМІСТ

ЗАВДАННЯ	2
РЕФЕРАТ	4
ВСТУП.....	6
1 АНАЛІЗ ПУБЛІКАЦІЙ ТА ДОСЛІДЖЕНЬ	7
1.1 Актуальність задачі про надійність	7
1.2 Поняття динамічного програмування.....	8
1.3 Області застосування моделей динамічного програмування.....	10
1.4 Огляд задачі про рюкзак	12
1.5 Загальні методи розрахунку надійності	13
2 ЗАДАЧА ПРО НАДІЙНІСТЬ ЕЛЕКТРОННОГО ПРИСТРОЮ	19
2.1 Постановка завдання	19
2.2 Програмна реалізація задачі про надійність з нормально розподіле- ними параметрами.....	20
2.2 Результати розрахунків задачі про надійність.....	23
2.2 Аналіз розрахунків задачі про надійність з нормально розпо- діленими параметрами.....	31
ВИСНОВОК.....	36
ПЕРЕЛІК ПОСИЛАНЬ	37
ДОДАТОК А Приклад програмної реалізації	40

ВСТУП

Проблема надійності продовжує залишатися однією з основних у сучасній техніці. Це пояснюється тим, що постійно ускладнюються розв'язувані завдання та одночасно підвищуються вимоги до надійності їх виконання. Інженери, фізики та математики доклали чимало спільних зусиль на розробку сучасної теорії надійності. Найкращим методом її вивчення та закріплення теоретичних знань є вирішення практичних завдань. Для того щоб на початковій стадії створення пристрою впевнитись в тому, що виріб буде відповідати встановленим вимогам, та мати можливість заздалегідь визначити можливі помилки, проводять випробування на надійність.

Надійність не є постійною величиною, вона змінюється в залежності від конструкції приладу та його компонентів. Оскільки завдання конструкторів розробити прилад з достатньою величиною надійності, виникає необхідність знайти оптимальні рішення задачі про надійність в умовах статистичної невизначеності і проаналізувати їх згідно з заданими критеріями оптимальності.

1 АНАЛІЗ ПУБЛІКАЦІЙ ТА ДОСЛІДЖЕНЬ

1.1 Актуальність задачі про надійність

Техніка сьогодення - породження науково-технічного прогресу, вона включає в себе величезну кількість найскладніших елементів, кожен з яких може зламатися (відмовити), і це може призвести до загибелі людей та створених ними дорогих і складних пристроїв. Для того, щоб це не сталося існує наука - теорія надійності, що дає можливість заздалегідь розрахувати появи в заданому пристрої "відмови", аварії. Ці розрахунки базуються на застосуванні теорії ймовірності.

Приблизні розрахунки дозволяють оцінити базовий потенціал для забезпечення необхідної вам надійності для вашого продукту. Цей розрахунок використовується для перевірки вимог до надійності, встановлених замовником у технічному завданні при порівнянні надійності окремих версій продукту на ранніх етапах розробки.

Вся промислова продукція, в тому числі напівпровідникові прилади та інтегральні мікросхеми характеризуються таким параметром як якість, що представляє собою сукупність властивостей продукції, які обумовлюють її придатність задовольняти певні потреби відповідно до призначення. На початку 50-х років розвиток техніки призвів до створення складних радіоелектронних приладів. Фахівці в цей період зіткнулися з частими відмовами апаратури за рахунок її недосконалості та неякісних деталей. Науково обґрунтований підхід був необхідний для збереження високої працездатності.

Враховуючи важливість сучасних технологічних інструментів у житті людини, вимоги надійності зростають, тому цьому питанню приділяють багато часу на всіх етапах виготовлення.[1].

1.2 Поняття динамічного програмування

Динамічне програмування — це метод розв'язання задачі шляхом її поділу на кілька однакових підзадач.

Розглянемо систему, що спостерігається протягом кінцевого або нескінченного часового горизонту, розділеного на періоди або етапи. На кожному етапі спостерігається стан системи, і має бути прийнято рішення (або дія) щодо системи. Рішення впливає (детерміновано чи стохастично) на стан, який слід спостерігати на наступному етапі, і залежно від стану та прийнятого рішення отримується негайна винагорода. Очікувана загальна винагорода від поточного етапу до кінця горизонту планування виражається функцією цінності. Співвідношення між функцією значення на сучасному етапі та функцією на наступному виражається функціональним рівнянням. Оптимальні рішення залежно від стадії та стану визначаються назад крок за кроком як рішення, що максимізують праву частину функціонального рівняння. Цей спосіб визначення оптимальної політики заснований на принципі оптимальності Беллмана, який говорить: «Оптимальна політика має властивість, що незалежно від початкового стану та початкового рішення, інші рішення повинні становити оптимальну політику щодо стану впливає з першого рішення» (Беллман, 1957 р.). Протягом наступних років Беллман опублікував кілька книг на цю тему (Bellman, 1961; Bellman and Dreyfus, 1962; Bellman and Kalaba, 1965). Книги були з великим ентузіазмом, і очікувалося, що метод буде рішенням дуже широкого кола проблем прийняття рішень реального світу. Очікування були настільки великі, і вони були висунуті з такою впевненістю, що Джонстон (1965) іронічно порівняв динамічне програмування з новою релігією. Інші розглядали метод як досить тривіальний обчислювальний пристрій. [2].

Усі класи задач динамічного програмування вирішуються методом рекурентних співвідношень, які складені за принципом оптимальності,

розробленого Р. Беллманом у 1953 р — один з провідних фахівців у галузі математики та обчислювальної техніки. Оптимальна стратегія має властивість, що хоч би якими були початковий стан і початкове рішення, наступні рішення мають становити оптимальний курс дій щодо стану, отриманого в результаті першого рішення. Суттєвою особливістю підходу динамічного програмування є структурування задач оптимізації на кілька етапів, які вирішуються послідовно по одному етапу. Хоча кожна проблема одноетапна вирішується як звичайна оптимізаційна задача, її вирішення допомагає визначити характеристики наступної одноетапна задача в послідовності. Часто етапи представляють різні періоди часу в горизонті планування проблеми. Наприклад, проблему визначення рівня запасів окремого товару можна сформулювати як динамічну програму. Змінна рішення — це сума для замовлення на початку кожного місяця; мета — мінімізувати загальні витрати на замовлення та ведення запасів; Основне обмеження вимагає, щоб попит на продукт був задоволений. Якщо ми можемо замовляти лише на початку кожного місяця і хочемо оптимальну політику замовлення для у наступному році ми могли б розкласти проблему на 12 етапів, кожен з яких представляє рішення щодо замовлення на початку відповідного місяця. Іноді етапи не мають тимчасових наслідків. Наприклад, у представленій простій ситуації у попередньому розділі проблема визначення маршрутів мінімальної затримки від будинків с пасажирів на паркінги в центрі міста була сформульована як динамічна програма. Змінна рішення була чи вибрати вгору чи вниз на будь-якому перетині, а етапи процесу були визначені як число перехрестя. Задачі, які можна сформулювати як динамічні програми з етапами, які не мають часові наслідки часто важко розпізнати. Кінцевою загальною характеристикою підходу динамічного програмування є розвиток рекурсивного оптимізаційна процедура, яка розв'язує загальну N-етапну задачу, спочатку розв'язуючи одноетапну задачі і послідовно включаючи один етап за раз і розв'язуючи одноетапні задачі до загального знайдено оптимум. Ця процедура може бути заснована на зворотному процесі індукції, де проводиться перша стадія підлягає аналізу — це останній етап проблеми, і проблеми

вирішуються, повертаючись на один етап за раз, поки включені всі етапи. В якості альтернативи, рекурсивна процедура може бути заснована на процесі прямого індукції, де першим етапом, який необхідно вирішити, є початкова стадія проблеми, а проблеми вирішуються в подальшому по одному етапу, доки не будуть включені всі етапи. У певних налаштуваннях проблеми тільки одна з цих індукції можуть бути застосовані процеси (наприклад, у більшості проблем, пов'язаних із невизначеністю, допускається лише зворотна індукція). Очевидними прикладами послідовних рішень у тваринництві є заміна тварин (важливо через регулярні проміжки часу розглядати, чи слід замінити поточний актив чи його слід зберігати на додатковий період), запліднення та медичне лікування. Динамічне програмування є відповідним інструментом, але якщо риси тварин добре визначені і їх точна поведінка з часом відома заздалегідь, існують інші методи, які можна застосувати для визначення оптимальних рішень аналітичним шляхом. З іншого боку, якщо на ознаки тварини впливають випадкові зміни в часі та серед тварин, рішення залежатимуть від поточних спостережень за ознаками. У цьому випадку динамічне програмування є очевидною технікою, яку потрібно використовувати для визначення оптимальних рішень і політик. Домінуючою сферою застосування у тваринництві було вирішення проблеми заміни тварин як окремо, так і у зв'язку із заплідненням та медичним лікуванням. Проте очікується, що останні методологічні розробки розширять сферу застосування.[3].

