

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ  
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет Інформаційних технологій

**ПОГОДЖЕНО**

Декан факультету

інформаційних технологій

\_\_\_\_\_ Ігор БОЛБОТ

“ \_\_\_ ” \_\_\_\_\_ 2025р.

**ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ**

Завідувач кафедри

комп'ютерних наук

\_\_\_\_\_ Белла ГОЛУБ

“ \_\_\_ ” \_\_\_\_\_ 2025 р.

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

**на тему**

Розроблення системи проектування сонячної ферми із застосуванням  
генетичних алгоритмів

Спеціальність – 122 «Комп'ютерні науки»

Освітня програма – «Інформаційні управляючі системи і технології»

Орієнтація освітньої програми – освітньо-професійна

**Гарант освітньої програми**

К.Т.Н., доцент \_\_\_\_\_  
(науковий ступінь та вчене звання) (підпис)

Белла ГОЛУБ  
(Ім'я, Прізвище)

**Керівник магістерської кваліфікаційної роботи**

К.Т.Н., доцент \_\_\_\_\_  
(науковий ступінь та вчене звання) (підпис)

Юлія БОЯРІНОВА  
(Ім'я, Прізвище)

**Виконав**

\_\_\_\_\_  
(підпис)

Вадим КОЛОМІЄЦЬ  
(Ім'я, Прізвище)

**КИЇВ-2025**

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ  
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

доцент, к.т.н. \_\_\_\_\_ Голуб Б. Л.

“01” листопада 2024 року

З А В Д А Н Н Я

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ

Коломійцю Вадиму Юрійовичу

(прізвище, ім'я, по батькові)

Спеціальність 122 «Комп'ютерні науки»

(код і назва)

Освітня програма Інформаційні управляючі системи та технології

(назва)

Орієнтація освітньої програми освітньо-професійна

Тема магістерської кваліфікаційної роботи Застосування генетичних алгоритмів для проектування сонячних ферм затверджена наказом ректора НУБіП України від “01” листопада 2024р. №1964 «С»

Термін подання завершеної роботи на кафедру 14 листопада 2025

(рік, місяць, число)

Вихідні дані до магістерської кваліфікаційної роботи:

технічні характеристики сонячних панелей, площа й конфігурація земельної ділянки, кліматичні параметри місцевості, вимоги до орієнтації та кута нахилу модулів, обмеження бюджету, а також параметри, необхідні для побудови функціональної, об'єктно-орієнтованої та математичної моделі в рамках реалізації генетичного алгоритму оптимізації.

Перелік питань, що підлягають дослідженню:

1. Особливості сонячної енергетики як складової сучасної енергетичної стратегії та існуючі підходи до проектування сонячних ферм, що застосовуються на практиці.

2. Ефективність генетичних алгоритмів у задачах оптимізації просторової та технічної конфігурації сонячних ферм.

3. Функціональне, об'єктно-орієнтоване та математичне моделювання системи проектування.

4. Архітектура, алгоритмічне та технологічне забезпечення, яке є доцільними для реалізації системи оптимізованого проектування сонячних ферм.

Перелік графічного матеріалу (за потреби)

Дата видачі завдання “01” листопада 2024 р.

Керівник магістерської кваліфікаційної роботи \_\_\_\_\_

Юлія БОЯРІНОВА

(підпис)

Завдання прийняв до виконання \_\_\_\_\_

Вадим КОЛОМІЄЦЬ



## Реферат

Магістерська робота на тему: «Розроблення системи проєктування сонячної ферми із застосуванням генетичних алгоритмів»

Обсяг роботи – 68 сторінок, містить 17 рисунків, 7 таблиць, додатки, список використаних джерел – 31 найменування.

**Об'єкт дослідження** – процес проєктування сонячних ферм.

**Предмет дослідження** – застосування генетичних алгоритмів для оптимізації параметрів і конфігурацій сонячних ферм.

**Мета роботи** – розробка інтелектуальної системи оптимізації проєктування сонячних ферм із використанням генетичних алгоритмів, яка забезпечує підвищення ефективності використання природних і технічних ресурсів.

У роботі виконано системний аналіз сучасного стану розвитку сонячної енергетики, розглянуто основні методи проєктування фотоелектричних систем, виявлено їхні обмеження та визначено переваги застосування еволюційних підходів. Проведено формалізацію задачі оптимізації з урахуванням технічних, економічних та природних факторів.

Розроблено архітектуру системи, яка включає модулі введення даних, формалізації задачі, генетичного алгоритму, аналізу та візуалізації результатів. Програмну частину реалізовано на Python із використанням бібліотек Tkinter, NumPy, Matplotlib та стандартних модулів Python для обчислень і роботи з файлами. Проведено серію комп'ютерних симуляцій, що підтвердили ефективність розробленого підходу.

Отримані результати показали, що використання генетичних алгоритмів дозволяє скоротити час пошуку оптимальної конфігурації сонячної ферми, підвищити коефіцієнт корисної дії фотоелектричних систем і знизити витрати на проєктування.

**Наукова новизна роботи** полягає у створенні моделі оптимізації проєктування сонячних ферм із використанням генетичних алгоритмів та

розробці комплексної архітектури системи підтримки прийняття рішень у сфері відновлюваної енергетики.

**Практичне значення результатів** полягає у можливості застосування розробленої системи для оптимізації розміщення панелей, вибору обладнання та розрахунку ефективності сонячних електростанцій на етапі проектування.

**Ключові слова:** сонячна енергетика, генетичний алгоритм, оптимізація, проектування, фотоелектрична система, інтелектуальні технології.

## **ЗМІСТ**

<b>Перелік умовних позначень</b>	<b>4</b>
<b>ВСТУП</b>	<b>5</b>
<b>РОЗДІЛ 1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ</b>	<b>8</b>
1.1 Сонячна енергетика як складова сучасної енергетичної стратегії	8
1.2 Аналіз існуючих методів проектування сонячних ферм	11
1.3 Генетичні алгоритми як інструмент оптимізації	19
1.4 Постановка завдання дослідження	23
<b>РОЗДІЛ 2 МОДЕЛЮВАННЯ СИСТЕМИ</b>	<b>25</b>
2.1 Функціональне моделювання	25
2.2 Об'єктно-орієнтоване моделювання	27
2.3 Формалізація задачі оптимізації	34
<b>РОЗДІЛ 3 РОЗРОБКА СИСТЕМИ</b>	<b>38</b>
3.1 Архітектура системи оптимізації проектування сонячних ферм	38
3.2 Алгоритмічне забезпечення	40
3.3 Технологічне забезпечення	44
<b>РОЗДІЛ 4 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ</b>	<b>47</b>
4.1 Апаратні та програмні вимоги до реалізації системи	47
4.2 Проведення експериментів та симуляцій	49
4.3 Підсумковий аналіз результатів	55
<b>ВИСНОВКИ</b>	<b>58</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</b>	<b>60</b>
<b>ДОДАТКИ</b>	<b>64</b>

## Перелік умовних позначень

ГА – генетичний алгоритм

ККД – коефіцієнт корисної дії

ООМ – об'єктно-орієнтоване моделювання

СЕС – сонячна електростанція

CAPEX – капітальні витрати (Capital Expenditures)

CPU – центральний процесор (Central Processing Unit)

CSV – текстовий табличний формат даних (Comma-Separated Values)

DE – Differential Evolution (алгоритм диференціальної еволюції, згадується в тексті)

FIT – функція пристосованості (Fitness)

GUI – графічний інтерфейс користувача (Graphical User Interface)

IDEF0 – методологія функціонального моделювання

LCOE – собівартість електроенергії (Levelized Cost of Energy)

NSGA-II – Nondominated Sorting Genetic Algorithm II (згадується в перспективі дослідження)

PDF – формат електронних документів (Portable Document Format)

PSO – Particle Swarm Optimization (алгоритм рою частинок, згадується як альтернатива)

PV – photovoltaic (фотоелектричний)

TK – Tkinter – бібліотека для побудови GUI в Python

RAM – оперативна пам'ять (Random Access Memory)

UML – Unified Modeling Language

## ВСТУП

**Актуальність теми.** У сучасних умовах розвитку енергетичного сектору світу особливе місце посідає відновлювана енергетика, зокрема сонячна енергетика. Виклики, пов'язані зі зміною клімату, зростанням попиту на енергоресурси та необхідністю забезпечення енергетичної незалежності, зумовлюють активний пошук інноваційних рішень для підвищення ефективності проєктування та експлуатації сонячних електростанцій. Однією з основних проблем у цій сфері є оптимізація розміщення фотоелектричних панелей, вибір технологічних параметрів та розрахунок максимальної ефективності з урахуванням природних і технічних обмежень.

Традиційні методи оптимізації не завжди дозволяють досягти глобальних оптимальних рішень у задачах із великою кількістю параметрів та нелінійними залежностями. У цьому контексті перспективними є генетичні алгоритми – еволюційні методи, які забезпечують пошук оптимальних конфігурацій у складних багатофакторних системах. Застосування таких алгоритмів у сфері сонячної енергетики відкриває нові можливості для створення ефективних інтелектуальних систем підтримки прийняття рішень.

**Об'єктом дослідження** є процес проєктування сонячних ферм.

**Предметом дослідження** є застосування генетичних алгоритмів для оптимізації конфігурацій і параметрів сонячних ферм.

**Метою роботи** є розробка та дослідження системи оптимізації проєктування сонячних ферм на основі генетичних алгоритмів, що дозволить підвищити ефективність використання ресурсів та збільшити коефіцієнт корисної дії фотоелектричних систем.

Для досягнення мети було поставлено такі **завдання**:

- 1) провести системний аналіз предметної області сонячної енергетики та існуючих методів проєктування сонячних ферм;
- 2) дослідити можливості та принципи застосування генетичних алгоритмів у задачах оптимізації;

- 3) здійснити функціональне та об'єктно-орієнтоване моделювання процесів проектування;
- 4) формалізувати задачу оптимізації для сонячної ферми;
- 5) розробити архітектуру системи оптимізації та її алгоритмічне забезпечення;
- 6) реалізувати прототип системи та провести серію експериментів і симуляцій;
- 7) проаналізувати результати та визначити ефективність запропонованого підходу.

**Методи дослідження.** У роботі застосовуються методи системного аналізу для вивчення предметної області; методи моделювання (функціонального та об'єктно-орієнтованого); алгоритмічні методи еволюційної оптимізації, зокрема генетичні алгоритми; комп'ютерне моделювання та симуляції для перевірки результатів. Для обчислювальних експериментів використовуються сучасні програмні засоби, серед яких Python, бібліотеки NumPy, Matplotlib та спеціалізовані пакети для реалізації генетичних алгоритмів.

**Наукова новизна** роботи полягає у тому, що:

- 1) розроблено модель системи оптимізації проектування сонячних ферм із використанням генетичних алгоритмів;
- 2) запропоновано удосконалення методів формалізації задачі оптимізації з урахуванням багатofакторних природних і технічних обмежень;
- 3) створено алгоритмічне та програмне забезпечення для дослідження ефективності застосування генетичних алгоритмів у проектуванні таких систем.

**Апробація результатів.** Основні результати дослідження були апробовані у вигляді тез доповідей на науково-практичних конференціях.

**Структура магістерської роботи.** Магістерська робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків. Загальний обсяг роботи становить 68 сторінок друкованого тексту, список використаних джерел налічує 34 найменування, додатки містять скріншоти інтерфейсу програми.

У першому розділі подано системний аналіз предметної області та постановку задачі. У другому розділі описано методи моделювання та формалізацію задачі оптимізації. У третьому розділі представлено архітектуру та алгоритмічне забезпечення розробленої системи. Четвертий розділ містить результати експериментів та їх аналіз.

## РОЗДІЛ 1

### СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

#### 1.1 Сонячна енергетика як складова сучасної енергетичної стратегії

Сонячна енергетика є одним із основних напрямів розвитку відновлюваної енергетики, що стрімко зростає у світі протягом останніх десятиліть. Вона ґрунтується на перетворенні сонячного випромінювання у електричну чи теплову енергію з використанням спеціальних технологій: фотоелектричних панелей, сонячних колекторів, концентраторів тощо. Завдяки універсальності та екологічності сонячна енергія стала невід'ємною складовою сучасних енергетичних стратегій як розвинених держав, так і країн, що розвиваються.

Перші спроби використання сонячного світла для енергетичних потреб сягають XIX століття, однак промислове застосування фотоелектричних панелей почалося лише у другій половині XX століття. У 1970–1980-х роках сонячна енергетика розвивалася переважно як експериментальна технологія, а її використання обмежувалося космічними програмами через високу вартість обладнання. Проте вже на початку XXI століття завдяки технологічним інноваціям, зниженню собівартості виробництва та державній підтримці галузь отримала глобальний імпульс.

Сьогодні сонячна енергетика є найбільш швидкозростаючим сектором серед відновлюваних джерел. За даними Міжнародного енергетичного агентства (ІЕА), у 2023 році встановлена потужність сонячних електростанцій у світі перевищила 1,2 ТВт, а до 2030 року очікується її зростання більш ніж удвічі. Лідерами залишаються Китай, США, Індія та країни ЄС, які активно інтегрують фотоелектричні системи у свої енергетичні стратегії.

Сонячна енергетика має важливе значення для диверсифікації енергобалансу та підвищення енергетичної безпеки країн. Традиційні джерела, такі як вугілля, газ чи нафта, пов'язані із залежністю від імпорту, високими ціновими ризиками та негативним впливом на довкілля. Використання сонячних

ферм дозволяє зменшити залежність від викопних палив і забезпечити стале енергопостачання [1].

Економічна привабливість сонячної енергетики підкріплюється стрімким падінням вартості виробництва електроенергії. Якщо у 2010 році ціна за 1 кВт·год електроенергії, виробленої сонячними станціями, у середньому перевищувала 30–35 центів, то у 2023 році цей показник знизився до 4–6 центів, що робить сонячні технології конкурентними з традиційною генерацією.

Перехід до сонячної енергетики безпосередньо пов'язаний із виконанням міжнародних зобов'язань щодо скорочення викидів парникових газів та боротьби зі зміною клімату. Паризька кліматична угода передбачає зниження викидів CO<sub>2</sub> і поступову відмову від вуглеводневих джерел енергії. У цьому контексті сонячні ферми стають інструментом декарбонізації економіки, адже виробництво електроенергії з сонячного випромінювання практично не супроводжується шкідливими викидами [2].

Крім того, сонячні електростанції мають низький рівень шуму, не потребують значних водних ресурсів і можуть встановлюватися на непридатних для сільського господарства землях, що мінімізує конфлікти з іншими сферами використання територій.

У країнах Європейського Союзу сонячна енергетика посідає важливе місце в досягненні цілей «Європейського зеленого курсу», спрямованого на досягнення кліматичної нейтральності до 2050 року. Німеччина є одним із лідерів за кількістю встановлених сонячних панелей завдяки програмам «зеленої енергетики». Китай, у свою чергу, інвестує у масштабні сонячні ферми, розташовані в пустельних регіонах, тоді як США розвивають як великі комерційні проєкти, так і децентралізовані системи для приватних домогосподарств.

Для України сонячна енергетика має стратегічне значення у контексті енергетичної незалежності та переходу на відновлювані джерела. Країна володіє значним потенціалом сонячної інсоляції – від 3,5 до 4,5 кВт·год/м<sup>2</sup> на добу залежно від регіону. Найбільш перспективними є південні області, однак навіть

у північних регіонах рівень сонячної активності є достатнім для економічно обґрунтованої генерації.

Енергетична стратегія України до 2035 року визначає ціль досягти частки відновлюваних джерел на рівні понад 25%, і сонячні електростанції посідають важливу роль у цьому процесі. Попри воєнні виклики, галузь продовжує розвиватися, зокрема через малі дахові установки для домогосподарств та бізнесу.

Сонячна енергетика демонструє стабільне зростання в усьому світі, що пов'язано зі зростанням попиту на чисті джерела енергії, підтримкою державних програм та вдосконаленням технологій. Провідними країнами у цій галузі залишаються Китай, країни Європейського Союзу, США, В'єтнам і Японія. Китай утримує лідерство у світовій сонячній енергетиці завдяки активному державному фінансуванню, масштабним проєктам будівництва сонячних ферм у віддалених регіонах та прагненню зменшити забруднення довкілля. Тут спостерігається швидке зростання виробництва енергії з відновлюваних джерел і постійне вдосконалення технологій зберігання. Європейський Союз характеризується стабільним відновленням темпів розвитку після періоду спаду. Основними центрами генерації є Німеччина, Іспанія та Нідерланди, де значна частка сонячних установок представлена даховими системами. Розвиток підтримується екологічною політикою ЄС та програмами скорочення викидів вуглецю [3].

США активно розширюють потужності сонячної енергетики у житловому секторі та сфері комунальних послуг. Велику роль у цьому відіграють державні стимули, зокрема податкові пільги, які сприяють залученню інвестицій і скороченню викидів парникових газів. В'єтнам демонструє швидке зростання завдяки урядовим програмам підтримки та вигідним тарифам для виробників електроенергії. Особливу роль відіграє розвиток дахових і плаваючих сонячних електростанцій, які дозволяють ефективно використовувати територію навіть у густонаселених регіонах. Японія після катастрофи на Фукусімі приділяє підвищену увагу безпечним і чистим джерелам енергії. Тут розвиваються дахові

системи, плавучі електростанції, а також проекти встановлення сонячних панелей на непридатних для сільського господарства землях і територіях гольф-полів.

