

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

Кваліфікаційна наукова праця на
правах рукопису

МАРТИНЮК ВАДИМ ІГОРОВИЧ

УДК 621.314

ДИСЕРТАЦІЯ
СИСТЕМА ДИНАМІЧНОЇ КОМУТАЦІЇ ТОПОЛОГІЇ СОНЯЧНИХ
ПАНЕЛЕЙ З ВРАХУВАННЯМ ОСОБЛИВОСТЕЙ ХМАРНОГО ПОКРИВУ

171 – Електроніка

17 – Електроніка та телекомунікації

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей,
результатів і текстів інших авторів мають посилання на відповідне джерело

_____ Мартинюк В. І.

Науковий керівник – Жуйков Валерій Якович
доктор технічних наук, професор

Київ-2024

АНОТАЦІЯ

Мартинюк В.І. Система динамічної комутації топології сонячних панелей з врахуванням особливостей хмарного покриву. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 171 Електроніка – Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» МОН України, Київ, 2024.

Дисертаційна робота присвячена створенню системи динамічної комутації топології сонячних панелей на основі створення їх уточненої вольт-амперної характеристики та моделювання хмарного покриву, що дозволяє визначити топологію комутації сонячних панелей зі максимізацією вихідної потужності.

У **першому розділі** проведено аналіз наявних методів оцінки потенціалу сонячної енергії та її основних параметрів. Виконано дослідження сучасного стану розвитку напівпровідникових перетворювачів сонячної енергії, особливостей їх функціонування в умовах часткового затінення, а також способів зменшення його негативного впливу. Проаналізовано стан розвитку еквівалентних схем заміщення, математичних моделей та топологій. Детально розглянуто математичні моделі хмарного покриву. Проведено класифікацію та порівняльний аналіз методів генерації зображень хмарного покриву на основі використання шумів.

У **другому розділі** наведено математичні співвідношення, які описують удосконалену модель сонячної панелі, що була отримана на основі аналізу еквівалентних схем заміщення.

Запропоновано два способи визначення параметрів моделі, а саме спосіб перебору коефіцієнтів ідеальності та опорів. Розглянуто особливості алгоритмів запропонованих способів та наведено рекомендації щодо використання вагових функцій, що дозволяє отримати більш точну оцінку

значень параметрів моделі. Запропоноване використання методу простої ітерації та методу релаксації для вирішення трансцендентних рівнянь і забезпечення збіжності чисельних методів. Показано, що найменші значення похибок апроксимації забезпечує Гаусівська віконна функція. Наведено розраховані значення параметрів моделі для сонячних панелей KC200GT та ST40.

Проведено порівняльний аналіз запропонованих способів визначення параметрів моделі з сучасними методами, представленими в літературі. Порівняння здійснювалося для сонячних панелей KC200GT та ST40 і встановлено, що запропоновані способи визначення параметрів забезпечують кращу апроксимацію вольт-амперних характеристик.

Проведено визначення залежності параметрів удосконаленої моделі сонячної панелі від рівня опроміненості та температури з наведенням відповідних значень похибок.

Третій розділ присвячений створенню математичної моделі хмарного покриття. Описано особливості запропонованої моделі хмарного покриття, заснованої на аналізі спектральних характеристик зображень хмар різного виду. Показано, що на основі даної моделі та вихідного зображення хмарного покриття здійснюється генерація зображення хмар, які відтворюють основні особливості вихідного.

Запропоноване використання вагової функції для спрощення аналізу вихідних зображень та зменшення впливу явища витікання спектру. Розглянуто спосіб визначення параметрів створеної моделі та досліджено особливості фазово-частотної характеристики і способу її генерації. Для апроксимації амплітудно-частотної характеристики використано методи регресійного аналізу та головних компонент. Наведено приклад використання з визначенням значень параметрів моделі для досліджуваного зображення хмарного покриття.

Проведено порівняльний аналіз запропонованої моделі з алгоритмом спектрального синтезу. Визначено похибку апроксимації, які забезпечують

розглянуті моделі. Проведено порівняння значень параметрів моделі зі значеннями, отриманими методом перебору, що вказує на ефективність розробленої моделі.

Четвертий розділ присвячений комп'ютерному моделюванню вольт-амперних характеристик та характеристик вихідної потужності масиву сонячних панелей на основі панелі KC200GT в програмному середовищі Matlab Simulink.

Розроблено та описано схемні моделі некомутованого та комутованого масивів сонячних панелей та наведено особливості їх використання з наведенням необхідних розрахунків параметрів.

Змодельовані робота системи динамічної комутації та перемикання топології масиву сонячних панелей під час функціонування для забезпечення максимальної вихідної потужності в умовах часткового затінення. Отримано вихідні характеристики для досліджуваних масивів на основі значень параметрів моделей сонячної панелі та хмарного покриву, які були отримані в попередніх розділах.

Проведено порівняльний аналіз даної системи з некомутованим масивом з метою оцінки ефективності запропонованого алгоритму комутації, що підтвердило теоретичні розрахунки та правочинність розробленої системи динамічної комутації і запропонованих математичних моделей сонячних панелей та хмарного покриву.

Ключові слова: сонячні панелі, еквівалентна схема, вольт-амперна характеристика, часткове затінення, чисельні методи, алгоритми апроксимації, апроксимація, метод простої ітерації, метод релаксації, спектральний синтез, моделювання хмарного покриву, регресійний аналіз, метод головних компонент, аналіз частотних характеристик.

Список публікацій здобувача.

1. В.І. Мартинюк і К.С. Клен, «Лінійні функції на базі функцій Уолша», Мікросистеми, Електроніка та Акустика, т. 24, вип. 1, с. 29-34, Лютий 2019, doi:10.20535/2523-4455.2019.24.1.160960.

2. K. Klen, V. Martynyuk, and M. Yaremenko, «Prediction of the wind speed change function by linear regression method», *Computational Problems of Electrical Engineering*, vol. 9, no. 2, pp. 28–33, Nov 2019, doi: 10.23939/jcpee2019.02.028.
3. В.І. Мартинюк, К.С. Клен, і В. Я. Жуйков, «Визначення параметрів схем заміщення сонячних панелей за експериментальними даними», *Мікросистеми, Електроніка та Акустика*, т. 26, вип. 2, с. 237316-1, Серпень 2021, doi:10.20535/2523-4455.me.237316.
4. V. I. Martyniuk and V. Y. Zhuikov, «Generation of Anisotropic Cloud Cover», *Microsystems, Electronics and Acoustics*, vol. 28, no. 3, pp. 291803.1–291803.10, Dec 2023, doi: 10.20535/2523-4455.me.291803.
5. K. Klen, V. Martynyuk, and M. Yaremenko, «Prediction of primary energy flow considering its stochastic nature», *Maszyny Elektryczne. – Zeszyty Problemowe*, no. 1(125), pp. 107–110, 2021.
6. V. Martyniuk and K. Klen, «Estimation of Parameters of Photovoltaic Modules Based on Ideality Factor Variation», 2022 IEEE 41st International Conference on Electronics and Nanotechnology (ELNANO), IEEE, Oct. 2022, pp. 692-696, doi: 10.1109/ELNANO54667.2022.9927063.

ABSTRACT

Martyniuk V.I. Dynamic commutation system of the topology of solar panels taking into account the characteristics of the cloud cover. – Qualifying scientific work on manuscript rights.

Dissertation for the degree of Doctor of Philosophy in specialty 171 Electronics – National Technical University of Ukraine "Ihor Sikorskyi Kyiv Polytechnic Institute" MES of Ukraine, Kyiv, 2024.

The dissertation work is devoted to creating optimal conditions for the operation of an array of solar power modules and simultaneously maximizing energy production under existing weather conditions by creating a system of dynamic commutation of the topology of solar panels, which determines the optimal structure of an array of solar panels taking into account the parameters of cloud cover.

In the **first chapter**, an analysis of the available methods for assessing the potential of solar energy and its main parameters is carried out. A study of the current state of development of semiconductor solar energy converters, features of their operation in conditions of partial shading, as well as ways of reducing its negative impact is carried out. The state of development of equivalent circuits, mathematical models and topologies is analyzed. Mathematical models of cloud cover are analyzed in details. The classification and comparative analysis of cloud cover image generation methods based on the use of noise is carried out.

In the **second chapter**, mathematical relations are given that describe the improved model of the solar panel, which was obtained on the basis of the analysis of equivalent substitution schemes.

Two methods of determining the parameters of the model are proposed, namely, the method of selecting ideality coefficients and resistances. The features of the algorithms of the proposed methods are considered and recommendations are given for the use of weight functions, which allows to obtain a more accurate estimate of the values of the model parameters. The use of the simple iteration method and the relaxation method for solving transcendental equations and ensuring

the convergence of numerical methods is proposed. It is shown that the smallest approximation error values are provided by the Gaussian window function. Calculated values of model parameters for KC200GT and ST40 solar panels are given.

A comparative analysis of the proposed methods of determining model parameters with modern methods presented in the literature is carried out. The comparison is made for KC200GT and ST40 solar panels and it is established that the proposed methods of determining the parameters provide a better approximation of the I-V curves.

The dependence of the parameters of the improved model of the solar panel on the level of irradiance and temperature is determined, with the corresponding error values.

The **third chapter** is devoted to the creation of a mathematical model of cloud cover. The features of the proposed cloud cover model, based on the analysis of the spectral characteristics of the image of various types of clouds, are described. It is shown that proposed model based on the original image of the cloud cover generates the cloud image, which reproduces the main features of the original image.

The use of a weight function is proposed to simplify the analysis of the original images and reduce the influence of the phenomenon of spectrum leakage. The method of determining the parameters of the created model is considered, and the peculiarities of the phase-frequency characteristic and the method of its generation are investigated. The methods of regression analysis and principal component analysis are used to approximate the amplitude-frequency characteristic. An example of use is given to determine the values of the model parameters for the cloud cover image under study.

A comparative analysis of the proposed model with the spectral synthesis algorithm is carried out. The approximation error that provides the presented models is determined. The values of the model parameters are compared with the values obtained by the brute force method, which indicates the effectiveness of the developed model.

The **fourth chapter** is devoted to the computer modeling of the I-V and output power characteristics of the solar panel array for the KC200GT panel in the Matlab Simulink software environment.

Schematic models of non-switched and switched arrays of solar panels are developed and described, and the features of their use are given with the necessary parameter calculations.

The operation of the dynamic switching system and switching of the topology of the array of solar panels during operation to ensure the maximum output power in conditions of partial shading are simulated. The initial characteristics for the studied arrays were obtained based on the parameter values of the solar panel and cloud cover models, which were obtained in the previous sections.

A comparative analysis of this system with a non-switched array is carried out in order to evaluate the efficiency of the proposed switching algorithm, which confirmed the theoretical calculations and feasibility of the developed dynamic switching system and the proposed mathematical models of solar panels and cloud cover.

Keywords: solar cells, equivalent circuit, current-voltage characteristic, partial shading, approximation algorithms, method of simple iteration, relaxation method, numerical methods, spectral synthesis, cloud cover modeling, regression analysis, principal component analysis, frequency characteristic analysis.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	11
ВСТУП	12
РОЗДІЛ 1. СИСТЕМИ ЕЛЕКТРОЖИВЛЕННЯ З ВИКОРИСТАННЯМ СОНЯЧНИХ ПАНЕЛЕЙ	16
1.1. Актуальність використання відновлюваних джерел енергії.....	16
1.2. Потенціал сонячної енергії.....	19
1.3. Параметри та характеристики сонячного елементу	30
1.4. Ефект часткового затінення.....	34
1.5. Математична модель сонячного елементу.....	46
1.6. Математична модель хмарного покриву	51
Висновки до розділу	68
РОЗДІЛ 2. ВИЗНАЧЕННЯ ПАРАМЕТРІВ ЕКВІВАЛЕНТНОЇ СХЕМИ ЗАМІЩЕННЯ	70
2.1. Визначення параметрів рівняння вольт-амперних характеристик.....	70
2.2. Спосіб перебору значень коефіцієнтів ідеальності	75
2.3. Спосіб перебору значень опорів	89
2.4. Порівняльний аналіз способів визначення параметрів	93
2.5. Залежність ВАХ від параметрів опроміненості і температури	101
Висновки до розділу	106
РОЗДІЛ 3. МОДЕЛЬ ХМАРНОГО ПОКРИВУ	108
3.1. Визначення параметрів моделі хмарного покриву	108
3.2. Порівняльний аналіз	131
Висновки до розділу	135
РОЗДІЛ 4. КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ	136
4.1. Моделювання масиву сонячних панелей.....	136
4.2. Моделювання комутованого масиву сонячних панелей.....	143
4.3. Порівняльний аналіз	148
Висновки до розділу	154
ЗАГАЛЬНІ ВИСНОВКИ.....	155

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	157
ДОДАТОК А. Рівняння математичної моделі сонячної панелі	171
ДОДАТОК Б. Програмна реалізація ВАХ сонячної панелі	175
ДОДАТОК В. Програмна реалізація моделі хмарного покриву	186
ДОДАТОК Г. Програмна реалізація комп'ютерного моделювання	205
ДОДАТОК Д. Акт впровадження.....	213
ДОДАТОК Е. Заявка на патент на винахід	214

ПЕРЕЛІК СКОРОЧЕНЬ

ВДЕ – відновлювані джерела енергії;
АМ0 – air mass zero spectrum;
АМ1.5 – air mass zero spectrum;
СЕ – сонячний елемент;
СП – сонячна панель;
ВАХ – вольт-амперна характеристика;
ХХ – холостий хід;
КЗ – коротке замикання;
ККД – коефіцієнт корисної дії;
СЕС – сонячна електростанція;
ТМП – точка максимальної потужності;
ВТМП – відслідковування точки максимальної потужності;
СПП – спектральна густина потужності;
СС – спектральний синтез;
ШПФ – швидке перетворення Фур'є;
МОК – метод основних компонент.

ВСТУП

Актуальність теми. Стратегічною метою розвитку електроенергетики України є забезпечення сталого та ефективного розвитку енергетичного сектору, що сприяє економічному зростанню країни, енергетичній безпеці, екологічній стійкості та технологічному прогресу. Ця мета визначена в більш широкому контексті національної енергетичної політики [1] та міжнародних зобов'язань [2], які є частиною нещодавно прийнятої Енергетичної стратегії України 2050 [3], [4].

Розвиток сектору електроенергетики передбачає комплексний підхід, який базується на створенні нових розподілених генеруючих потужностей на основі використання відновлюваних джерел енергії (ВДЕ) [4], а також на впровадженні заходів з енергоефективності у сфері виробництва, передачі та споживання електроенергії.

Роль ВДЕ, серед яких особливе місце посідають сонячні електростанції (СЕС), у стратегічному розвитку електроенергетики України є надзвичайно важливою. Проте використання електростанцій на базі ВДЕ характеризується низкою суттєвих недоліків, серед яких найбільш істотними є залежність потужності від погодних умов і низька щільність потоку енергії первинного носія. Інтеграція ВДЕ в енергетичний сектор нівелює ці недоліки за умови одночасного забезпечення оптимальних умов експлуатації та максимізації видобутку енергії за наявних погодних умов.

Значний внесок у дослідження забезпечення ефективної роботи енергетичної системи зі сонячними електростанціями зробили такі вітчизняні вчені як: акад. Кириленко О.В., член-кор. Кудря С.О., проф. Блінов І.В., проф. Шавьолкін О.О. с.н.с. Суржик Т.М., с.н.с. Матях С.В., с.н.с. Кузнєцов М.П., доц. Будько В.І., проф. Ямненко Ю.С., проф. Ромашко В.Я., проф. Вербицький Є.В. Та іноземні вчені: проф. Мартін Грін, проф. Міхаель Гретцель та проф. Генрі Снейт.

При побудові систем генерації з використанням відновлюваних джерел енергії необхідно проводити аналіз ефективності роботи з урахуванням найбільш можливих факторів впливу на системи електроживлення на основі сонячних панелей (СП) для максимізації відбору первинної енергії.

Тому розробка система динамічної комутації топології сонячних панелей з врахуванням особливостей хмарного покриття є актуальною.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконана в Національному технічному університеті України «Київський політехнічний інститут імені Ігоря Сікорського» на кафедрі електронних пристроїв та систем в рамках держбюджетної прикладної науково-дослідної роботи № 2314-п «Система енергозабезпечення високочастотних вентильно-індукторних двигунів дрона з багатокомірковими перетворювачами і просторово-часовою модуляцією», № державної реєстрації 0120U102131, у відповідності до пріоритетного тематичного напрямку наукових досліджень і науково-технічних розробок 3. Енергетика та енергоефективність.

Мета та задачі дослідження. Метою дисертаційної роботи є подальший розвиток теорії моделювання хмарного покриття для створення системи динамічної комутації сонячних панелей з урахуванням нелінійності вольт-амперних характеристик (ВАХ) та підвищенням ефективності використання сонячної енергії.

Для досягнення поставленої мети в роботі вирішуються наступні задачі:

1. Аналіз сучасних математичних моделей, топологій, методів побудови та способів максимізації відбору енергії у відновлюваних системах, що будуються на основі сонячних панелей, з урахуванням особливостей хмарного покриття.

2. Розробка удосконаленої нелінійної математичної моделі СП з урахуванням їх робочої температури, а також способів визначення її параметрів.

3. Розробка математичної моделі хмарного покриття з урахуванням зміни параметрів хмар.

4. Комп'ютерне моделювання роботи системи динамічної комутації топології сонячних панелей, що працює в умовах наближених до реальних.

Об'єктом дослідження є процеси, що протікають в системах генерації електроенергії на основі сонячних панелей, та їх режими роботи в залежності від топології та зовнішніх факторів.

Предметом дослідження є система електроживлення на основі СП.

Методи дослідження. При вирішенні поставлених задач для побудови математичної моделі первинного джерела енергії та хмарного покритву використано теорію інтегральних перетворень, чисельні методи вирішення трансцендентних рівнянь, статистичні методи, методи регресійного аналізу та теорії електричних кіл. Розрахунки та моделювання проводились з використанням мови програмування Python та програмного пакету Matlab Simulink.

Наукова новизна отриманих результатів полягає в наступному:

1. На основі аналізу еквівалентної схеми заміщення розроблено удосконалену математичну модель СП, що дозволяє покращити точність апроксимації ВАХ.
2. Розроблено спосіб визначення параметрів удосконаленої математичної моделі на основі оцифрованої ВАХ та табличних даних СП.
3. Вперше розроблено метод генерації моделі хмарного покритву для різних типів хмар з урахуванням випадкової зміни їх щільності на основі визначення параметрів по їх спектральним характеристикам.

Практичне значення отриманих результатів

1. Розроблено математичну модель системи динамічної комутації масиву СП.
2. Розроблено програмне забезпечення для визначення параметрів для відтворення ВАХ СП, генерації хмарного покритву, а також визначення топології, яка забезпечує максимальну вихідну потужність при наявності заданого хмарного покритву.

Особистий внесок автора. Дисертаційна робота є узагальненням результатів досліджень, проведених автором самостійно. Автору належать обґрунтування задачі, проведення досліджень, аналіз і обробка результатів, висновки за отриманими результатами роботи. У роботах, опублікованих зі співавторами, дисертанту належить: [5] – створення лінійних функцій Уолша за допомогою процедури Грама-Шмідта; [6], [7] – використання лінійних функцій Уолша у процедурі прогнозування; [8], [9] – створення удосконаленої математичної моделі СП та способу визначення її параметрів; [10] – розробка математичної моделі хмарного покриву.

Апробація результатів дисертації. Основні положення дисертаційної роботи обговорювались на вітчизняних і міжнародних науково-технічних конференціях: науковій конференції молодих вчених та студентів «Generation – Transmission – Use GPW» (2019, 2021), VI міжнародній науково-технічній конференції «SMART-ТЕХНОЛОГІЇ В ЕНЕРГЕТИЦІ ТА ЕЛЕКТРОНІЦІ – 2021» (STEE-2021), 2022 IEEE 41st International Conference on Electronics and Nanotechnology (ELNANO), на наукових семінарах НАН України «Аналіз процесів у приладах, пристроях та системах електроніки», секція «Перетворення параметрів електричної енергії» від 30 листопада 2022.

Публікації. Основні положення роботи відображено у 6 публікаціях: 5 статтях у наукових фахових виданнях (1 стаття у зарубіжному виданні, 3 у вітчизняних фахових виданнях категорії Б, 1 у виданні, яке включено до міжнародних наукометричних баз).