1.3 Области застосування моделей динамічного програмування

Модель динамічного програмування використовують для вирішення задач по типу управління запасами, розподіл дефіцитних капітальних вкладень, принцип календарного планування виробництва, вибір методів проведення рекламних компаній для знайомства покупця з продукцією, систематизація методів пошуку цінного ресурсу.

У підручнику [4] розглянуто розв'язання задачі про оптимальне розподілення однорідного ресурсу та завдання про оптимальне завантаження транспортного засобу неподільними предметами методом динамічного програмування.

Сутність інвестицій в умовах ринкової економіки полягає у поєднанні двох сторін інвестиційної діяльності: витрат ресурсу та отримання результату. Інвестиції здійснюються з метою отримання доходу (результату) у майбутньому і стають марними, якщо вони цього доходу (результату) не приносять. Зазвичай підприємство-інвестор інвестує діяльність якогось одного підприємства, а кількох підприємств різного профілю. У такому разі перед будь-яким підприємством-інвестором постає питання: «Як визначити, яку кількість ресурсів потрібно виділити кожному інвестованому підприємству для максимізації прибутку?». Саме знаходження відповіді це питання і розроблено метод розподілу ресурсів між підприємствами в роботі [5].

У роботі [6] розглянуті операції, економічні показники, схемо утворення прибутку, структуру зв'язку економічних і математичних методів, методи і моделі вивчення, аналізу та прогнозування попиту, аналізу та формування товарних запасів, лінійного програмування, теорії ігор, теорії графів та мережевого планування, теорії масового обслуговування, призначені для вирішення задач розподілу ресурсів, планування продажу та купівлі товарів, комівояжера, призначення, перевезення вантажів, аналізу стійкості комерційної діяльності підприємства, конфліктних ситуацій, фінансово-комерційних операцій, оцінки комерційних ризиків.

В роботі [7] на прикладах показано, як вирішувати економічні завдання, які зводяться до динамічного програмування. Завдання вирішуються не лише вручну, а й пропонуються відповідні програми у системі MATLAB. Ці програми дозволяють вирішувати реальні економічні завдання, що містять багато тисяч змінних. Програми є досить простими, тому вони можуть бути легко адаптовані та для вирішення аналогічних економічних завдань.

1.4 Огляд задачі про рюкзак

Завдання про рюкзак є актуальним і досить затребуваним з погляду його застосування в реальному житті.

Задача може бути використана в наступних областях:

- оптимальне управління в економіко-фінансових сферах;
- при складанні телекомунікаційних систем обмежених за масою/вартістю;
- в системах підтримки управління портфелем з метою пошуку найкращого балансу між ризиками та ефективністю;
- вибір багажів для оптимального завантаження транспорту, тощо.

Є багато вчених, які досліджували цю задачу, наприклад, Беллман Р. [8], Скляр Б. [9], Таха Х.[10], Лежньов А.[11], Грешилов А. [12], Солтіс М.[13], тощо.

Математична постановка задачі формулюється так: нехай є n предметів. Для кожного i -го предмета задана його вага $p_i > 0$ та вартість (цінність) $c_i > 0$ $i = 1, 2, \dots, n$. Вказано обмеження на максимальну вагу рюкзака – P . Кожен x_i може набувати лише одне з двох значень: $x_i = 1$, якщо i -й предмет упаковують в рюкзак, або $x_i = 0$, в іншому випадку.

Потрібно вибрати із заданої множини предметів набір з максимальною сумарною вартістю $\sum_{i=1}^n c_i x_i$ при одночасному дотриманні обмеження на сумарну вагу знайденого набору $\sum_{i=1}^n p_i x_i \leq P$. [14].

Існує кілька різновидів задачі про ранець, відмінності між якими полягають в умовах, накладених на рюкзак, предмети або їх вибір:

- Рюкзак 0-1 або Класична задача – маємо можливість тільки один раз використати один предмет;
- Обмежений рюкзак – маємо можливість декілька разів використати кожен предмет;

- Необмежений рюкзак – маємо можливість безліч разів використовувати кожен предмет;
- Рюкзак з мультिवибором – є кілька класів предметів, з яких можна брати одного представника, причому деякі речі мають більший пріоритет, ніж інші;
- Мультиплікативний рюкзак – є кілька рюкзаків з різною місткістю;
- Багатовимірний рюкзак – є більше одного обмеження на рюкзак.

Існує ряд методів, що дозволяють знаходити розв'язання задачі про ранець. В літературі [15] ми можемо розглянути дослідження за трьома алгоритмами, які були призначені для вирішення задачі про ранець. Першим є метод динамічного програмування та другим метод гілок і меж, вони дозволяють знайти максимально точний результат. Евристичний метод являється третім та дає лиш наближений розв'язок. Також в даній роботі розміщена порівняльна характеристика трьох методів та надано рекомендації щодо їх використання на практиці.

В роботі [16] задачу про рюкзак вирішують методом динамічного програмування.

В роботі [17] можемо розглянути реалізацію жадібним та генетичним алгоритмом, а також метод перебору та їх порівняння між собою.

1.5 Загальні методи розрахунку надійності

У підручнику [18] описано основи теорії надійності, способи обчислення показників надійності роботи, метрологічну надійність, програмну надійність тощо.

Щоб пояснити надійність, розглянемо дані методи:

- інтегральні рівняння;
- диференціальні рівняння;
- за графом можливих станів пристрою.

Перший метод можна використовувати для розрахунку надійності будь-якого пристрою при будь-якому розподілі безвідмовного часу роботи та часу відновлення.

Таким чином, індекс надійності визначається шляхом складання та розв'язування інтегральних або інтегрально - диференціальних рівнянь. Складаючи інтегральні рівняння, відмічають невеликі часові інтервали. Для цих інтервалів ми розглядаємо складні події, які відбуваються, коли кілька подій відбуваються одночасно.

Ці рівняння легко створити, але важко вирішити. Іноді рішення необхідно отримати чисельно за допомогою комп'ютера. У зв'язку з цим інтегральних рівнянь на даний момент не дуже поширений.

Метод диференціальних рівнянь був створений з припущенням, що на час від відмови до відновлення впливає розподіл показника.

Розраховуючи показник надійності, складається та розв'язується система диференціальних рівнянь для ймовірностей станів (рівнянь Колгоморова). Зазвичай прийнято для мінімізації витрат праці на розрахунки, висувати наступні припущення:

- відразу відновлюються об'єкти, які відмовили;
- кількість оновлень не обмежена;
- засіб контролю має ідеальну надійність.

Математична модель представлена у вигляді графіка стану. На цьому графіку можливий стан пристрою в разі несправності компонента зображено колом. Напрямок, який може змінюватися з одного в інший стан - стрілка. Інтенсивність переходів вказано поруч зі стрілкою. (Рисунок 1.1).

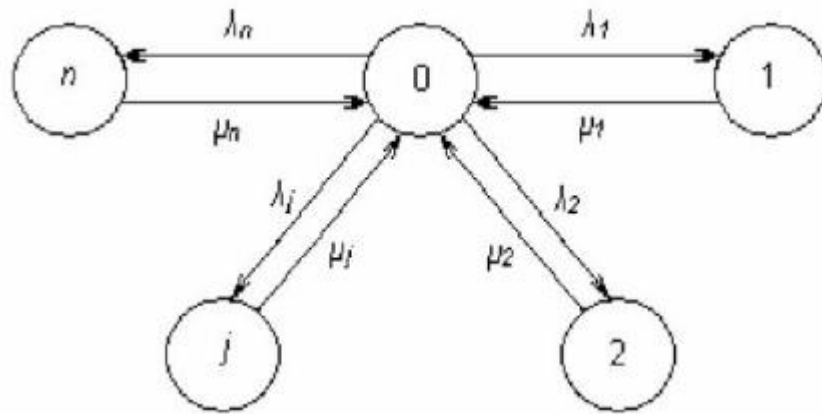


Рисунок 1.1 – Граф станів

Отримані системи диференціальних рівнянь доповнюються нормованою умовою:

$$\sum_{j=0}^{n+1} P_j(t) = 1$$

де $P_j(t)$ – вірогідність що пристрій знаходиться в j -му стані;

$n + 1$ – кількість вірогідних станів.