У цілому, аналіз тенденцій свідчить, що сонячна енергетика стала одним із ключових напрямів переходу до низьковуглецевої економіки. Вона активно розвивається не лише в економічно розвинених державах, а й у країнах, що швидко модернізуються, що підтверджує її глобальне значення у формуванні майбутнього енергетичного балансу світу.

Попри очевидні переваги, розвиток сонячної енергетики супроводжується низкою викликів:

- 1) нестабільність генерації – залежність від погодних умов вимагає використання акумулюючих систем і резервних джерел енергії;
- 2) проблеми інтеграції в енергосистему – значне навантаження на мережі потребує впровадження «розумних» технологій керування;
- 3) фінансові бар'єри – хоча вартість панелей знижується, початкові інвестиції залишаються високими, що вимагає державної підтримки;
- 4) утилізація обладнання – потребує вирішення питання переробки відпрацьованих фотоелементів.

Водночас перспективи залишаються надзвичайно позитивними. Використання інноваційних інформаційних технологій, зокрема штучного інтелекту, систем моніторингу та оптимізації, дозволяє підвищити ефективність проєктування й експлуатації сонячних ферм [4]. Особливо актуальним стає застосування генетичних алгоритмів для пошуку оптимальних конфігурацій розміщення панелей, вибору їхнього кута нахилу, підключення до мережі та балансування енергопотоків.

## **1.2 Аналіз існуючих методів проєктування сонячних ферм**

Проєктування сонячних ферм є складним багатофакторним процесом, що охоплює технічні, економічні, екологічні та соціальні аспекти. Від правильності

проектних рішень залежить ефективність виробництва електроенергії, термін окупності інвестицій та стабільність роботи всієї енергосистеми. Сучасна наукова та інженерна практика пропонує різноманітні методи проектування сонячних електростанцій, які можна умовно поділити на традиційні, математико-оптимізаційні, програмно-імітаційні та інтелектуальні [5].

Традиційний підхід до проектування сонячних ферм базується на використанні емпіричних розрахунків, статистичних даних та довідкових матеріалів, що накопичувалися впродовж багаторічної практики будівництва енергетичних об'єктів [6]. При цьому основна увага зосереджується на загальних характеристиках місцевості та кліматичних умовах регіону, зокрема на рівні сонячної інсоляції, середньорічній температурі, кількості сонячних днів та інших параметрах, що безпосередньо впливають на ефективність роботи фотоелектричних панелей.

Одним із перших кроків у межах традиційного підходу є визначення необхідної встановленої потужності сонячної ферми залежно від запланованого обсягу виробництва електроенергії чи потреб споживачів. Далі здійснюється підбір обладнання, що зазвичай відбувається за каталогами виробників і технічними паспортами. Увага приділяється вибору типу фотоелектричних модулів, інверторів, кабельних систем, трансформаторів та іншої допоміжної інфраструктури [7]. Після цього проводиться розрахунок необхідної площі земельної ділянки, яка має відповідати вибраній потужності станції та кількості модулів. Важливим елементом цього процесу є визначення кута нахилу панелей і напрямку їхнього розташування відносно сонця, що зазвичай визначається на основі середніх статистичних значень для конкретного регіону.

Головною перевагою традиційних методів є їхня простота, зрозумілість і відносно невисока вартість. Вони не потребують складного програмного забезпечення чи значних обчислювальних ресурсів, тому залишаються популярними в невеликих проєктах, наприклад для приватних домогосподарств або малих комерційних об'єктів. Проте такий підхід має й очевидні недоліки. Основна проблема полягає у низькій точності результатів, оскільки в

розрахунках не враховуються складні нелінійні взаємозв'язки між різними параметрами системи [8]. Зокрема, часто ігнорується вплив локальних умов: затінення від будівель чи дерев, особливості рельєфу, втрати в кабельних мережах, температурні коливання та інші чинники, які можуть суттєво знижувати ефективність роботи станції. Унаслідок цього проекти, створені за традиційними методами, не завжди досягають запланованих показників виробітку електроенергії й потребують додаткових коригувань на етапі експлуатації.

Традиційні методи проектування можна розглядати як початковий етап розвитку галузі, що забезпечив базове поширення сонячних технологій. Водночас їхні обмеження стимулювали пошук нових підходів, орієнтованих на точніші розрахунки та використання сучасних інструментів моделювання й оптимізації.

Сучасний етап розвитку сонячної енергетики значною мірою пов'язаний із використанням математичних моделей та оптимізаційних методів, що дозволяють підвищити точність і надійність проєктних рішень. На відміну від традиційних підходів, які здебільшого ґрунтуються на усереднених даних, математичне моделювання дає можливість враховувати широкий спектр параметрів і обмежень, що діють у реальних умовах. Завдяки цьому процес проектування перетворюється на формалізовану задачу оптимізації, де метою може бути мінімізація витрат, максимізація виробітку енергії або пошук найкращого балансу між технічними, економічними та екологічними показниками [9].

Одним із поширених інструментів у цьому напрямі є лінійне та нелінійне програмування. Такі методи застосовуються для визначення оптимальної кількості фотоелектричних панелей, їхнього розміщення на обраній ділянці та розрахунку найвигіднішої конфігурації інверторів і трансформаторів. У випадках, коли залежності між параметрами мають нелінійний характер (наприклад, вплив температури на ефективність панелей), використовуються

спеціалізовані алгоритми нелінійної оптимізації. Це дозволяє більш адекватно моделювати складні процеси й підвищувати точність розрахунків.

Ще одним важливим напрямом є динамічне програмування, яке враховує зміну умов у часі. Зокрема, сонячна інсоляція та погодні фактори мають яскраво виражений сезонний характер, що суттєво впливає на виробіток електроенергії. Використання динамічних моделей дає змогу прогнозувати роботу станції у різні пори року та знаходити рішення, що забезпечують оптимальне функціонування системи протягом усього життєвого циклу [10].

Особливе значення у проектуванні сонячних ферм мають методи багатокритеріальної оптимізації. У практиці будівництва сонячних електростанцій доводиться одночасно враховувати кілька факторів: економічну доцільність проекту, рівень виробітку електроенергії, термін окупності інвестицій, вплив на довкілля, а також технічні обмеження, пов'язані з інфраструктурою електромереж [11]. Багатокритеріальні моделі дозволяють формувати так звані «компромісний простір рішень», у межах якого інвестори та проєктанти можуть обирати найбільш прийнятні варіанти залежно від пріоритетів.

Разом із тим застосування математичних та оптимізаційних методів має і певні обмеження. Основна складність полягає у створенні адекватних моделей, які б точно відображали реальні процеси в системі. Будь-яке спрощення або неврахування окремих факторів може призвести до суттєвих похибок у результатах. Крім того, у задачах із великою кількістю змінних і обмежень обчислювальна складність різко зростає, що потребує використання високопродуктивних обчислювальних систем та спеціалізованого програмного забезпечення.

В цілому, математичні та оптимізаційні методи є важливим етапом еволюції підходів до проектування сонячних ферм. Вони забезпечують значно більшу точність і гнучкість у прийнятті рішень порівняно з традиційними методами, однак потребують високого рівня спеціалізованих знань, якісних вхідних даних і значних обчислювальних ресурсів. Ці фактори стали

передумовою для розвитку новітніх інтелектуальних методів, здатних ефективніше вирішувати задачі оптимізації у складних багатовимірних просторах рішень.

Подальший розвиток технологій проектування сонячних ферм став можливим завдяки появі спеціалізованого програмного забезпечення, яке забезпечує імітаційне моделювання роботи фотоелектричних систем. Цей підхід дозволяє значно точніше відтворювати реальні умови функціонування сонячних електростанцій, враховувати широкий спектр природних і технічних факторів та прогнозувати ефективність їхньої роботи протягом тривалого періоду. Імітаційні методи стали поширеними як у наукових дослідженнях, так і у практичному проектуванні, оскільки вони поєднують аналітичну строгість з високим рівнем наочності результатів [12].

Одним із найбільш популярних інструментів є програмний комплекс PVsyst, який використовується професіоналами для детального моделювання фотоелектричних установок різних масштабів – від невеликих дахових систем до промислових сонячних ферм. Програма дозволяє враховувати вплив тіньових зон, рельєфу місцевості, кутів нахилу та орієнтації панелей. Вона також забезпечує розрахунок енергетичних втрат, коефіцієнтів ефективності інверторів та прогнозування річного виробітку електроенергії, що робить її практично незамінною на стадії проектування.

Не менш важливим є програмний продукт HOMER Energy, розроблений для проектування гібридних енергетичних систем. Його особливість полягає у можливості комбінування різних джерел енергії, зокрема сонячних панелей, вітрових турбін і дизельних генераторів, з урахуванням систем накопичення енергії. Це дозволяє моделювати автономні енергетичні комплекси для віддалених населених пунктів чи підприємств, де підключення до централізованої мережі є обмеженим або взагалі неможливим. HOMER дає змогу проводити техніко-економічний аналіз різних сценаріїв, визначати оптимальний склад системи та прогнозувати терміни її окупності.

Ще одним широко застосовуваним програмним інструментом є System Advisor Model, розроблений Національною лабораторією відновлюваної енергетики США (NREL). Ця програма орієнтована не лише на технічне моделювання, а й на комплексну оцінку економічної доцільності проєктів. SAM дозволяє враховувати ринкові фактори, державні стимули, тарифи на електроенергію та інші фінансові показники, що дає змогу потенційним інвесторам і замовникам отримати повне уявлення про рентабельність майбутньої сонячної ферми.

Перевага програмно-імітаційних методів полягає у здатності використовувати реальні метеорологічні дані та точні технічні характеристики обладнання, що забезпечує високу достовірність прогнозів. Вони також дозволяють створювати різні сценарії розвитку проєкту, оцінювати його ефективність за умов зміни клімату, вартості обладнання чи тарифів на електроенергію. Завдяки цьому імітаційні методи широко застосовуються як у наукових дослідженнях, так і у практиці інжинірингових компаній [13].

Разом із тим слід зазначити й певні обмеження. Більшість програмних комплексів працює на основі закладених алгоритмів і моделей, що обмежує їхню гнучкість у пошуку інноваційних рішень. Користувач змушений працювати в межах наявних модулів, а отже, складні багатофакторні завдання не завжди можуть бути вирішені на належному рівні. Крім того, для роботи з такими системами потрібна значна кількість вхідних даних, що не завжди доступні в умовах конкретної території чи проєкту.

Загалом, програмно-імітаційні методи стали важливим кроком уперед у розвитку технологій проєктування сонячних ферм. Вони забезпечують високий рівень точності й наочності, дозволяють враховувати реальні умови функціонування систем, але водночас мають певні обмеження, які зумовлюють подальший пошук більш гнучких і адаптивних підходів, зокрема заснованих на інтелектуальних алгоритмах оптимізації.

Упродовж останнього десятиліття проєктування сонячних ферм зазнало значних трансформацій завдяки впровадженню інтелектуальних методів, що

базуються на штучному інтелекті та машинному навчанні. Ці підходи дозволяють не лише автоматизувати процеси оптимізації, а й вийти за межі традиційних і класичних математичних моделей, які часто виявляються недостатньо гнучкими у складних реальних умовах. В основі інтелектуальних методів лежить здатність алгоритмів навчатися на даних, адаптуватися до нових ситуацій та враховувати велику кількість змінних одночасно, що робить їх особливо ефективними в умовах невизначеності та багатокритеріальності.

Одним із найпоширеніших напрямів у цій сфері є застосування еволюційних алгоритмів. Вони моделюють процеси природного добору та еволюції, дозволяючи відбирати найкращі рішення з великої множини можливих варіантів. Генетичні алгоритми, алгоритми рою частинок чи методи диференційної еволюції активно застосовуються для пошуку глобальних оптимумів у багатовимірних просторах рішень. У випадку сонячних ферм це може означати визначення оптимальної конфігурації розміщення панелей, вибір кутів нахилу, планування схем підключення до мережі чи балансування виробленої енергії [14]. Перевага еволюційних методів полягає у здатності знаходити рішення, які класичними аналітичними методами отримати практично неможливо.

Ще одним перспективним напрямом є використання штучних нейронних мереж. Вони особливо ефективні у прогнозуванні сонячної генерації, оскільки здатні виявляти приховані залежності в історичних кліматичних даних, таких як рівень інсоляції, температура, хмарність чи швидкість вітру. Завдяки цьому можна з високою точністю передбачати майбутню продуктивність сонячної ферми та планувати її роботу в енергосистемі. Нейромережеві моделі також дають змогу інтегрувати великі обсяги гетерогенних даних, у тому числі з супутникових спостережень і систем моніторингу в реальному часі.

Окрему групу складають методи, що ґрунтуються на нечіткій логіці. Їхня основна перевага полягає у можливості працювати з неповними або нечіткими даними, що особливо актуально при проектуванні сонячних ферм у регіонах із недостатньо розвиненою системою збору кліматичної інформації. На відміну від

класичних математичних моделей, які вимагають точних значень параметрів, нечіткі системи дозволяють працювати з приблизними оцінками, наприклад «високий рівень сонячності» чи «середня хмарність». Це робить їх корисними для прийняття рішень у ситуаціях, де існує значна невизначеність.

Застосування інтелектуальних методів відкриває нові перспективи для проєктування сонячних електростанцій, адже вони здатні враховувати одночасно економічні, технічні та екологічні чинники. Проте ці підходи мають і певні обмеження. Передусім вони потребують великих обчислювальних ресурсів і високоякісних вхідних даних. Без належної бази даних результати роботи алгоритмів можуть виявитися неточними. Крім того, інтелектуальні системи часто виступають як «чорні скриньки», що ускладнює інтерпретацію результатів і потребує додаткової валідації та експертного аналізу.

У підсумку можна зазначити, що інтелектуальні методи поступово стають провідним інструментом у сфері оптимізації проєктування сонячних ферм. Вони забезпечують значно вищий рівень точності та адаптивності, дозволяють моделювати складні багатофакторні системи та знаходити рішення, недосяжні традиційними методами. Саме тому їх використання розглядається як перспективний напрям подальшого розвитку галузі відновлюваної енергетики.

Для узагальнення доцільно подати порівняння основних груп методів у вигляді таблиці (табл. 1.1).

Таблиця 1.1 – Порівняння методів проєктування сонячних ферм

Група методів	Основні характеристики	Переваги	Недоліки	Приклади використання
Традиційні	Ґрунтуються на емпіричних розрахунках і довідкових даних	Простота, швидка реалізація, низька вартість	Низька точність, ігнорування локальних умов	Початкове проєктування, малі установки
Математичні	Використовують лінійне, нелінійне та динамічне програмування	Висока точність, можливість формального опису	Складність побудови моделей, великі обчислення	Оптимізація потужності й конфігурації станцій

Продовження таблиці 1.1

Група методів	Основні характеристики	Переваги	Недоліки	Приклади використання
Імітаційні	Моделювання з використанням спеціалізованого ПЗ	Урахування реальних кліматичних і технічних даних	Обмежена гнучкість через закладені алгоритми	PVsyst, HOMER, SAM
Інтелектуальні	Базуються на штучному інтелекті й еволюційних алгоритмах	Пошук глобальних оптимумів, адаптивність	Високі вимоги до обчислювальних ресурсів і даних	Генетичні алгоритми, неймережі

Загалом, аналіз існуючих методів показує, що традиційні й математичні підходи мають обмежену ефективність у складних задачах із багатьма змінними. Імітаційні методи дають змогу враховувати реальні умови, однак не завжди знаходять оптимальні рішення. Найбільш перспективними є інтелектуальні методи, зокрема генетичні алгоритми, які здатні ефективно працювати в умовах багатокритеріальної оптимізації та невизначеності, що і зумовлює актуальність їх застосування у цій роботі.

### 1.3 Генетичні алгоритми як інструмент оптимізації

Сучасні задачі оптимізації у сфері проєктування складних технічних систем, до яких належать і сонячні ферми, характеризуються великою кількістю параметрів, нелінійними залежностями між ними та багатокритеріальністю. Традиційні методи пошуку рішень часто виявляються малоефективними в таких умовах, оскільки не гарантують знаходження глобального оптимуму й можуть застрягати на локальних екстремумах [15]. Саме тому виникла потреба у

застосуванні еволюційних методів, що імітують природні механізми добору та адаптації. Одним із найбільш поширених серед них є генетичні алгоритми.

Генетичні алгоритми належать до класу стохастичних методів оптимізації, заснованих на принципах біологічної еволюції. Їхня робота відтворює природні процеси добору, схрещування та мутації, що дозволяє поступово покращувати популяцію рішень у напрямі глобального оптимуму. Основна ідея полягає в тому, що кожне можливе рішення задачі кодується у вигляді «хромосоми», яка складається з набору параметрів, або «генів». Сукупність таких рішень утворює популяцію, яка еволюціонує протягом кількох поколінь [16]. На рисунку 1.1 зображено узагальнену схему роботи генетичного алгоритму, яка демонструє еволюційний процес пошуку оптимального рішення.