Структура та обсяг дисертації. Дисертація складається зі змісту, вступу, 4 розділів, висновків, списку використаної літератури із 143 найменувань та 6 додатків. Загальний обсяг роботи становить 214 сторінок, у тому числі 142 сторінки основного тексту, 87 рисунків та 19 таблиць.

РОЗДІЛ 1. СИСТЕМИ ЕЛЕКТРОЖИВЛЕННЯ З ВИКОРИСТАННЯМ СОНЯЧНИХ ПАНЕЛЕЙ

1.1. Актуальність використання відновлюваних джерел енергії

Глобальне зростання енергоспоживання [11] стало причиною занепокоєння щодо енергетичної стійкості, негативного впливу на навколишнє середовище та загострення проблеми кліматичних змін [12]. Цей тренд є результатом кількох взаємопов'язаних факторів, які включають стрімке збільшення населення, прискорену індустріалізацію та урбанізацію. Значна частка енергетичних потреб у сучасному світі припадає на транспорт. Використання викопного палива в транспортній галузі призводить до великого споживання енергії та спричиняє серйозні екологічні проблеми [13], [14]. Прагнення перейти на електричні види транспорту сприяє збільшенню попиту на електроенергію. Хоча такий перехід має потенційні екологічні переваги у вигляді зменшення викидів парникових газів, проте це також призводить до додаткового навантаження на систему електропостачання [15].

Прогнозується, що глобальний попит на електроенергію продовжуватиме зростати приблизно на 2,3% щорічно (рис. 1.1), що призведе до подвоєння споживання електроенергії до 2050 року [16], [17]. Подібне зростання споживання електроенергії потребує стратегічного підходу, який забезпечуватиме як енергетичну безпеку, так і екологічну стійкість, що в кінцевому результаті дозволить реалізувати концепцію сталого розвитку людства [18].

Зростаючий попит на електроенергію, якщо він задовольнятиметься виключно за рахунок традиційного електроенергії на основі викопного палива, створює значні екологічні, економічні та соціальні проблеми. Крім того, залежність від обмежених ресурсів викопного палива викликає занепокоєння щодо енергетичної безпеки [14].

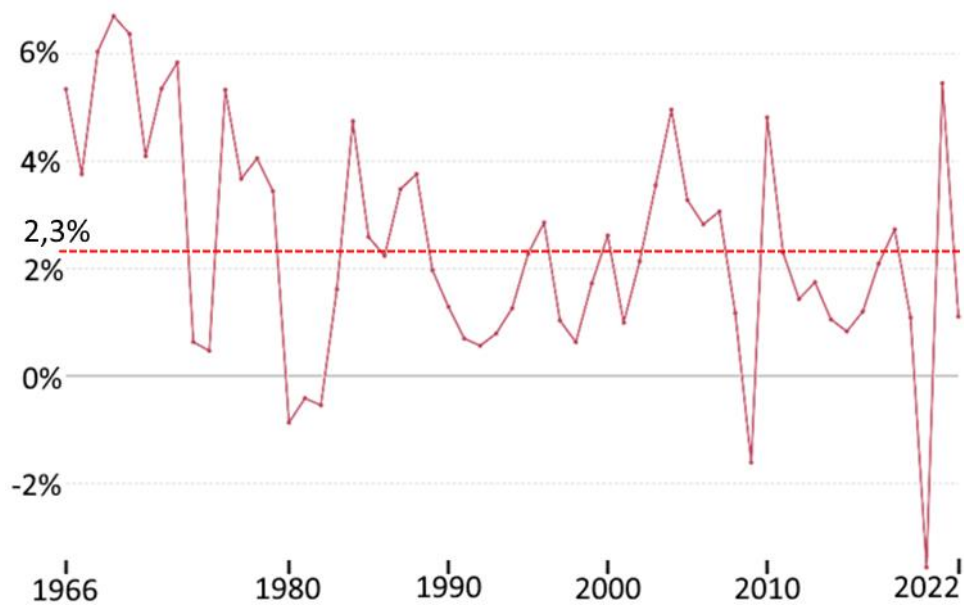


Рис. 1.1. Графік зміни споживання електроенергії по рокам

Відновлювані джерела енергії (ВДЕ), стали багатообіцяючим рішенням для задоволення зростаючого попиту на електроенергію, які одночасно зменшують негативний вплив на навколишнє середовище [19]. ВДЕ є одним із основних шляхів досягнення стратегії сталого розвитку, насамперед завдяки їх мінімальному вуглецевому сліду та практично невичерпній природі первинного носія енергії. Впровадження ВДЕ має фундаментальне значення для відмови від викопного палива та досягнення глобальних кліматичних цілей [20].

Основними перевагами використання ВДЕ є [21]:

1. Значне скорочення викидів шкідливих речовин в атмосферу в процесі експлуатації на відміну від традиційних джерел на основі викопного палива.
2. Легка масштабованість, що дозволяє створювати електростанції різноманітної потужності, починаючи від електростанцій для живлення індивідуальних домогосподарств і закінчуючи централізованими електростанціями для живлення великих міст чи районів.
3. ВДЕ є основою для переходу до розосередженої генерації.

Попри це ВДЕ характеризуються значною кількістю недоліків, серед яких [21]:

1. Виробництво електроенергії залежить від погодних умов, що призводить до переривчастої генерації, що може потребувати рішень для накопичення енергії для постійного електропостачання.

2. Низька щільність потоку енергії первинного носія, що збільшує потреби в площі.

3. Відносно висока вартість електроенергії у порівнянні з викопними видами палива.

Серед інших видів ВДЕ джерела на основі сонячної енергії займають ключові позиції [22]. Це пов'язано з тим, що первинний енергоносіє є широкодоступними, а перетворювачі сонячної панелі не містять механічних деталей, що спрощує їх експлуатацію в рамках індивідуальних домогосподарств.

Окрім того, вартість сонячних панелей помітно знизилася, що зробило сонячну енергію конкурентоспроможною або навіть дешевшою за звичайне викопне паливо в багатьох регіонах [23]. Подібне зниження витрат у поєднанні зі сприятливою державною політикою стимулювало впровадження сонячної енергії в глобальному масштабі.

У зв'язку з цим, починаючи з 90-х років, світовий ріст сукупної потужності сонячних перетворювачів енергії показував експоненціальний ріст, що робить сонячну енергетику, поряд з вітроенергетикою, найбільш швидко зростаючими за встановленою потужністю видами ВДЕ (рис. 1.2) [24].

Крім того, сонячна енергія добре узгоджується з концепцією розподіленого виробництва енергії [25]. Подібна децентралізація виробництва енергії зменшує навантаження на централізовані мережі та підвищує енергетичну безпеку, особливо під час пікового попиту або нестабільності мережі, а також нівелює витрати на транспортування електроенергії. Це також дає можливість регіонам з обмеженим доступом до традиційної електроенергетичної інфраструктури зробити ривок у чистіше енергетичне майбутнє [26].

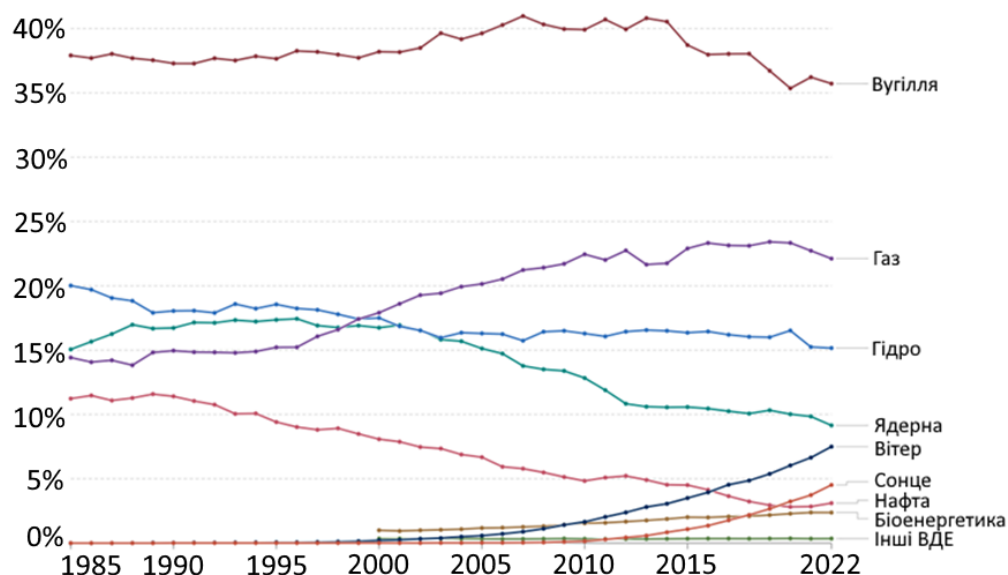


Рис. 1.2. Графік частка виробленої електроенергії по джерелам

Таким чином, глобальний перехід до використання ВДЕ, зокрема сонячної, є відповіддю на взаємопов'язані виклики зростання споживання електроенергії та зміни клімату. Цей перехід характеризується фундаментальним переосмисленням енергетичного сектору з використанням більш екологічно чистих альтернатив, які здатні забезпечити сталий розвиток і енергетичну безпеку.

Хоча фотоелектричні системи пропонують безліч переваг, вони також володіють рядом недоліків, які необхідно вирішити для створення ефективної сонячної електростанції. У зв'язку з цим постає потреба в визначенні об'єктивних параметрів для оцінки потенціалу сонячної енергії перед впровадженням системи електроживлення, що будуються на основі СП.

1.2. Потенціал сонячної енергії

Сонячна енергія – електромагнітне випромінювання, яке генерує Сонце в процесі протікання термоядерних реакцій. Це відновлюване та стійке джерело енергії, яку можна використати та перетворити на інші, придатні для використання форми енергії. Сонячна енергія є єдиним і найбільшим джерелом енергії у розпорядженні людства, яке визначає клімат та перебіг всіх

процесів на нашій планеті. Також в значній мірі всі інші ВДЕ є похідними від сонячної енергії.

Сонячне випромінювання характеризується наступними фізичними величинами [27]:

1. Енергія випромінювання (Q_e , Дж) – кількість енергії, яку випромінює джерело в процесі емісії електромагнітного випромінювання.

2. Потік (потужність) випромінювання (Φ_e , Вт) – потужність, яка передається електромагнітним випромінюванням:

$$\Phi_e = \frac{dQ_e}{dt}$$

Спектральний потік (потужність) випромінювання – потужність випромінювання джерела, яка припадає на одиницю частоти ($\Phi_{e,\nu}$, Вт/Гц):

$$\Phi_{e,\nu} = \frac{dQ_e}{d\nu}$$

Або довжини хвилі ($\Phi_{e,\lambda}$, Вт/нм):

$$\Phi_{e,\lambda} = \frac{dQ_e}{d\lambda}$$

3. Щільність потоку випромінювання – фізична величина, яка характеризує поверхневу щільність потужності випромінювання, яке падає на поверхню (E_e , Вт/м²):

$$E_e = \frac{dQ_e}{dS},$$

де dS – площа поверхні.

4. Спектральна щільність потоку випромінювання – фізична величина, яка визначає поверхневу потужність випромінювання, яка припадає на одиницю частоти ($E_{e,\nu}$, Вт/Гц·м²):

$$E_{e,\nu} = \frac{dE_e}{d\nu}$$

Або довжини хвилі ($E_{e,\nu}$, Вт/нм·м²):

$$E_{e,\lambda} = \frac{dE_e}{d\lambda}$$

Серед наведених параметрів особливо важливими є щільність потоку випромінювання E_e та спектральна щільність потоку випромінювання $E_{e,\nu}$, оскільки ці величини характеризують частку випромінювання, яка досягає поверхні Землі.

Враховуючи періодичність обертання Землі навколо Сонця, лише миттєвих значення наведених вище параметрів недостатньо. Тому доволі часто використовують їх усереднені значення за різні проміжки часу. Прикладом такого усереднення є сонячна стала G_{SC} [28]. Для її визначення використовують експериментально отримані значення щільності потоку випромінювання, які приведені до середньої річної відстані між Сонцем і Землею (рис. 1.3). Результатом усереднення нормалізованих даних є значення сонячної сталої G_{SC} [29].

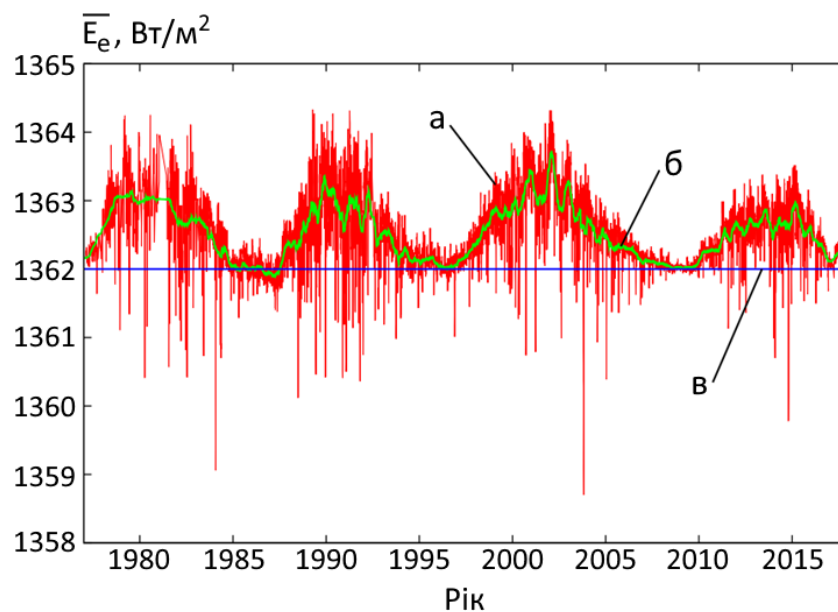


Рис. 1.3. Графік експериментальних даних щільності потоку випромінювання на межі атмосфери: а – денне середнє значення, б – ковзне середнє за 121 день, в – сонячна стала

Незважаючи на недавні дослідження, які мали на меті уточнення значення даної константи у зв'язку зі значним розвитком супутникових засобів вимірювання, її значення залишається в діапазоні $G_{SC} = 1361 \dots 1362 \text{ Вт/м}^2$ [29], [30]. При цьому змінами, які виникають в наслідок 11-річного сонячного циклу або інших факторів, можна знехтувати, оскільки вони становлять всього 0,1 % від значення сталої.

Для оцінки потенціалу сонячного випромінювання спектральні характеристики сонячного випромінювання відіграють чималу роль, оскільки сонячне випромінювання має неперервний спектр з нерівномірним розподілом потужності по довжинам хвиль. Окрім того, фотоелектричні перетворювачі використовують напівпровідникові матеріали, які можуть мати різну ефективність в залежності від частоти електромагнітного випромінювання.

Експериментально виміряне значення спектральної щільності потоку випромінювання нормують аналогічним чином, як і значення сонячної константи G_{SC} . Даний спектр отримав назву AM0 (air mass zero spectrum) і він має вирішальне значення при дослідженні, розробці та виготовленні напівпровідникових перетворювачів сонячної енергії як для космічного застосування, та і для використання в земній атмосфері.

На основі спектру AM0 можна отримати сонячну сталу G_{SC} , яка є усередненим значенням щільності потоку випромінювання за рік. Дана стала визначається наступним чином:

$$G_{SC} = \int \overline{E_{e,\lambda}} d\lambda,$$

де $\overline{E_{e,\lambda}}$ – спектральна щільність потоку випромінювання, визначення для середньорічної відстані між Землею та Сонцем.

Таким чином спектр AM0 є способом охарактеризувати сонячну сталу G_{SC} з точки зору її спектрального розподілу. Враховуючи цей зв'язок і важливість спектральних характеристик та сонячної константи, спектр AM0

був стандартизований для уніфікації та узгодження досліджень у цій сфері по всьому світі [31], [32].

Спектр АМ0 можна розрахувати теоретично, використовуючи закон випромінювання Планка. При цьому теоретично розрахований спектр сонячного випромінювання на межі атмосфери для абсолютно чорного тіла з ефективною температурою 5800 К (рис. 1.4, а) добре узгоджується з експериментально виміряним спектром (рис. 1.4, б). Відмінності в спектральному розподілі зумовлені поглинанням у більш холодних, верхніх частинах фотосфери Сонця.

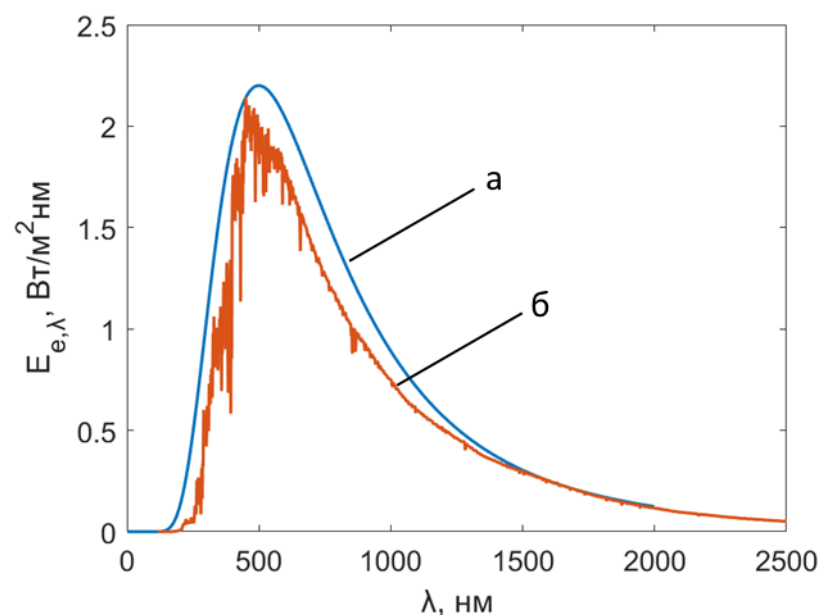


Рис. 1.4. Графіки спектрів сонячного випромінювання: а – теоретичний на межі атмосфери, б – експериментальний на межі атмосфери (АМ0)

У процесі проходження сонячного випромінювання через атмосферу мають місце різні види взаємодії випромінювання з атмосферою:

1. Відбивання. При проходженні сонячного світла через атмосферу велику частину світла відбивають хмари, різноманітні частинки та аерозолі (рис. 1.5, б).

Кількість сонячної випромінювання, яку відбивають аерозолі в повітрі, також відома як альбеда, залежить від різних факторів, включаючи розмір, склад, концентрацію та оптичні властивості частинок аерозолі. У середньому

аерозолі можуть відбивати значну частину падаючого сонячного випромінювання, що називається прямим аерозольним ефектом (DAE – direct aerosol effect) [33], [34].

Важливо відзначити, що аерозолі також відіграють певну роль у непрямому аерозольному ефекті (IAE – indirect aerosol effect), діючи як осередки конденсації хмар, потенційно впливаючи на властивості хмар і опосередковано впливаючи на їх альбедо [34].

Загальний вплив аерозолів на енергетичний баланс і клімат Землі є складним через їх різноманітні характеристики та взаємодію з хмарами та сонячним випромінюванням. Точна кількість сонячного випромінювання, відбитого аерозолями, може змінюватися залежно від регіональних і місцевих умов, типів присутніх аерозолів та інших атмосферних факторів.

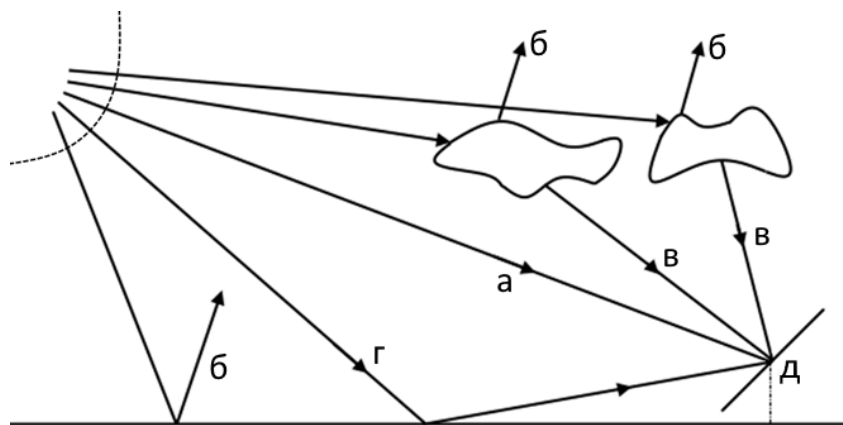


Рис. 1.5. Ілюстрація механізмів впливу атмосфери на сонячне випромінювання: а – падаюче пряме випромінювання (DNI); б – відбите у космос атмосферою та поверхнею землі випромінювання; в – розсіяне або поглинуте і повторно емітоване випромінювання (DHI); г – відбите від поверхні випромінювання (GRR), д – прийомна площадка. Штрихова лінія позначає умовну межу атмосфери

В свою чергу альбедо хмарного покриву може значно змінюватися залежно від сукупності факторів, включаючи тип хмар, їхньої висоти, товщини, а також розміру і складу частинок хмари. Хмари відіграють вирішальну роль у кліматичній системі Землі, відбиваючи сонячне світло та

затримуючи тепло, що виділяється поверхнею Землі (парниковий ефект). Баланс між цими двома ефектами має значний вплив на загальний енергетичний потенціал сонячної енергії та клімат.