Потім розбираємо на дві під множини множину станів:

а) n_1 – підмножина стану, нероботоздатного приладу;

б) n_2 – підмножина стану, роботоздатного приладу.

Розглянемо формулою для функції готовності:

$$G(t) = \sum_{j=0}^n P_j(t)$$

де $P_j(t)$ – вірогідність що пристрій знаходиться в j -му стані, який являється робото здатним.

До роботи [19] включені завдання з розрахунку надійності невідновлюваних та відновлюваних виробів при основному та резервному з'єднанні елементів, а також завдання щодо оцінки надійності за даними про їх

відмови, отримані за результатами випробувань. У кожному розділі наведено короткі відомості з теорії надійності, типові приклади та завдання з відповідями. Посібник включає як прості завдання, корисні при початковому вивченні теорії надійності, так і більш складні, вирішення яких сприятиме виробленню практичних навичок.

Прикидний розрахунок надійності ґрунтується на наступних припущеннях:

- всі елементи виробу рівнонадійні;
- інтенсивність відмов всіх виробів не залежить від часу, $\lambda = \text{const}$;
- відмова всього прибору може настати за рахунок відмови одного з елементів.

Прикидковий розрахунок надійності виконується:

- при перевірці вимог щодо надійності, які були вказані в технічному завданні;
- при розрахунку нормативних даних щодо надійності окремих блоків та пристроїв;
- при розрахунку мінімально допустимого рівня надійності;
- для порівняння надійності перних варіантів виробу на етапі ескізного проектування.

Орієнтувальний розрахунок надійності враховує вплив на надійність лише кількості та типів, що входять до системи елементів та ґрунтується за наступними правилами:

- всі елементи цього типу рівнонадійні;
- всі елементи працюють у номінальному (нормальному) режимі, зазначеними технічною умовою;
- інтенсивності відмов всіх елементів залежить від часу;
- відмови елементів виробу є подіями випадковими та незалежними;
- усі елементи виробу працюють одночасно.

Щоб розрахувати надійність потрібно мати наступні дані:

- за яким видом з'єднані елементи;
- типи елементів, що входять у виріб та число елементів кожного типу;
- величина інтенсивності відмови елементів, що входять у виріб, вибір для кожного типу елементів здійснюється за відповідними таблицями. [19].

Посібник [20] містить основні відомості по курсу «Надійність приладів систем». Викладено основні поняття та визначення теорії та надійності. Розглянуто випадки аналізу за статистичними даними, аналітичного визначення характеристик надійності, послідовного з'єднання елементів у систему, постійного резервування, не відновлюваних систем відновленням. Наведено приклади розрахунку на надійності систем.

В роботі [21] викладено основні поняття, визначення та критерії, що використовуються в теорії надійності, методи розрахунку надійності нерезервованої та резервованої апаратури та надійності автоматичних систем управління, технічної діагностики апаратури, планування та обробки результатів випробувань на надійність та забезпечення експлуатаційної надійності автоматичних систем управління.

В матеріалі [22] вводяться поняття інформаційної відмови та інформаційної надійності навігаційної системи. Розглядаються процедури розрахунку інформаційної надійності навігаційної системи, засновані на апроксимації її похибки випадковими дифузійними та стрибкоподібними марківськими процесами та які використовують методи теорії викидів та вирішення рівняння Фоккера-Планка-Колмогорова. Пропонуються та досліджуються ефективні алгоритми контролю та діагностики інформаційних відмов та порушень навігаційної системи. Алгоритми засновані на методах нелінійної багатоальтернативної фільтрації та передбачають використання банку фільтрів Калмана. Наводяться приклади реалізації контролю та діагностики для інерційних та супутникових навігаційних систем.

Робота [23] присвячена питанням надійності автоматизованих систем. Описано особливості оцінки та розрахунку надійності. Значну увагу приділено дослідженню надійності за даними наближення до відмов. Викладено практичні

питання застосування моделювання для передбачення надійності. Описуються характеристики резервованих систем та методи розрахунку їх надійності. Викладено практичні питання формування показників надійності систем при їх проектуванні. Розглянуто шляхи підвищення надійності автоматизованих систем при їх проектуванні, виготовленні та експлуатації.

2 ЗАДАЧА ПРО НАДІЙНІСТЬ ЕЛЕКТРОННОГО ПРИСТРОЮ

2.1 Постановка завдання

Розглянемо наступну задачу про надійність електронного пристрою. Конструюється електронний прилад, що складається з j основних компонентів. Компоненти мають послідовне з'єднання, це свідчить про те що якщо одна з них вийде з ладу, то це призведе до відмови всього пристрою. Є можливість покращити надійність пристрою дублюванням компоненти, яка може складатися з x_i блоків. Максимальна ціна всього пристрою обмежена у грошових одиниць. Кожна компонента j , що складається з x_i блоків, характеризується вартістю C_{ij} та надійністю R_{ij} . При вирішенні задачі потрібно підібрати вдалу кількість блоків x_j в компоненті j , таким чином, щоб не вийти за межі вказаної вартості та щоб надійність приладу була максимальною.

На практиці надійність компонентів приладу може змінюватися в деякому діапазоні. Було прийнято рішення розв'язувати задачу в умовах невизначеності.

Для прикладу маємо задачу з наступними вхідними даними: $y = 10, n = 3, m = 3$, C_{ij} та R_{ij} підпорядковуються закону нормального розподілу з 10%, 20% та 30% відхиленнями від значень, наведених в таблиці 2.1.

Таблиця 2.1 – Вхідні дані

x_i	$j = 1$		$j = 2$		$j = 3$	
	C_1	R_1	C_2	R_2	C_3	R_3
1	1	0.6	3	0.7	2	0.5
2	2	0.8	5	0.8	4	0.7
3	3	0.9	6	0.9	5	0.9

2.2 Програмна реалізація задачі про надійність з нормально розподіленими параметрами

Було прийнято рішення розв'язувати задачу методом перебору, використовуючи середовище для програмування MATLAB. MATLAB (від matrix laboratory) — мова програмування, що інтерпретується, призначена для вирішення обчислювальних завдань. Особливостями мови є орієнтація на роботу з матрицями (звідси неофіційне гасло мови — Think vectorized, тобто, «думай векторами»), готові до реалізації низки корисних алгоритмів, багаті можливості візуалізації даних та взаємодії з іншими мовами [24].

Вхідні дані :

- y – загальна вартість приладу,
- n – кількість компонентів в приладі,
- m – максимальна кількість блоків в компоненті,
- C – вартість,
- R – надійність.

Вхідну матрицю R_{ij} , що підпорядковується нормальному закону розподілу з 10%, 20%, 30% відхиленням від значень, задаємо використовуючи функцію `normrnd(mu,sigma)` пакету Statistics Toolbox пакету MATLAB. Функція `normrnd` призначена для генерації чисел, розподілених за нормальним законом, який відрізняється від стандартного нормального розподілу. Вона має такі синтаксичні конструкції:

- `normrnd(mu, sigma)` - повертає одне випадкове число;
- `normrnd(mu, sigma, M, N)` – повертає матрицю чисел розмірністю $M \times N$, де μ – математичне очікування, σ – стандартне відхилення.

Використавши вище вказану функцію, потрібно визначити значення SIGMA для кожного числа всіх варіантів відхилення та вписати матриці.