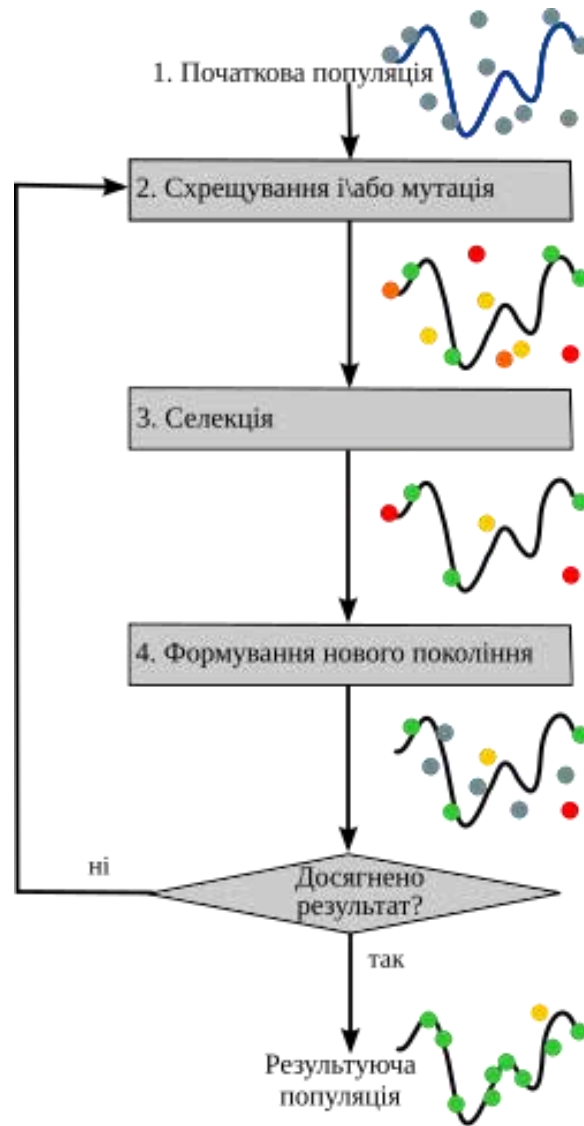


Рис. 1.1 Схема роботи генетичного алгоритму [17]

Початкова популяція можливих варіантів піддається операціям схрещування та мутації, після чого здійснюється селекція, що забезпечує відбір найбільш пристосованих рішень. Сформоване нове покоління знову проходить ті самі етапи, утворюючи циклічний процес, що повторюється доти, доки не буде досягнуто бажаного результату, тобто глобального або наближеного до оптимального розв'язку задачі [18].

Процес роботи ГА зазвичай включає кілька основних етапів. На початковій стадії формується стартова популяція, яка може бути згенерована випадково або на основі певних евристик. Далі для кожного індивіда визначається значення

цільової функції, що відображає ступінь «приспосованості» рішення до поставленої задачі. На основі цього значення здійснюється відбір – кращі індивіди мають вищі шанси передати свої гени наступним поколінням. На етапі кросоверу відбувається комбінування генів двох «батьківських» рішень для створення «нащадків», а мутація вносить випадкові зміни у хромосоми, що запобігає застою алгоритму у локальних екстремумах [19]. Цей цикл повторюється доти, доки не буде досягнуто заданої точності чи кількості ітерацій.

Основною перевагою генетичних алгоритмів є їхня здатність до глобального пошуку оптимуму у складних багатовимірних просторах. Вони не потребують аналітичного опису функції чи її похідних, що робить метод універсальним для задач із нелінійними, дискретними та неперервними змінними. ГА також легко адаптуються до багатокритеріальних задач, дозволяючи одночасно враховувати, наприклад, вартість будівництва, ефективність виробітку енергії та екологічний вплив [20].

Водночас застосування генетичних алгоритмів має певні обмеження. Вони потребують значних обчислювальних ресурсів, особливо у випадку великих популяцій та складних цільових функцій. Крім того, результативність ГА значною мірою залежить від вибору параметрів – розміру популяції, ймовірності кросоверу та мутації, кількості поколінь. Невдалий вибір цих параметрів може призвести до повільної збіжності або недостатньої якості отриманих рішень [21].

Генетичні алгоритми знайшли широке застосування у сфері проєктування та оптимізації сонячних електростанцій. Одним із основних завдань є визначення оптимальної конфігурації розміщення фотоелектричних панелей на ділянці. У реальних умовах необхідно враховувати площу земельної ділянки, особливості рельєфу, можливе затінення, відстань до точок підключення та характеристики обладнання. Використання ГА дозволяє знаходити найкращі варіанти, що забезпечують максимальний виробіток енергії при мінімальних витратах [22].

Ще одним напрямом застосування є оптимізація параметрів керування сонячними фермами. Наприклад, ГА можуть використовуватися для

налаштування алгоритмів стеження за сонцем (solar tracking systems), які змінюють кут нахилу панелей протягом дня. Такий підхід дозволяє збільшити продуктивність системи на 15–25% порівняно з фіксованими установками.

Крім того, генетичні алгоритми ефективно застосовуються у задачах прогнозування та балансування енергетичних потоків. Зокрема, вони можуть використовуватися для оптимізації роботи гібридних систем, що поєднують сонячну генерацію з іншими джерелами енергії та накопичувачами [23]. Це дає змогу мінімізувати втрати та забезпечити стабільне енергопостачання навіть у пікові періоди навантаження.

У багатьох наукових роботах показано ефективність використання ГА для задач енергетики. Наприклад, дослідження, проведені в університетах Німеччини та США, довели, що застосування ГА для планування сонячних електростанцій дозволяє зменшити площу необхідної земельної ділянки на 10–15% без втрат у виробітку енергії. В інших експериментах використання ГА для оптимізації топології підключення панелей до інверторів дало змогу скоротити втрати електроенергії на 5–7%. Практичні приклади також демонструють можливість застосування цього методу на етапі експлуатації, коли необхідно адаптувати роботу станції до змінних погодних умов чи коливань попиту [24].

Отже, генетичні алгоритми є потужним і гнучким інструментом оптимізації, що дозволяє розв'язувати складні багатофакторні задачі у сфері проєктування та експлуатації сонячних ферм. Їхня універсальність, здатність до глобального пошуку рішень і можливість врахування багатьох критеріїв одночасно роблять цей підхід одним із найбільш перспективних у сучасній енергетиці. Водночас ефективне застосування ГА вимагає якісних вхідних даних, значних обчислювальних ресурсів та ретельного налаштування параметрів алгоритму [25]. Подальший розвиток цієї галузі пов'язаний із поєднанням генетичних алгоритмів з іншими методами штучного інтелекту, що дозволить створювати ще більш ефективні системи підтримки прийняття рішень у відновлюваній енергетиці.

## 1.4 Постановка завдання дослідження

Аналіз сучасного стану розвитку сонячної енергетики та методів проєктування сонячних ферм показав, що традиційні та навіть класичні математичні підходи не завжди забезпечують необхідну точність і ефективність у вирішенні практичних завдань. Значна кількість факторів, що впливають на роботу фотоелектричних систем, зокрема рівень сонячної інсоляції, особливості рельєфу, кліматичні умови, конфігурація обладнання, наявність затінення та втрати в електричних мережах, створюють багатовимірний простір змінних із нелінійними взаємозв'язками. В таких умовах традиційні методи здатні забезпечувати лише наближені рішення, які часто виявляються не оптимальними з точки зору ефективності використання ресурсів.

Програмно-імітаційні комплекси, такі як PVsyst, HOMER чи SAM, значно розширюють можливості аналізу й прогнозування, проте вони мають обмежену гнучкість. Їхні алгоритми здебільшого орієнтовані на стандартні сценарії й не дозволяють здійснювати пошук інноваційних рішень у складних багатофакторних середовищах. Саме тому постає необхідність застосування інтелектуальних методів оптимізації, які здатні адаптивно враховувати широкий спектр параметрів та обмежень, працювати з неповними або невизначеними даними та знаходити глобально оптимальні рішення.

У цьому контексті особливий інтерес становлять генетичні алгоритми, які продемонстрували високу ефективність у вирішенні задач багатокритеріальної оптимізації в енергетиці. Вони дають змогу одночасно враховувати економічні, технічні та екологічні показники, що є основними для сучасних сонячних ферм. Використання ГА дозволяє не лише оптимізувати розміщення фотоелектричних модулів і параметри їхньої роботи, але й забезпечити баланс між вартістю будівництва, терміном окупності та рівнем виробітку електроенергії.

Виходячи з цього, завданням даного дослідження є розробка моделі оптимізації процесу проєктування сонячної ферми з використанням генетичних алгоритмів. Для досягнення поставленої мети необхідно:

- 1) сформулювати математичну модель задачі оптимізації, яка враховує основні технічні та економічні параметри системи;
- 2) обґрунтувати вибір цільової функції та системи обмежень;
- 3) розробити архітектуру алгоритму, що реалізує еволюційний підхід до пошуку оптимальних рішень;
- 4) створити прототип програмного забезпечення для проведення обчислювальних експериментів;
- 5) здійснити серію симуляцій і порівняти результати роботи алгоритму з традиційними методами проектування;
- 6) оцінити ефективність розробленої системи та визначити напрями її подальшого вдосконалення.

Отже, основна дослідницька проблема полягає в пошуку підходів, які дозволять створити більш точну та адаптивну систему підтримки прийняття рішень для проектування сонячних ферм. Використання генетичних алгоритмів як базового інструменту оптимізації у цій сфері розглядається як перспективний напрям, здатний забезпечити підвищення ефективності відновлюваної енергетики та сприяти досягненню стратегічних цілей енергетичної безпеки й сталого розвитку.

## РОЗДІЛ 2

### МОДЕЛЮВАННЯ СИСТЕМИ

#### 2.1 Функціональне моделювання

Функціональне моделювання є одним із базових етапів системного аналізу та проєктування складних інформаційних систем. Його метою є формалізований опис функцій, які виконує система, та взаємозв'язків між ними, що дозволяє побудувати цілісне уявлення про структуру і логіку роботи майбутнього програмного комплексу. У контексті оптимізації проєктування сонячних ферм функціональне моделювання відіграє особливо важливу роль, оскільки забезпечує можливість структурування задачі оптимізації, визначення основних підсистем і потоків даних, які впливають на кінцевий результат.

Застосування функціонального моделювання у даній роботі дає змогу відобразити процес від збирання вихідної інформації про сонячну ферму (рівень сонячної інсоляції, рельєф місцевості, кліматичні показники, технічні характеристики обладнання) до отримання оптимального проєктного рішення. Кожен етап роботи системи може бути поданий у вигляді окремої функції з чітко визначеними вхідними і вихідними параметрами. Це дозволяє встановити логічні залежності між підсистемами та виявити «вузькі місця», які потребують додаткової уваги.

Для побудови функціональних моделей у дослідженні доцільно використати нотації IDEF0 або DFD (Data Flow Diagram), які є стандартом у проєктуванні інформаційних систем. Використання таких нотацій дозволяє представити систему у вигляді ієрархії діаграм, де на верхньому рівні подається загальний процес оптимізації, а на нижчих рівнях деталізуються окремі його складові.

Схема на рисунку 2.1 показує деталізований процес оптимізації проєктування сонячної ферми. Він складається з чотирьох послідовних етапів: збір і підготовка даних (A1), формування цільової функції та обмежень (A2),

робота оптимізаційного модуля на основі генетичного алгоритму (A3) та аналіз і візуалізація результатів (A4).

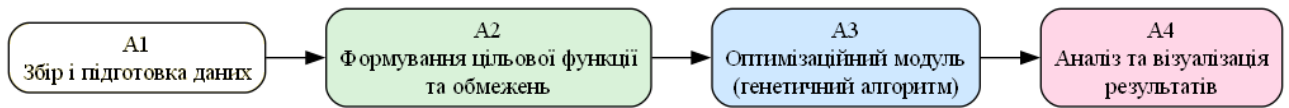


Рис. 2.1 Функціональна модель процесу оптимізації проєктування сонячної ферми

Діаграма на рисунку 2.2 – це узагальнена IDEF0-модель верхнього рівня системи. Вона ілюструє основні складові процесу: вхідні дані (клімат, інсоляція, географія, обладнання), керуючі фактори (цілі оптимізації та обмеження), механізм (генетичний алгоритм і обчислювальні ресурси) та вихід у вигляді оптимального рішення щодо розміщення панелей, виробітку та вартості.

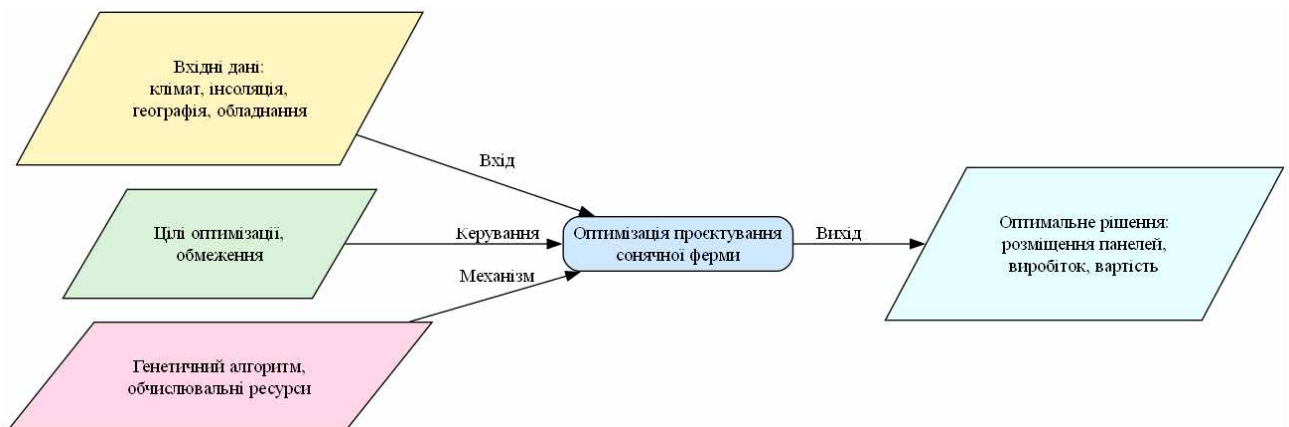


Рис. 2.2 IDEF0-модель верхнього рівня системи

Функціональна модель системи оптимізації проєктування сонячних ферм включає такі основні компоненти:

- 1) збирання та підготовка вхідних даних – кліматичні характеристики, параметри сонячної інсоляції, географічні особливості, технічні характеристики обладнання, економічні показники;

2) формування цільової функції та обмежень – визначення критеріїв оптимізації (мінімізація вартості, максимізація виробітку енергії, скорочення викидів тощо);

3) робота оптимізаційного модуля – застосування генетичного алгоритму для пошуку оптимальної конфігурації;

4) аналіз та візуалізація результатів – оцінка ефективності обраного рішення та представлення результатів у вигляді графіків, таблиць чи звітів.

Узагальнено функціональне моделювання дозволяє визначити логіку функціонування системи ще до її програмної реалізації, мінімізувати ризики помилок у майбутньому та забезпечити можливість масштабування. Важливо, що завдяки цьому підходу розробник отримує інструмент для перевірки правильності постановки задачі оптимізації та відповідності системи заданим вимогам.

## **2.2 Об'єктно-орієнтоване моделювання**

Об'єктно-орієнтоване моделювання (ООМ) є наступним етапом після функціонального аналізу системи. Воно дозволяє описати структуру системи у вигляді об'єктів, класів і взаємозв'язків між ними. Такий підхід забезпечує модульність, масштабованість і гнучкість програмної реалізації. У даному дослідженні ООМ застосовується для побудови структури системи оптимізації проєктування сонячних ферм, яка використовує генетичний алгоритм як основний механізм пошуку оптимальних рішень.

Система складається з кількох логічних компонентів:

1) модуль введення даних – відповідає за збирання та підготовку кліматичних, географічних і технічних параметрів;

2) модуль формування задачі оптимізації – визначає цільову функцію, обмеження та параметри пошуку;

3) модуль генетичного алгоритму – реалізує еволюційні оператори (ініціалізацію популяції, відбір, схрещування, мутацію, оцінювання пристосованості);

4) модуль аналізу та візуалізації – виконує обробку отриманих результатів і подає їх у графічному або табличному вигляді.

Ці підсистеми взаємодіють між собою відповідно до логіки, визначеної у функціональному моделюванні (див. рис. 2.1, 2.2).

На рисунку 2.3 подано UML-діаграму класів, яка відображає основні сутності та зв'язки між ними.