У середньому атмосфера відбиває близько 30% сонячного випромінювання [35]. При цьому хмарний покрив відбиває 20% сонячного випромінювання, хоча це значення може коливатися у діапазоні від 10% для тонких перистих хмар до понад 90% для густих купчастих хмар із високим відбиттям, аерозолі – 6%, а земна поверхня – 4% [36]. Наявність даних механізмів взаємодії сонячного випромінювання з атмосферою призводить до того, що лише половина сонячного випромінювання досягає земної поверхні. Таким чином вплив хмарного покриву є визначальним у механізмі відбивання.

Окрім того, поза атмосферою сонячне випромінювання характеризується наявністю лише прямого направленого потоку від Сонця до Землі, тоді як наявність атмосфери і земної поверхні створює наступні складові сонячного випромінювання на поверхні довільно орієнтованої прийомної площадки:

1. Сумарне сонячне випромінювання (TSI – total solar irradiance) – це сумарна потужність сонячного випромінювання по всіх довжинах хвиль на одиницю площі у верхніх шарах атмосфери. Чисельно рівна сонячній сталій G_{sc} .

2. Пряме нормальне випромінювання (DNI – direct normal irradiance) – щільність потоку випромінювання на прийомній площадці на поверхні землі, орієнтованій під прямим кутом до випромінювання. Чисельне значення даної складової дорівнює сумарному сонячному випромінюванню з урахуванням втрат, які виникають внаслідок взаємодії атмосфери з випромінюванням (рис. 1.5, а).

3. Дифузійне горизонтальне випромінювання (DHI – diffuse horizontal irradiance) – щільність потоку випромінювання на прийомній площадці від світла, розсіяного атмосферою (рис. 1.5, в)

4. Відбите випромінювання (G_{RR} – ground reflected radiation) – щільність потоку випромінювання на прийомній площадці від світла, відбитого від поверхні землі та інших об'єктів (рис. 1.5, д)

5. Сумарне горизонтальне випромінювання (G_{HI} – global horizontal irradiance) – сумарна щільність потоку випромінювання на прийомній площадці.

У фотовольтаїці прийнято позначати щільності потоку випромінювання за допомогою G , тому надалі будемо використовувати його. Чисельне значення сумарної горизонтальної щільності потоку випромінювання визначається наступним чином:

$$G_{GHI} = G_{DNI} + G_{DHI} + G_{GRR}$$

Цей параметр є ключовим при оцінці потенціалу сонячної енергії, оскільки він забезпечує загальну індикацію доступного сонячного ресурсу в конкретному місці. Розподіл середньорічного сумарного горизонтального випромінювання G_{GHI} на території України зображено на рис. 1.6 [37].

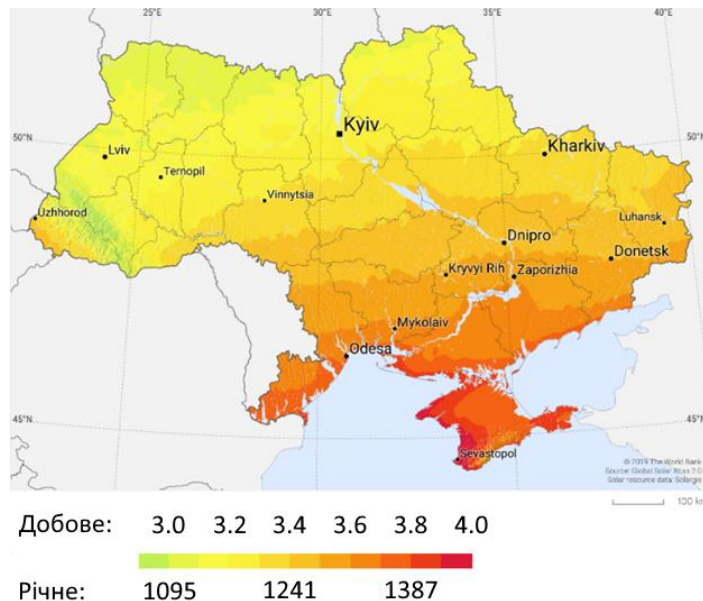


Рис. 1.6. Карта розподілу середнього значення G_{HI} по території України

Описані вище атмосферні явища також впливають на спектральну характеристику сонячного випромінювання, що призводить до того, що спектр

сонячного випромінювання доволі суттєво відрізняється від теоретичного і експериментального на межі атмосфери – AM0.

Окрім того, на спектр в значній мірі впливає географічне положення, а саме широта. Від широти залежить довжина шляху, яке світло повинне подолати для досягнення поверхні землі. Також ця відстань змінюється протягом доби за рахунок руху сонця по небесній сфері, приймаючи мінімальне значення під час сонячного полудня, тобто при мінімальному значенні сонячного зенітного кута θ (рис. 1.7).

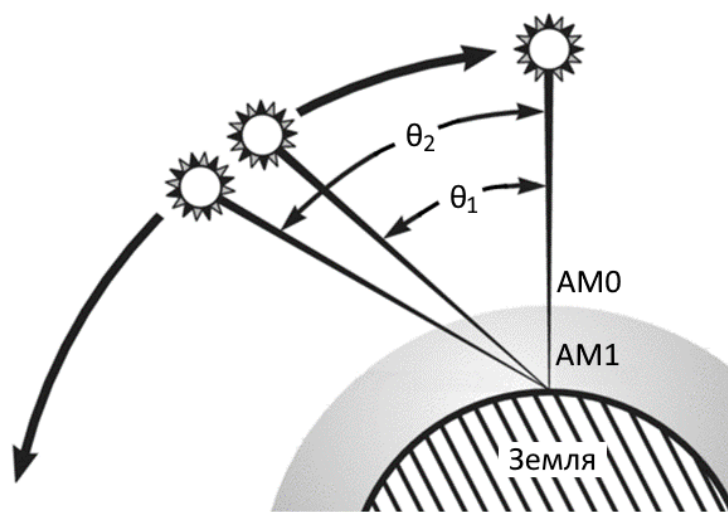


Рис. 1.7. Ілюстрація залежності відстані, яку сонячне випромінювання проходить, від зенітного кута

У зв'язку з цим вводять поняття коефіцієнту маси повітря (air mass coefficient), який визначається наступним чином [38]:

$$AM = \frac{L}{L_0} \approx \frac{1}{\cos \theta},$$

де L – довжина шляху сонячного випромінювання через атмосферу при куті падіння випромінювання θ , L_0 – довжина шляху при мінімальному значенні $\theta = 0^\circ$.

Очевидно, що при $\theta = 0^\circ$ сонячне випромінювання зазнає найменшого впливу зі сторони атмосфери, тому значення коефіцієнту маси повітря є мінімальним і дорівнює $AM = 1$. Зі збільшенням значення цього коефіцієнту

спектру також відносять спектр AM1.1 ($\theta = 25^\circ$). Дані спектри є корисними для оцінки продуктивності перетворювачів сонячної енергії в екваторіальних і тропічних регіонах.

2. AM1.5 – спектр, сонячне випромінювання якого проходить півтори відстані спектру AM1. Відповідає $\theta = 48,2^\circ$. Даний спектр є корисним для представлення загального середнього річного значення щільності випромінювання для середніх широт. Специфічне значення коефіцієнту маси повітря 1,5 було обране в 1970-х роках для цілей стандартизації на основі аналізу даних про сонячне опромінення в Сполучених Штатах [39]. З тих пір сонячна промисловість використовує AM1.5 для всіх стандартизованих випробувань або оцінки сонячних елементів або панелей в умовах земної атмосфери. Останнім стандартом AM1.5, що стосуються фотоелектричних застосувань, є ASTM G-173-03 [40]. Даний стандарт визначає два додаткові спектри для наземного використання:

2.1. AM1.5G (global spectrum) – спектр, який представляє сонячне випромінювання на рівні моря і який враховує складові сумарного горизонтального випромінювання. Спектр AM1.5G (рис. 1.9, б) зазвичай використовується для тестування напівпровідникових перетворювачів сонячної енергії та оцінки їх продуктивності, тому що він відображає умови роботи перетворювачів, які є наближеними до реальних. Має інтегровану потужність 1000 Вт/м^2 .

2.2. AM1.5D (direct spectrum) – спектр, який представляє сонячне випромінювання на рівні моря і який враховує лише пряму нормальну складову випромінювання без будь-якого розсіювання чи дифузії із-за наявності атмосфери. Спектр AM1.5D (рис. 1.9, в) часто використовується для теоретичних розрахунків і дослідницьких цілей, де основна увага приділяється продуктивності фотоелектричних матеріалів, пристроїв, а також сонячних концентраторів в ідеальних умовах прямого сонячного світла. Має інтегровану потужність 900 Вт/м^2 .

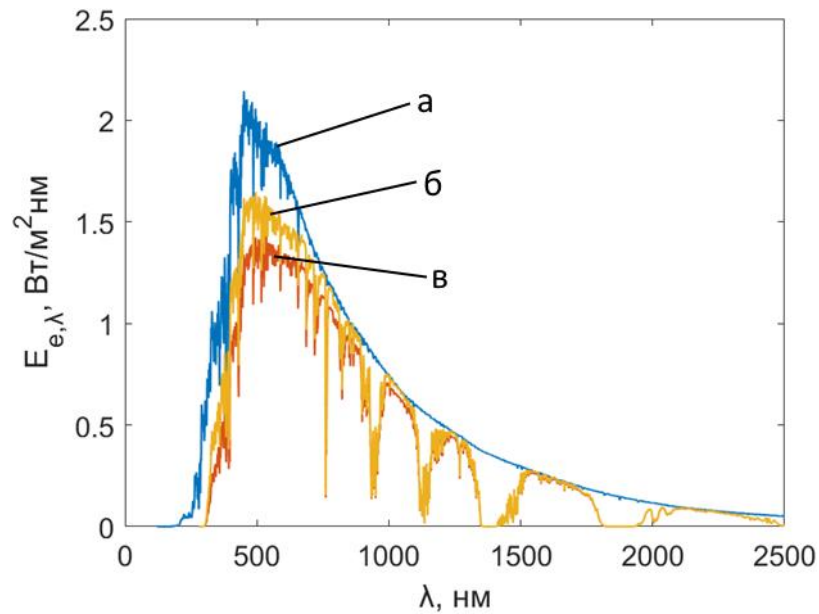


Рис. 1.9. Графіки спектральних характеристик сонячного випромінювання: а – AM0, б – AM1.5G, в – AM1.5D

Таким чином, для оцінки потенціалу сонячної найчастіше використовуються такі параметри як щільність потоку випромінювання і спектральні характеристики. Дані параметри дозволяють проводити аналіз потенціалу певної області в залежності від її географічного положення. Проте для визначення реального потенціалу необхідно обов'язково враховувати вид та характеристики хмарного покриву.

1.3. Параметри та характеристики сонячного елементу

Основним елементом фотоелектричних перетворювачів є напівпровідниковий сонячний елемент (СЕ), який безпосередньо перетворює сонячне випромінювання в електроенергію за рахунок явища внутрішнього фотоефекту.

Під явищем внутрішнього фотоефекту розуміють поглинання фотону електромагнітного випромінювання напівпровідниковим матеріалом зі збудженням електронів, що призводить до утворення пари вільних носіїв заряду дірка-електрон. Їх присутність призводить до створення різниці

потенціалів та генерації електричного струму, якщо створено електричне коло для протікання струму [41].

Класифікацію СЕ прийнято проводити на основі покоління технології їх виготовлення [42], [43]:

1. Перше покоління. СЕ даного покоління, які також називають звичайними або традиційними, виготовляються з пластин кристалічного кремнію, комерційно переважаючої фотоелектричної технології, яка включає такі матеріали, як моно- та полікристалічний кремній [44].

2. Друге покоління. При виготовленні СЕ другого покоління використовується технологія тонкоплівкового напилення, яка дозволяє скоротити використання матеріалів, необхідних для виробництва, що дозволяє зменшити вартість виготовлення. Окрім того дані СЕ є набагато тоншими і можуть бути гнучкими. Для виготовлення СЕ цього покоління використовують аморфний кремній та інші напівпровідникові сполуки.

3. Третє покоління. Дане покоління СЕ характеризується використанням нещодавно розроблених хімічних сполук з використанням тонкоплівкового напилення. СЕ даного покоління ще не були комерційно застосовані та все ще перебувають на стадії дослідження чи розробки, метою яких є створення недорогих та високоефективних СЕ.

Основною характеристикою СЕ є його вольт-амперна характеристика (ВАХ). За домовленістю ВАХ СЕ є дзеркальним відображенням ВАХ фотодіоду відносно осі напруги V (рис. 1.10).

Вирізняють наступні параметри СЕ:

1. Напруга холостого ходу V_{oc} – максимальна напруга, яку може генерувати СЕ в режимі холостого ходу (ХХ).

2. Струм короткого замикання I_{sc} – максимальний струм, який забезпечує СЕ в режимі короткого замикання (КЗ).

3. Точка максимальної потужності (V_{MPP}, I_{MPP}) – точка ВАХ СЕ, яка відповідає максимальному значенню вихідної потужності, яку може генерувати СЕ.

4. Температурний коефіцієнт K – параметр, який визначає відносну зміну іншої параметра при зміні температури. Для СЕ визначають температурні коефіцієнти для струму КЗ K_I , напруги ХХ K_V .

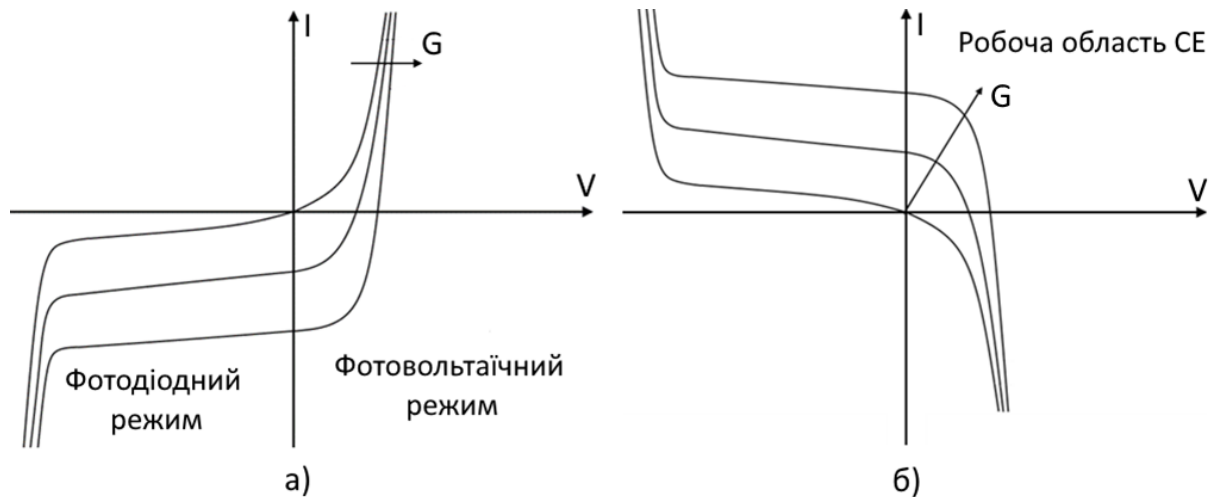


Рис. 1.10. Схематичні ВАХ фотодіоду (а) та СЕ (б)

Коефіцієнт корисної дії – параметр, який визначає частку падаючої потужності сонячного випромінювання, яка перетворюється в електроенергію. Коефіцієнт корисної дії (ККД) СЕ в значній мірі визначається матеріалом, з якого виготовляється СЕ. ККД сучасних СЕ з одним $p-n$ переходом наближається до теоретичної межі ефективності Шоклі-Квайсера, яка становить 33,16% [45]. Тому для підвищення ККД використовують СЕ, які складаються з кількох напівпровідникових шарів, кожен з яких призначений для поглинання певного діапазону довжин хвиль із спектру [46], [47]. Використовуючи різні напівпровідникові матеріали, кожен з яких оптимізований для різного діапазону довжин хвиль, СЕ з декількома $p-n$ переходами можуть досягти вищої ефективності, поглинаючи ширший спектр світла. Рекордне значення ефективності подібних СЕ сягає 47%, типове ж значення ККД для комерційно поширених СЕ коливається в діапазоні від 15 до 22% [48].

5. Іншим параметром для оцінки ефективності СЕ на основі розглянутих раніше параметрів є коефіцієнт заповнення FF , який визначається наступним чином:

$$FF = \frac{V_{MPP} I_{MPP}}{V_{OC} I_{SC}}$$

6. Спектр поглинання – вказує на те, наскільки СЕ чутливий до різних довжин хвиль світла. Різні матеріали для СЕ по-різному реагують на різні частини сонячного спектру, що в результаті визначає ефективність роботи.

Типова вихідна потужність СЕ доволі мала і становить до 10 Вт, при цьому типова вихідна напруга СЕ становить 0,5 В, що не задовольняє потреби більшості споживачів. Тому для збільшення вихідної напруги СЕ з'єднують послідовно в герметичну і стійку до атмосферного впливу структуру – сонячну панель – СП (рис. 1.11). Зазвичай кількість СЕ в СП кратна 12, а саме: 12, 24, 36, 48 і 72. Номінальна потужність однієї такої панелі лежить в діапазоні від 30 до 700 Вт [41], [42].

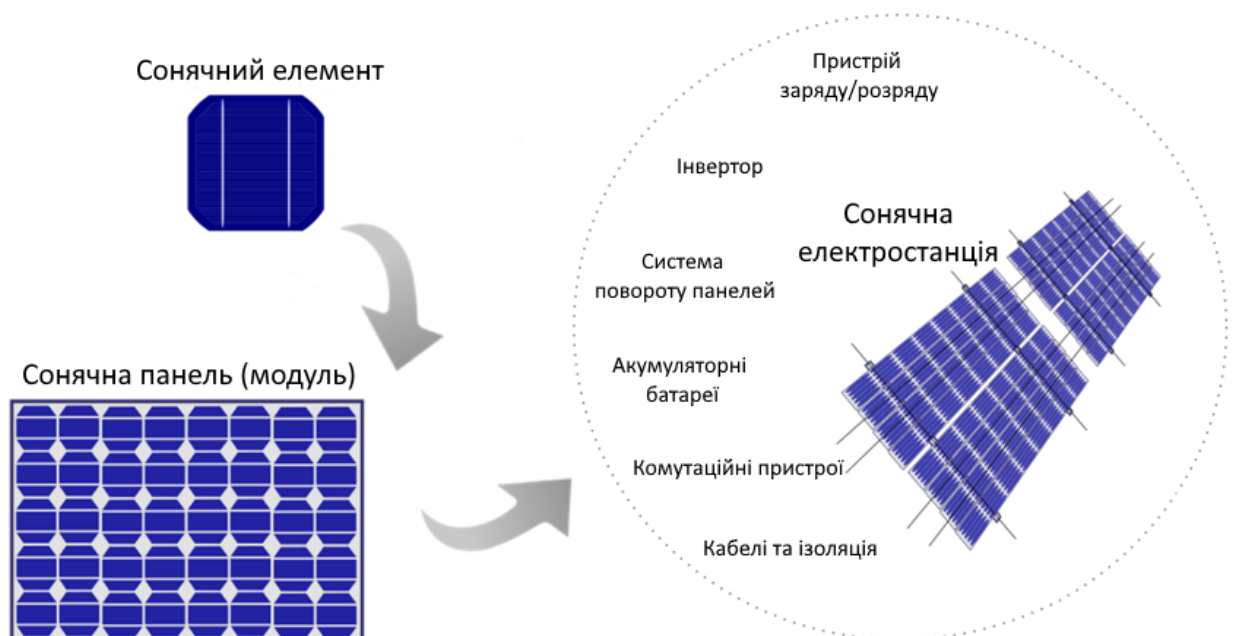


Рис. 1.11. Ілюстрація прикладу масштабування вихідної потужності СЕС

Для отримання необхідних параметрів електричної енергії (вихідні струм і напруга) СП з'єднують послідовно-паралельно, що в результаті

утворює масив сонячних панелей, а разом з перетворювальним обладнанням все це утворює сонячну електростанцію (СЕС). Даний перехід від СЕ до СЕС (рис. 1.11) є доволі легким і це дозволяє масштабувати вихідну потужність СЕС у великому діапазоні, що безперечно є перевагою сонячної енергетики.