Для прикладу проаналізуємо розрахунок одного з елементів матриці R для 10% відхилення по нормальному закону. Розглянемо елемент, який

спочатку дорівнює 0,7. 10% відхилення від числа 0,7 вказує на те, що значення повинно змінюватися в межах від $0,7 * 0,9$ до $0,7 * 1,1$ (від 0,63 до 0,77). На Рисунку 2.1 наведено приклад підбору SIGMA.

```
>> R = normrnd (0.7,0.0120,1000,1000);
>> max(max(R))

ans =

    0.7600

>> min(min(R))

ans =

    0.6392
fx >> |
```

Рисунок 2.1 – Скріншот прикладу знаходження SIGMA для R

Виконавши всі розрахунки ми отримали наступні матриці:

- Матриця R для 10% відхилення:

$$R = [\text{normrnd}(0.6,0.0125) \text{normrnd}(0.7,0.0120) \text{normrnd}(0.5,0.0120) \\ \text{normrnd}(0.8,0.0179) \text{normrnd}(0.8,0.0179) \text{normrnd}(0.7,0.0120) \\ \text{normrnd}(0.9,0.0190) \text{normrnd}(0.9,0.0190) \text{normrnd}(0.9,0.0190)] ;$$

- Матриця R для 20% відхилення:

$$R = [\text{normrnd}(0.6,0.026) \text{normrnd}(0.7,0.028) \text{normrnd}(0.5,0.020) \\ \text{normrnd}(0.8,0.034) \text{normrnd}(0.8,0.034) \text{normrnd}(0.7,0.028) \\ \text{normrnd}(0.9,0.036) \text{normrnd}(0.9,0.036) \text{normrnd}(0.9,0.036)] ;$$

- Матриця R для 30% відхилення:

$$R = [\text{normrnd}(0.6,0.039) \text{normrnd}(0.7,0.044) \text{normrnd}(0.5,0.029) \\ \text{normrnd}(0.8,0.05) \text{normrnd}(0.8,0.05) \text{normrnd}(0.7,0.044) \\ \text{normrnd}(0.9,0.055) \text{normrnd}(0.9,0.055) \text{normrnd}(0.9,0.055)] ;$$

Далі потрібно провести розрахунки C за таким же алгоритмом що і матрицю R .

Для прикладу проаналізуємо розрахунок елемента матриці C для 30% відхилення по нормальному закону. Розглянемо елемент, який спочатку дорівнює 6. 30% відхилення від числа 6 вказує на те що значення повинно змінюватися в межах від $0,7*6$ до $1,3*6$, точніше, від 4,2 до 7,8.

```
>> R=normrnd(6,0.38,1000,1000);
>> min(min(R))

ans =

    4.2376

>> max(max(R))

ans =

    7.8261

fx >> |
```

Рисунок 2.2 – Скріншот прикладу знаходження SIGMA для C

Значення 0.38 ідеально підходить під наш діапазон (Рисунок 2.2). Таким же чином було підібрано всі значення.

- Матриця C для 10% відхилення виглядає наступним чином:

$$C = [\text{normrnd}(1,0.02) \text{ normrnd}(3,0.064) \text{ normrnd}(2,0.04) \\ \text{normrnd}(2,0.04) \text{ normrnd}(5,0.1) \text{ normrnd}(4,0.082) \\ \text{normrnd}(3,0.064) \text{ normrnd}(6,0.12) \text{ normrnd}(5,0.1)];$$

- Матриця C для 20% відхилення виглядає наступним чином:

$$C = [\text{normrnd}(1,0.041) \text{ normrnd}(3,0.143) \text{ normrnd}(2,0.084) \\ \text{normrnd}(2,0.084) \text{ normrnd}(5,0.2) \text{ normrnd}(4,0.15) \\ \text{normrnd}(3,0.143) \text{ normrnd}(6,0.25) \text{ normrnd}(5,0.2)];$$

- Матриця С для 30% відхилення виглядає наступним чином:

$$C = [\text{normrnd}(1,0.045) \text{normrnd}(3,0.18) \text{normrnd}(2,0.12) \\ \text{normrnd}(2,0.12) \text{normrnd}(5,0.33) \text{normrnd}(4,0.25) \\ \text{normrnd}(3,0.18) \text{normrnd}(6,0.38) \text{normrnd}(5,0.33)];$$

В подальшому за допомогою мови MATLAB було створено програмний код. Його цінність в тому, що після того як розрахунки закінчуються в пакеті MATLAB, всі дані заносяться до файлу .csv, який створюється автоматично за вказаним нами шляхом. (див. додаток А).

При написанні коду було враховано що x_1 вказує на кількість блоків в кожній компоненті, яка за умовою може містити 1-3 блоки, відповідно x_1 може приймати значення від 1 до 3, x_2 и x_3 так само. Сформулювали вектор вірогідних планів та вказуємо умову, при якій вони перебираються та заносять до файлу Excel, провівши 1000, 4000, 8000 та 12000 запусків програмного коду для кожного з варіантів відхилень за для того щоб вислідити зміни результатів та бути більш впевненим в тому, що результат буде якісним.

2.2 Результати розрахунків задачі про надійність

Так як всі розрахунки були занесені до файлу Excel, то результат ми можемо побачити на зображенні таблиць (Рисунок 2.3-2.14). Червоним виділені можливі варіанти планів, значення нижче – надійність.

Спочатку розглянемо результати для 10% відхилення.

	H	I	J	K	L		H	I	J	K	L	M
1	132	133	211	212	213	979					0,50448	
2					0,516063	980					0,489627	
3					0,532295	981					0,498286	
4					0,502766	982					0,510686	
5					0,507974	983					0,495559	
6					0,484384	984					0,534772	
7					0,520404	985					0,530068	
8					0,531897	986					0,539427	
9					0,493343	987					0,504262	
10					0,49557	988					0,491844	
11					0,504806	989					0,525668	
12					0,503824	990					0,515191	
13					0,484304	991					0,504919	
14					0,491897	992					0,519677	
15					0,504292	993					0,492089	
16					0,510557	994					0,517512	
17					0,508787	995					0,52353	
18					0,478835	996					0,505597	
19					0,497096	997					0,515755	
20					0,525525	998					0,489185	
21					0,508011	999					0,52966	
22					0,485829	1000					0,505449	
23					0,488059	1001					0,507818	ваці:

Рисунок 2.3 – Скріншот таблиці значень вибірки для 1000 запусків для 10% відхилення

Можемо зробити висновок, що для 10% відхилення, результат 213 являється 100% (Рисунок 2.3)

	L	T
1	213	312
2	0,517192	0,454131
3	0,509061	
4	0,465738	
5	0,50998	
6	0,492481	
7	0,545939	
8	0,504754	
9	0,489054	
10	0,47333	
11	0,525377	
12	0,507842	
13	0,493359	
14	0,480715	
15	0,501394	
16	0,504398	
17	0,5179	
18	0,493691	
19	0,50236	
20	0,50983	

Рисунок 2.4 – Скріншот таблиці значень вибірки для 4000 запусків для 10% відхилення

При 4000 запусків з 10% відхилення ми отримали 2 плани 213 та 312.

	L	T
1	213	312
2	0,536453	0,483315
3	0,51234	0,470286
4	0,506445	0,474391
5	0,525784	0,477734
6	0,483286	0,489504
7	0,505603	0,491256
8	0,502673	
9	0,481117	
10	0,492395	
11	0,483607	
12	0,542342	
13	0,485456	
14	0,512823	
15	0,502271	
16	0,512991	

Рисунок 2.5 – Скріншот таблиці значень вибірки для 8000 запусків для 10% відхилення

При 8000 запусків з 10% відхилення бачимо що результат не змінився від попереднього, два плани 213 та 312.

	L	T		L	T
1	213	312	11973	0,489617	
2	0,485943	0,476508	11974	0,497985	
3	0,519287	0,480986	11975	0,468723	
4	0,511715	0,466511	11976	0,520878	
5	0,496164	0,461948	11977	0,519884	
6	0,49915	0,490864	11978	0,508909	
7	0,478514	0,456845	11979	0,516743	
8	0,497244	0,463965	11980	0,499589	
9	0,532569		11981	0,499212	
10	0,507358		11982	0,481265	
11	0,504842		11983	0,500203	
12	0,526518		11984	0,475496	
13	0,507689		11985	0,503446	
14	0,518406		11986	0,496013	
15	0,501209		11987	0,502951	
16	0,509149		11988	0,510862	
17	0,516618		11989	0,507832	
18	0,504178		11990	0,482516	
19	0,489834		11991	0,512799	
20	0,52754		11992	0,508952	
21	0,481167		11993	0,542046	
22	0,522857		11994	0,482757	
23	0,498716		11995		

Рисунок 2.6 Скріншот таблиці значень вибірки для 12 000 запусків 10%

При 12000 запусків з 10% відхилення бачимо що результат не змінився від попередніх, також два плани 213 та 312. Можна зробити висновок що при збільшенні кількості запусків результат може змінюватись, але не дуже вагомо.

Далі розглянемо розрахунки для 20% відхилення.

L	M	N	O	P	Q	R	S	T
213	221	222	223	231	232	233	311	312
0,538721	0,655447	0,446626					0,615568	0,474932
0,499285	0,447542							0,440681
0,487596	0,624123							0,527207
0,540639								0,491235
0,555745								0,534637
0,443214								0,501531
0,462794								0,509525
0,548052								0,565178
0,517512								0,404622
0,489587								0,445057
0,527717								0,624845
0,547474								0,463821
0,544331								0,478184
0,54372								0,55263
0,498402								0,517005
0,528873								0,457636
0,501111								0,487635

Рисунок 2.7 – Скріншот таблиці значень вибірки для 1000 запусків 20%

На Рисунку 2.7 зображена вибірка для 20% відхилення, порівнюючи з результатами 10%, можемо спостерігати, що кількість варіантів плану збільшилась. Маємо 5 варіантів плану.