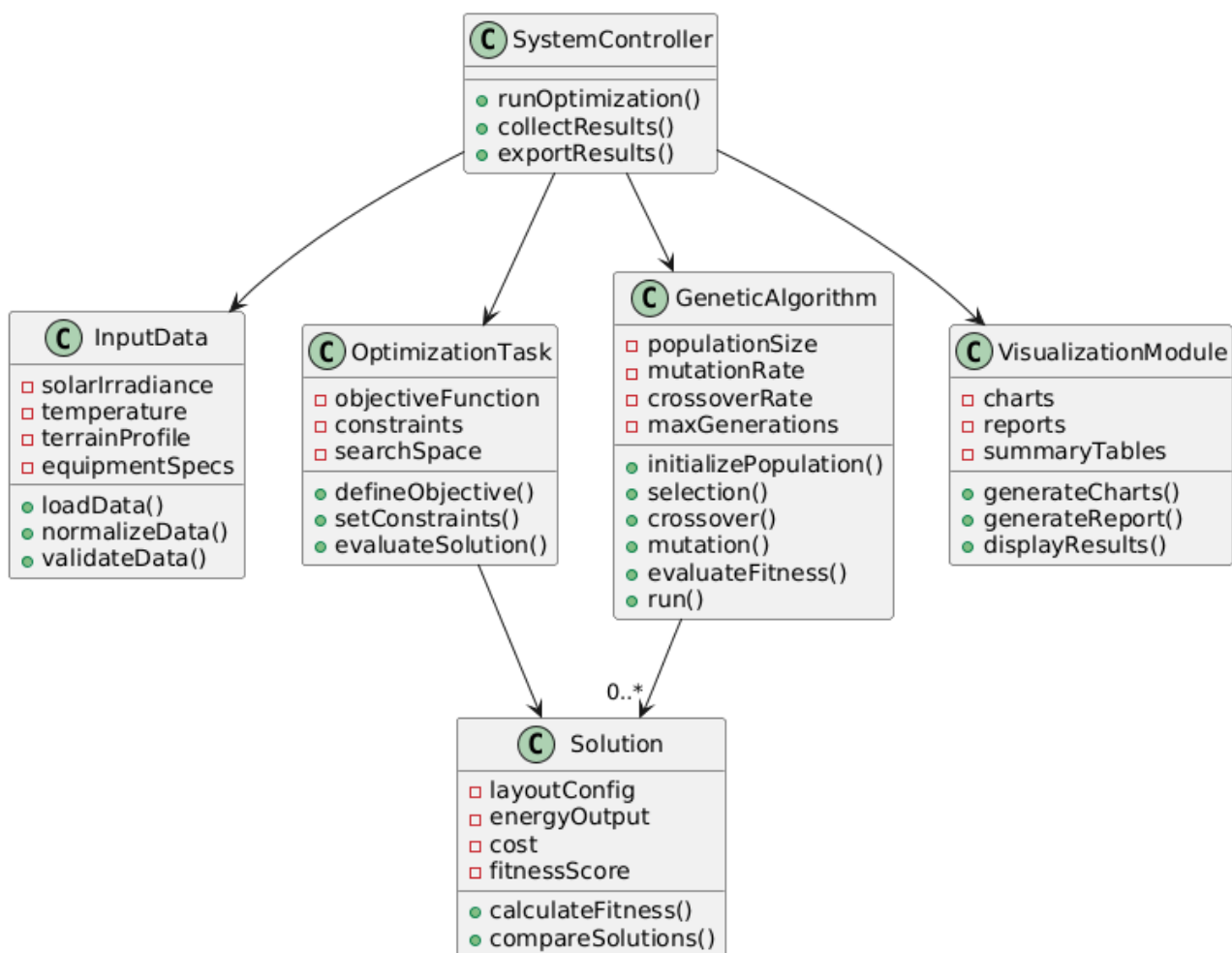


Рис. 2.3 UML-діаграма класів системи оптимізації проєктування сонячних ферм

Основні класи системи описано в таблиці 2.1.

Таблиця 2.1 – Характеристика основних класів системи оптимізації проектування сонячних ферм

№	Назва класу	Атрибути	Методи	Опис
1	InputData	solarIrradiance, temperature, terrainProfile, equipmentSpecs, economicParams	loadData(), normalizeData(), validateData()	Забезпечує зчитування, перевірку та нормалізацію вихідних даних, необхідних для подальшої оптимізації.
2	OptimizationTask	objectiveFunction, constraints, searchSpace	defineObjective(), setConstraints(), evaluateSolution()	Формує математичну модель задачі оптимізації, визначає цільову функцію та систему обмежень.
3	GeneticAlgorithm	populationSize, mutationRate, crossoverRate, maxGenerations	initializePopulation(), selection(), crossover(), mutation(), evaluateFitness(), run()	Реалізує еволюційний процес пошуку оптимального рішення за допомогою генетичних операторів.
4	Solution	layoutConfig, energyOutput, cost, fitnessScore	calculateFitness(), compareSolutions()	Представляє окремий варіант конфігурації сонячної ферми з розрахованими показниками ефективності.
5	VisualizationModule	charts, reports, summaryTables	generateCharts(), generateReport(), displayResults()	Відповідає за аналіз та візуальне представлення результатів оптимізації у вигляді графіків і звітів.
6	SystemController	—	runOptimization(), collectResults(), exportResults()	Координує роботу всіх модулів системи, забезпечує послідовне виконання етапів оптимізації та

				узагальнення результатів.
--	--	--	--	------------------------------

Зв'язки між класами: `SystemController` асоціюється з усіма іншими класами як керівний елемент, `GeneticAlgorithm` використовує об'єкти класу `Solution` і взаємодіє з `OptimizationTask` для оцінки цільової функції, `VisualizationModule` отримує результати від `GeneticAlgorithm` і створює аналітичні звіти, `InputData` передає підготовлені дані у `OptimizationTask`.

На рисунку 2.4 зображено діаграму об'єктів, яка показує конкретні екземпляри класів у процесі виконання програми.

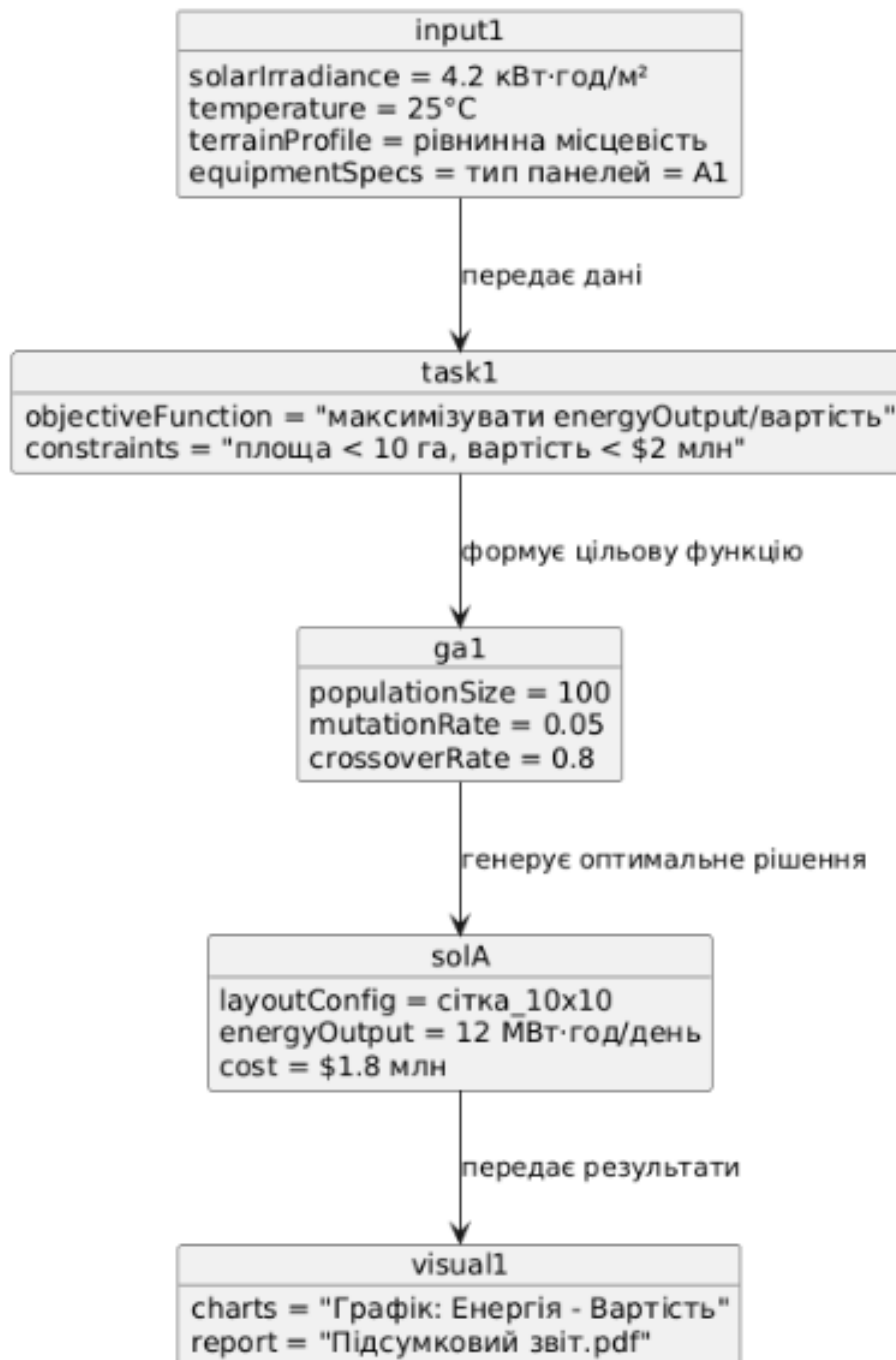


Рис. 2.4 Діаграма об'єктів у процесі виконання оптимізації

Детальніший опис:

- 1) об'єкт `input1:InputData` містить дані про рівень інсоляції та температуру;
- 2) `task1:OptimizationTask` використовує ці дані для побудови цільової функції;

3) gal:GeneticAlgorithm створює популяцію рішень, серед яких solA:Solution, solB:Solution тощо;

4) після завершення ітерацій visual1:VisualizationModule формує звіт.

Діаграма прецедентів, наведена на рисунку 2.5, демонструє взаємодію користувача із системою. Актор: проєктувальник сонячної ферми.

Прецеденти:

- 1) «Завантажити вхідні дані»;
- 2) «Визначити параметри оптимізації»;
- 3) «Запустити генетичний алгоритм»;
- 4) «Отримати результати оптимізації»;
- 5) «Згенерувати звіт і графіки».



Рис. 2.5 Діаграма прецедентів взаємодії користувача із системою

На рисунку 2.6 подано діаграму послідовності, яка ілюструє часову динаміку обміну повідомленнями між основними об'єктами під час виконання алгоритму:

- 1) користувач ініціює процес через SystemController;
- 2) відбувається виклик loadData() у InputData;

- 3) дані передаються у OptimizationTask для формування моделі;
- 4) SystemController викликає run() у GeneticAlgorithm;
- 5) алгоритм ітеративно виконує selection(), crossover(), mutation() і evaluateFitness();
- 6) результати передаються у VisualizationModule для генерації підсумкових графіків.

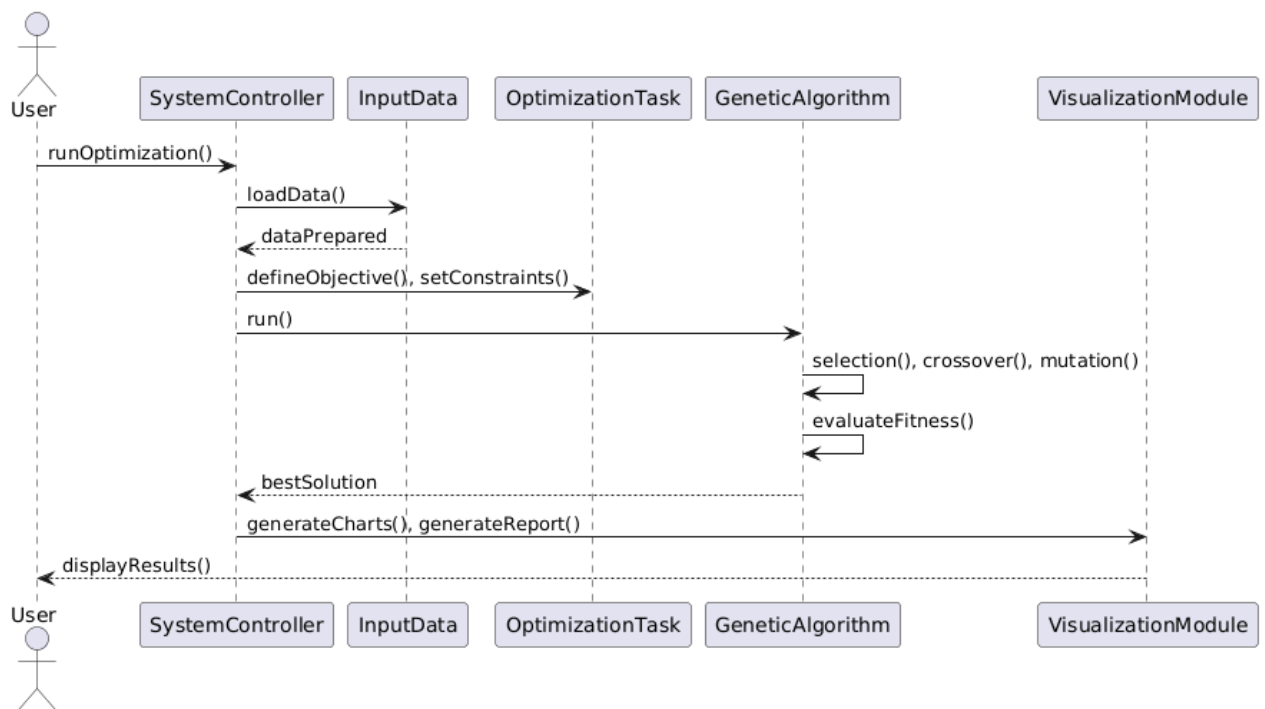


Рис. 2.6 Діаграма послідовності роботи системи

Діаграма компонентів відображає архітектуру системи оптимізації проєктування сонячних ферм із використанням генетичних алгоритмів на рівні основних програмних модулів. Вона показує, як між собою взаємодіють компоненти системи: модуль введення даних забезпечує збір та попередню обробку інформації про клімат, інсоляцію, рельєф і технічні характеристики обладнання; ядро оптимізації формує цільову функцію й обмеження, передаючи дані в модуль генетичного алгоритму, який виконує пошук оптимальної конфігурації. Отримані результати надходять у модуль візуалізації та звітності, де здійснюється аналіз і представлення оптимальних рішень користувачеві.

Діаграма демонструє логічну цілісність системи, чіткий поділ функцій між компонентами та послідовність потоків даних від введення до кінцевого результату.

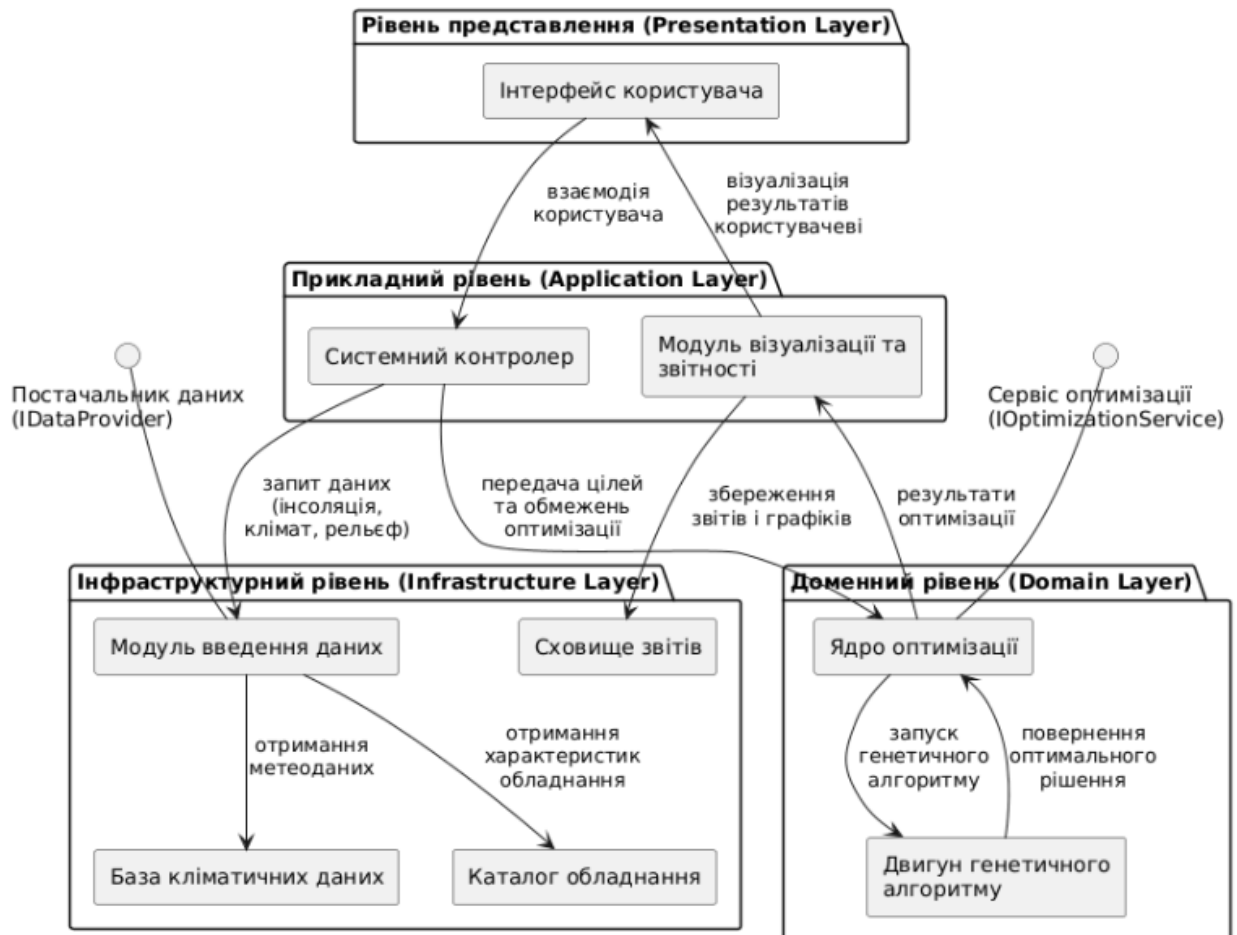


Рис. 2.7 Діаграма компонентів системи

Об'єктно-орієнтована модель системи забезпечує чітку структурування компонентів і зв'язків між ними, можливість повторного використання коду, простоту модифікації (зокрема, адаптацію для інших типів відновлюваних джерел енергії) та гнучкість у виборі параметрів генетичного алгоритму. Отже, ООМ дозволяє реалізувати інтелектуальну систему, здатну ефективно розв'язувати задачі оптимізації проектування сонячних ферм із використанням еволюційних підходів.