Однак одночасно з відносно легкою масштабованістю проявляється один з найбільших недоліків фотоелектричних перетворювачів – ефект часткового затінення

1.4. Ефект часткового затінення

Ефект часткового затінення – зниження вихідної потужності СП за рахунок зниження вихідної потужності одного або декількох СЕ в результаті нерівномірної щільності потоку випромінювання на поверхні СП. Затінення може виникати через різні природні і штучні причини: тінь від будівель, кабелів, дерев, але найчастішою причиною є хмарний покрив.

При затіненні індивідуального СЕ ефективна щільність потоку сонячної енергії зменшується, що спричиняє падіння значень струму і напруги фотоперетворювача (рис. 1.12). В рамках індивідуального СЕ це не критично і призводить лише до зниження значення вихідної потужності.

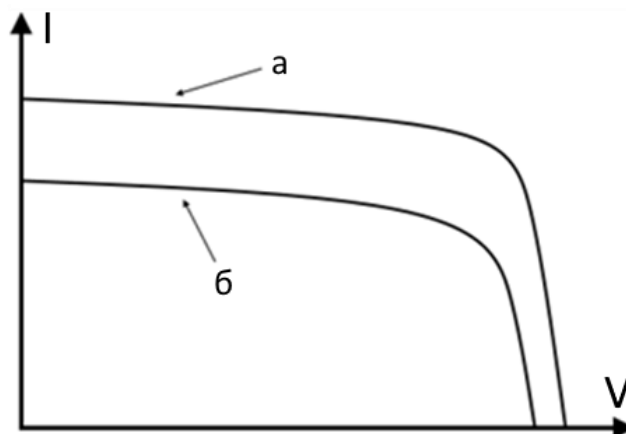


Рис. 1.12. Схематична ВАХ незатіненого (а) та затіненого (б) СЕ

Зовсім інакше ситуація виглядає у випадку СП, тобто послідовного з'єднання декількох СЕ. Якщо СЕ перебувають у умовах різної освітленості, то

у випадку послідовного з'єднання струм вітки буде визначатися точкою перетину ВАХ затіненого й віддзеркаленою ВАХ незатіненого елементів, оскільки кожен СЕ виступає джерелом струму (рис. 1.13). При цьому затінений СЕ виступає навантаженням для незатіненого. Це призводить до значного нагрівання затіненого СЕ і якщо робоча точка (V_1, I_1) знаходиться близько до точки КЗ, то це може спричинити руйнівні ефекти, такі як розтріскування елемента або скла, розплавлення припою або, навіть, до повного руйнування [49], [50].

Аналогічним чином визначають напругу ХХ у випадку паралельного з'єднання СЕ. Дана напругу визначають точкою перетину віддзеркаленої ВАХ незатіненого СЕ та ВАХ затіненого елемента. Якщо ж робоча точка (V_1, I_1) знаходиться близько до точки ХХ, то частина струму незатіненого елемента протікатиме через затінений елемент, що може привести до його перегрівання, або пошкодження. Проте, оскільки струм поблизу точки холостого ходу є доволі малим, вихід з ладу СЕ є малоімовірним.

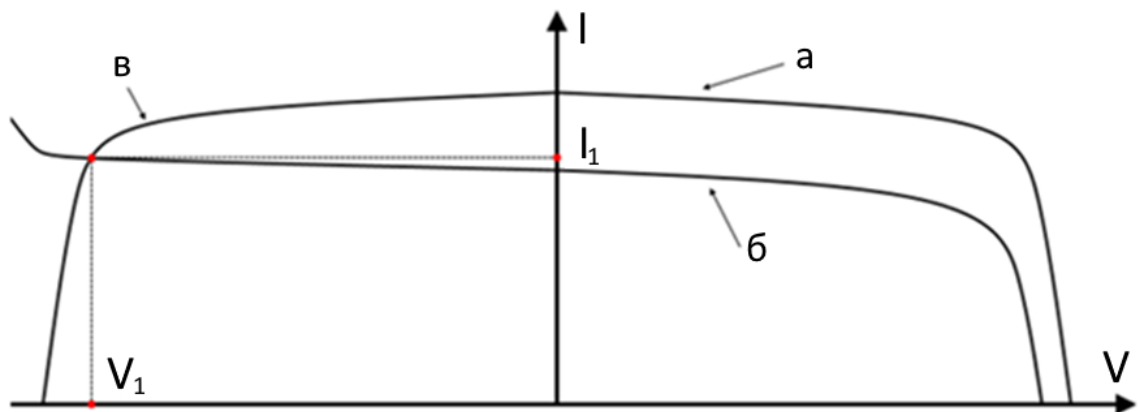


Рис. 1.13. Ілюстрація способу визначення струму КЗ послідовно з'єднаних СЕ за умов різної освітленості: ВАХ незатіненого (а), затіненого (б) та віддзеркалена затіненого (в) СЕ

Ефект частково затінення призводить до значних втрат електроенергії, а також пришвидшеної деградації СП через надмірний нагрів або їх виходу з ладу. Тому розроблено чимало способів зменшення втрат енергії та

негативного впливу на перетворювальне обладнання та СП через ефект часткового затінення.

В загальному випадку методи зменшення впливу ефекту часткового затінення можна поділити на дві великі групи: пасивні та активні (рис. 1.14) [51].

Пасивні методи передбачають використання пасивних компонентів (шунтуючі та блокуючі діоди), а також підбір оптимальної схеми увімкнення СП в масиві. Це дозволяє запобігти появі перегрівання і виходу з ладу СП при роботі в умовах часткового затінення [52].

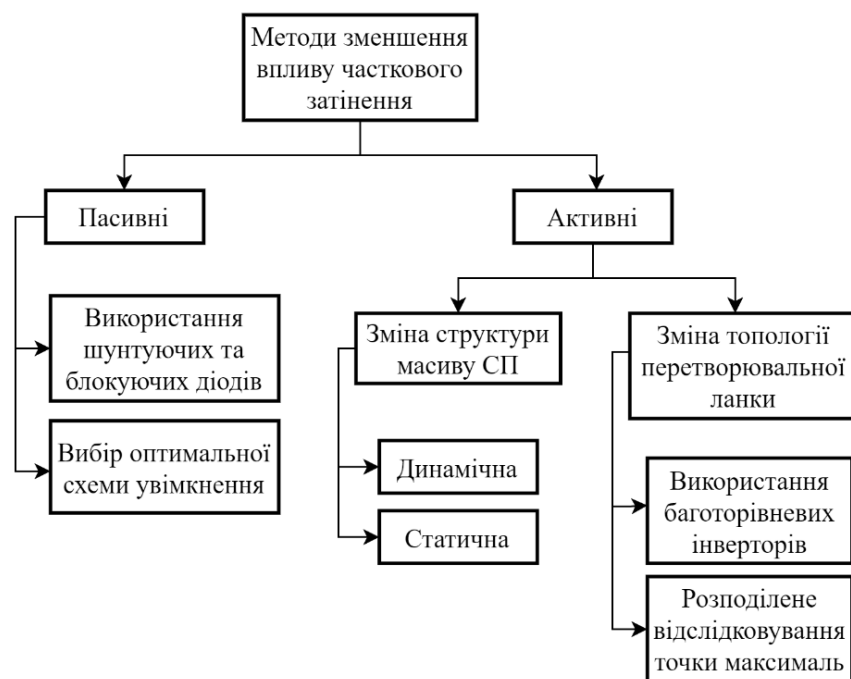


Рис. 1.14. Класифікація методів зменшення впливу ефекту часткового затінення

В контексті методів зменшення негативного впливу ефекту часткового затінення на роботу СП, пасивні методи відіграють важливу роль. Ці методи базуються на використанні пасивних компонентів, таких як шунтуючі та блокуючі діоди. Окрім використання цих компонентів, пасивні методи також включають у себе підбір оптимальної схеми підключення СП у масиві. Оптимізація конфігурації підключення може значно знизити вплив часткового затінення, особливо у великих фотоелектричних системах. Розробка такої

схеми вимагає глибокого аналізу можливих варіантів з'єднань СП, їх взаємодії та поведінки під час різних умов освітлення. Це може включати використання різних типів з'єднань, або їх комбінацій, для досягнення оптимального балансу між ефективністю та надійністю системи.

Шунтуючі діоди вмикають паралельно СЕ, тим самим створюючи додатковий шлях для протікання струму у випадку, коли одна та/або декілька елементів є затіненими. При однакових умовах освітленості напруга, яку генерують СЕ, зворотно зміщує шунтуючі діоди і вони ніяк не впливають на режим роботи послідовно з'єднаних елементів (рис. 1.15, а). При виникненні затінення один та/або декілька СЕ отримують меншу кількість сонячного випромінювання, що призводить до того, що вони переходять у фотодіодний режим. Це змінює полярність напруги на виводах даних СЕ, тобто шунтуючий діод є прямо зміщеним і через нього починає протікати надлишковий струм (рис. 1.15, б).

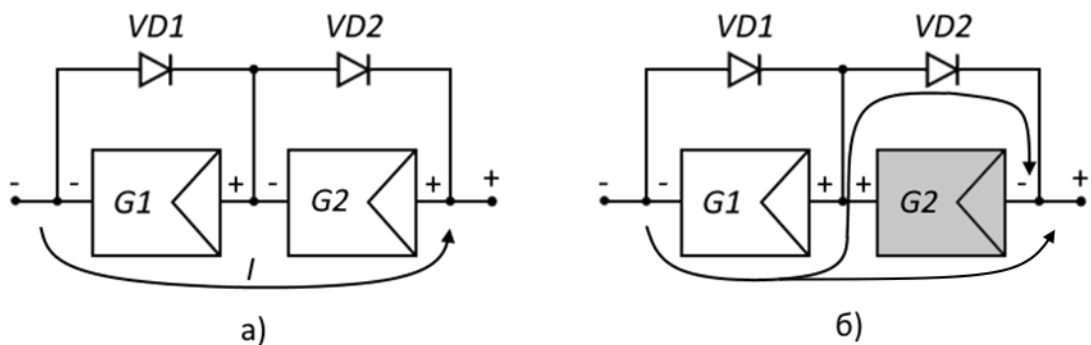


Рис. 1.15. Ілюстрація протікання струму (позначено стрілками) через шунтуючі діоди: при рівномірній освітленості панелей (а) та у випадку часткового затінення (б).

Блокуючі діоди використовуються в масиві СП і вмикаються послідовно у паралельних вітках масиву СП (рис. 1.16, а). Вони захищають вітку від протікання зворотнього струму, який може утворюватися при нерівномірному рівні освітленості паралельних віток.

В умовах рівномірної освітленості діоди практично не впливають на ВАХ та характеристику вихідної потужності (рис. 1.16, б) – втрати в

блокуючих діодах в даному випадку є малими і ними можна знехтувати. Проте при нерівномірному освітленні характеристики масиву СП змінюються доволі суттєво – на кривій вихідної потужності виникають два піки: локальний та глобальний (рис. 1.16, в). Це може призвести до того, що пристрій відслідковування точки максимальної потужності (ВТМП) «застрягає» в точці локального піку, вважаючи його глобальним піком, що призводить до значного недоотримання енергії [53]. Вирішення цього питання потребує встановлення досконаліших пристроїв ВТМП, що вимагає додаткових витрат [54].

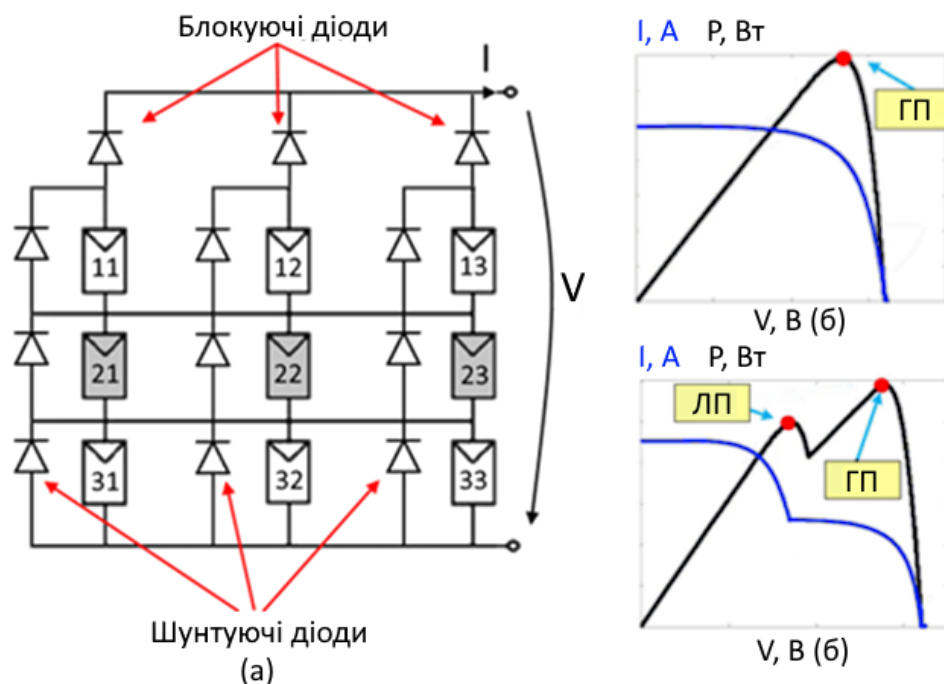


Рис. 1.16. Типова схема масиву сонячних панелей з використанням діодів (а), а також характеристики масиву в умовах рівномірного освітлення (б) та умовах нерівномірного освітлення (в). ГП – глобальний пік, ЛП – локальний пік, сірим позначено затінені панелі

Використання блокуючих і шунтуючих діодів дозволяє усунути проблему перегрів та виходу з ладу СЕ та СП і знизити втрати в умовах нерівномірної освітленості. Проте даний підхід характеризується наступними недоліки [55]:

1. Вплив на ВАХ та криву вихідної потужності. Використання діодів у масив СП вносить зміни у ВАХ і криву вихідної потужності. Це призводить

до формування додаткових локальних піків потужності, що може ускладнити процес відбору максимальної потужності з системи. Такі зміни вимагають більш складних методів керування та оптимізації для ефективного використання сонячної енергії [56].

2. Додаткові втрати енергії: Незважаючи на захисну роль, використання діодів може призвести до певних втрат. При протіканні струму через діод, частина енергії втрачається у вигляді тепла, що знижує загальну ефективність системи. Важливо враховувати ці втрати під час проектування та експлуатації СЕС.

Незважаючи на ці недоліки, використання шунтуючих і блокуючих діодів залишається найпоширенішим пасивним методом мінімізації негативного впливу часткового затінення на СП та СЕ.

Виробники зазвичай виконують шунтування СЕ всередині панелі, використовуючи один шунтуючий діод на кожні кілька десятків СЕ, що дозволяє оптимізувати захист та ефективність системи (рис. 1.17).

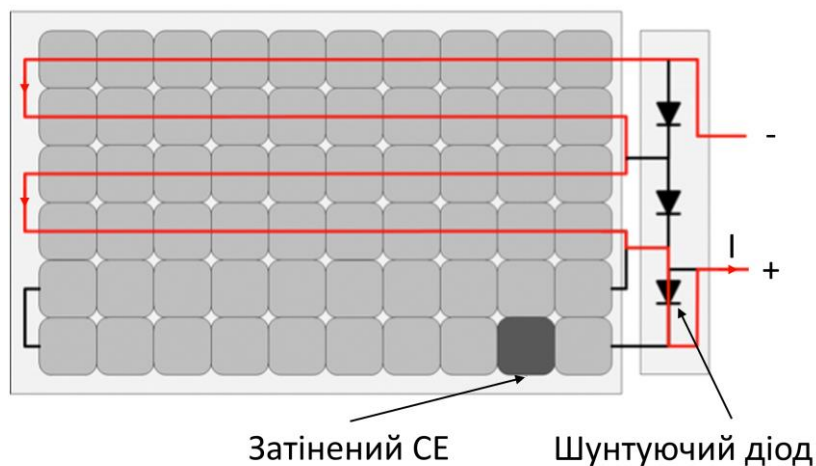


Рис. 1.17. Ілюстрація використання шунтуючих діодів всередині СП

Іншим пасивним способом зменшення впливу ефекту затінення є підбір оптимальної схеми увімкнення СП. Ряд досліджень вказує на те, що різні схеми увімкнення проявляють різну ефективність при різних умовах освітленості [57], [58]. Типові схеми увімкнення масиву СП зображені на рис. 1.18, кожна з яких має свої особливості та оптимальні умови використання:

1. Послідовне з'єднання. У цій конфігурації, всі СП з'єднуються послідовно. Це дозволяє досягти високої вихідної напруги, оскільки загальна напруга системи дорівнює сумі напруг кожного модуля. Проте, струм системи обмежений струмом одного модуля, що може бути недостатньо для деяких застосувань.

2. Паралельне з'єднання. В даній схемі увімкнення всі модулі підключаються паралельно. Це збільшує загальний струм системи, який дорівнює сумі струмів кожного модуля, але напруга залишається на рівні одного модуля. Таке з'єднання є корисним для систем, що потребують високого струму.

3. Послідовно-паралельне з'єднання. Є поєднанням двох попередніх підходів, що дозволяє збільшити як вихідну напругу, так і струм системи. Таке з'єднання забезпечує більшу гнучкість у виборі параметрів системи та часто використовується у великих СЕС.

4. Total-Cross-Tied (ТСТ). Є більш складною конфігурацією, що оптимізує роботу системи при різних умовах освітлення. Ця схема увімкнення дозволяє краще впоратися з частковим затіненням, забезпечуючи вищу продуктивність у порівнянні з традиційними схемами. Проте використання даної схеми увімкнення потребує великої кількості додаткових з'єднань і кабелів, що значно збільшує вартість системи.

5. Bridge Link (BL) та Honey Comb (НС). Ці схеми представляють собою проміжні варіанти між послідовно-паралельним з'єднанням та ТСТ. Вони забезпечують баланс між економічністю, ефективністю та здатністю адаптуватися до різних умов освітлення.

6. Гібридні схеми увімкнення, які є поєднанням або варіаціями попередніх схем, що дозволяє комбінувати переваги кожної з них для досягнення необхідного режиму роботи навантаження в умовах часткового затінення.