	L	M	N	S	T	V
1	213	221	222	311	312	321
2	0,484851	0,4723447	0,446626	0,5143729	0,479132	0,442318
3	0,477632	0,4895139		0,5064935	0,507629	0,466941
4	0,492809	0,4589151			0,609806	
5	0,503714	0,5544503			0,449876	
6	0,493378	0,5258249			0,48475	
7	0,498361	0,4642529			0,452441	
8	0,503071	0,4597661			0,410719	
9	0,485726	0,4523576			0,537292	
10	0,513507				0,541865	
11	0,569407				0,415874	
12	0,471441				0,459015	
13	0,529023				0,458837	
14	0,519888				0,474784	
15	0,493813				0,489569	
16	0,504503				0,466173	
17	0,499886				0,475389	
18	0,551437				0,492752	
19	0,550818				0,49072	
20	0,516479				0,455685	

Рисунок 2.8 – Скріншот таблиці значень вибірки для 4000 запусків 20% Порівнюючи з 1000 запусків, зараз додався шостий варіант виборки 321.

	E	F	L	M	N	S	T	V	Y
1	122	123	213	221	222	311	312	321	331
2	0,468236	0,438464	0,503015	0,503717	0,436688	0,464951	0,453115	0,451119	0,389891
3	0,509127	0,494793	0,544062	0,664423	0,484156	0,511431	0,501649	0,444107	0,417638
4	0,469851		0,461946	0,473486	0,432503	0,44013	0,502344	0,58457	0,433889
5	0,465917		0,461473	0,493051	0,471277	0,502986	0,468512	0,419318	0,436944
6	0,464757		0,503094	0,481664	0,428033		0,403979		
7	0,487355		0,470589	0,524319	0,535161		0,535167		
8			0,491602	0,502424	0,450869		0,479485		
9			0,524057	0,430511	0,479727		0,554651		
10			0,556595	0,549763	0,520445		0,502041		
11			0,557953	0,459622			0,466919		
12			0,548608	0,661869			0,418104		
13			0,484455	0,48009			0,510665		
14			0,543419	0,411936			0,466408		
15			0,532102	0,471608			0,539927		
16			0,513292	0,445294			0,418538		
17			0,490957				0,511156		
18			0,489364				0,439582		

Рисунок 2.9 – Скріншот таблиці значень вибірки для 8000 запусків 20% Маємо 9 варіантів плану.

	E	F	G	L	M	N	P	S	T	V	Y
1	122	123	131	213	221	222	231	311	312	321	331
2	0,496655	0,43846388	0,419254	0,559624	0,519747	0,452354	0,40232	0,441103	0,425597	0,367048	0,481702
3	0,443963	0,49479301		0,500603	0,470776	0,485976	0,461096	0,523106	0,491147	0,430679	
4	0,441296			0,516206	0,494044	0,476262	0,407028	0,439761	0,47365	0,420464	
5	0,468673			0,46774	0,453611	0,460645	0,404709	0,470442	0,507091	0,406226	
6	0,454822			0,53863	0,476947	0,48476		0,500729	0,473898		
7				0,555688	0,474019	0,539827		0,453065	0,436314		
8				0,478681	0,423313	0,514213		0,37348	0,457094		
9				0,491357	0,50295	0,531586		0,410321	0,482535		
10				0,502842	0,466269	0,496813		0,472234	0,499938		
11				0,466419	0,486849	0,423395		0,473063	0,53873		
12				0,480614	0,473639			0,446214	0,470389		
13				0,505231	0,455273			0,450265	0,39568		
14				0,514018	0,50791			0,522158	0,466167		
15				0,516326	0,430612			0,458801	0,444385		
16				0,448608	0,488355			0,424964	0,47327		
17				0,485834	0,455041			0,484917	0,427934		
18				0,544575	0,446599				0,472848		
19				0,507691					0,522345		
20				0,464209					0,472013		
21				0,470816					0,504733		
22				0,522111					0,474081		
23				0,550342					0,461417		

Рисунок 2.10 – Скріншот таблиці значень вибірки для 12000 запусків 20%
 При збільшенні запусків, отримали 11 варіантів плану.
 Розглянемо результати для 30% відхилення.

	E	F	G	H	I	L	M	N	O	P	S	T	V	W	Y
1	122	123	131	132	133	213	221	222	223	231	311	312	321	322	331
2	0,405359	0,56868	0,408732	0,491454	0,490398	0,464509	0,473664	0,515248	0,613032	0,401372	0,574312	0,460709	0,442238	0,521136	0,449826
3		0,459175	0,42411	0,389397	0	0,549402	0,510904	0,518533		0,436028	0,362106	0,44145	0,412279		0,435856
4		0,463043	0,463212			0,473224	0,462779	0,575221		0,565463	0,485997	0,595155	0,415549		0,424968
5		0,532174				0,562109	0,45682	0,533232		0,476674	0,573833	0,467271	0,411252		0,459481
6						0,631826	0,445996	0,440357			0,461385	0,596125	0,461471		0,428585
7						0,640015	0,64786	0,562341				0,560083	0,527476		0,468092
8						0,509168	0,581578	0,441005				0,46104	0,408095		
9						0,516603	0,407213	0,453114				0,501757	0,455975		
10						0,523876	0,540857	0,435268				0,467339			
11						0,472463	0,553089	0,442021				0,40424			
12						0,469068	0,508365					0,539949			
13						0,47997	0,457422					0,507236			
14						0,490679	0,434706					0,553525			
15						0,527184	0,585365					0,633254			
16						0,597651						0,404592			
17						0,514407						0,523151			
18						0,51128						0,532863			
19						0,529425						0,538325			
20						0,571724						0,465841			
21						0,499648						0,421788			
22						0,50669						0,415177			
23						0,5493						0,455362			

Рисунок 2.11 – Скріншот таблиці значень вибірки для 1000 запусків 30%
 Для 30% відхилення ми отримали 15 варіантів плану.

A	D	E	F	G	H	I	L	M	N	O	P	Q	S	T	V	W	Y
1	121	122	123	131	132	133	213	221	222	223	231	232	311	312	321	322	331
2	0,464168	0,504036	0,53886	0,461253	0,45733	0,554693	0,505008	0,447042	0,5112	0,639434	0,382198	0,484404	0,480206	0,440044	0,487966	0,546498	0,456216
3	0,411417	0,471523	0,424104	0,568243	0,503802	0,543674	0,41337	0,55613	0,493612	0,641377	0,431054		0,54135	0,645939	0,430227		0,414572
4		0,465423	0,504542	0,583769	0,451573		0,569284	0,497293	0,459375	0,685514	0,391635		0,437018	0,555092	0,395542		0,416378
5		0,448664	0,404044		0,521717		0,585622	0,461276	0,423413		0,489427		0,481391	0,431589	0,379116		0,471469
6		0,600763	0,549096		0,459716		0,456758	0,470635	0,50417		0,387269		0,568368	0,517142	0,449521		0,446574
7		0,448884	0,449081		0,448959		0,483939	0,61887	0,531502		0,393308		0,399021	0,557381	0,511248		0,476565
8		0,432346	0,489949		0,495403		0,480165	0,562757	0,542729		0,435979		0,461489	0,489475	0,388908		0,491497
9		0,419164	0,586642				0,491006	0,484759	0,526367		0,405761		0,484302	0,591578	0,413894		0,48312
10		0,568134	0,517399				0,504284	0,447115	0,59775		0,536702		0,478769	0,588633	0,450645		0,449933
11		0,518648	0,515106				0,448187	0,465569	0,543116		0,427351		0,462008	0,628196	0,510212		0,436048
12		0,505189	0,466082				0,50352	0,515513	0,551195		0,511038		0,640119	0,445351	0,494336		0,540405
13		0,483534	0,399375				0,582843	0,57443	0,5108		0,417095		0,576321	0,55316	0,403867		
14			0,568513				0,49924	0,466044	0,405723		0,44193		0,55997	0,466871	0,436639		
15			0,539985				0,470833	0,368913	0,460537		0,417316		0,470961	0,38172	0,371413		
16			0,503898				0,519696	0,486989	0,476932		0,413214		0,535429	0,549784	0,404632		
17			0,500756				0,571681	0,536128	0,40364		0,474186		0,520986	0,471589	0,531201		
18							0,563118	0,581748	0,451251		0,442954		0,611729	0,411949	0,426484		
19							0,458303	0,584344	0,479372		0,407275		0,587144	0,4554	0,450763		
20							0,522039	0,569453	0,575756		0,517735		0,435103	0,443392	0,399519		
21							0,475019	0,6012	0,416452		0,454816		0,609511	0,388346	0,636849		
22							0,585669	0,444718	0,4649		0,396969		0,457497	0,534565	0,411365		Активация V
23							0,44927	0,59613	0,503424		0,422187		0,451865	0,457722	0,411107		Чтобы активирс

Рисунок 2.12 – Скріншот таблиці значень вибірки для 4000 запусків 30%
Маємо 17 варіантів.