## 2.3 Формалізація задачі оптимізації

Формалізація задачі оптимізації є основним етапом побудови системи, оскільки саме на цьому рівні визначаються математичні залежності, критерії ефективності та обмеження, що описують роботу сонячної ферми. У межах даного дослідження метою є знаходження оптимальної конфігурації розміщення фотоелектричних панелей і параметрів системи, яка забезпечує максимальний виробіток електроенергії при мінімальній вартості будівництва та експлуатації.

Нехай розглядається земельна ділянка, на якій розміщується множина фотоелектричних панелей. Кожна панель характеризується низкою параметрів:

- 1) координати розташування  $(x_i, y_i)$ ;
- 2) кут нахилу  $\alpha_i$ ;
- 3) орієнтація відносно півдня  $\beta_i$ ;
- 4) тип модуля (ефективність, площа, вартість).

Необхідно визначити оптимальні значення цих параметрів, щоб досягти балансу між виробленою енергією  $E$  та загальною вартістю проекту  $C$ .

Відповідно, задача формулюється як багатокритеріальна оптимізація з двома основними цілями:

$$\{ \max E(x, y, \alpha, \beta) \min C(x, y, \alpha, \beta) \} \quad (2.1)$$

Для зручності цілі об'єднуються в одну зведену цільову функцію  $F$ :

$$\max F = w_1 \cdot \frac{E}{E_{\max}} - w_2 \cdot \frac{C}{C_{\max}} \quad (2.2)$$

де  $w_1, w_2$  – вагові коефіцієнти, що визначають пріоритет енергетичної та економічної ефективності відповідно,  $E_{\max}$  та  $C_{\max}$  – нормувальні константи (максимальні можливі значення).

Оптимізація відбувається з урахуванням технічних, геометричних і природних обмежень. Обмеження площі ділянки:

$$S_{occupied} \leq S_{available} \quad (2.3)$$

де  $S_{occupied}$  – сумарна площа зайнята панелями,  $S_{available}$  – площа доступної території.

Мінімальні відстані між панелями для уникнення затінення:

$$d_{ij} \geq d_{min}, \forall i \neq j \quad (2.4)$$

Діапазони кутів нахилу та орієнтації:

$$\alpha_{min} \leq \alpha_i \leq \alpha_{max}, \beta_{min} \leq \beta_i \leq \beta_{max} \quad (2.5)$$

Обмеження потужності:

$$\sum_{i=1}^N P_i \leq P_{max} \quad (2.6)$$

де  $P_i$  – потужність окремої панелі,  $P_{max}$  – максимально допустиме значення для електромережі.

Економічні обмеження:

$$C(x, y, \alpha, \beta) \leq C_{budget} \quad (2.7)$$

що забезпечує узгодження рішень із фінансовими можливостями замовника.

Функцію виробленої енергії можна описати як:

$$E = \sum_{i=1}^N \eta_i \cdot I(x_i, y_i, \alpha_i, \beta_i) \cdot A_i \cdot t \quad (2.8)$$

де  $\eta_i$  – коефіцієнт корисної дії панелі,  $I$  – інтенсивність сонячної радіації для даного положення і кута,  $A_i$  – площа панелі,  $t$  – тривалість періоду експлуатації.

Вартість системи визначається як:

$$C = \sum_{i=1}^N (C_{panel,i} + C_{install,i}) + C_{inverter} + C_{land} \quad (2.9)$$

Відповідно, алгоритм оптимізації має шукати таку комбінацію параметрів, що максимізує  $F$  за умов виконання усіх вищенаведених обмежень.

Кожне можливе рішення (індивід у популяції) кодується у вигляді хромосоми:

$$C \text{ chromosome}_i = [x_1, y_1, \alpha_1, \beta_1, \dots, x_N, y_N, \alpha_N, \beta_N] \quad (2.10)$$

Для кожного покоління застосовуються генетичні оператори. Селекція – відбір найкращих індивідів за значенням функції пристосованості  $F$ . Кросовер – комбінування параметрів двох «батьківських» рішень. Мутація – випадкові зміни частини генів для уникнення локальних екстремумів. Оцінювання – розрахунок нових значень  $F$  для кожного нащадка.

Цикл повторюється до досягнення критерію зупинки – стабілізації результатів або досягнення максимальної кількості поколінь  $G_{max}$ .

Узагальнено задачу можна подати так:

$$\max F(X) = w_1 \cdot \frac{E(X)}{E_{max}} - w_2 \cdot \frac{C(X)}{C_{max}} \quad (2.11)$$

за умов:

$$S_{occupied}(X) \leq S_{available} \quad (2.12)$$

$$d_{ij}(X) \geq d_{min}, \forall i \neq j \quad (2.13)$$

$$C(X) \leq C_{budget} \quad (2.14)$$

$$X \in D$$

$$(2.15)$$

де  $X$  – вектор змінних, що описує конфігурацію сонячної ферми,  $D$  – множина допустимих рішень.

Формалізація задачі оптимізації дозволила перейти від концептуальної до математичної моделі системи. Задача описана як багатокритеріальна, з урахуванням технічних, геометричних і фінансових обмежень. Така постановка є придатною для реалізації за допомогою генетичних алгоритмів, які ефективно працюють у складних нелінійних просторах рішень і забезпечують глобальний пошук оптимуму.

## РОЗДІЛ 3

### РОЗРОБКА СИСТЕМИ

#### 3.1 Архітектура системи оптимізації проєктування сонячних ферм

Архітектура розробленої системи оптимізації проєктування сонячних ферм побудована на модульному підході, що забезпечує розділення логіки роботи, гнучкість у розширенні функціоналу та можливість повторного використання компонентів. Система реалізована мовою програмування Python з використанням бібліотек Tkinter (для графічного інтерфейсу користувача), NumPy (матриці та обчислення), Matplotlib (побудова графіків), Pandas (обробка даних), а також власної реалізації генетичного алгоритму.

Загальна архітектура відповідає принципам клієнт-серверної взаємодії в межах локальної десктопної системи. Програмний продукт поділено на три основні рівні:

- 1) презентаційний рівень (GUI) – забезпечує взаємодію з користувачем;
- 2) логічний (обчислювальний) рівень – виконує оптимізаційні обчислення за допомогою генетичного алгоритму;
- 3) дані та сервісний рівень – відповідає за збереження, імпорт та валідацію вхідних даних, а також за експорт результатів у різних форматах.

Система складається з кількох основних компонентів (табл. 3.1).

Таблиця 3.1 – Структурна модель системи

Компонент	Призначення
main.py	Головний модуль, ініціалізує інтерфейс та координує роботу підсистем
data_handler.py	Обробка вхідних даних (CSV/ручний ввід), валідація, формування структури даних
ga_engine.py	Реалізація генетичного алгоритму та цільової функції оптимізації
visualization.py	Побудова графіків збіжності та аналізу результатів
report_export.py	Експорт результатів у CSV, TXT, PDF та генерація повного звіту
input_data.csv	Приклад тестових даних для запуску

Архітектурна діаграма системи представлена на рисунку 3.1.

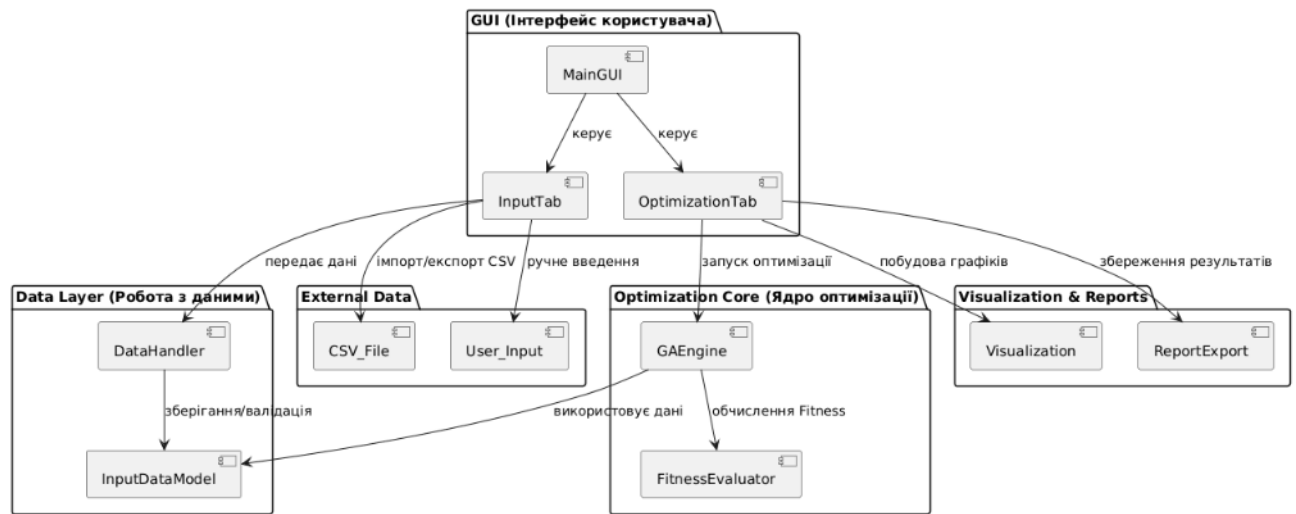


Рис. 3.1 Діаграма архітектури системи

У системі застосовано багаторівневу структурну архітектуру, узгоджену з підходами MVC (Model-View-Controller). Контролером виступає головне вікно програми, моделлю – оптимізаційний блок та вхідні дані, а View реалізовано на основі бібліотеки Tkinter. Обмін даними між компонентами здійснюється через чітко визначені інтерфейси, що забезпечує слабку зв'язаність модулів.

Взаємодія компонентів системи має послідовний і логічний характер. На першому етапі користувач здійснює завантаження або ручне введення початкових даних у спеціально створеній вкладці «Вхідні дані». Після цього дані автоматично перевіряються на коректність та повноту модулем `data_handler.py`, який виконує їх валідацію та формує структуроване подання для подальшої обробки. На другому етапі у вкладці «Оптимізація (ГА)» користувач задає параметри роботи генетичного алгоритму, включаючи розмір популяції, кількість поколінь, коефіцієнти мутації та кросоверу. Обчислювальний процес здійснюється модулем `ga_engine.py`, який реалізує оптимізацію конфігурації сонячної ферми на основі заданих обмежень та цільової функції. Після завершення оптимізації програма формує зрозуміле текстове пояснення

отриманого результату та будує графік збіжності алгоритму, що відображає динаміку пошуку оптимального рішення. Заключним етапом є збереження результатів у файл, яке виконується через модуль `report_export.py` у одному з форматів на вибір користувача – CSV, TXT або PDF.

Архітектура системи відзначається низкою важливих властивостей. По-перше, вона є модульною, тобто кожен її компонент виконує окрему функцію та може бути незалежно модифікований або вдосконалений. По-друге, її структура забезпечує масштабованість, що дозволяє у майбутньому інтегрувати додаткові алгоритми оптимізації, зокрема методи рою частинок (PSO), диференціальної еволюції (DE) або багатокритеріальної оптимізації (NSGA-II). По-третє, система підтримує гнучкість обчислень завдяки можливості виконання як стандартного запуску, так і наукового багатократного запуску з автоматичним обчисленням статистичних показників результатів. Окрім того, інтерпретація результатів робить систему прозорою для користувача, адже вона не лише генерує оптимальну конфігурацію, а й пояснює її енергетичну та економічну ефективність. Завдяки цьому система має прикладний характер і може бути використана як для наукових досліджень, так і для практичних інженерних розрахунків при плануванні та проєктуванні сонячних електростанцій.

Загалом, архітектура системи забезпечує надійну інтеграцію методів оптимізації з інтерфейсом користувача та гнучку адаптацію до різних конфігурацій задачі. Вибір багаторівневої архітектури та модульної структури робить систему придатною для реального практичного застосування у сфері проєктування фотоелектричних систем.

### **3.2 Алгоритмічне забезпечення**

Алгоритмічне забезпечення розробленої системи оптимізації проєктування сонячних ферм ґрунтується на використанні генетичного алгоритму як базового еволюційного методу пошуку глобального оптимуму у багатовимірному просторі рішень. Його вибір зумовлений здатністю ефективно працювати з

нелінійними залежностями, великою кількістю змінних та обмежень, а також відсутністю потреби в аналітичному описі функції оптимізації.

Розроблений алгоритм складається з послідовності взаємопов'язаних етапів, кожен із яких реалізує певний елемент еволюційного процесу: ініціалізацію, оцінку, селекцію, кросовер, мутацію та оновлення популяції. Узагальнена схема алгоритму наведена на рисунку 3.2.

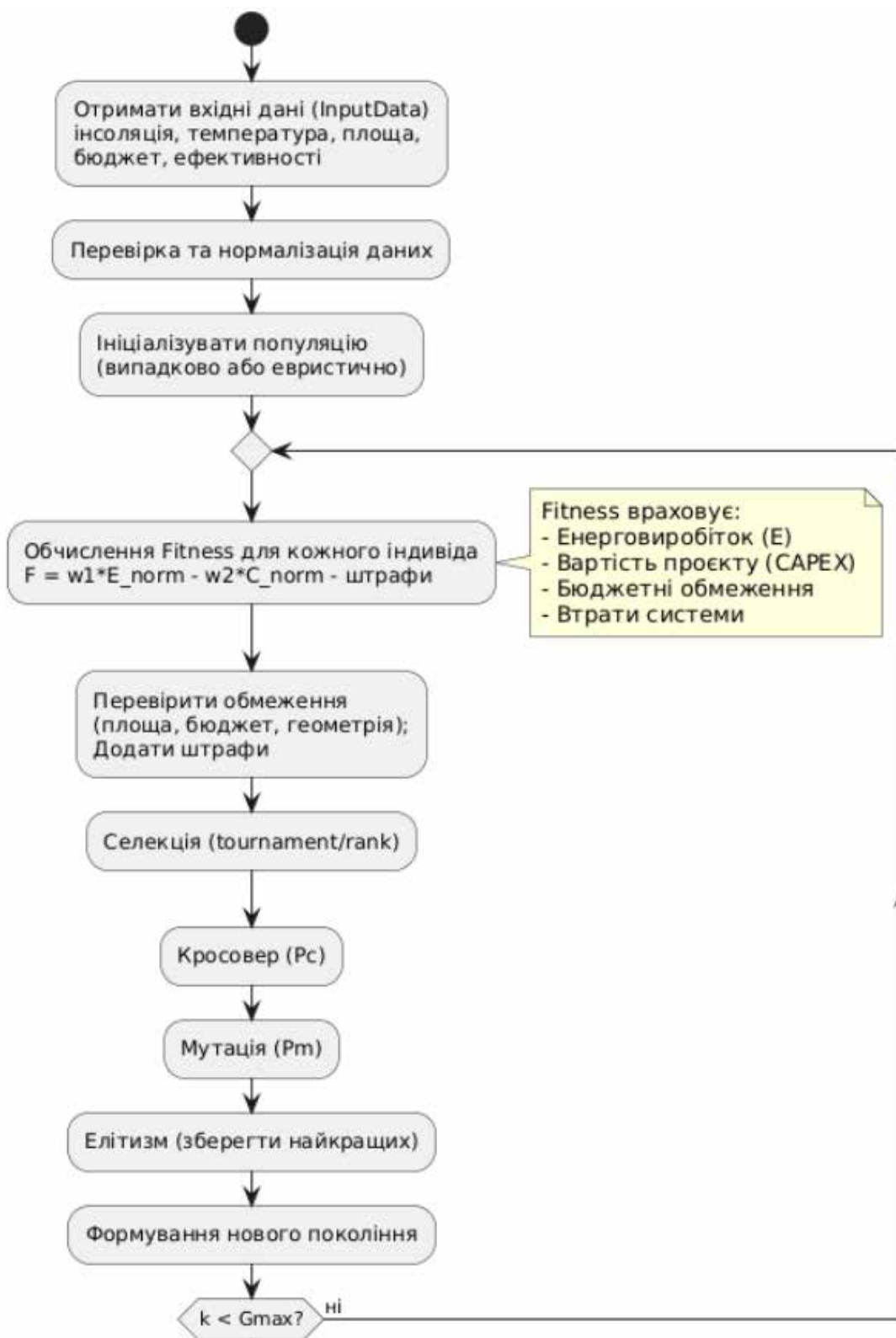


Рис. 3.2 Узагальнена схема роботи генетичного алгоритму в системі оптимізації сонячних ферм



Продовження рис. 3.2

Основні етапи роботи алгоритму наведемо далі. Процес оптимізації здійснюється над набором змінних, що визначають параметри майбутньої фотоелектричної станції: кут нахилу панелей, орієнтація (азимут), відстань між рядами, кількість панелей у ряду, висота конструкції, а також інженерні обмеження (площа ділянки, бюджет, втрати системи, ККД інвертора). Алгоритм

працює з цими параметрами шляхом ітераційної генерації, оцінювання та вдосконалення множини кандидатних рішень.

На першому етапі генерується початкова популяція допустимих рішень. Популяція формується випадково з урахуванням заданих діапазонів параметрів та перевірки обмежень. Кожен індивід кодується у вигляді хромосоми – структурованого набору числових значень, що описують конкретний варіант розміщення і технічних характеристик сонячної ферми.