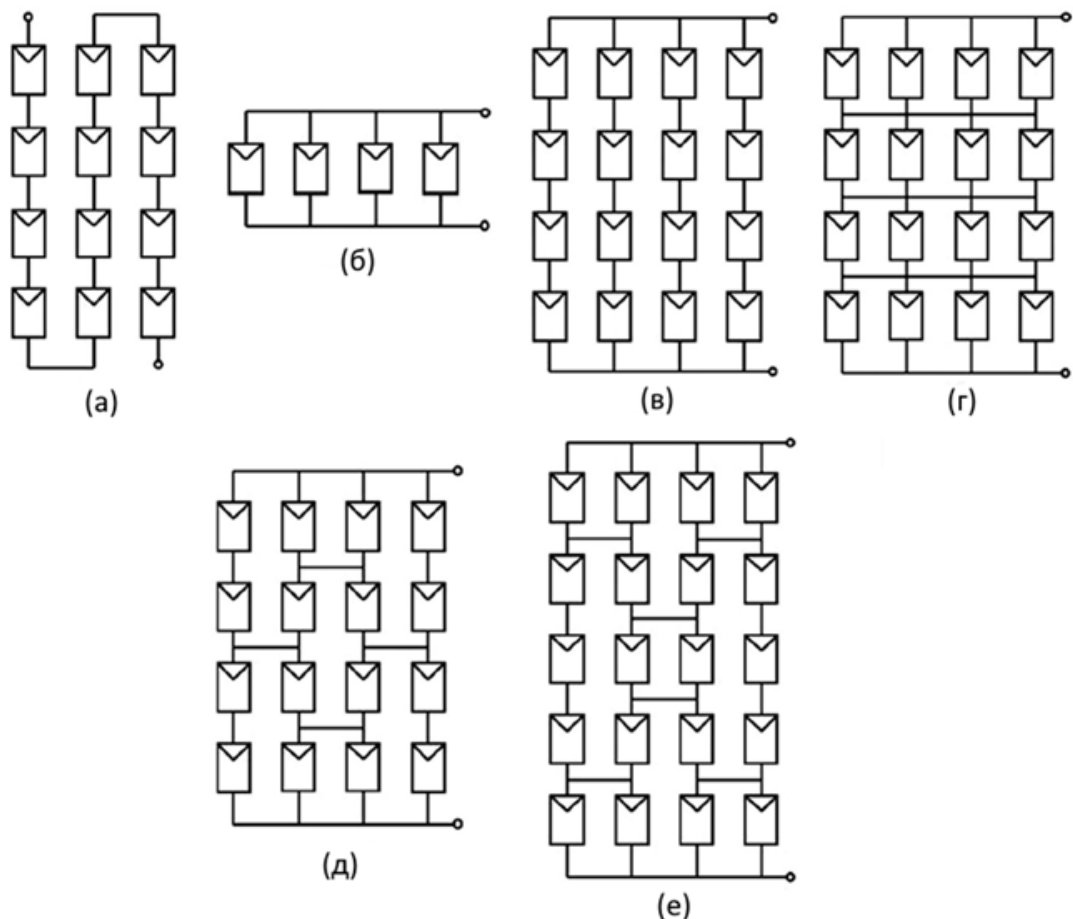


Рис. 1.18. Схеми увімкнення СП: а – послідовне, б – паралельне, в – послідовно-паралельне, г – Total-Cross-Tied, д – Bridge-Link, е – Honey-Comb

У реальних умовах, залежно від рівномірності освітленості, послідовно-паралельне з'єднання та ТСТ можуть забезпечувати найбільшу вихідну потужність. Проте, при нерівномірному освітленні, ТСТ ефективніше зменшує втрати та підвищує максимальну потужність, порівняно з іншими конфігураціями. Вибір оптимальної конфігурації залежить від специфічних умов розташування, доступності сонячного світла та вимог до електричних характеристик системи. На практиці ТСТ рідко використовують, через збільшені витрати на її встановлення. Тому найбільшого поширення здобуло послідовно-паралельне увімкнення, а також ВЛ та НС [58].

Активні методи зменшення впливу часткового затінення поділяють на дві великі групи: методи вибору оптимальної топології перетворювальної

частини та методи засновані на зміні структури масиву сонячних панелей (рис. 1.14) [51].

На сьогодні створено чималу кількість топологій перетворювальної частини СЕС. Наведемо перелік найпоширеніших топологій (Рис. 1.19) та їх сфер використання.

Топологія з центральним інвертором (рис. 1.19, а). Головною перевагою є простота системи, але вона має обмеження з точки зору масштабування, оскільки у випадку досягнення певного значення потужності (до 5 кВт) стає більш економічно вигідніше перейти до інших топологій.

Топології з розподіленими інверторами (рис. 1.19, б). Ця топологія є більш підходящою для СЕС середньої потужності. Вона дозволяє краще управляти виробленням енергії, оскільки кожен інвертор може оптимізувати вихідну потужність окремої групи панелей. Недоліком є необхідність узгодження рівнів напруг інверторів, які працюють на спільну шину змінної напруги, що інколи є складною та дорогавартісною задачею.

У зв'язку з цим доволі часто також використовують топологію з центральним інвертором та розподіленими перетворювачами в ланці постійного струму (рис. 1.19, в). Ця схема поєднує переваги централізованої та розподіленої систем. Вона дозволяє знизити вартість центрального інвертора та підвищити надійність окремих секцій.

Найбільш ефективним варіантом є використання топології з індивідуальними інверторами (рис. 1.19, г). Це забезпечує максимальну ефективність та оптимізацію кожного модуля окремо, проте через свою високу вартість дана топологія не здобула поширення.

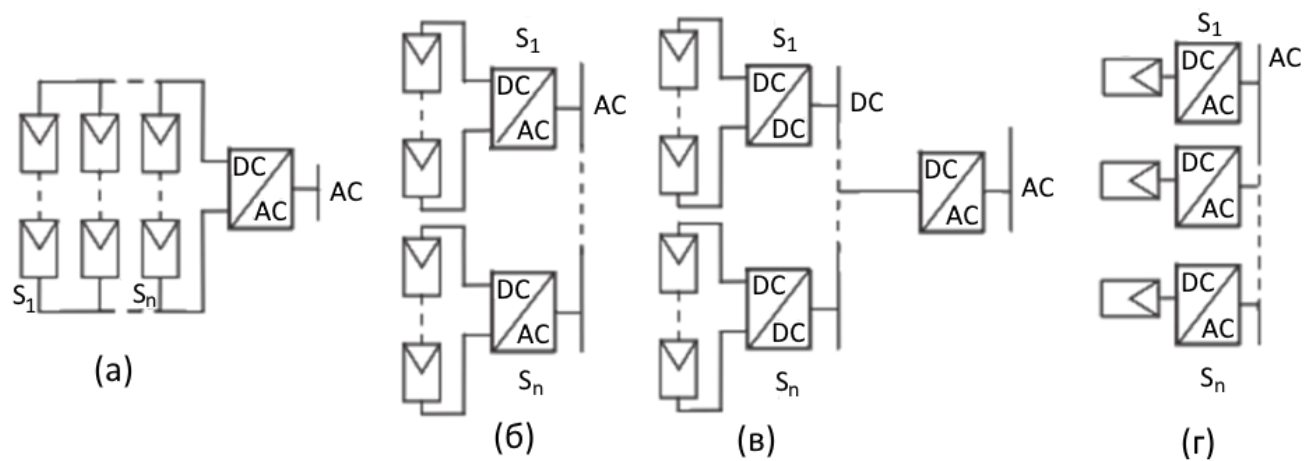


Рис. 1.19. Топології перетворювальних частин: з центральним інвертором (а), з розподіленими інверторами, які працюють на спільну шину змінної напруги (б), з розподіленими перетворювачами постійного струму, які працюють на центральний інвертор (в), з індивідуальними інверторами, які працюють на спільну шину змінної напруги (г)

Методи зміни структури масиву сонячних панелей передбачають зміну схеми увімкнення панелей в процесі експлуатації з метою максимізації відбору потужності за наявних погодних умов. При цьому виділяють два види подібних методів: статичні та динамічні.

Статичні методи зміни структури (реконфігурації) набули широкого поширення в СЕС великої потужності [59]. Процес оптимізації роботи СЕС шляхом використання статичної реконфігурації зменшення впливу часткового затінення передбачає детальний аналіз та зміну структури масиву СП на основі зібраних статистичних даних. Суть даних методів полягає у наступних кроках:

1. Збір статистичних даних. Відбувається систематичний збір даних про ефективність роботи СЕС протягом тривалого періоду, який може охоплювати тижні або місяці. Ці дані включають показники вихідної потужності, інформацію про погодні умови, кількість сонячного випромінювання та інші важливі параметри.

2. Аналіз даних та визначення оптимальної схеми увімкнення. На наступному етапі проводиться аналіз зібраних даних з метою визначення найбільш ефективної конфігурації увімкнення сонячних панелей, яка

забезпечить максимальну вихідну потужність СЕС. Цей аналіз враховує різні умови експлуатації та вплив зовнішніх факторів.

3. Реалізація змін у масиві сонячних панелей. Відповідно до результатів аналізу, вносяться зміни у фізичну конфігурацію масиву сонячних панелей. Це може включати зміну місцезположення панелей або реорганізацію їх з'єднань за допомогою додаткових кабелів.

Вказаний процес статичної реконфігурації, який детально ілюструється на рис. 1.20, є фундаментальним для підвищення ефективності та надійності сонячних енергетичних систем, особливо у контексті часткового затінення та змінних умов експлуатації.

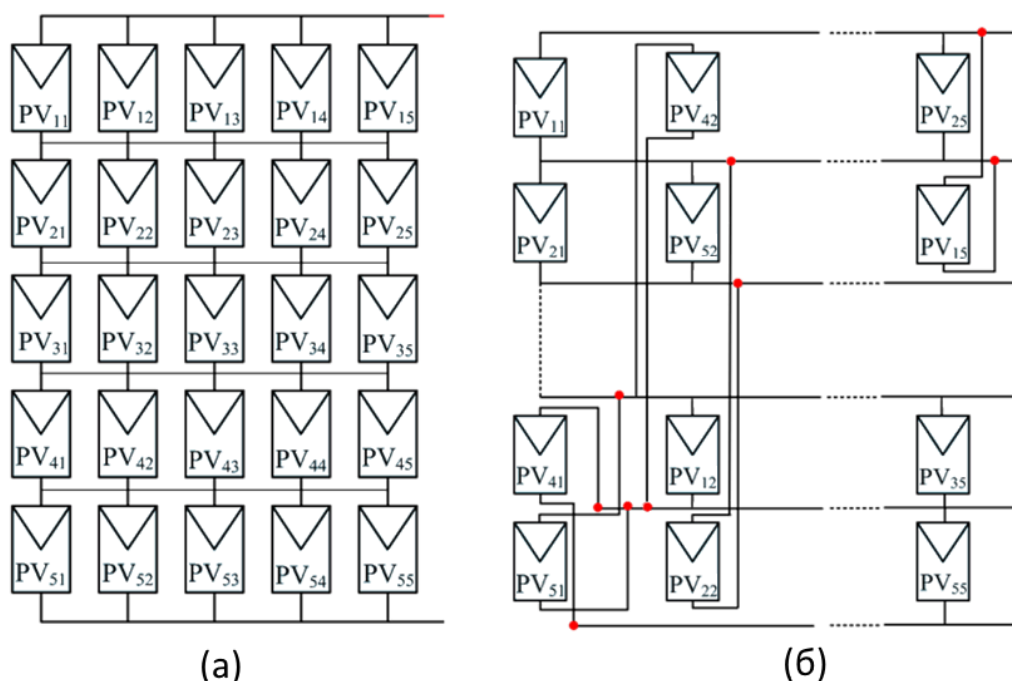


Рис. 1.20. Етапи процесу статичної реконфігурації: початкова схема увімкнення (а), яку використовують для збору даних про експлуатацію; схема після застосування методу (б)

Процес реконфігурації виконується декілька разів на рік з урахуванням сезонних змін у кількості сонячного випромінювання. Такий підхід дозволяє адаптувати систему до змінних умов освітленості та погодних факторів.

Динамічні методи рекомбінації передбачають використання комутуючих матриць (зазвичай напівпровідникових), а також мікропроцесорних систем

керування. При цьому при використанні динамічних методів комутації найчастіше використовують схему увімкнення ТСТ (рис. 1.18, г). На основі даних, отриманих від датчиків вимірювання рівня сонячного випромінювання, система керування генерує сигнали перемикання для комутуючої матриці, що дозволяє змінювати структуру масиву доволі легко та часто. Також іноді відбувається короткострокове прогнозування умов експлуатації [6], [7], що спрощує інтеграцію СЕС в загальну енергосистему.

Таким чином динамічні методи дозволяють максимізувати відбір потужності на доволі коротких відрізках часу. Проте через дороговизну необхідного для цього обладнання, дані методи використовуються в малопотужних СЕС – зазвичай електростанціях приватних домогосподарств. Типова схема масиву сонячних панелей та комутуючої матриці зображена на рис. 1.21.

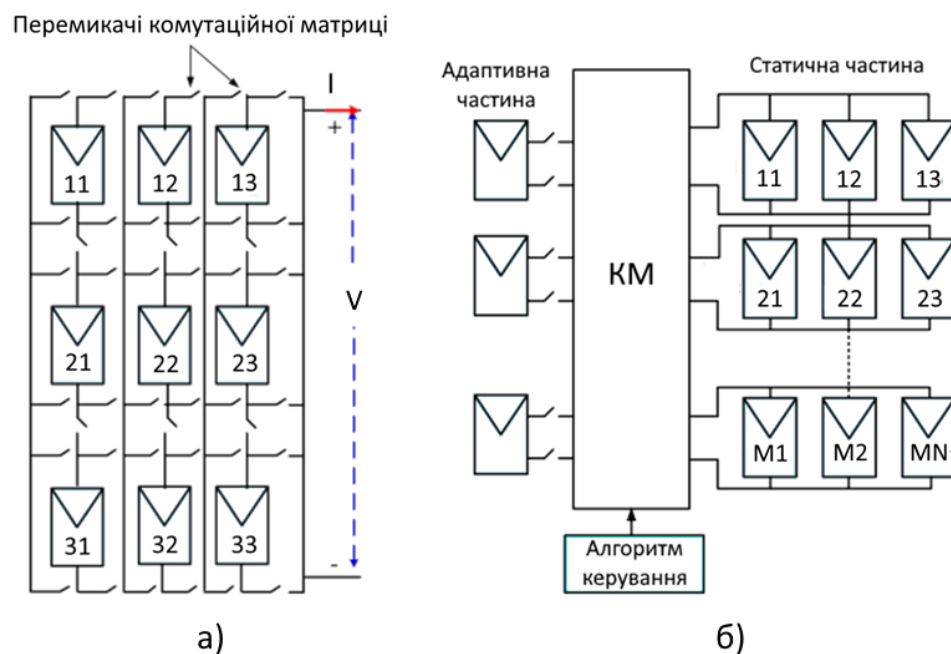


Рис. 1.21. Типова схема масиву сонячних панелей з динамічною комутацією (а), схема ААР методу

В залежності від алгоритму комутації динамічні методи рекомбінації поділяють на:

1. EAR (electrical array reconfiguration) – реконфігурація масиву СП з метою вирівнювання струму паралельних віток.

2. Irradiation equalization (IE) reconfiguration – вирівнювання рівня освітленості паралельно підключених віток масиву СП.

3. Adaptive array reconfiguration (AAR) – метод вирівнювання струму паралельно підключених панелей, для чого використовується адаптивний масив резервних сонячних панелей (рис. 1.21, б)

4. Гібридні методи – методи, які засновані на використанні комбінації попередніх.

Важливо зазначити, що для аналізу та оптимізації роботи СЕС в умовах часткового затінення необхідно:

1. Створити детальну та точну імітаційну модель СЕ та СП. Така модель дозволяє аналізувати різні сценарії експлуатації та вибирати найбільш ефективні стратегії управління системою.

2. Створити модель хмарного покриття, що дозволить використовувати комп'ютерне моделювання, і дасть важливі знання для розуміння поведінки системи та її відповіді на зміни умов часткового затінення.

1.5. Математична модель сонячного елемента

Імітаційне моделювання – підхід до моделювання реальних явищ і предметів, який передбачає заміну оригіналу на наближену модель, яка буде відтворювати поведінку оригіналу зі заданою точністю.

В електроніці одним зі найпоширеніших методів імітаційного моделювання є використання еквівалентних схем, яка є електрично еквівалентною реальному об'єкту дослідження та базується на дискретних ідеальних електричних компонентах, поведінка яких добре досліджена та чітко визначена. Оскільки фотодіод та СЕ є фотоелектричними приладами з подібною структурою, вони мають спільну схему заміщення (рис. 1.22).

Кожна схема заміщення характеризується своїми перевагами та недоліками і розрахована на моделювання СЕ в доволі вузькому діапазоні умов

навколишнього середовища, і містить мінімум два елементи: джерело фотоструму I_{PV} та паралельно увімкненого діоду VD1, який характеризує зворотній струм p - n переходу (рис. 1.22, а). Назви схем заміщення СЕ характеризують кількість діодів в схемі та наявності врахування втрат. Тому дана схема заміщення отримала назву одnodіодної ідеалізованої або ж спрощеною. Вона не враховує неідеальність напівпровідникової структури p - n переходу та паразитні явища в ньому.

Перші схеми заміщення [60] враховували ємнісні параметри напівпровідникової структури, які використовувалися для аналізу їх роботи при протіканні перехідних процесів. Для цього еквівалентна схема СЕ містить конденсатори, які моделюють дифузійну C_D та бар'єрну ємність C_J p - n переходу (рис. 1.22, а разом з г). Проте для СЕ, які працюють в режимі постійного струму, дані ємності є надлишковими і призводять до ускладнення еквівалентної схеми. Тому більшість актуальних еквівалентних схем заміщення не містять даних елементів.

Першою сферою використання СЕ було живлення штучних супутників, тому велика кількість праць даного періоду присвячена аналізу впливу космічного випромінювання на СЕ та спробам відтворити умови космічного випромінювання (спектру АМ0) в земних умовах [61], [62]. Одночасно з цим проводяться перші дослідження, які призначені аналізу проблеми часткового затінення та роботи СЕ в даних умовах [63].

В реальних СЕ існують втрати енергії (опір омичних контактів та струми витікання, тощо), тому для їх врахування до схеми заміщення додають послідовно з'єднаний резистор з опором R_s та шунтуючий з опором R_p (рис. 1.22, а і в). Схеми, які враховують втрати за рахунок даних резисторів, отримали назву практичних. Одночасно з цим створюються перші методи вимірювання та визначення параметрів наявних схем заміщення [64].

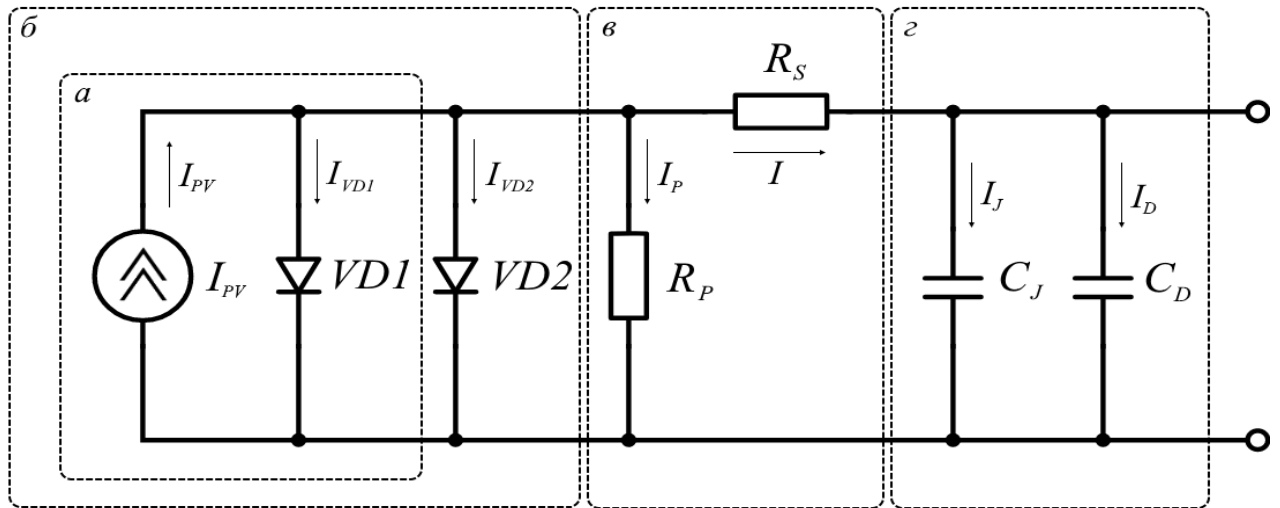


Рис. 1.22. Еквівалентні схеми заміщення СЕ (а – однодіодна ідеалізована або спрощена; б – двохдіодна ідеалізована; в – опори, які разом з контуром а або б дають практичну схему); г – бар'єрна та дифузійна ємності, які разом з контуром а або б дають схему для аналізу перехідних процесів

З метою підвищення точності апроксимації в еквівалентну схему додають додатковий діод VD2 (рис. 1.22, контур б) [65]. Дана схема отримала назву двохдіодної (інколи ще з подвійним експоненціалом). Додавання в схему додаткового діода VD2 дозволяє врахувати рекомбінацію носіїв заряду у зоні збіднення p - n переходу. В цей же час пропонуються способи визначення параметрів даної схеми заміщення [66], [67].

На сучасному етапі досліджень використовується як однодіодна [68], так і двохдіодна схеми заміщення [69]. При цьому увага досліджень сфокусована на створенні нових аналітичних і чисельних методів визначення параметрів схем заміщення, а не на удосконаленні їх структури [70]–[72]. При цьому існують роботи, присвячені збільшенню точності апроксимації еквівалентної схеми за рахунок введення додаткових елементів схеми [73]. Також існує чимала кількість робіт, які присвячені порівняльному аналізу наявних схем заміщення [74], які вказують на те, що з одного боку двохдіодна модель забезпечує дещо більшу точність апроксимації, проте з іншого боку вона також є складнішою для розрахунків. Велику кількість робіт присвячено визначенню коефіцієнту ідеальності діодів схем заміщення СЕ [75], [76].