D	E	F	G	H	I	L	M	N	O	P	Q	S	T	V	W	Y	
1	121	122	123	131	132	133	213	221	222	223	231	232	311	312	321	322	331
2	0,464168	0,486726	0,53937	0,500238	0,384718	0,48523	0,575954	0,534914	0,522	0,56499	0,388343	0,496499	0,449207	0,409358	0,423636	0,519182	0,495611
3	0,411417	0,409036	0,528721	0,461253	0,45733	0,554693	0,390985	0,588177	0,454822	0,639434	0,408477	0,484404	0,516495	0,59434	0,58633	0,546498	0,448709
4		0,504036	0,53437	0,568243	0,503802	0,543674	0,632193	0,604658	0,589436	0,641377	0,419704		0,414458	0,567749	0,507204	0,521136	0,398903
5		0,471523	0,436583	0,583769	0,451573	0,490398	0,431796	0,500349	0,435379	0,685514	0,571681		0,45795	0,503118	0,502133	0,433154	0,471327
6		0,465423	0,588649	0,408732	0,521717	0,576207	0,533302	0,530119	0,532878	0,613032	0,550266		0,538443	0,462142	0,416023	0,496247	0,390871
7		0,448664	0,456779	0,42411	0,459716	0,540708	0,498186	0,522831	0,461051		0,467665		0,550234	0,520479	0,534176		0,496618
8		0,600763	0,499395	0,463212	0,448959		0,531092	0,497132	0,455355		0,418841		0,463114	0,461542	0,414803		0,623031
9		0,448884	0,463986		0,495403		0,450089	0,457779	0,509334		0,418333		0,480206	0,446727	0,51906		0,452509
10		0,432346	0,53886		0,491454		0,482306	0,549713	0,501549		0,382198		0,54135	0,598056	0,484266		0,370606
11		0,419164	0,424104		0,389397		0,569715	0,487181	0,485246		0,431054		0,437018	0,504336	0,513608		0,459052
12		0,568134	0,504542		0,491293		0,532427	0,508938	0,466352		0,391635		0,481391	0,596907	0,361128		0,441555
13		0,518648	0,404044				0,548441	0,649376	0,488715		0,489427		0,568368	0,561448	0,437742		0,456216
14		0,505189	0,549096				0,480681	0,50774	0,504799		0,387269		0,399021	0,479789	0,423426		0,414572
15		0,483534	0,449081				0,516759	0,557737	0,49837		0,393308		0,461489	0,579083	0,394113		0,416378
16		0,405359	0,489949				0,469892	0,427942	0,497199		0,435979		0,484302	0,484716	0,56154		0,471469
17		0,442711	0,586642				0,59808	0,738244	0,422473		0,405761		0,478769	0,454419	0,393565		0,446574
18		0,454927	0,517399				0,532938	0,447042	0,467718		0,536702		0,462008	0,521318	0,425815		0,476565
19		0,518884	0,515106				0,405522	0,55613	0,496974		0,427351		0,640119	0,455646	0,408066		0,491497
20		0,524018	0,466082				0,667294	0,497293	0,58227		0,511038		0,576321	0,399731	0,439435		0,48312
21		0,504772	0,399375				0,522459	0,461276	0,431722		0,417095		0,55997	0,543029	0,468198		0,449933
22		0,530137	0,568513				0,52024	0,470635	0,506066		0,44193		0,470961	0,518897	0,386509		0,436048
23		0,464572	0,539985				0,541807	0,61887	0,541124		0,417316		0,535429	0,570343	0,487966		0,540405

Рисунок 2.13 – Скріншот таблиці значень вибірки для 8000 запусків 30%
Бачимо що при збільшенні запусків результат не змінився від попереднього,
маємо 17 планів.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	121	122	123	131	132	133	213	221	222	223	231	232	311	312	321	322	331
2	0,464168	0,552493	0,449571	0,500238	0,527017	0,510038	0,388788	0,525054	0,566863	0,567509	0,51733	0,522728	0,519375	0,464023	0,455955	0,519182	0,536149
3	0,411417	0,416901	0,479834	0,461253	0,476154	0,48523	0,518661	0,472688	0,519918	0,564044	0,576332	0,496499	0,425704	0,503778	0,38667	0,546498	0,519786
4		0,424141	0,48288	0,568243	0,384718	0,554693	0,54444	0,451071	0,500499	0,654192	0,388343	0,484404	0,499702	0,477276	0,393546	0,521136	0,485427
5		0,454776	0,485155	0,583769	0,45733	0,543674	0,441454	0,466109	0,504553	0,56499	0,408477		0,45853	0,46194	0,420054	0,433154	0,427814
6		0,456867	0,456945	0,408732	0,503802	0,490398	0,505695	0,5707	0,449739	0,639434	0,419704		0,531431	0,391623	0,370501	0,496247	0,577778
7		0,496233	0,475498	0,42411	0,451573	0,576207	0,519048	0,542294	0,547388	0,641377	0,571681		0,514763	0,623184	0,413415	0,452522	0,423263
8		0,474121	0,481238	0,463212	0,521717	0,540708	0,559867	0,489626	0,44576	0,685514	0,550266		0,510705	0,429989	0,509993		0,495761
9		0,486726	0,471304	0,493276	0,459716		0,44697	0,52545	0,496356	0,613032	0,467665		0,529615	0,644409	0,373955		0,495611
10		0,409036	0,48151		0,448959		0,621921	0,426497	0,505037	0,613387	0,418841		0,495426	0,402847	0,524547		0,448709
11		0,504036	0,504857		0,495403		0,495866	0,511899	0,482888		0,418333		0,444936	0,527298	0,425215		0,398903
12		0,471523	0,53937		0,491454		0,556705	0,479374	0,512773		0,382198		0,675845	0,398258	0,468674		0,471327
13		0,465423	0,528721		0,389397		0,56136	0,551989	0,441883		0,431054		0,583103	0,505642	0,387438		0,390871
14		0,448664	0,53437		0,491293		0,578213	0,42785	0,529337		0,391635		0,487427	0,608957	0,455009		0,496618
15		0,600763	0,436583		0,442177		0,544104	0,382254	0,44011		0,489427		0,449207	0,57114	0,542674		0,623031
16		0,448884	0,588649		0,466316		0,522425	0,598307	0,436955		0,387269		0,516495	0,49912	0,408003		0,452509
17		0,432346	0,456779				0,472449	0,479331	0,494492		0,393308		0,414458	0,53046	0,4192		0,370606
18		0,419164	0,499395				0,653945	0,620201	0,489216		0,435979		0,45795	0,578743	0,405313		0,459052
19		0,568134	0,463986				0,512049	0,492088	0,530317		0,405761		0,538443	0,45488	0,388898		0,441555
20		0,518648	0,53886				0,51737	0,50639	0,461905		0,536702		0,550234	0,517682	0,423636		0,456216
21		0,505189	0,424104				0,517068	0,525842	0,384996		0,427351		0,463114	0,62451	0,58633		0,414572
22		0,483534	0,504542				0,410059	0,542168	0,441871		0,511038		0,480206	0,580987	0,507204		0,416378
23		0,405359	0,404044				0,614538	0,534914	0,506023		0,417095		0,54135	0,539682	0,502133		0,471469

Рисунок 2.14 – Скріншот таблиці значень вибірки для 12000 запусків 30% Маємо 17 планів. В випадку 30% можемо зробити висновок, що результат може змінюватися при збільшенні запусків.

2.2 Аналіз розрахунків задачі про надійність з нормально розподіленими параметрами

Для того щоб рекомендувати план, який зможе забезпечити максимальну надійність, потрібно привести аналіз за наступними критеріями:

- кількість відсотків появи кожного з плану;
- середньоарифметична надійність;
- ймовірність того, що буде отримано результат вище ніж 0,5.