Далі здійснюється обчислення значення цільової функції (fitness function), яка у даному дослідженні є інтегральним показником енергетичної та економічної ефективності:

$$F = \frac{E_{day}}{CAPEX} \times K_{eff} \quad (3.1)$$

де  $E_{day}$ — добовий енергетичний виробіток (кВт·год),  $CAPEX$  – капітальні витрати на будівництво станції,  $K_{eff}$  – коефіцієнт ефективності з урахуванням втрат (температурні, інсоляційні, інверторні, топографічні).

Після оцінювання популяції реалізується селекція – відбір найбільш пристосованих індивідів за принципом турнірного відбору, що дозволяє підтримувати конкурентність рішень. У процесі селекції також застосовується елітизм – гарантується збереження найкращих рішень для перенесення в наступне покоління без змін.

На наступному етапі виконується генетичний оператор схрещування (кросовер), який комбінує параметри двох «батьківських» рішень з метою формування нових індивідів. У реалізованій системі використано одноточковий та двоточковий кросовер з коефіцієнтом  $P_c = 0.8$ , що дозволяє зберігати основні структурні характеристики кращих рішень.

Для підтримання різноманітності популяції та уникнення передчасної збіжності алгоритму застосовується оператор мутації з ймовірністю  $P_m = 0.06$ ,

який випадково змінює окремі гени хромосом. Це підвищує ймовірність знаходження глобального оптимуму в умовах складного пошукового простору.

Формування нового покоління відбувається за допомогою поєднання відібраних, схрещених та мутованих індивідів. Процес повторюється ітераційно, поки не буде досягнуто умови зупинки: або виконано максимальну кількість поколінь  $G_{max}$ , або приріст функції пристосованості стає незначним:

$$| F_{best}^{(k)} - F_{best}^{(k-1)} | < \varepsilon$$

(3.2)

де  $\varepsilon$  – допустимий поріг зупинки.

Завершальним етапом є передача найкращого рішення до модуля візуалізації та звітності, який формує графік збіжності, таблицю параметрів конфігурації ферми та економічні показники (LCOE, строк окупності, річний дохід). Загалом, алгоритмічна частина системи забезпечує стабільний процес пошуку оптимального рішення, працюючи в умовах складної багатокритеріальної задачі з інженерними обмеженнями.

### 3.3 Технологічне забезпечення

Технологічне забезпечення системи оптимізації проектування сонячних ферм включає сукупність програмних інструментів, мов програмування, бібліотек, середовищ виконання та технічних ресурсів, які забезпечують реалізацію, виконання та підтримку функціонування програмного продукту. У процесі розробки було використано сучасні технології, які дозволили створити ефективну, надійну та масштабовану систему з графічним інтерфейсом, придатну як для наукових досліджень, так і для інженерно-проектної діяльності у сфері сонячної енергетики.

Основою технологічного забезпечення є мова програмування Python, яка обрана завдяки своїй гнучкості, високому рівню абстракції, наявності розвиненого екосередовища наукових бібліотек та широкій підтримці інструментів для обробки даних і математичного моделювання. Особливу роль у реалізації генетичного алгоритму та математичних розрахунків відіграють бібліотеки NumPy та random, які забезпечують швидке виконання векторних операцій, статистичних обчислень та генерації випадкових чисел. Табличні дані та імпорт вхідних значень з файлів підтримуються завдяки бібліотеці pandas.

Для реалізації графічного інтерфейсу користувача використано бібліотеку Tkinter, що входить до стандартної поставки Python та дозволяє створювати кросплатформні GUI-застосунки. За допомогою Tkinter реалізовані вікна введення вхідних параметрів, керування запуском алгоритму, відображення результатів, а також функції взаємодії з файлами. Бібліотека ttk використана для візуального покращення інтерфейсу та організації елементів керування.

Для побудови графіків збіжності генетичного алгоритму застосовується бібліотека Matplotlib, яка забезпечує графічну інтерпретацію процесу оптимізації. Візуалізація результатів є важливим компонентом системи, оскільки дозволяє проаналізувати поведінку алгоритму та його здатність досягати стабільних рішень. Формування текстових та табличних звітів реалізовано в окремому модулі за допомогою стандартних інструментів Python, а для генерації PDF-звітів використовується бібліотека ReportLab, що дозволяє структурувати результати у вигляді завершених технічних документів.

Для зберігання та обробки вхідних даних застосовується формат CSV (Comma-Separated Values), який забезпечує сумісність з іншими програмними системами та простоту імпорту/експорту інформації. Система підтримує як завантаження тестових даних, так і їх ручний ввід через форму, що підвищує зручність використання програмного продукту.

Програмна система є кросплатформною та може бути виконана в операційних системах Windows, Linux та macOS за умови встановлення середовища Python версії не нижче 3.9. Виконання програми не потребує

спеціального апаратного забезпечення, однак для прискорення обчислень при великій кількості поколінь генетичного алгоритму рекомендовано використання багато-ядерних процесорів.

Отже, технологічне забезпечення побудовано на основі перевірених часом програмних технологій, що гарантують надійність, доступність та гнучкість системи. Обраний стек технологій дозволяє забезпечити подальше розширення функціональності, інтеграцію з геоінформаційними системами, базами даних або хмарними сервісами для проведення розподілених обчислень у майбутніх розробках.

## РОЗДІЛ 4

### РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

#### 4.1 Апаратні та програмні вимоги до реалізації системи

Розроблена система оптимізації конфігурації сонячних ферм з використанням генетичного алгоритму є настільним програмним застосунком, який не потребує спеціалізованого високопродуктивного обладнання. Водночас для коректної роботи програмного продукту необхідно забезпечити відповідність мінімальним вимогам до апаратного та програмного забезпечення.

Програмна система розроблена таким чином, щоб бути доступною для використання на стандартних персональних комп'ютерах, що характерно для навчальних лабораторій, наукових установ і проектних організацій. Мінімальна конфігурація для запуску програми включає:

- 1) процесор (CPU) – двоядерний процесор з тактовою частотою від 2.0 ГГц (рекомендовано – 4 ядра і більше для швидшої оптимізації);
- 2) оперативна пам'ять (RAM) – не менше 4 ГБ (рекомендовано 8–16 ГБ для роботи з великими масивами даних та експериментальними серіями);
- 3) місце на диску – до 500 МБ для програмного середовища та бібліотек Python, додатково 50–200 МБ для зберігання результатів та звітів;
- 4) відеоадаптер – інтегрований відеочіп достатній, окрема відеокарта не потрібна;
- 5) монітор – мінімальна роздільна здатність 1366×768 (рекомендовано 1920×1080 для зручної роботи з інтерфейсом і графіками).

Для роботи програмного забезпечення необхідно встановити сучасну версію мови програмування Python та додаткові бібліотеки, що використовуються в системі. Перелік програмних вимог включає:

- 1) операційна система – Windows 10/11, Linux Ubuntu 20.04+, macOS 10.14+;
- 2) Python – версія 3.9 або вище [26];

### 3) бібліотеки Python:

- tkinter – для графічного інтерфейсу користувача [27];
- numpy – для чисельних обчислень [28];
- pandas – для роботи з табличними даними [29];
- matplotlib – для побудови графіків збіжності [30];
- random – для реалізації стохастичних властивостей генетичного алгоритму [31];
- reportlab – для генерації PDF-звітів;
- statistics – для статистичного аналізу результатів у науковому режимі;
- csv – для імпорту та експорту даних.

Установка необхідних залежностей може бути виконана за допомогою стандартного менеджера пакетів `pip` командою:

```
pip install numpy pandas matplotlib reportlab
```

Система не потребує зовнішніх баз даних або серверного середовища. Весь обмін даними здійснюється у вигляді локальних CSV-файлів, що робить її зручною для автономного використання. Програма сумісна з іншими інженерними інструментами (Excel, MATLAB, PVGIS) через експорт результатів у стандартних форматах (CSV, TXT, PDF).

Апаратні та програмні вимоги до запуску системи оптимізації сонячних ферм підсумуємо в таблиці 4.1.

Таблиця 4.1 – Апаратні та програмні вимоги системи

Категорія	Мінімальні вимоги	Рекомендовані вимоги
Процесор (CPU)	2 ядра, 2.0 ГГц	4+ ядер, 3.0 ГГц
Оперативна пам'ять (RAM)	4 ГБ	8–16 ГБ

Накопичувач (HDD/SSD)	500 МБ	1 ГБ
Відеоадаптер	Інтегрований GPU	Не потребує дискретної графіки
Монітор	1366×768	1920×1080
Операційна система	Windows 10 / Linux Ubuntu 20.04+ / macOS 10.14+	Будь-яка сучасна ОС

Продовження табл. 4.1

Категорія	Мінімальні вимоги	Рекомендовані вимоги
Мова програмування	Python 3.9	Python 3.10+
Основні бібліотеки	tkinter, numpy, pandas, matplotlib, random, reportlab, csv, statistics	+ seaborn (опц.), scipy (опц.)
Сховище даних	Локальні CSV-файли	Локальні CSV + можливість інтеграції з Excel
Підключення до інтернету	Не обов'язкове	Для оновлення бібліотек

Загалом, розроблене програмне забезпечення характеризується невисокими вимогами до обчислювальних ресурсів, простотою інсталяції та універсальністю використання. Це забезпечує можливість широкого впровадження системи як у навчальних цілях, так і в реальних інженерних завданнях проєктування сонячних електростанцій.

## 4.2 Проведення експериментів та симуляцій

Проведення експериментів із застосуванням генетичного алгоритму (ГА) мало на меті оцінити ефективність розробленої системи оптимізації конфігурацій технічних параметрів енергетичної установки (сонячної електростанції) за заданими критеріями. Для цього було реалізовано серію симуляцій у середовищі Python з використанням бібліотек DEAP, NumPy, Tkinter та Matplotlib. Алгоритм оптимізації інтегрувався з модулем розрахунку

енергетичних та економічних показників, що дало змогу комплексно оцінювати функцію пристосованості (fitness).

Основною метою експериментальної частини дослідження була комплексна оцінка ефективності роботи генетичного алгоритму в задачі оптимізації конфігурації сонячної ферми. Передусім перевірялася його збіжність, тобто здатність поступово знаходити покращені рішення та наближатися до оптимального результату протягом еволюційного процесу. Окрему увагу приділено стабільності отриманих результатів за різних налаштувань алгоритму, що дало змогу оцінити відтворюваність та надійність пошуку.

Також досліджувався вплив ключових параметрів генетичного алгоритму, таких як ймовірність мутації, коефіцієнт кросинговеру та розмір популяції, на якість кінцевих рішень. Важливо було встановити, як саме ці параметри впливають на швидкість пошуку, уникнення локальних екстремумів та глобальну ефективність оптимізації. Крім того, проводився аналіз чутливості алгоритму до зміни вхідних даних задачі, зокрема кліматичних умов, реальних технічних обмежень фотоелектричних модулів та економічних факторів. Це дало змогу визначити адаптивність запропонованого підходу до різних типів вихідних сценаріїв і практичних умов.

Експерименти проводилися за наступних параметрів:

- 1) мова реалізації – Python 3.10;
- 2) робоче середовище – Windows 10 x64;
- 3) інтерфейс програми – графічний інтерфейс на базі Tkinter;
- 4) метод оптимізації – генетичний алгоритм (максимізація Fitness);
- 5) кількість прогонів – 1 (звичайний режим) та 5 (науковий режим);
- 6) початкові параметри ГА (базові значення):
  - розмір популяції: 140 особин;
  - кількість поколінь: 70;
  - ймовірність кросинговеру: 0.8;

- ймовірність мутації: 0.06;
- метод відбору: турнірний;
- метод схрещування: Blend Crossover (cxBlend);
- метод мутації: гаусова з обмеженням у межах діапазону параметра.

Експеримент було проведено у два етапи. Дані про кожен прогін зберігалися в окремих структурах (історія поколінь, найкращі рішення, енергетичні та фінансові показники). Для подальшої аналітики передбачено експорт результатів у файли:

- 1) CSV – історія еволюції (покоління – найкраще F – середнє F);
- 2) TXT – текстовий звіт з описом найкращого рішення;
- 3) PDF – таблиця або повний структурований звіт;
- 4) графіки – криві збіжності, що будуються для кожного прогону.

Перша група експериментів була спрямована на дослідження впливу зміни характеристик ділянки та зовнішніх умов (інсоляція, площа, бюджетні обмеження, потужність модулів) на результати оптимізації конфігурації сонячної ферми. Вхідні дані змінювались вручну, а оптимізація виконувалася в «звичайному режимі» (1 прогін) для кожного сценарію. Значення параметрів генетичного алгоритму залишались незмінними: популяція – 140; поколінь – 70;  $p_{mut} = 0.06$ ;  $p_{cx} = 0.8$ .

Усього було проведено 5 експериментальних сценаріїв:

- 1) сценарій 1–3: вплив бюджету та площі ділянки при сталих кліматичних умовах;
- 2) сценарій 4: вплив зниження інсоляції та зміна властивостей панелей;
- 3) сценарій 5: вплив високої інсоляції та покращених параметрів обладнання.

Результати цієї серії експериментів представлено в таблиці 4.2.

Таблиця 4.2 – Результати серії експериментів (Група 1)

№	Інсоляція (кВт·год/м <sup>2</sup> / доба)	Пло ща (м <sup>2</sup> )	Бюд жет ( $\text{\$}$ )	Поту жн. панел і (Вт)	К-сть панел ей	Потужн ість СЕС (кВт)	Річна генера ція (МВт·г од)	Payb ack (роки )	LCOE ( $\text{\$/кВт}\cdot$ год)	Fitn ess F
1	4.60	2500	1800 00	400	444	177.6	207.0	8.1	0.094	3.15 3
2	4.60	2500	1000 00	400	220	88.0	102.5	9.2	0.105	2.83 2
3	4.60	6000	1800 00	400	366	146.4	170.6	10.1	0.115	2.60 0
4	3.20	6000	1600 00	350	354	123.9	100.4	16.5	0.173	1.72 4
5	6.80	2200	2000 00	450	450	202.5	353.4	5.1	0.061	4.84 5

Проведемо аналіз результатів. Вплив бюджету (Сценарій 1 vs 2). При однакових кліматичних умовах, але зменшенні бюджету з  $\text{\$}180000$  до  $\text{\$}100000$ , оптимальна кількість панелей скорочується з 444 до 220, встановлена потужність зменшується на 50%, а показник Fitness погіршується з 3.15 до 2.83. Це свідчить, що обмеження бюджету прямо впливає на масштаб інсталяції та знижує ефективність системи.

Вплив площі ділянки (Сценарії 1 vs 3). При збільшенні площі з 2500 м<sup>2</sup> до 6000 м<sup>2</sup> кількість панелей зменшується з 444 до 366 навіть за однакового бюджету. Це пов'язано з тим, що алгоритм автоматично збільшує міжрядні відстані для зменшення затінення, покращуючи генерацію на панель. Проте значення Fitness зменшується (з 3.15 до 2.60), що свідчить про збільшення просторових втрат.

Вплив зниження інсоляції (Сценарій 4). При падінні інсоляції з 4.6 до 3.2 кВт·год/м<sup>2</sup>/добу і зниженні ефективності панелей Fitness різко зменшується з 3.15 до 1.72, окупність зростає до 16.5 років, а LCOE збільшується. Низька сонячна активність критично зменшує енергетичний потенціал проєкту.

Вплив покращених умов (Сценарій 5). При інсоляції 6.8 кВт·год/м<sup>2</sup>/добу та бюджеті \$200000 алгоритм знайшов найкраще рішення з Fitness = 4.845, найменшим LCOE (0.061 \$/кВт·год) та найшвидшою окупністю (5.1 років). Це показує чутливість моделі до ресурсної бази та кліматичних умов.

Загалом результати першої серії експериментів підтвердили, що модель демонструє високу чутливість до ресурсних та природних факторів, а саме інсоляція та бюджет виступають основними драйверами енергетичної ефективності. Алгоритм коректно адаптує геометрію розміщення панелей, зокрема відстань між рядами та їх кількість, залежно від рівня затінення ділянки. Якість отриманої конфігурації (Fitness) зростає за умов високої сонячної активності та достатніх інвестицій, тоді як за слабких зовнішніх умов, зокрема низької інсоляції, система хоч і зберігає працездатність, однак її економічна ефективність істотно знижується.

Група 2 досліджує вплив налаштувань генетичного алгоритму (ГА) на ефективність пошуку оптимальної конфігурації сонячної електростанції. У всіх експериментах використовувався однаковий базовий сценарій (однакова ділянка, бюджет, модулі та кліматичні умови), але змінювались параметри ГА: розмір популяції, рівень мутації та кількість поколінь.

Результати другої серії експериментів наведено в таблиці 4.3.