Математична модель еквівалентної схеми заміщення СЕ (рис. 1.22) описується рівнянням вихідного струму схеми:

$$I = I_{PV} - I_{01} \left(e^{\frac{V + IR_S}{\alpha_1 V_T}} - 1 \right) - I_{02} \left(e^{\frac{V + IR_S}{\alpha_2 V_T}} - 1 \right) - \frac{V + IR_S}{R_p}, \quad (1.1)$$

де I – вихідний струм; I_{PV} – фотострум; I_{01} , I_{02} – струми діода VD1 та VD2 відповідно; α_1 та α_2 – коефіцієнти ідеальності діода VD1 та VD2 відповідно; V – вихідна напруга; $V_T = kT/q$ – теплова напруга СЕ; R_S – послідовний опір; R_p – шунтуючий опір, T – температура, k – стала Больцмана, q – заряд електрона.

Для рівняння (1.1) характерні наступні особливості:

1. Наведене рівняння також описує вихідний струм однодіодної схеми заміщення за рахунок граничного переходу $\alpha_2 \rightarrow \infty$.
2. Дане рівняння є трансцендентним – струм одночасно є незалежною змінною і аргументом трансцендентної функції. Тому для його вирішення необхідно використовувати чисельні методи.
3. Для моделювання зовнішніх характеристик СЕ необхідно знайти сім параметрів $(I_{PV}, I_{01}, I_{02}, \alpha_1, \alpha_2, R_S, R_p)$ у випадку двохдіодної схеми заміщення та п'ять параметрів $(I_{PV}, I_{01}, \alpha_1, R_S, R_p)$ у випадку однодіодної.

Сучасні методи визначення параметрів еквівалентної схеми заміщення використовуються дані, які надають виробники СП. Зазвичай це табличні дані та характеристики, які отримані за стандартних тестових умов (температура панелі 25 °С, спектр випромінювання AM1.5G та щільність потоку випромінювання 1000 Вт/м²). Проте лише табличних даних не достатньо для визначення всіх необхідних параметрів, тому доволі часто використовують також ВАХ. Окрім того вдаються до спрощення схеми заміщення або ж її математичної моделі.

Серед параметрів еквівалентної схеми заміщення особливе місце посідає коефіцієнт ідеальності α , який значною мірою впливає на точність

апроксимації ВАХ СЕ. Для оцінки значення даного параметру існують наступні методи:

1. Чисельні методи [77].
2. Аналітичні методи, серед яких поширення набуло використання W-функції Ламберта [78].
3. Графічний метод або іноді метод прямого вимірювання.

Коефіцієнт ідеальності можна оцінити безпосередньо за кривою ВАХ шляхом обчислення нахилу лінійних ділянок темної кривої ВАХ у напівлогарифмічній шкалі:

$$\alpha = \frac{1}{V_T} \frac{dV}{d \ln(I)}$$

У загальному випадку величина коефіцієнта ідеальності залежить від вихідної напруги СЕ. Перевагою графічного методу є те, що він дозволяє встановити дану залежність [79]. Також даний підхід дозволяє мінімізувати вплив паразитних опорів та при визначенні значення коефіцієнта ідеальності. Однак виробники СП не надають темної ВАХ, тому цей метод не є поширеним.

Перевагою аналітичних методів є їх досить точна оцінка величини коефіцієнта ідеальності, але ці методи досить складні в комп'ютерній реалізації і мають велику обчислювальну складність.

У зв'язку з перерахованими вище недоліками найбільшого поширення здобули саме чисельні методи визначення значення коефіцієнту ідеальності.

Перехід від математичної моделі та схеми заміщення окремого СЕ до СП є ключовим етапом у розумінні та оптимізації СЕС. Основою цього переходу є розширення простої моделі одного елементу до більш складної системи, яка включає багато подібних елементів.

Цей перехід від одиничних елементів до модуля є важливим для проектування та аналізу реальних сонячних панелей, оскільки він дозволяє точно передбачати поведінку системи під час експлуатації. Завдяки цьому,

можна ефективно оптимізувати конструкцію та управління СП для досягнення максимальної продуктивності.

Еквівалентна схема заміщення СП залишається незмінною і повністю дублює СЕ (рис. 1.22). Враховуючи, що СП складається з послідовного з'єднання певної кількості СЕ, для отримання математичної моделі СП в рівнянні (1.1) необхідно визначати теплову напругу наступним чином [80]:

$$V_T = N_s kT/q,$$

де N_s – кількість послідовно з'єднаних СЕ.

Хмарний покрив та його моделювання мають вирішальне значення при розробці та експлуатації СП. Він характеризується високою динамікою та непередбачуваністю. Це створює складнощі у точному прогнозуванні кількості сонячного випромінювання, що буде доступною для СЕС. Моделі хмарного покриву допомагають в аналізі цих динамічних умов і дозволяють більш точно оцінювати потенційну продуктивність системи.

1.6. Математична модель хмарного покриву

Для кількісної оцінки впливу хмарного покриву на падаюче на СП сонячне випромінювання використовують оптичну товщину хмарного покриву τ , яка визначається наступним чином:

$$\tau = \ln \frac{E_{e,i}}{E_{e,t}} = \ln T, \quad (1.2)$$

де $E_{e,i}$ – щільність потоку падаючого на поверхню випромінювання, $E_{e,t}$ – щільність потоку випромінювання, яке пройшло через поверхню, T – прозорість поверхні.

На практиці даний параметр вимірюється експериментально за допомогою наземного обладнання або супутників [81]–[83]. Недоліком використання експериментально отриманих даних є низька частота вимірів, що не дозволяє проводити моделювання роботи масиву СП у режимі реального

часу. У зв'язку з цим найбільшого поширення набув підхід генерації хмарного покриття на основі математичної моделі хмарного покриття, яка описує оптичну товщину хмарного покриття [84], [85].

Як і більшість природних процесів, зображення хмарного покриття має характеристику спектральної густини потужності $S(f)$ (СГП) з законом розподілу [86]:

$$S(f) \propto c/f^{\alpha m}, \quad (1.3)$$

де m – розмірність простору, f – норма m -вимірного частотного вектору (f_1, \dots, f_m) , $\alpha = 1 \dots 2$ – параметр, який визначає колір процесу (шуму), стала пропорційності.

В m -вимірному просторі норма частотного вектору f визначається наступним чином:

$$f = \sqrt{\sum_{i=1}^m f_i^2}, \quad (1.4)$$

де f_i – частотна складова вздовж i -ої осі.

В залежності від значення параметру α визначають наступні кольори шумів:

1. При $\alpha \approx 1$ рожевий.
2. При $\alpha \approx 2$ червоний або коричневий.

Через характер частотної характеристики даний процес отримав назву $1/f$ шум. Подібні процеси зустрічаються в електроніці [87], біології [88] та багатьох інших сферах людської діяльності. У зв'язку з цим, проведено чимало досліджень з метою аналізу їх природи, а також створення математичної моделі [89].

Варто відзначити, що поняття шум досить часто застосовується замість термінів випадковий процес або сигнал, тому в подальших викладках дані визначення будуть вважатися взаємозамінними. Дано визначення терміну шум [90].

Шум – це випадковий процес, який характеризується наступними властивостями:

1. Стаціонарність – відсутність залежності статистичних параметрів даного процесу (такі як середнє значення та дисперсія) від часу. Це спрощує аналіз, оскільки властивості процесу залишаються постійними.

2. Нормальний закон розподілу – в будь-який момент часу процес відповідає нормальному розподілу, який характеризується середнім значенням і стандартним відхиленням.

3. Можливість керування спектральною щільністю. Спектр потужності сигналу відображає, як його енергія розподіляється між різними частотами. Керуючи спектром потужності, можна керувати розподілом енергії на різних частотних компонентах шуму. Це може бути важливо в різних програмах, включаючи обробку сигналів і системи зв'язку.

4. Обмеженість діапазону спектру. Дана умова визначає, що шум має ненульові складові спектру лише в межах певного частотного діапазону або смуги пропускання. Це поширене явище в багатьох сценаріях реального світу, коли шум виникає в певних частотних межах.

Здатність керуванням спектральною щільністю дозволяє маніпулювати розподілом енергії на різних частотних компонентах шуму. Це може бути важливо для відтворення властивостей реальних процесів і явищ. Це може бути досягнуто напряду або ж за допомогою підсумовування незалежних нормованих шумів.

Існує чимала кількість методів генерації шумів, класифікація яких наведена на рис. 1.23.

Решітчасті методи отримали свою назву через використання сітки випадкових значень у процесі своєї роботи, на основі якої генеруються значення шуму в проміжних точках між вузлами сітки [90].

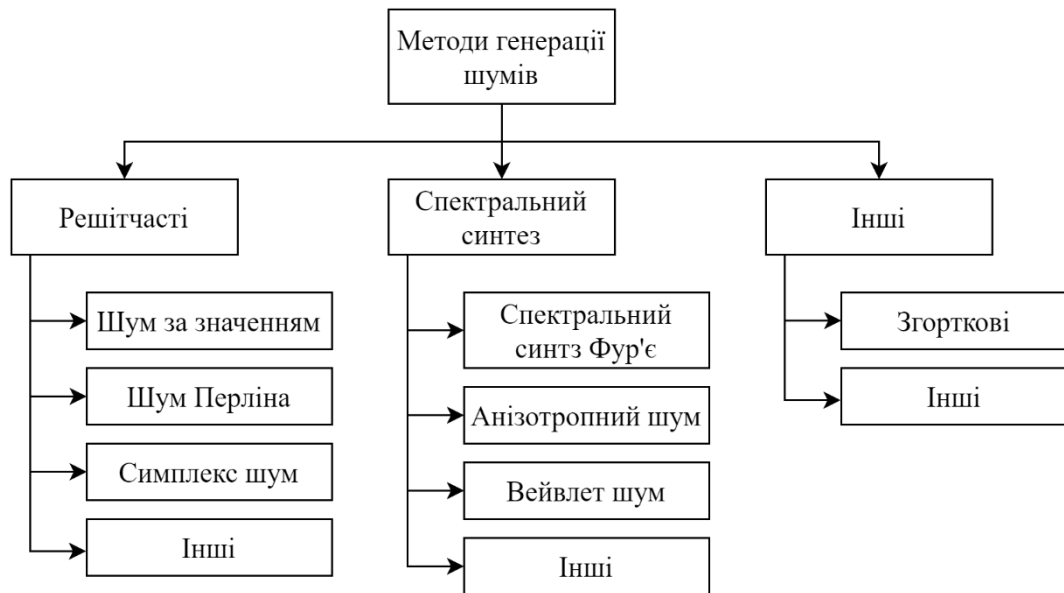


Рис. 1.23. Класифікація методів генерації шумів

Шум за значенням (надалі використовуватиметься англomовний термін Value Noise) є історично першим процедурним методом генерації шумів. Розглянемо алгоритм його роботи. Першим кроком є генерації решітки з випадковими значеннями, після чого відбувається інтерполяція значень між вузлами (рис. 1.24). Для цього можуть використовуватися різноманітні способи, проте зазвичай використовується поліноміальна інтерполяція порядку якої не перевищує п'ятого ступеню [91].

В дискретному випадку, який набув найбільшого поширення в комп'ютерному моделюванні, для генерації шуму необхідно визначити кінцевий розмір шуму N та кількість проміжних значень n , які визначаються за допомогою інтерполяції між вузлами з псевдовипадковими значеннями. При цьому кількість випадкових значень сітки s , які необхідно згенерувати, визначається наступним чином:

$$s = \left\lfloor \frac{N}{n} \right\rfloor + 1$$

Зазвичай значення s має бути в діапазоні 5...15.

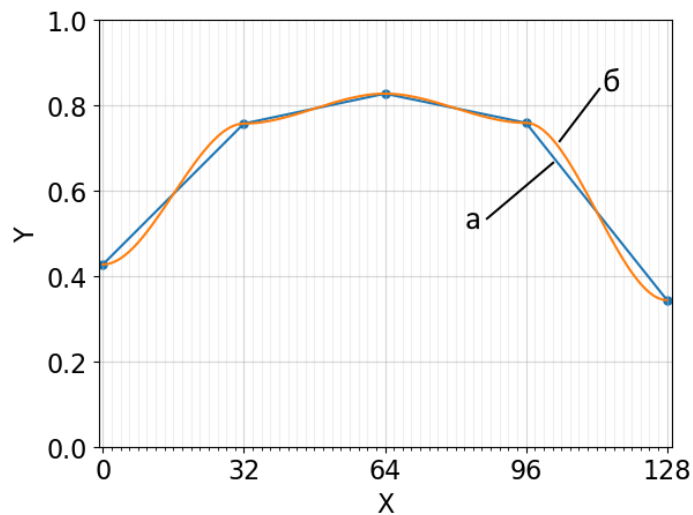


Рис. 1.24. Графік прикладу лінійної (а) та кубічної (б) інтерполяції для шуму з параметрами $N = 128$, $n = 16$ та $s = 9$

Для надання більш реалістичного зовнішнього вигляду згенерованому шуму і можливості керування спектральними характеристиками використовують метод з використанням октав [92]. Суть методу полягає в генерації шуму, який є суперпозицією певної кількості d октав з нормованим значенням амплітуди. В даному контексті під октавою мається на увазі складова результуючого шуму, частота якої вдвічі більша за частоту попередньої складової.

Для реалізації можливості керування спектром згенерованого шуму необхідно створити закон зміни амплітуди в залежності від порядкового номеру октави, який відповідає СГП типу шуму, який необхідно отримати. Наприклад, для одновимірного ($m = 1$) дискретного $1/f$ шуму залежність СГП $S(k)$ від порядкового номеру частотної складової описується виразом:

$$S(k) = \frac{1}{k^\alpha}, \quad (1.5)$$

де k – порядковий номер частотної складової у спектрі, α – параметр, який визначає колір шуму.

З іншого боку СГП визначається наступним чином:

$$S(k) = |X(k)|^2, \quad (1.6)$$

де $X(k)$ – спектр дискретного сигналу $x(t)$.

З виразів (1.5) та (1.6) маємо наступний вираз для амплітудно-частотної характеристики (АЧХ) $1/f$ шуму:

$$|X(k)| = \frac{1}{\sqrt{k^\alpha}}$$

Для октави з довільним порядковим номером k_i маємо:

$$|X(k_i)| = \frac{1}{\sqrt{k_i^\alpha}}$$

Враховуючи визначення октави, для октави з порядковим номером k_{i+1} :

$$|X(k_{i+1})| = \frac{1}{\sqrt{k_{i+1}^\alpha}} = \frac{1}{\sqrt{2k_i^\alpha}}$$

Звідки, відношення амплітуд двох сусідніх октав для $1/f$ шуму:

$$c = \frac{|X(k_{i+1})|}{|X(k_i)|} = \frac{\sqrt{k_i^\alpha}}{\sqrt{2k_i^\alpha}} = \frac{1}{\sqrt{2}} \quad (1.7)$$

Таким чином, для одновимірного $1/f$ шуму амплітуда кожної наступної октави менша в $\sqrt{2}$ разів за амплітуду попередньої октави. Аналогічним чином можна отримати відношення амплітуд двох сусідніх октав для шумів, які мають інші закони розподілу спектральної густини потужності.

Також вираз (1.7) не залежить від розмірності простору, тому для рожевого шуму довільної розмірності залежність амплітуд сусідніх октав залишається незмінною.

На рис. 1.25 наведено графік прикладу з використанням трьох октав (а-в) та результуючого шуму (г), який є їх сумою. В даному випадку використовувалася інтерполяція кубічним поліномом.

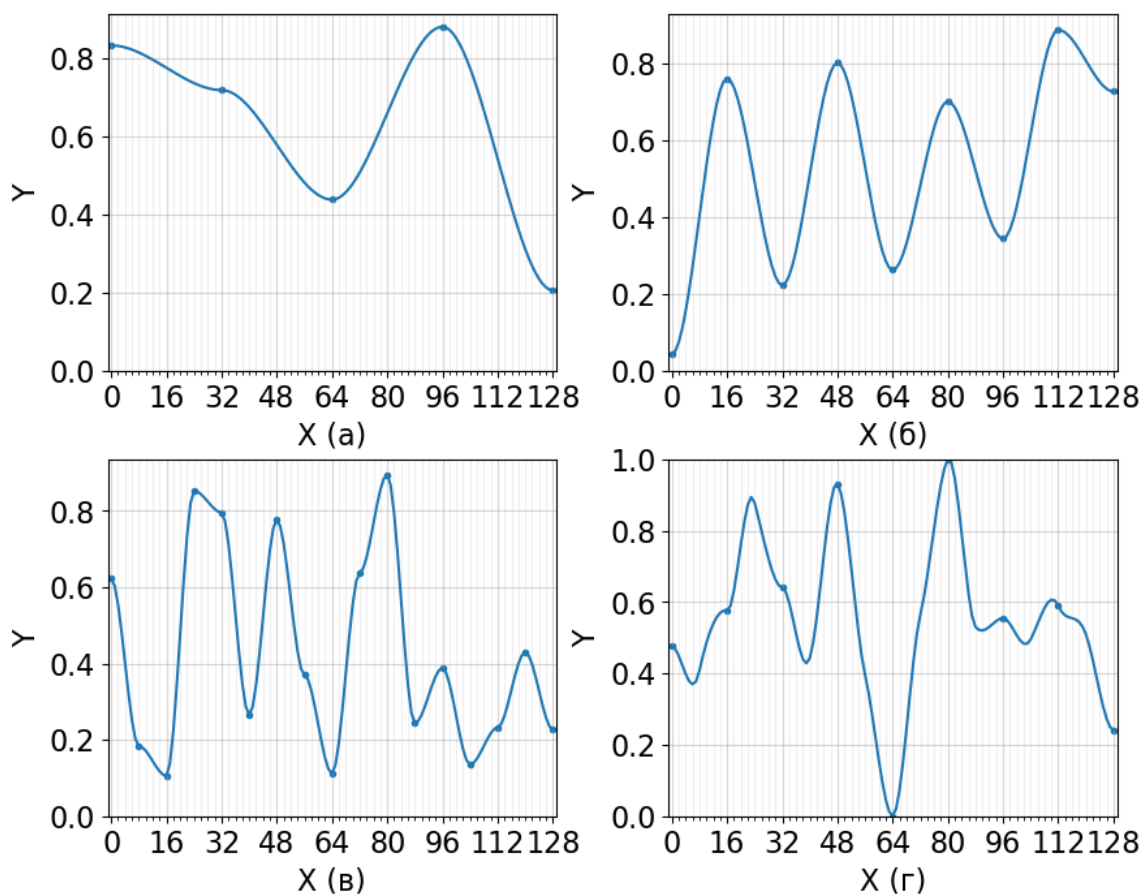


Рис. 1.25. Графік прикладу методу генерації шуму з використанням октав з параметрами $N = 128$, $n = 32$ та $c = \sqrt{2}$: октави (а-в) та результуючий шум (г)

Аналіз впливу використання описаного методу на результуючий шум доволі часто проводять графічним способом, порівнюючи АЧХ спектру та його огибаючої в логарифмічному масштабі по обох осях. При цьому, перед виконанням подібного порівняння, з шуму необхідно виключити постійну складову для покращення наочності графіку.

На рис. 1.26 продемонстровано вплив використання октав на спектр. У звичайного шуму (рис. 1.26, а) основна потужність припадає на низькочастотні складові, тоді як у шумі з використанням трьох складових присутня більша кількість високочастотних гармонік (рис. 1.26, б). Збільшення кількості гармонік дозволяє покращити відповідність заданій спектральній характеристиці, проте це негативним чином впливає на швидкодію [92].

Варто зауважити, що перше й останнє значення згенерованого шуму з великою вірогідністю матимуть різні значення, що при використанні

перетворення Фур'є призводить до появи високочастотних складових у спектрі. Дане явище отримало назву витікання спектру і для зменшення його впливу використовують віконне перетворення Фур'є [93]. Повністю усунути даний ефект можна за рахунок використання алгоритму генерації шуму, який забезпечуватиме рівність значень на початку та наприкінці сигналу.