Спочатку розглянемо дані з 10% відхиленням.

Таблиця 2.2 – Результати за обраними критеріями для 10% відхилення 1000 запусків

	213
Відсоток появи	100 %
Середнє значення	0,503129
Вище ніж 0,5	57%

Таблиця 2.3 – Результати за обраними критеріями для 10% відхилення 4000 запусків

	213	312
Відсоток появи	99,975%	0,025 %
Середнє значення	0,50313	0,45413
Вище ніж 0,5	56,81 %	0%

Таблиця 2.4 – Результати за обраними критеріями для 10% відхилення 8000 запусків

	213	312
Відсоток появи	99,925%	0,075%
Середнє значення	0,504086	0,481081
Вище ніж 0,5	58,85 %	0

Таблиця 2.5 – Результати за обраними критеріями для 10% відхилення 12000 запусків

	213	312
Відсоток появи	99,94%	0,06%
Середнє значення	0,504036	0,471089
Вище ніж 0,5	58,82%	0%

Проаналізувавши плани для 10% відхилення, можемо зробити висновок, що план 213 має найкращі показники, його відсоток появи дорівнює від 99,92% до 100%, відсоток, що надійність буде вище ніж 0,5 становить від 56,81% до 58,85% в залежності від кількості запусків. Середнє значення = 0,50.

Далі розглянемо розрахунки для 20% відхилення.

Таблиця 2.6 – Результати за обраними критеріями для 20% відхилення 1000 запусків

	213	221	222	311	312
Відсоток появи	93,2 %	0,3%	0,1%	0,1%	6,3%
Середнє значення	0,507648	0,575704	0,446626	0,615568	0,484002
Вище ніж 0,5	57,188%	66,666%	0%	100%	38,095%

Таблиця 2.7 – Результати за обраними критеріями для 20% відхилення 4000 запусків

	213	221	222	311	312	321
Відсоток появи	92,62%	0,2%	0,025%	0,05%	7,05%	0,05%
Середнє значення	0,5073	0,484678	0,446626	0,51043	0,4907	0,45463
Вище ніж 0,5	57,19%	25%	0%	100%	32,62%	0%

Так як кількість вибірок збільшується зі зміною відсотків та числа запусків, було прийнято рішення в подальшому аналізувати 5-6 планів, які мають більший відсоток появи, в нашому випадку, для 20% відхилення, починаючи з 0,025%.

Таблиця 2.8 – Результати за обраними критеріями для 20% відхилення 8000 запусків

	122	213	221	222	312
Відсоток появи	0,075 %	92,26%	0,18%	0,11%	6,82%
Середнє значення	0,47754	0,506728	0,430511	0,479727	0,485893
Вище ніж 0,5	16,66%	58,24%	40%	22,22%	30,21%

Таблиця 2.9 – Результати за обраними критеріями для 20% відхилення 12000 запусків

	213	221	222	311	312
Відсоток появи	91,8%	0,14%	0,08%	0,13%	7,3%
Середнє значення	0,505891	0,472115	0,531586	0,410321	0,487949
Вище ніж 0,5	55,90%	17,64%	30%	18,75%	31,16%

Провівши аналіз для результатів з 20% відхилень, план 213 ми можемо виділити як фаворита по заданим критеріям. Відсоток появи варіюється в залежності від кількості запусків, починаючи з 91,8% до 93,2 %, також можемо відмити закономірність, що результат зменшується при збільшенні ітерацій. Відсоток того що надійність може бути вище ніж 0,5 дорівнює 55,90% – 58,24%. Середнє значення = 0,5.

Перейдемо до аналізу результатів з 30% відхилення.

Таблиця 2.10 – Результати за обраними критеріями для 30% відхилення 1000 запусків

	213	221	222	312	321	331
Відсоток появи	70%	1,4%	1%	24%	0,8%	0,6%
Середнє значення	0,51228	0,50475	0,49163	0,51188	0,44179	0,44446
Вище ніж 0,5	58,42%	50%	50%	51,25%	12,5%	0%

Таблиця 2.11 – Результати за обраними критеріями для 30% відхилення 4000 запусків

	213	221	222	231	311	312
Відсоток появи	69,45%	1,1%	1,125%	0,67%	0,67%	22,5%
Середнє значення	0,51416	0,51786	0,4896	0,44660	0,51075	0,50871
Вище ніж 0,5	59,35%	52,27%	40%	14,81%	51,85%	50,11%

Таблиця 2.12 – Результати за обраними критеріями для 30% о відхилення 8000 запусків

	213	221	222	231	312	321
Відсоток появи	69,06%	0,98%	1,15%	0,57%	22,82%	0,77%
Середнє значення	0,51312	0,52261	0,49009	0,45174	0,50998	0,44906
Вище ніж 0,5	58,66%	56,96%	41,30%	19,56%	50,65%	20,96%

Таблиця 2.13 – Результати за обраними критеріями для 30% відхилення 12000 запусків

	213	221	222	231	311	312	321
Відсоток появи	68,9%	1%	1,11%	0,52%	0,58%	23%	0,75%
Середнє значення	0,5133	0,5180	0,4904	0,4523	0,5050	0,5086	0,4452
Вище ніж 0,5	58,73%	54,16%	44,02%	20,63%	50%	49,43%	21,97%

Для 30% відхилення план 213 являється найкращим за встановленими критеріями. Відсоток появи має діапазон 68,9% – 70%. Вірогідність що результат буде вище ніж 0,5 варіюється від 58,42% до 59,35%. Середнє значення = 0,51.

Провівши аналіз по заданим критеріям, можна винести рішення, що для всіх розглянутих варіантів відхилення, найвищу надійність може надати план (2 1 3). Це означає що пристрій має містити в собі три компоненти, які складаються з двох, одного та трьох блоків.

Наданий приклад було вирішено методом перебору і визначено оптимальний план, який задовольняє вказаному бюджету та дає найвищу надійність.

ВИСНОВОК

Виконавши дипломну роботу методом перебору, ми отримали оптимальний розв'язок задачі про надійність з нормально розподіленими вхідними даними та трьома варіантами відхилення. Було розроблено програмний код за допомогою пакету прикладних програм MATLAB, який розраховує план та відповідно до нього значення надійності, також має можливість автоматично створювати файл .csv та заносити до нього результат.

Результат задачі проаналізовано за трьома критеріями оптимальності та затверджено наступний кінцевий результат:

- для 10% відхилення рекомендовано план 213, середнє значення надійності = 0,50, відсоток появи = 99,92% – 100%, вірогідність що надійність буде вище ніж 0,5 становить 56,81% – 58,85% в залежності від кількості запусків.
- для 20% відхилення рекомендовано план 213, середнє значення надійності = 0,50, відсоток появи = 91,8% – 93,2%, вірогідність що надійність буде вище ніж 0,5 становить 55,90% – 58,24% в залежності від кількості запусків.
- для 30% відхилення рекомендовано план 213, середнє значення надійності = 0,51, відсоток появи = 68,9% – 70% вірогідність що надійність буде вище ніж 0,5 становить 58,42% – 59,35% в залежності від кількості запусків.

ПЕРЕЛІК ПОСИЛАНЬ

1. **Маліков І.** Основи теорії і розрахунку надійності [Текст] / І. М. Маліков, А. М. Половко, – Л. : Судромгіз, 1959. – 325 с.
2. **Hadley, G.** Nonlinear and dynamic programming. [Текст] / Reading, Massachusetts: Addison-Wesley. Publishing Company, 1964, 484 с.
3. **Окулов, С. М.** Динамическое программирование [Текст] / С. М. Окулов, О. А. Пестов. – 3-е изд. – М. : Лаборатория знаний, 2020. – 299 с.
4. **Балдин, К.В.** Математическое программирование [Текст] / К. В. Балдин, Н. А. Брызгалов, А. В. Рукосуев. – М. : Дашков и К°, 2016. – 2198 с.
5. **Романовская, А. М.** Динамическое программирование [Текст] : Учебное пособие / А. М. Романовская, М. В. Мендзив. – Омск: Издатель Омский институт (филиал) РГТЭУ, 2010. – 58 с.
6. **Фомин, Г. П.** Экономико-математические методы и модели в коммерческой деятельности [Текст] : учебник / Г. П. Фомин. – М. : Издательство Юрайт, 2013. — 462 с.
7. **Визгунов, Н. П.** Динамическое программирование в экономических задачах с применением системы MATLAB [Текст] / Н. П. Визгунов. – Н. Новгород: ННГУ, 2006. – 72 с.
8. **Беллман, Р.** Теория динамического программирования [Текст] Пер. с англ. – И.М. Андреевой, А.А. Корбута, И.В. Романовского, И.Н. Соколовой. Под редакцией Н.Н.Воробьева. – М: Издательство иностранной литературы, 1960. — 400 с.
9. **Скляр, Б.** Цифровая связь. Теоретические основы и практическое применение [Текст] / Б. Скляр; Изд. 2-е, испр.: Пер. с англ. – М. : Издательский дом «Вильямс», 2003. – 1104 с.
10. **Таха, Х. А.** Введение в исследование операций [Текст] / Х. А. Таха; пер. с англ. – 7-е изд. – М. : "Вильямс", 2005. – 912 с.