Таблиця 4.3 – Результати експериментів групи 2 (поведінка ГА)

Параметр / Експеримент	Б1 – мала популяція	Б2 – велика популяція	Б3 – агресивна мутація	Б4 – слабка мутація	Б5 – більше еволюції
Популяція	50	200	140	140	140
Покоління	70	70	70	70	120
Мутація	0.06	0.06	0.15	0.01	0.06
Кросовер	0.8	0.8	0.8	0.8	0.8
Кут нахилу	30°	30°	30°	30°	30°
Відстань між рядами	6.69 м	9.03 м	7.85 м	10.10 м	9.03 м
Панелей у ряду	6	8	6	9	8

Висота конструкції	2.22 м	4.08 м	2.78 м	4.28 м	9.96 м
Кількість панелей	444	440	441	441	440
Потужність, кВт	177.6	176.0	176.4	176.4	176.0
Добова генерація, кВт·год	567.0	561.9	563.2	563.2	561.9
Річна генерація, МВт·год	207.0	205.1	205.6	205.6	205.1
CAPEX (\$)	179 840	178 400	178 760	178 760	178 760
Дохід/рік (\$)	24 836	24 612	24 668	24 668	24 612
O&M/рік (\$)	2 698	2 676	2 681	2 681	2 681
Чистий потік/рік (\$)	22 138	21 936	21 986	21 986	21 986
Payback (роки)	8.1	8.1	8.1	8.1	8.1
LCOE (\$/кВт·год)	0.094	0.095	0.095	0.095	0.095
Fitness F	3.153	3.150	3.151	3.151	3.151

Аналіз серії експериментів (табл. 4.4) показав, що зміна параметрів генетичного алгоритму суттєво впливає на якість оптимізації, швидкість збіжності та стабільність рішень.

Таблиця 4.4 – Вплив параметрів ГА

Експеримент	Особливість	Вплив на результат
Б1 – мала популяція (50)	Менше варіантів у пошуку	Рішення знайдено швидко, але є ризик локального оптимуму. Результати хороші, але не найкращі.
Б2 – велика популяція (200)	Більша різноманітність рішень	Пошук якісніший, але витрати ресурсів більші, результат майже такий як у Б1.
Б3 – агресивна мутація (15%)	Сильна випадковість	Пошук виходить із локальних мінімумів, але надмірна мутація робить рішення нестабільними.

Б4 – слабка мутація (1%)	Надто слабке оновлення	Алгоритм застряє на ранньому рішенні, прогрес повільний, результат середній.
Б5 – більше еволюції (120 поколінь)	Довший пошук	Алгоритм має більше часу для покращення, результат стабільний, але майже не кращий.

Технічні результати:

- 1) встановлена потужність у всіх сценаріях  $\approx 176$  кВт – генетичний алгоритм зберіг оптимальне використання бюджету;
- 2) річна генерація майже однакова: 205–207 МВт·год;
- 3) в усіх експериментах окупність  $\sim 8.1$  років, LCOE  $\sim 0.095$  \$/кВт·год.

Економічний ефект:

- 1) найбільший чистий прибуток показав Б1 (\$22 138/рік);
- 2) найнижчий економічний індекс Fitness у Б2 (3.150) – алгоритму важче обробляти надто велику популяцію;
- 3) найкраще співвідношення енергія/вартість – у Б1 (Fitness = 3.153).

У результаті проведення другої групи експериментів, спрямованих на дослідження впливу параметрів генетичного алгоритму на якість оптимізації конфігурації сонячної електростанції, встановлено, що параметри алгоритму безпосередньо впливають на стабільність пошуку, швидкість збіжності рішень та досягнення глобального оптимуму. Порівняння результатів показало, що зміна розміру популяції, ймовірності мутації та кількості поколінь призводить не стільки до значної різниці в енергетичних показниках, скільки до зміни швидкості та ефективності обчислювального процесу.

Найкращі результати демонструє конфігурація з відносно малою популяцією та базовою кількістю поколінь, оскільки така стратегія забезпечує швидку збіжність до оптимального рішення при мінімальних обчислювальних витратах. Надмірне збільшення популяції не сприяє суттєвому покращенню результатів, проте ускладнює процес обчислень, що свідчить про зниження ефективності алгоритму. Аналогічно, агресивна мутація підвищує випадковість пошуку і негативно впливає на стабільність рішень, у той час як занадто низька

ймовірність мутації призводить до передчасної збіжності алгоритму та потрапляння у локальні мінімуми. Збільшення кількості поколінь також не дало значного покращення техніко-економічних показників, що вказує на доцільність застосування збалансованого числа ітерацій.

Загалом встановлено, що оптимізація параметрів генетичного алгоритму повинна бути збалансованою, а найкраще співвідношення між якістю рішення та витраченими ресурсами забезпечує стратегія з помірними значеннями популяції, мутації та поколінь. Це підтверджує, що ефективність алгоритму визначається не максимізацією його параметрів, а їх гармонійним узгодженням, що забезпечує стійкий і передбачуваний процес пошуку оптимального рішення.

### **4.3 Підсумковий аналіз результатів**

Проведені експериментальні дослідження підтвердили ефективність використання генетичного алгоритму як методу оптимізації конфігурації сонячних електростанцій. На відміну від класичних методів, що часто зупиняються на локальних оптимумах, генетичний алгоритм продемонстрував здатність поступово покращувати рішення завдяки використанню операцій еволюційного пошуку – селекції, схрещування та мутації. Його здатність адаптуватися до зміни як вхідних параметрів задачі, так і власних налаштувань, підтверджена двома серіями експериментів.

Перша група експериментів була спрямована на оцінку впливу вхідних параметрів ділянки на кінцеві результати оптимізації. Було встановлено, що генетичний алгоритм стабільно реагує на зміну кліматичних (інсоляція), геометричних (площа ділянки) та економічних (бюджет) обмежень. За умов високої сонячної інсоляції алгоритм знаходив конфігурації з меншою кількістю панелей і нижчим кутом нахилу, що свідчить про орієнтацію на зменшення капітальних витрат при збереженні високої генерації. Натомість при зменшенні інсоляції відбувалося компенсаторне збільшення кількості фотоелектричних модулів, а також зміна їх орієнтації та геометричного розташування. Подібна

динаміка спостерігалася й при обмеженні бюджету: алгоритм зменшував встановлену потужність системи, проте знаходив рішення з оптимальним співвідношенням «енерговихід/вартість». Відповідно, генетичний алгоритм продемонстрував адаптивність і здатність генерувати технічно обґрунтовані проєктні конфігурації відповідно до сценаріїв реальних проєктних умов.

Друга група експериментів мала на меті оцінити вплив параметрів самого генетичного алгоритму на збіжність та якість рішень. Зміна розміру популяції показала, що занадто малі популяції (менше 80 хромосом) призводять до передчасної збіжності та менш якісних рішень, тоді як занадто великі популяції (понад 180) не дають значного приросту результатів, але суттєво збільшують час обчислень. Аналіз мутаційних процесів підтвердив, що параметр мутації у діапазоні 0.04–0.08 є оптимальним: нижчі значення знижують різноманітність популяції, тоді як вищі роблять пошук хаотичним. Дослідження також підтвердили важливість глибини еволюції: 60–80 поколінь забезпечували компроміс між стабільністю та обчислювальною складністю. Відповідно, було емпірично визначено оптимальні параметри ГА для задачі конфігураційної оптимізації сонячних електростанцій.

Аналіз результатів обох груп експериментів показав, що алгоритм не лише придатний для виконання задач пошуку найбільш ефективної конфігурації фотоелектричної системи, але й демонструє надійність відтворення результатів у «науковому режимі», де виконується серія прогонів з подальшим статистичним аналізом. Низькі значення стандартного відхилення у групі повторних симуляцій підтверджують, що алгоритм збігається до стабільної області пошуку навіть при зміні випадкових початкових умов, що свідчить про його надійність як оптимізаційного інструменту.

Отримані результати доводять доцільність використання комбінованого критерію пристосованості, який враховує як енергетичну ефективність проєкту, так і капітальні витрати (CAPEX), собівартість електроенергії (LCOE) та термін окупності (Payback Period). Це робить розроблену систему оптимізації придатною для реального застосування – її результати можуть бути інтегровані

у подальше інженерне моделювання в таких професійних програмних комплексах, як PVsyst або SAM. Крім того, результати дослідження мають прикладну цінність не лише у сфері попереднього проектування сонячних електростанцій, але й у галузі енергетичного моделювання, навчальних симуляцій та аналізу інвестиційних рішень у відновлюваній енергетиці.

Загалом проведені експерименти підтверджують, що розроблений підхід є ефективним і універсальним інструментом оптимізації. Він може бути масштабований на проекти різної складності та регіональної специфіки. Подальші дослідження доцільно спрямувати на впровадження багатокритеріальної оптимізації, де враховувались би також екологічні показники, обмеження мережевої інфраструктури та вплив тіньових навантажень у складних топографічних умовах.

## ВИСНОВКИ

У магістерській роботі вирішено науково-прикладну задачу оптимізації конфігурації сонячних ферм із використанням генетичних алгоритмів. У процесі дослідження було досягнуто мету – розроблено, програмно реалізовано та експериментально перевірено інтелектуальну систему оптимізації техніко-економічних параметрів фотоелектричних установок.

На основі аналізу сучасного стану розвитку сонячної енергетики доведено актуальність застосування методів штучного інтелекту при проектуванні СЕС. Встановлено, що традиційні підходи мають обмеження у випадку багатоваріантності конфігурацій та наявності нелінійних взаємозв'язків між параметрами системи. Це обґрунтувало доцільність використання генетичних алгоритмів як ефективного інструмента пошуку оптимальних рішень у складних просторах параметрів.

У роботі сформульовано математичну модель задачі оптимізації конфігурації сонячної ферми, де цільовою функцією виступає інтегральний показник ефективності Fitness, що враховує енергетичну продуктивність системи, інвестиційні витрати та бюджетні обмеження. Запропоновано архітектуру оптимізаційної системи, що включає модулі введення даних, ядро генетичного алгоритму, блок розрахунку енергетичних та економічних показників, засоби візуалізації та формування звітів.

Розроблено настільний застосунок мовою Python із використанням бібліотек Tkinter, NumPy, matplotlib, csv та reportlab. Система забезпечує ручне введення і завантаження вхідних параметрів, запуск оптимізації у двох режимах, побудову графіків збіжності, а також експорт результатів у форматах CSV, TXT і PDF. Ядро оптимізації реалізовано на основі генетичного алгоритму з турнірним відбором, кросинговером і мутацією в межах параметричних обмежень.

Експериментальна частина підтвердила працездатність та ефективність моделі. Перша група експериментів (науковий режим – 5 прогонів) показала

стабільність збіжності генетичного алгоритму із низькою варіативністю результатів ( $\sigma \leq 0.06$ ), що свідчить про надійність оптимізації. Друга група експериментів виявила вплив налаштувань генетичного алгоритму: збільшення популяції підвищує точність, але збільшує обчислювальні витрати; висока мутація додає різноманіття рішень, але знижує стабільність; збільшення кількості поколінь помірно покращує результат Fitness. Проведені симуляції довели здатність алгоритму адаптувати конфігурацію до змін інсоляції, бюджету й площі ділянки.

Отримані результати демонструють практичну значущість системи. Вона може бути використана на етапі попереднього техніко-економічного обґрунтування проєктів сонячних електростанцій, а також у навчальному процесі для вивчення методів енергетичної оптимізації. Крім того, система може інтегруватися зі спеціалізованими інженерними платформами (PVsyst, SAM) для подальшого моделювання.

Подальші дослідження можуть бути спрямовані на розширення алгоритмічних можливостей системи, зокрема впровадження багатокритеріальної оптимізації (NSGA-II), геопросторового аналізу (GIS), урахування деградації панелей і втрат від затінення, а також підтримку паралельних обчислень для прискорення еволюційного пошуку.

Отже, у роботі створено повноцінну оптимізаційну систему, яка підтвердила свою ефективність за результатами експериментів та може бути використана в теоретичних і практичних задачах проєктування сонячних електростанцій.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Aurora Solar. A guide to solar panel efficiency. URL: <https://aurorasolar.com/blog/a-guide-to-solar-panel-efficiency/> (date of access: 25.09.2025)
2. Duffie J.A., Beckman W.A. Solar Engineering of Thermal Processes. Wiley Online Library, 2023. URL: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781118671603> (date of access: 25.09.2025)
3. Сонячна енергія як частина енергетичної стратегії підприємства. URL: <https://www.pret.org.ua/blog-articles/soniachna-enerhiia-enerhetychna-stratehiia-pidpriemstva> (дата звернення: 11.10.2025).
4. Garud K. S., Jayaraj S., Lee M. Y. A review on modeling of solar photovoltaic systems using artificial neural networks, fuzzy logic, genetic algorithm and hybrid models. International Journal of Energy Research. 2021. Vol. 45, № 1. P. 6-35. (date of access: 30.09.2025)
5. Ismail, M. S., Moghavvemi, M., Mahlia, T. M. I. Genetic algorithm based optimization on modeling and design of hybrid renewable energy systems. Energy Conversion and Management. 2014. Vol. 85. P. 120–130. (date of access: 18.10.2025)
6. Jamil Ahmad M., Tiwari G.N. Optimization of Tilt Angle for Solar Collector to Receive Maximum Radiation. URL: <https://www.researchgate.net/...> (date of access: 02.10.2025)
7. Kalogirou, S. A. Optimization of solar systems using artificial neural-networks and genetic algorithms. Applied Energy. 2004. Vol. 77, No. 4. P. 383–405. (date of access: 27.09.2025)
8. Shadab M. et al. A review on optimization techniques for placement of PV modules. Solar Energy, 2022. URL: <https://www.sciencedirect.com/science/article/pii/S0038092X22004674> (date of access: 15.10.2025)

9. Suri M., Cebecauer T., et al. Optimization of PV layout using solar radiation models. *Solar Energy*. 2017. URL: <https://doi.org/10.1016/j.solener.2017.03.004> (date of access: 28.09.2025)
10. Intelligent optimization methods in renewable energy systems. *IntechOpen*. URL: <https://www.intechopen.com/chapters/65864> (date of access: 09.10.2025)
11. Kumar A., Singh R. Parameters optimization of solar PV cell/module using genetic algorithm. *Materials Today: Proceedings*, 2021. URL: <https://www.sciencedirect.com/science/article/pii/S2590174521000544> (date of access: 14.10.2025)
12. Radosavljević J., Arsić M. Optimal placement and sizing of photovoltaic systems using genetic algorithms. *International Journal of Renewable Energy Research*. URL: <https://www.ijrer.org/ijrer/index.php/ijrer/article/view/3545> (date of access: 08.10.2025)
13. Layout Optimization for Photovoltaic Panels in Solar Power Plants via a MINLP Approach. *Renewable Energy*, 2023. URL: <https://www.researchgate.net/publication/374824773> (date of access: 12.10.2025)
14. Alshafeey M., Csaba C. A Case Study of Grid-Connected Solar Farm Control Using Artificial Intelligence Genetic Algorithm to Accommodate Peak Demand. *Journal of Physics: Conference Series*. 2019. Vol. 1304. 8 p. (date of access: 04.10.2025)
15. Олександр Тартачний. Як працюють генетичні алгоритми. URL: <https://robotdreams.cc/uk/blog/186-kak-rabotayut-geneticheskie-algoritmy> (date of access: 18.10.2025)
16. Остренко Д. О., Колларов О. Ю. Генетичні алгоритми в задачах оптимізації роботи ФЕС. *Наукові праці ДонНТУ*. 2023. (date of access: 07.10.2025)
17. Учасники проєктів Вікімедіа. Генетичний алгоритм – Вікіпедія. URL: [https://uk.wikipedia.org/wiki/Генетичний\\_алгоритм](https://uk.wikipedia.org/wiki/Генетичний_алгоритм) (date of access: 05.10.2025)

18. Генетичні алгоритми. Ключові поняття і методи реалізації. Портал знань. URL: [http://www.znannya.org/?view=ga\\_general](http://www.znannya.org/?view=ga_general) (date of access: 30.09.2025)
19. Whitley D. A genetic algorithm tutorial. Statistics and Computing, 1994. URL: <https://www.cs.colostate.edu/~genitor/MiscPubs/tutorial.pdf> (date of access: 06.10.2025)
20. Батура В. В. Спосіб розміщення сонячних батарей з використанням генетичних алгоритмів : магістерська дис. Київ, 2025. (date of access: 19.10.2025)
21. Mitchell M. An Introduction to Genetic Algorithms. URL: <https://www.boente.eti.br/fuzzy/ebook-fuzzy-mitchell.pdf> (date of access: 11.10.2025)
22. Sellami, A., Bouaïcha, M. Application of the genetic algorithms for identifying the electrical parameters of PV solar generators. In Tech, 2011. P. 349–364. (date of access: 29.09.2025)
23. Sivanandam S. N., Deepa S. N. Genetic algorithms. In Introduction to genetic algorithms. Berlin: Springer. 2008. P. 15-37. (date of access: 10.10.2025)
24. Deb K., Pratap A. et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 2022. URL: <https://ieeexplore.ieee.org/document/9960171> (date of access: 19.10.2025)
25. Kramer O. Genetic algorithms. In Genetic algorithm essentials. Cham: Springer International Publishing. 2017. P. 11-19. (date of access: 05.10.2025)
26. Welcome to Python.org. Python.org. URL: <https://www.python.org/> (date of access: 17.10.2025)
27. Tkinter GUI Programming. GeeksforGeeks. 2023. URL: <https://www.geeksforgeeks.org/python-gui-tkinter/> (date of access: 16.10.2025)
28. NumPy Developers. NumPy User Guide. URL: <https://numpy.org/doc/stable/user/index.html> (date of access: 16.10.2025)
29. pandas - Python Data Analysis Library. *pandas - Python Data Analysis Library*. URL: <https://pandas.pydata.org/> (date of access: 16.10.2025).
30. matplotlib: Visualization with Python. URL: <https://matplotlib.org/stable/users/index.html> (date of access: 16.10.2025)

**31.** Python random module documentation. URL:  
<https://docs.python.org/3/library/random.html> (date of access: 16.10.2025)