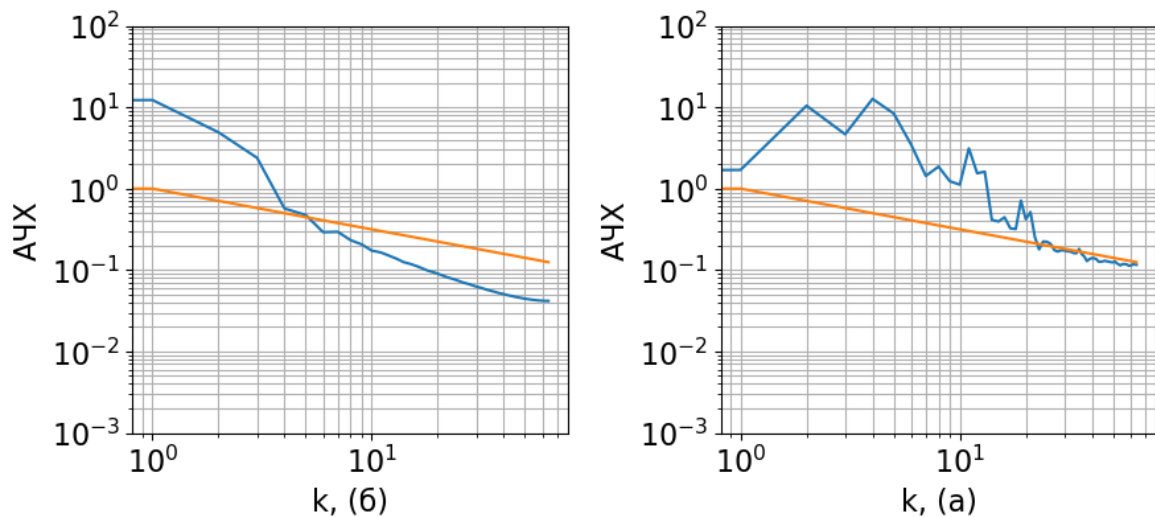


Рис. 1.26. Графіки СГП звичайного шуму (а) та шуму, який складається з трьох октав (б)

Алгоритм для двовимірного випадку ($m = 2$) складається з аналогічних кроків, проте інтерполяція здійснюється послідовно по кожній з осей. Для спрощення викладок приймемо, що двовимірний шум має однаковий розмір по обох осях.

Приклад роботи двовимірного випадку зображено на рис. 1.27. Очевидно, що використання октав (рис. 1.27, б) у двовимірному випадку дозволяє створити більш реалістичне зображення хмарного покриву, тоді як простий шум має недоліки у виді осередків одної інтенсивності з квадратними межами. Важливо зазначити, що в комп'ютерних науках при роботі зі зображеннями початок координат лежить у верхній лівій частині зображення, з позитивними напрямками осей як на рис. 1.27, а.

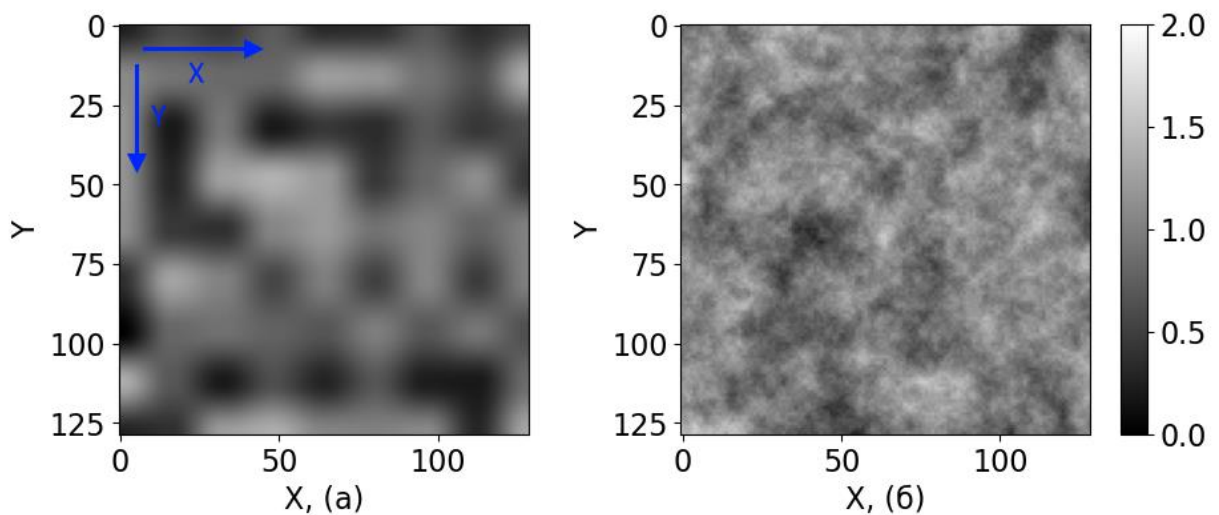


Рис. 1.27. Ілюстрації двовимірного шуму з параметрами $N = 128$, $n = 16$, $c = \sqrt{2}$ та кубічною інтерполяцією: без використання октав (а), з п'ятьма октавами (б)

Підсумовуючи, Value Noise характеризується наступними переваги [94]:

1. Простота реалізації.
2. Можливість керування спектром шуму при використанні октав.

Та наступні недоліки:

1. Наявність осередків однакової інтенсивності у згенерованому шумі з квадратними межами.

2. Для двовимірного випадку часова складність алгоритму Value Noise становить $O(dN^2)$, що з точки зору теорії алгоритмів вважається недоліком. Варто зауважити, що в даному випадку кількість октав d є вхідним параметром алгоритму, а не константою, тому складність алгоритму не може розглядатись як $O(N^2)$.

3. Керування спектром реалізовується за рахунок використання октав, що збільшує часову складність алгоритму.

Для виключення осередків з квадратними межами використовується модифікований алгоритм – шум Перліна, який має ряд покращень [95]. Основною відмінністю даного методу від Value Noise є те, що на сітці зі сталим кроком визначаються випадкові значення, які визначають значення градієнту.

Градiєнт g – векторна величина, яка вказує напрям найшвидшого зростання функції. В одновимірному випадку градієнт еквівалентний геометричному значенню похідної, тобто він визначає кут нахилу між дотичною в деякій точці та позитивним напрямом осі абсцис (рис. 1.28, штрихова лінія).

Після визначення випадкових значень градієнту відбувається інтерполяція проміжних значень між вузлами сітки (рис. 1.28, суцільна лінія). Іншою відмінністю шуму Перліна є встановлення формальних вимог до функції шуму, а відповідно й до інтерполяційних функцій, що має на меті покращити якість інтерполяції та створити більш реалістичні зображення.

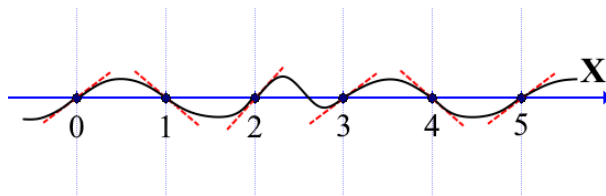


Рис. 1.28. Графік одновимірного шуму Перліна

До формальних вимог відносяться:

1. Перша похідна від функції шуму має бути неперервною. Ця умова гарантує безперервність функції шуму в точках сітки. По суті, ця вимога гарантує, що інтерполяція не створює раптових розривів у точках сітки.
2. Рекомендовано, щоб функція результуючого шуму мала неперервну похідну другого порядку, що дозволить створити більш плавний перехід між інтерполяційними поліномами у вузлах сітки.

Для задоволення даних вимог можуть використовуватися різноманітні функції, проте найбільшого поширення здобула поліноміальна функція плавного переходу (smoothstep) різних порядків (рис. 1.29):

1. Третього $s_3(t) = -2t^3 + 3t^2$, який задовольняє першу вимогу.
2. П'ятого $s_5(t) = 6t^5 - 15t^4 + 10t^3$, який задовольняє обидві вимоги.

Використання функції плавного переходу вищих порядків забезпечить більшу плавність переходу у вузлах сітки (рис. 1.29, б). Проте недоліком даного

рішення є те, що на кожній ітерації алгоритму доведеться розраховувати значення поліному вищого порядку, що в результаті знижує швидкість виконання алгоритму.

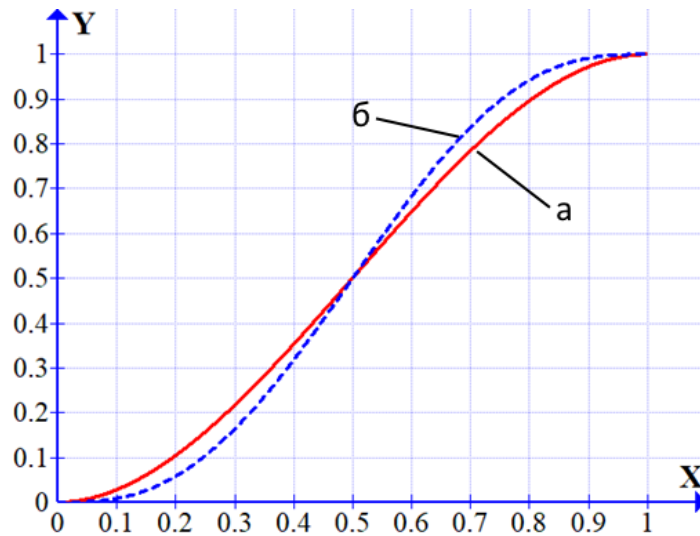


Рис. 1.29. Приклади графіків функції плавного переходу: третього (а) та п'ятого (б) порядків

У двовимірному випадку алгоритм також розпочинається зі задання випадкових значень у вузлах сітки (рис. 1.30). Після цього визначається зважена сума впливу градієнтів на кінцеве значення шуму в певній точці P за допомогою скалярного добутку градієнтних векторів g з відповідними векторами відстані від точок сітки до точки P :

$$\begin{aligned} n_{00} &= g_{00} \cdot \begin{pmatrix} u \\ v \end{pmatrix}, \\ n_{01} &= g_{01} \cdot \begin{pmatrix} u-1 \\ v \end{pmatrix}, \\ n_{10} &= g_{10} \cdot \begin{pmatrix} u \\ v-1 \end{pmatrix}, \\ n_{11} &= g_{11} \cdot \begin{pmatrix} u-1 \\ v-1 \end{pmatrix} \end{aligned}$$

Щоб функція шуму була відтворюваною, тобто завжди давала однакове значення для даної вхідної точки, градієнти мають бути псевдовипадковими. Хорошим вибором для двовимірного випадку є використання у якості

градієнтів одиничної довжини (кількістю від 8 до 16), рівномірно розподілених по одиничному колу [96]. Також у якості значень градієнтів можуть використовуватися ортонормовані набори функцій, наприклад лінійні функції на базі функцій Уолша [5] та інші.

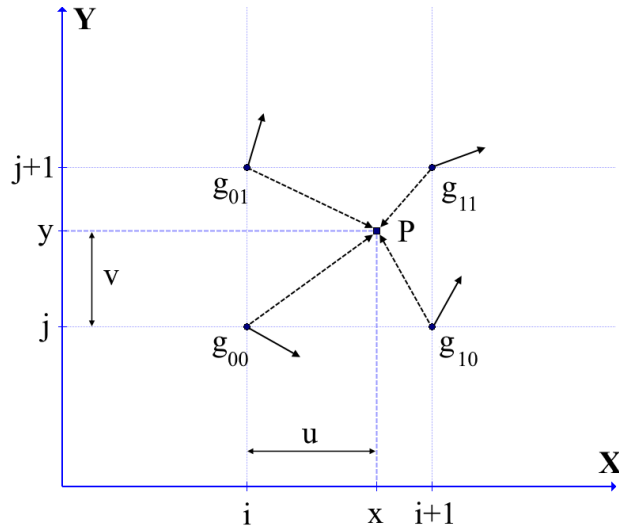


Рис. 1.30. Зображення двовимірної сітки та довільно обраної точки P

Далі проводиться інтерполяція за допомогою розглянутої раніше функції плавного переходу $s(t)$. Це робиться послідовно по кожній з осей в довільному порядку. Наприклад, спершу по осі абсцис:

$$n_{x0} = n_{00}(1 - s(u)) + n_{10}s(u),$$

$$n_{x1} = n_{01}(1 - s(u)) + n_{11}s(u)$$

Після цього по осі ординат, що в результаті дає кінцеве значення шуму в точці P :

$$n_{xy} = n_{x0}(1 - s(v)) + n_{x1}s(v)$$

Аналогічно Value Noise, шум Перліна може використовувати октави для створення більш реалістичного зображення хмарного покриву [97]. Приклад двовимірного випадку шуму Перліну зображено на рис. 1.31.

Шум Перліна характеризується наступними перевагами [97]:

1. Відсутність осередків однакової інтенсивності з прямокутними межами.
2. Можливість керування спектром шуму при використанні октав.

Та недоліками:

1. Для двовимірного випадку часова складність складає $O(dN^2)$.
2. Використання інтерполяційного поліному п'ятого порядку негативно впливає на швидкодію.

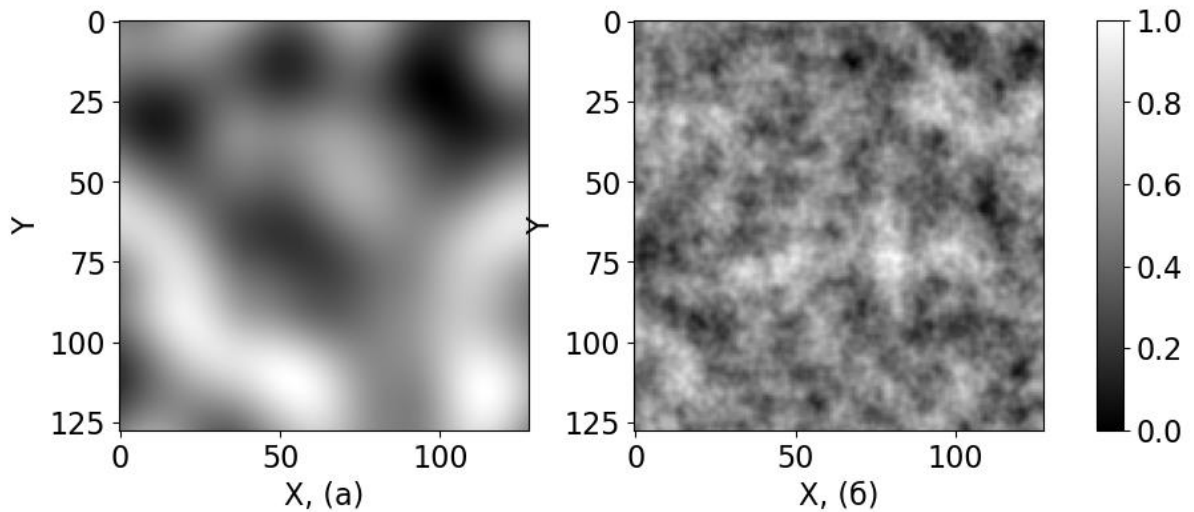


Рис. 1.31. Ілюстрація двовимірного шуму Перліна з параметрами $N = 128$, $n = 16$ та $c = \sqrt{2}$: без використання октав (а), з п'ятьма октавами (б)

Основним недоліком алгоритму генерації шуму Перліна є його низька швидкодія для великої розмірності простору m . У зв'язку з цим було розроблено чималу кількість модифікацій даного методу для покращення його швидкості виконання. Одним з таких методів є Симплекс метод. Основною його відмінністю є перехід до трикутної сітки та ряд спрощень в розрахунках, що в результаті дозволило зменшити складність алгоритму до $O(m^2)$, тоді як складність шуму Перліну від розмірності простору становить $O(2^m)$.

Іншим класом методів генерації шумів є методи спектрального синтезу, які засновані на використанні різноманітних інтегральних перетворень в процесі своєї роботи. Це, в свою чергу, дає можливість напряду оперувати зі спектральними характеристиками вихідного сигналу [98], [99].

Найбільшого поширення здобули методи засновані на використанні перетворення Фур'є. Розглянемо алгоритм роботи даного методу в одновимірному випадку.

Як і у випадку інших методів спектрального синтезу спершу генерується білий шум r у дійсній області необхідного розміру N , тобто шум з нормальним законом розподілу. Зазвичай прийнято використовувати шум з нульовим середнім $\mu = 0$ та одиничною дисперсією $\sigma^2 = 1$, який ще називають гаусівським, проте в залежності від потреб дані статистичні параметри можуть приймати інші значення.

Далі знаходиться спектр шуму за допомогою перетворювання Фур'є:

$$R = X + jY,$$

де X, Y – випадкові величини з нормальним законом розподілу.

У випадку використання рекомендованих значень моментів, частотне зображення білого шуму є також білим шумом. Тому в даному випадку можна упустити етап знаходження перетворення Фур'є і генерувати білий шум відразу в частотній області. Це дозволяє покращити швидкодію алгоритму, проте призводить до появи симетрії у вихідному шумі.

Також варто зауважити, що для сигналів з малим розміром N спектр може зазнавати суттєвих змін із-за явища витікання спектру [100]. В контексті швидкого перетворення Фур'є (ШПФ) це явище в основному спричинене кінцевою довжиною сигналу N та різними значеннями сигналу на кінцях інтервалу його визначення, що призводить до різких переходів (рис. 1.32, а). Оскільки перетворення Фур'є працює з періодичними функціями, для відтворення подібного різкого переходу необхідні високочастотні гармоніки. В результаті дане явище спричиняє появу високочастотних складових у спектрі (рис. 1.32, б). Щоб уникнути або мінімізувати явище витікання спектру під час ШПФ, використовуються різноманітні способи, такі як [101]:

1. Використання віконних функцій. Віконна функція – це математична функція, яка плавно модулює значення сигналу до нуля на його

краях, зменшуючи різкі переходи. До популярних функцій вікон належать вікна Хеммінга, Ханнінга, Блекмана, Кайзера та інші. Вибір даної функції залежить від конкретних вимог, проте їх використання може значно модифікувати спектр сигналу.

2. Доповнення нульовими значеннями. Даний спосіб передбачає додавання нульових значень напочатку та вкінці сигналу або ж дублювання першого значення сигналу наприкінці. Це дозволяє усунути причину появи даного ефекту, але й призводить до небажаної зміни сигналу.

3. Збільшення довжини сигналу N . За рахунок використання більшої кількості точок у часовій області зменшується вплив різких фронтів.

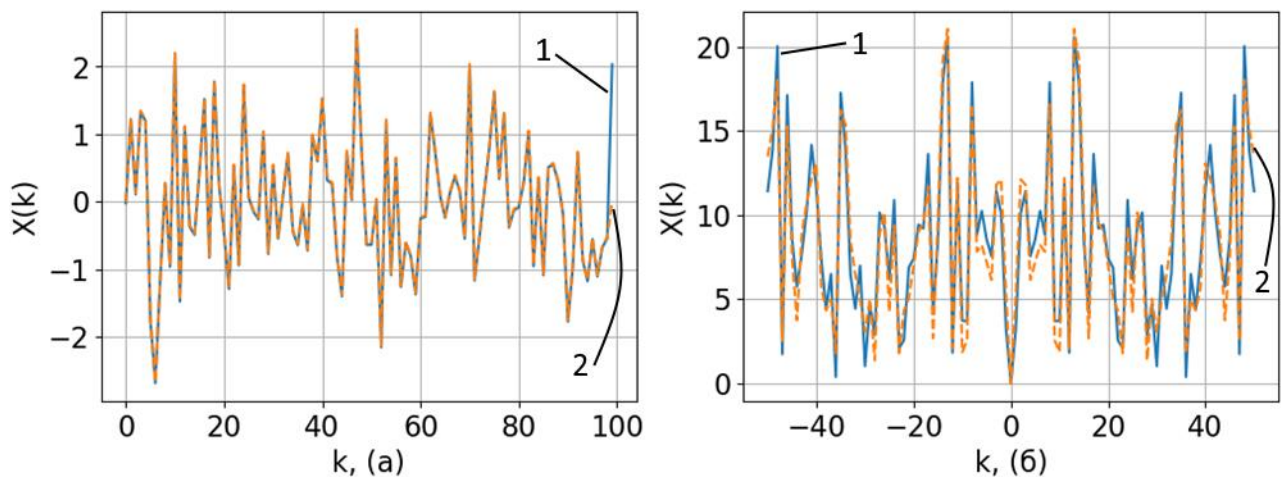


Рис. 1.32. Ілюстрація явища витікання спектру: оригінальний (1) та доповнений (2) сигнали (а) та спектральні характеристики відповідних сигналів (б)

У контексті генерації шумів необхідно отримати бажані спектральні характеристики, тому в даній галузі найбільшого поширення здобули останні два підходи для зменшення впливу витікання спектру.