11. **Лежнёв, А.В.** Динамическое программирование в экономических задачах: учебное пособие [Текст] / А.В. Лежнёв. – М. : БИНОМ. Лаборатория знаний, 2010. – 176 с.
12. **Грешилов, А.А.** Математические методы принятия решений: учеб. пособие [Текст] / А.А. Грешилов. – 2-е изд., испр. и доп. – М. : Изд-во МГТУ им. Н.Э. Баумана, 2014. – 647 с.
13. **Солтис, М.** Введение в анализ алгоритмов [Текст] / М. Солтис; пер. с англ. – М. : ДМК Пресс, 2019. – 278 с.
14. **Кагіров, Р. Р.** Багатомірна задача про ранець: нові методи вирішення [Текст] / Р. Р. Кагіров // Вісник.– 2015. – № 5. – С. 16 – 20.
15. **Канцедал, С. А.** . Конструирование и исследование алгоритмов решения задачи про рюкзак [Текст] / С. А. Канцедал // Автомобильный транспорт. – 2015. – № 36. – С. 154 – 160.
16. **Куприяшин, М.А.** Исследование алгоритма точного решения задачи о рюкзаке методом динамического программирования [Текст] / М. А. Куприяшин, Г. И. Борзунов // Вестник ПНИПУ. Электротехника, информационные технологии, системы управления. – 2016. – № 17. – С. 121 – 130.
17. **Monaci, M.** Exact solution of the robust knapsack problem [Текст] / M. Monaci, U. Pferschy, P. Serafini // Computers & Operations Research. – 2013. – V. 40, I. 11. – P. 2625 – 2631.
18. **Васілевський, О. М.** Нормування показників надійності технічних засобів [Текст]: навчальний посібник/О. М. Васілевський, В. О. Поджаренко. – Вінниця: ВНТУ, 2010. – 129 с.
19. **Горелик, А.В.** Основы теории надёжности в примерах и задачах[Текст] / А. В. Горелик, О. П. Ермакова. – М.: МИИТ, 2009. – 98 с.
20. **Бычков, А.А.** Надёжность приборов и систем [Текст] : учебное пособие. –Ростов-на-Дону, 2008. – 84 с.
21. **Глазунов, Л. П.** Основы теории надёжности автоматических систем управления [Текст] : учеб. пособие для вузов по спец. "Автоматика и

телемеханика" / Л. П. Глазунов, В. П. Грабовецкий, О. В. Щербаков. – Л. : Энергоатомиздат : Ленингр. отд-ние, 1984. – 207 с. : ил.; 22 см.

22. **Дмитриев, С.П.** Информационная надежность, контроль и диагностика навигационных систем. ГНЦ-РФ [Текст] / С. П. Дмитриев, Н. В. Колесов, А. В. Осипов – ЦНИИ «Электроприбор», С –Пб. 2003.-207 с.

23. **Дружинин, Г. В.** Надежность автоматизированных систем [Текст] / Г. В. Дружинин. – М. : «ЭНЕРГИЯ» 1977. – 536 с.

24. **Дьяконов, В.П.** MATLAB. Полный самоучитель [Текст] / В.П. Дьяконов // Пер. з англ. – М.: ДМК Пресс, 2012. – 768 с.

ДОДАТОК А

Приклад програмної реалізації

```

mas111=zeros(1,8000);
mas112=zeros(1,8000);
mas113=zeros(1,8000);
mas121=zeros(1,8000);
mas122=zeros(1,8000);
mas123=zeros(1,8000);
mas131=zeros(1,8000);
mas132=zeros(1,8000);
//дублювання всіх можливих варіантів
mas322=zeros(1,8000);
mas323=zeros(1,8000);
mas331=zeros(1,8000);
mas332=zeros(1,8000);
mas333=zeros(1,8000);
counter111=1;
counter112=1;
counter113=1;
counter121=1;
counter122=1;
//дублювання всіх можливих варіантів
counter322=1;
counter323=1;
counter331=1;
counter332=1;
counter333=1;
chek=0;
while chek<8000
y1 = 10 ;
C = [ normrnd(1,0.041) normrnd(3,0.143)
normrnd(2,0.084)
normrnd(2,0.084) normrnd(5,0.2) normrnd(4,0.15)
normrnd(3,0.143) normrnd(6,0.25) normrnd(5,0.2) ];
R = [ normrnd(0.6,0.026) normrnd(0.7,0.028)
normrnd(0.5,0.020)
normrnd(0.8,0.034) normrnd(0.8,0.034)
normrnd(0.7,0.028)
normrnd(0.9,0.036) normrnd(0.9,0.036)
normrnd(0.9,0.036)]; [d9,q] = size(C);
k9 = y1 + 1;
minimum_y = zeros(1,n+1);
for i = q : -1 : 1
minimum_y(i) = minimum_y(i + 1) + C(1,i);

```

```

end
max_y = zeros(1, q);
max_y(1) = y1;
for i = 2 : q
    max_y(i) = max_y(i - 1) - C(1, i - 1);
end
max_x = sum(isfinite(R));
k8 = max(max_y - minimum_y(1:q) + 1);
k8 = round(k8);
X = -inf * ones(k8, q + 1);
L = X;
L(:, q + 1) = ones(k8, 1);
for i = q : -1 : 1
    k7 = max_y(i) - minimum_y(i) + 1;
    k7 = round(k7);
    d7 = max_x(i);
    Fsu = -inf * ones(k7, d7);
    for k = 1 : k7
        for d = 1 : d7
            Yj = minimum_y(i) + k - 1;
            k1 = (Yj - C(d, i)) - minimum_y(i + 1) + 1;
            k1 = round(k1);
            if 1 <= k1 && k1 <= k7
                Fsu(k, d) = R(d, i) * L(k1, i + 1);
            end
        end
    end
end
end
[fs, ind] = max(Fsu');
L(1: k7, i) = fs';
X(1: k7, i) = ind';
%disp(['Таблиця' int2str(i)])
%disp([ blanks(d7*3 - 2) ])
%disp([(minimum_y(i) : max_y(i))', fs', ind', Fsu ])
end
nabor = zeros(1, q);
Yj = y1;
for i = 1 : q
    k = Yj - minimum_y(i) + 1;
    k = round(k);
    nabor(i) = X(k, i);
    Yj = Yj - C(nabor(i), i);
end
%disp('_____Розв'язок_____')
nabor;
reliability = L(k8, 1);

```

```

resultat=nabor(1)*100+nabor(2)*10+nabor(3);

if resultat==111
    mas111(counter111)=reliability;
    counter111=counter111+1;
end
if resultat==121
    mas121(counter121)=reliability;
    counter121=counter121+1;
end
if resultat==122
    mas122(counter122)=reliability;
    counter122=counter122+1;

    //дублирование всех возможных вариантов

end
if resultat==332
    mas332(counter332)=reliability;
    counter332=counter332+1;
end
if resultat==333
    mas333(counter333)=reliability;
    counter333=counter333+1;
end
chek=chek+1;
end
xlswrite('desvid.csv',[111 112 113 121 122 123 131
132 133 211 212 213 221 222 223 231 232 233 311 312 313
321 322 323 331 332 333 ],'A1:AA1');
xlswrite('desvid.csv',mas111,'A2:A8001');
xlswrite('desvid.csv',mas112,'B2:B8001');
xlswrite('desvid.csv',mas113,'C2:C8001');
xlswrite('desvid.csv',mas121,'D2:D8001');
xlswrite('desvid.csv',mas222,'N2:N8001');
//дублирование всех возможных вариантов
xlswrite('desvid.csv',mas331,'Y2:Y8001');
xlswrite('desvid.csv',mas332,'Z2:Z8001');
xlswrite('desvid.csv',mas333,'AA2:AA8001');

```