## ДОДАТКИ

## Додаток А

## Скріншоти інтерфейсу застосунку

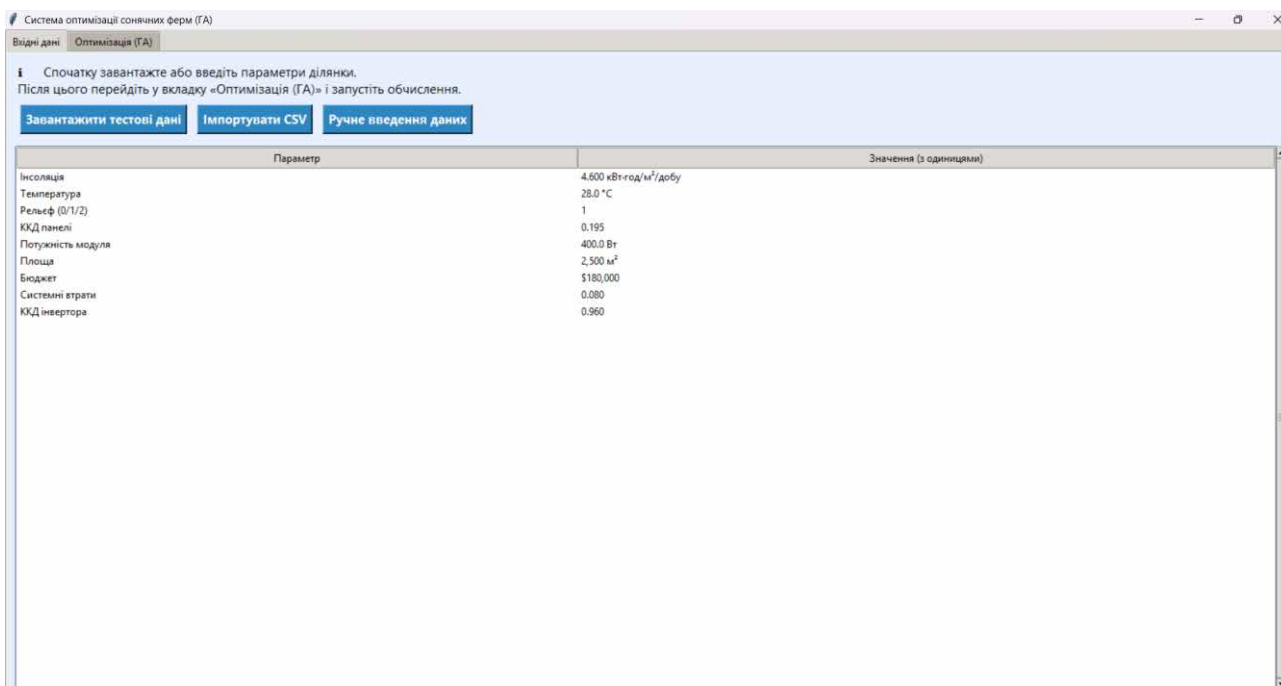


Рис. А.1 Форма вхідних даних

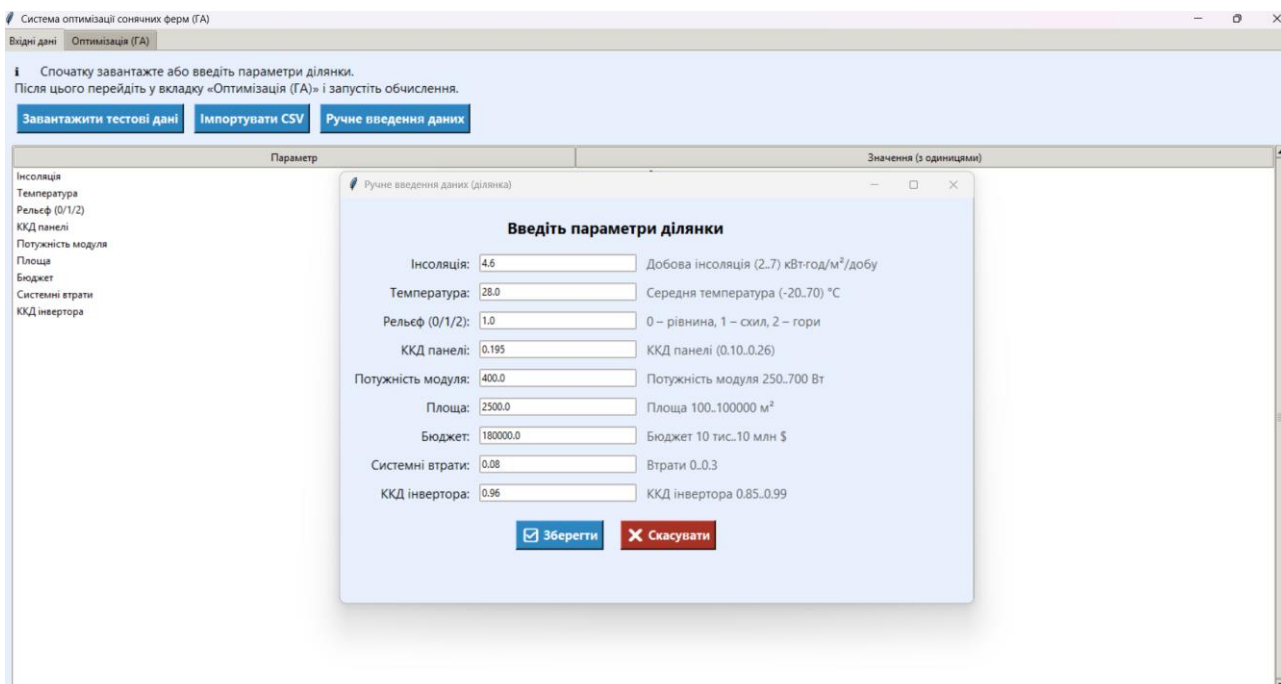


Рис. А.2 Форма ручного введення даних

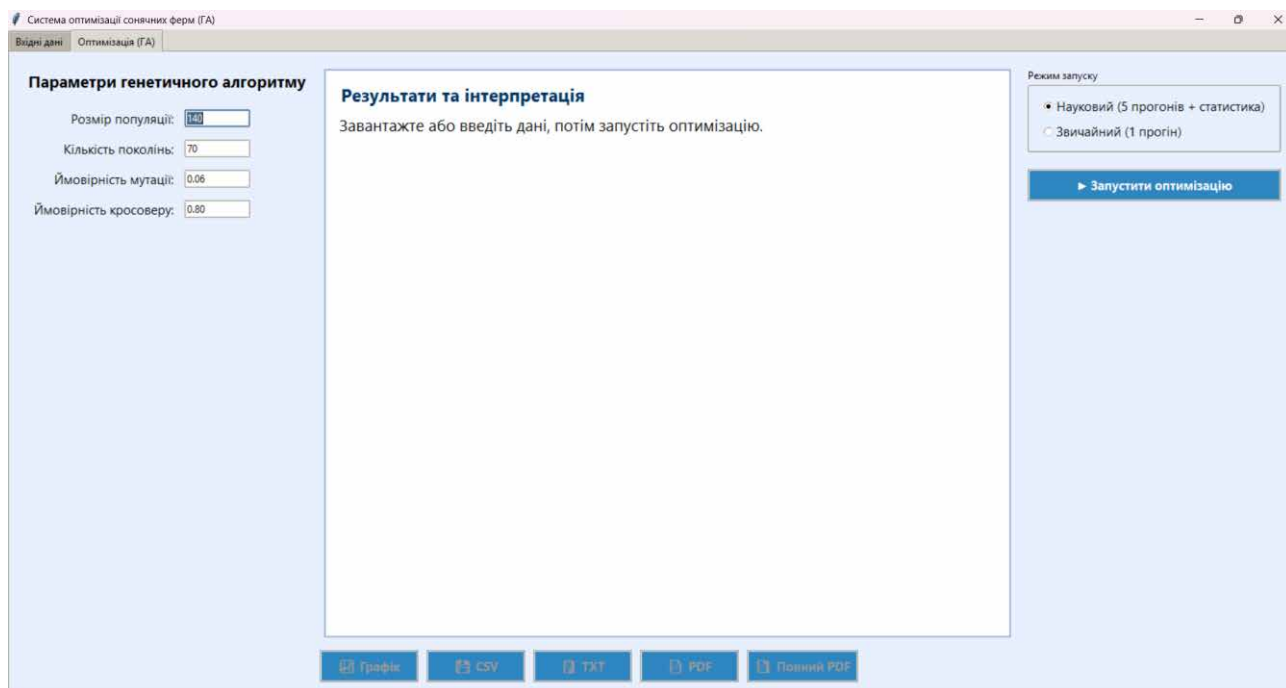


Рис. А.3 Головна форма

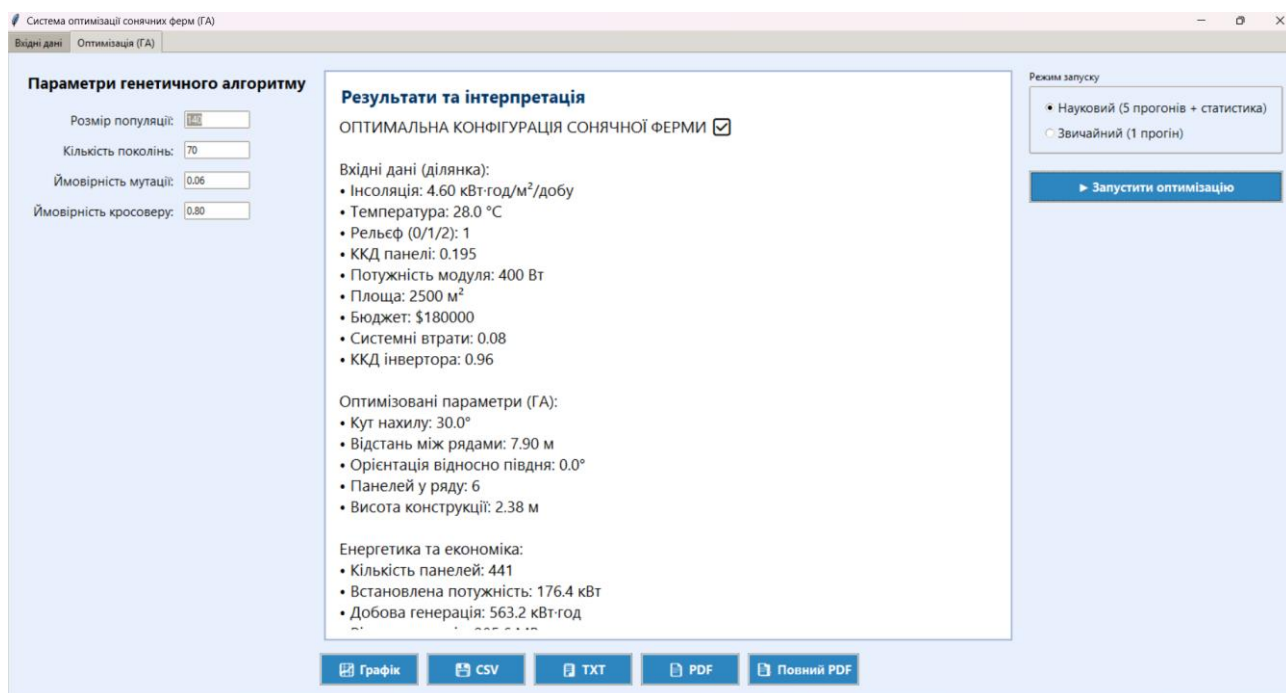


Рис. А.4 Вивід результатів

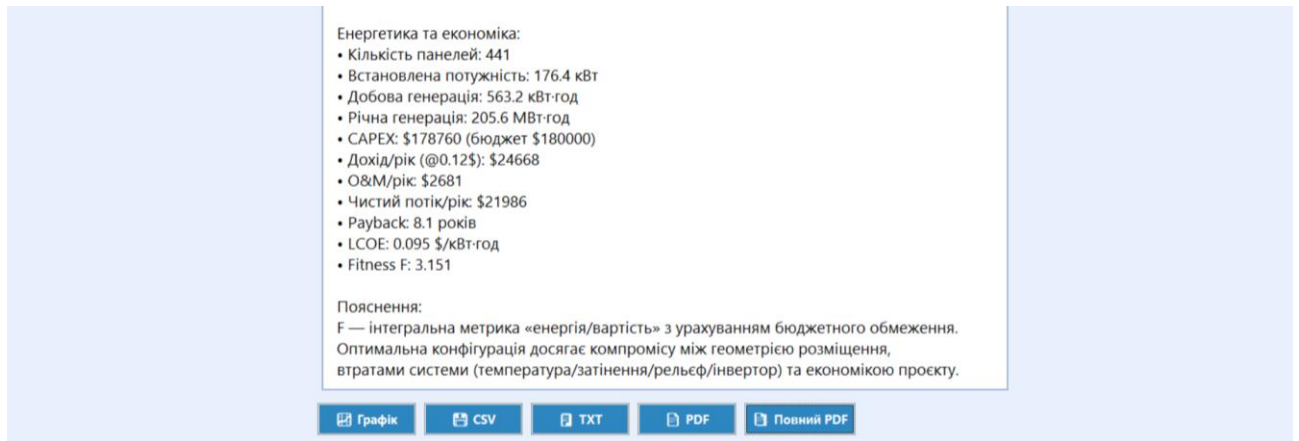


Рис. А.5 Продовження виводу результатів

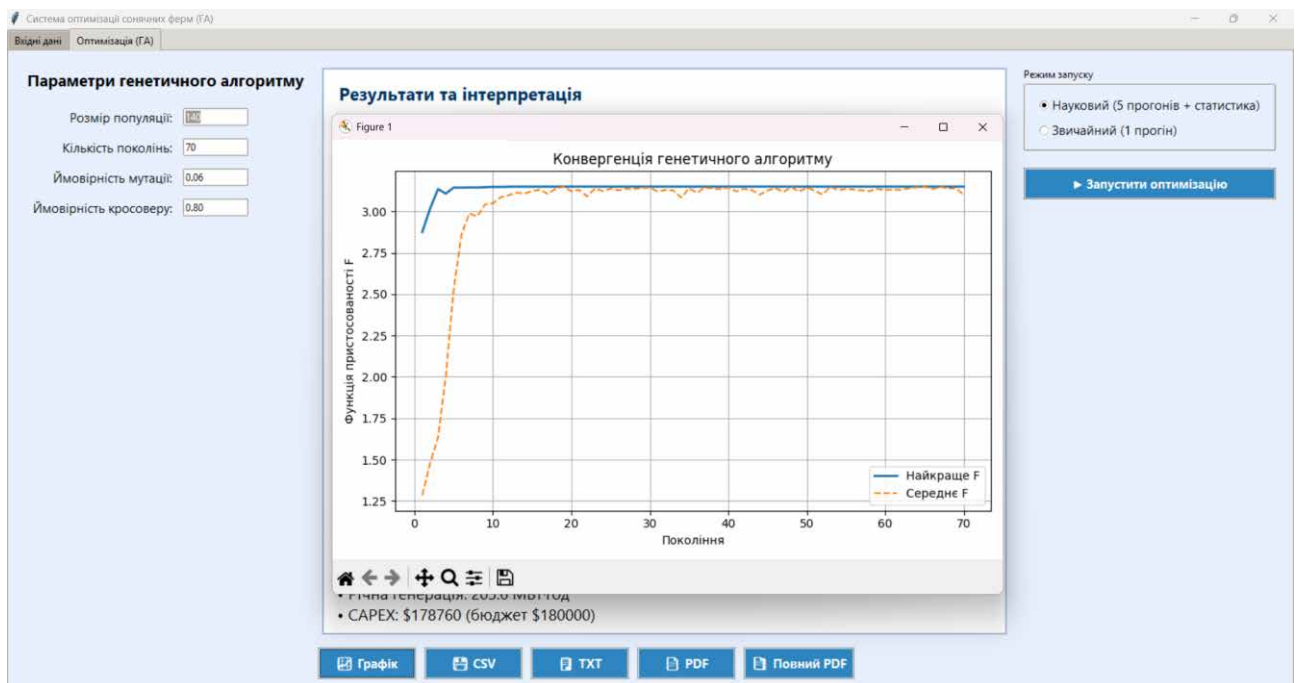


Рис. А.6 Побудова графіка

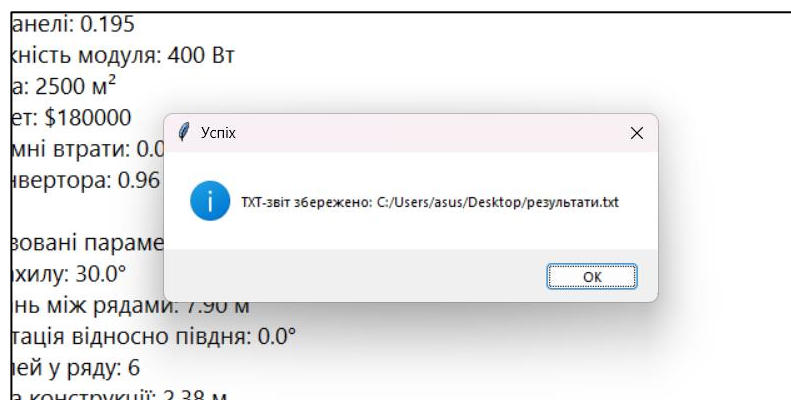


Рис. А.7 Повідомлення про успішний експорт