Після знаходження спектру білого шуму в частотній області задається фільтруюча функція F , яка дозволяє отримати бажану спектральну густину потужності. У якості подібних функцій може використовуватися ряд різноманітних функцій. У випадку дискретного $1/f$ шуму дана функція відповідає кореню з її СГП (1.5):

$$F(k) = \frac{1}{\sqrt{k^{\alpha_m}}}$$

Після чого знаходиться оригінал від добутку значень сітки R на фільтруючу функцію F за допомогою зворотнього ШПФ, що й в результаті дає шум з бажаною спектральною характеристикою:

$$p = F^{-1}[R \cdot F]$$

Приклад одновимірного спектрального синтезу з використанням ШПФ зображено на (рис. 1.33).

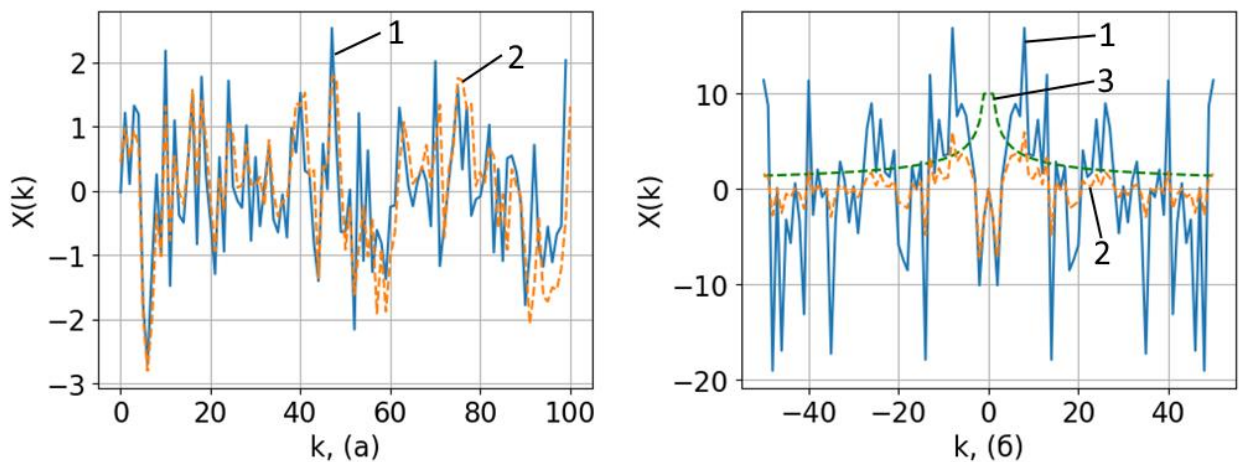


Рис. 1.33. Ілюстрація прикладу спектрального синтезу: вихідний білий (1) та отриманий рожевий (2) шум (а) та їх спектри (б). Фільтруюча функція позначена (3)

У двовимірному випадку задається двовимірна фільтруюча функція F . Якщо дана функція є асиметричною і не є кривою обертання, то це дозволяє генерувати анізотропний шум, параметри якого залежать від напрямку осі [102]. Прикладом подібного фільтру для двовимірного $1/f$ шуму є еліптичний фільтр:

$$F(k) = \frac{1}{k^{\alpha}} = \left(\sqrt{\left(\frac{k_x}{a}\right)^2 + \left(\frac{k_y}{b}\right)^2} \right)^{-\alpha},$$

де a та b – параметри, які визначають еліптичність фільтру.

Очевидно, що при $a = b$ еліптичний фільтр стає симетричним. Діапазон зміни значень даних параметрів визначається вимогою збереження спектральної характеристики $1/f$ уздовж кожної осі.

Зазвичай значення даних параметрів визначаються експериментально для отримання необхідних спектральних характеристик, проте доволі часто лежать в діапазоні $0,5 \dots 1,5$.

Приклад генерації двовимірного випадку спектрального синтезу за допомогою перетворення Фур'є зображено на рис. 1.34.

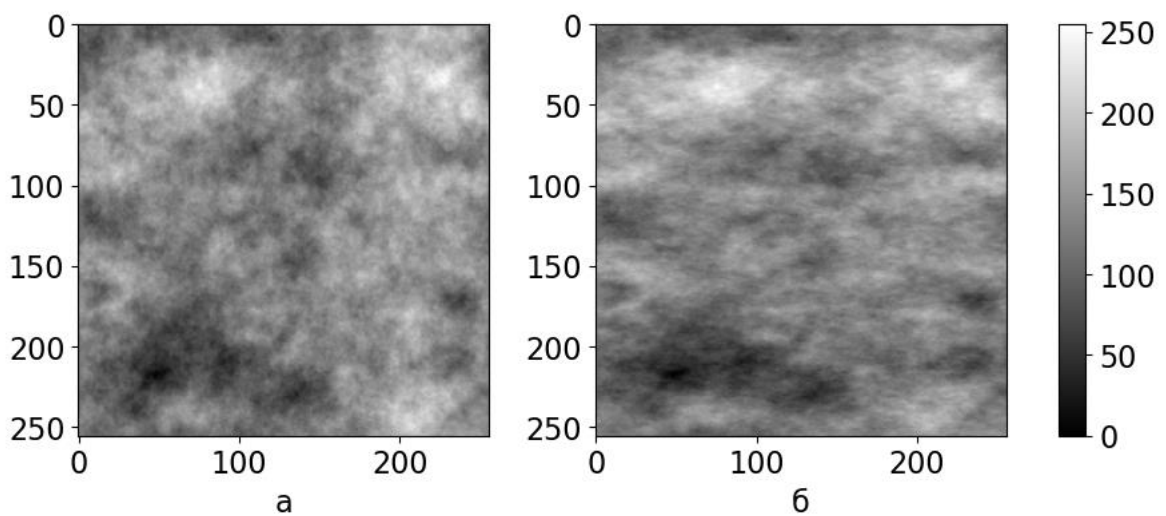


Рис. 1.34. Ілюстрація двовимірного спектрального синтезу з $N = 128$ та використанням симетричного (а) й еліптичного (б) фільтрів з $a = 0,5$

Метод спектрального синтезу на основі використання ШПФ характеризується наступними перевагами у порівнянні з решітковими методами [90]:

1. Відсутність осередків однакової інтенсивності.
2. Можливість керування спектром шуму без використання октав.
3. Для двовимірного випадку часова складність алгоритму генерації складає $O(N^2)$.

У загальному випадку даний метод, як і решіткові, має обмежену спектральну роздільну здатність, тобто кількість гармонік вздовж кожної осі

визначається розміром сигналу N . Це призводить до того, що для заданого розміру N існує обмеження на те, наскільки деталізованим може бути шум.

Окрім того, спектральний синтез у якості вихідних даних використовує випадкову змінну з нормальним розподілом (білий шум). Це створює деякі обмеження при моделюванні природніх явищ, які описуються законами розподілу іншого характеру. Тому для відтворення більшої кількості параметрів реальних процесів необхідно використовувати додаткові перетворення та маніпуляції зі шумом.

Не зважаючи на дані недоліки, метод спектрального синтезу на основі перетворення ШПФ характеризується рядом переваг у порівнянні з решітковими методами, що призвело до значного поширення даного методу для генерації та моделювання зображень хмарного покриву.

Висновки до розділу

1. Аналіз сучасних математичних моделей, топологій, методів побудови та способів максимізації відбору енергії у відновлюваних системах з використанням СП підтверджує визначний вплив наявності ефекту часткового затінення на ефективність та режим роботи сонячної електростанції.

2. Розглянуті параметри, характеристики та еквівалентна схема заміщення сонячного елементу вказують на потребу в розробці удосконаленої нелінійної математичної моделі сонячної панелі для підвищення точності моделювання ВАХ.

3. При дослідженні впливу ефекту часткового затінення на рівень вихідної потужності сонячної електростанції необхідно проводити врахування щільності та особливостей хмарного покриву, що вказує на необхідність розробки математичної моделі хмарного покриву.

4. Враховуючи вплив хмарного покриву на режим роботи та вихідну потужність сонячної електростанції, задача дослідження процесів динамічної комутації сонячних панелей в умовах часткового затінення актуальною є

РОЗДІЛ 2. ВИЗНАЧЕННЯ ПАРАМЕТРІВ ЕКВІВАЛЕНТНОЇ СХЕМИ ЗАМІЩЕННЯ

2.1. Визначення параметрів рівняння вольт-амперних характеристик

Для моделювання зовнішніх характеристик СП необхідно знайти параметри еквівалентної схеми заміщення СП (рис. 1.22).

Виробники СП надають параметри і характеристики своїх виробів, які були отримані за стандартних умов тестування (температура панелі 25 °С, спектр випромінювання AM1.5G та щільність потоку випромінювання 1000 Вт/м²). Зазвичай кількість наданих параметрів є недостатньою для визначення всіх параметрів еквівалентної схеми заміщення, тому більшість наявних методів визначення даних параметрів використовують ряд спрощень математичної моделі, які негативним чином впливають на точність моделі. Окрім того, в якості чисельного методу доволі часто використовується метод Ньютона, який характеризується рядом недоліків [103]:

1. Необхідність визначення похідної на кожній ітерації. Для функцій з великою обчислювальною складністю похідних це може бути суттєвим недоліком. Крім того, у випадках, коли похідна дорівнює нулю або невизначена, метод не може бути використаний.

2. Чутливість до поведінки функції. Збіжність методу може бути повільною або він може бути розбіжним, якщо функція має плоскі області або поводить нерівномірно біля кореня. У таких випадках дотична лінія може неточно відображати поведінку функції на даних ділянках.

3. Чутливість до якості даних. Якщо значення вихідних даних є зашумленими, то збіжність методу Ньютона не є гарантованою.

Альтернативним підходом до визначення параметрів еквівалентної схеми заміщення є використання ВАХ замість введення ряду спрощень в

математичну модель [66], [72]. Виробники СЕ надають всі характеристики у виді готових графіків, а не табличних даних. Тому для їх використання в розрахунках, необхідно провести оцифрування даних графіків, що вносить деяку похибку. Це, в свою чергу, призводить до того, що метод Ньютона не може бути використаний одночасно з оцифрованою ВАХ.

Для можливості застосування запропонованого методу як для одно-, так і для двохдіодної моделі, прийнемо струми діодів VD1, VD2 рівними:

$$I_{01} = I_{02} = I_0 \quad (2.1)$$

З урахуванням даного припущення, для моделювання ВАХ СП необхідно визначити шість параметрів $(I_{PV}, I_0, \alpha_1, \alpha_2, R_S, R_P)$ у випадку двохдіодної схеми заміщення та п'ять параметрів $(I_{PV}, I_0, \alpha_1, R_S, R_P)$ для однодіодної.

Для визначення параметрів еквівалентної схеми заміщення розглянемо рівняння вихідного струму СП (1.1). Враховуючи (2.1), дане рівняння набуде наступного вигляду:

$$I = I_{PV} - \frac{V + IR_S}{R_P} - I_0 \left[e^{\frac{V + IR_S}{\alpha_1 V_T}} + e^{\frac{V + IR_S}{\alpha_2 V_T}} - 2 \right] \quad (2.2)$$

Спершу проведемо аналіз отриманого рівняння поблизу точки КЗ. При доволі малій вихідній напрузі $V \approx 0$ вихідний струм СП має набагато вище значення, ніж зворотній струм $I \gg I_0$. Також ряд досліджень [104]–[106] показує те, що типові значення R_S та R_P для комерційних СЕ лежить в наступних межах:

1. Перше покоління – $R_S = 0,1 \dots 5 \text{ Ом}$, $R_P = 100 \dots 1000 \text{ Ом}$.
2. Друге покоління – $R_S = 1 \dots 10 \text{ Ом}$, $R_P = 100 \dots 1000 \text{ Ом}$.

Важливо зазначити, що ці діапазони є приблизними та можуть змінюватися залежно від конкретної технології в кожному поколінні, ефективності СЕ, матеріалів, що використовуються, і процесу виробництва. Проте для перших двох поколінь СЕ маємо:

$$R_P \gg R_S$$

Враховуючи вищесказане, аргумент експоненти в рівнянні (2.2) при малих значеннях напруги приймає значення близьке до нуля $V + IR_s \sim 0$. Відповідно рівняння (2.2) набуває вигляду:

$$I = I_{PV} - \frac{V + IR_s}{R_p} + O(V^2) = \frac{I_{PV}}{1 + R_s/R_p} - \frac{V}{R_s + R_p} + O(V^2) \quad (2.3)$$

Аналізуючи рівняння (2.3), можна сказати, що поблизу точки КЗ досліджуваний вид ВАХ носить здебільшого лінійний характер.

Проведемо дослідження похідної від вихідного струму поблизу точки КЗ. Диференціюючи (2.2), отримуємо вираз для похідної ВАХ (див. додаток а):

$$\frac{I'}{1 + I'R_s} = -\frac{1}{R_p} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V+IR_s}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V+IR_s}{\alpha_2 V_T}} \right] \quad (2.4)$$

Як було вказано раніше, при малій вихідній напрузі $V + IR_s \sim 0$, тому доданком, пропорційному I_0/V_T , можна знехтувати. Враховуючи це, маємо вираз для похідної поблизу точки КЗ:

$$I'_{SC} = -\frac{1}{R_s + R_p} \quad (2.5)$$

Експериментальні дослідження вказують, що табличне значення струму короткого замикання I_{SC} , яке надають виробники, доволі часто має завищене значення, і воно зазвичай повторює значення фотоструму I_{PV} . Чисельні методи є доволі чутливими до значення початкового наближення, тобто в даному випадку до значення струму короткого замикання I_{SC} . Враховуючи це, необхідно його перевизначити, для підвищення точності моделі. Для цього, підставивши $V = 0$ в (2.2), отримаємо початкове наближення для значення струму КЗ:

$$I_{SC}^n = \frac{I_{PV}}{1 + R_s/R_p} = I(0), \quad (2.6)$$

де $I(0)$ – значення струму КЗ, отримане з оцифрованої ВАХ.

Беручи до уваги отримані вирази, рівняння вихідного струму еквівалентної схеми СП (2.2) поблизу точки КЗ можна записати у вигляді:

$$I = I_{SC}'' + I_{SC}'V + O(V^2) \quad (2.7)$$

З впливає (2.7), що при відносно малому значенні вихідної напруги струм на виході СЕ описується лінійним законом. Враховуючи це, доцільно умовно розділити ВАХ сонячного елементу на три частини (ділянки) з наступними умовними межами:

1. Лінійну або пологую $(0 - 0,65V_{OC})$.
2. Робочу з точкою максимальної потужності $(0,65V_{OC} - 0,95V_{OC})$.
3. Спадну $(0,95V_{OC} - V_{OC})$.

Приклад подібного поділу ВАХ СЕ зображено на рис. 2.1.

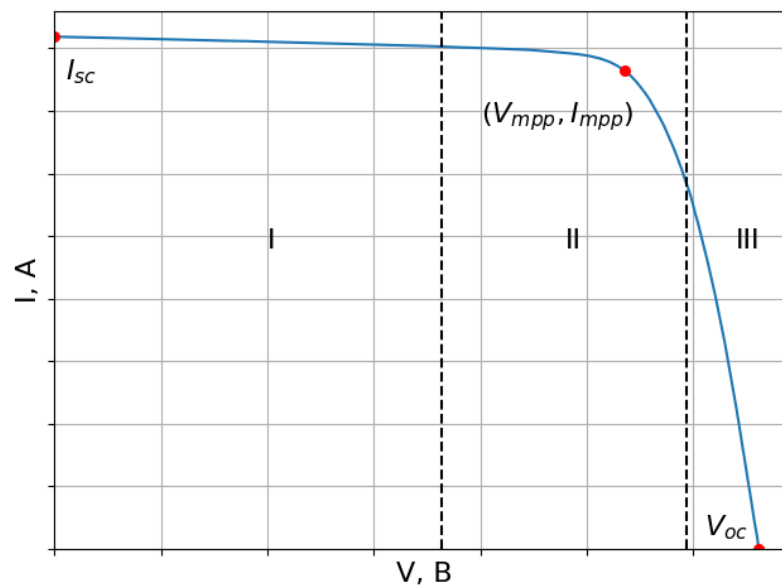


Рис. 2.1. Графік поділу ВАХ СЕ: I – лінійна частина, II – робоча частина, III – спадна частина

На основі отриманих виразів (2.5) та (2.6), маємо наступну систему рівнянь:

$$\begin{cases} I(0) = \frac{I_{PV}}{1 + R_S/R_P}, \\ I_{SC}' = -\frac{1}{R_S + R_P}. \end{cases} \quad (2.8)$$

Вирішивши дану систему рівнянь відносно опорів, отримуємо вираз для послідовного опору (див. додаток а):

$$R_s = -\frac{1}{I'_{SC}} \left(1 - \frac{I(0)}{I_{PV}} \right) \quad (2.9)$$

Та шунтуючого опору:

$$R_p = -\frac{I(0)}{I'_{SC} I_{PV}} \quad (2.10)$$

Перейдемо до аналізу (2.2) поблизу точки ХХ. Підставивши значення напруги і струму ХХ у даний вираз, маємо:

$$0 = I_{PV} - \frac{V_{OC}}{R_p} - I_0 \left[e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2 \right]$$

Звідки отримаємо вираз для зворотного струму діодів VD1, VD2:

$$I_0 = \frac{I_{PV} - \frac{V_{OC}}{R_p}}{e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2} \quad (2.11)$$

Підставивши значення напруги ХХ V_{OC} в (2.4), знайдемо похідну виразу (2.2) поблизу точки ХХ:

$$\frac{I'_{OC}}{1 + I'_{OC} R_s} = -\frac{1}{R_p} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V_{OC}}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V_{OC}}{\alpha_2 V_T}} \right] \quad (2.12)$$

Підставивши (2.11) в (2.12) і вирішивши відносно фотоструму, отримаємо наступний вираз (див. додаток а):

$$I_{PV} = \frac{I(0) I'_{SC} I'_{OC} V_T}{[I'_{SC} V_T - \beta I(0) - \beta I'_{SC} V_{OC}] [I'_{SC} + I'_{OC}]} - \frac{I(0) I'_{OC}}{I'_{SC} - I'_{OC}}, \quad (2.13)$$

де $\beta = \left(e^{V_{OC}/\alpha_1 V_T} / \alpha_1 + e^{V_{OC}/\alpha_2 V_T} / \alpha_2 \right) / \left(e^{V_{OC}/\alpha_1 V_T} + e^{V_{OC}/\alpha_2 V_T} - 2 \right)$.

Для моделювання ВАХ СП залишилося визначити коефіцієнт ідеальності α , який серед параметрів еквівалентної схеми заміщення посідає особливе місце, оскільки він значною мірою впливає на точність апроксимації ВАХ. Для його визначення найбільшого поширення здобули саме чисельні

методи. Таким чином для моделювання ВАХ СП були визначенні необхідні параметри (I_{PV}, I_0, R_S, R_P) , які залежать від табличних параметрів за стандартних тестових умов та оцифрованої ВАХ.

2.2. Спосіб перебору значень коефіцієнтів ідеальності

На відміну від інших параметрів моделі, значення коефіцієнту ідеальності α лежить у діапазоні 1...2 [107]. Також деякі досліджень свідчать, що для перших двох поколінь СЕ значення коефіцієнтів ідеальності приймають типові значення, а саме $\alpha_1 = 1$ та $\alpha_2 = 1,2$ [108].

Враховуючи вищесказане, для визначення значень параметрів схеми заміщення розроблено наступні два підходи [8], [9]:

1. Спосіб перебору значень коефіцієнтів ідеальності (ПЗКІ).
2. Спосіб перебору значень опорів (ПЗО).

Надалі результати, наведені для ПЗО та ПЗКІ, отримані за допомогою програмної реалізації запропонованої математичної моделі (див. додаток б). Спершу розглянемо ПЗКІ, блок-схема алгоритму якого зображена на рис. 2.2. Його суть полягає у наступному:

1. Перебір значень коефіцієнтів ідеальності, який включає систематичний перебір значень коефіцієнтів ідеальності α_1 та α_2 у діапазоні значень 1...2. Коефіцієнти ідеальності визначають, наскільки реальна поведінка сонячного елемента відрізняється від ідеальної. При необхідності використання однодіодної моделі СП виконується граничний перехід $\alpha_2 \rightarrow \infty$.

2. Визначення значення параметрів схеми заміщення на кожній ітерації за допомогою раніше встановлених залежностей. Одночасно з цим визначається похибка апроксимації між оцифрованою та побудованою ВАХ за допомогою отриманих значень параметрів.

3. Результатом перебору є визначення оптимальних значень параметрів схеми заміщення, які максимально точно апроксимують ВАХ СП.

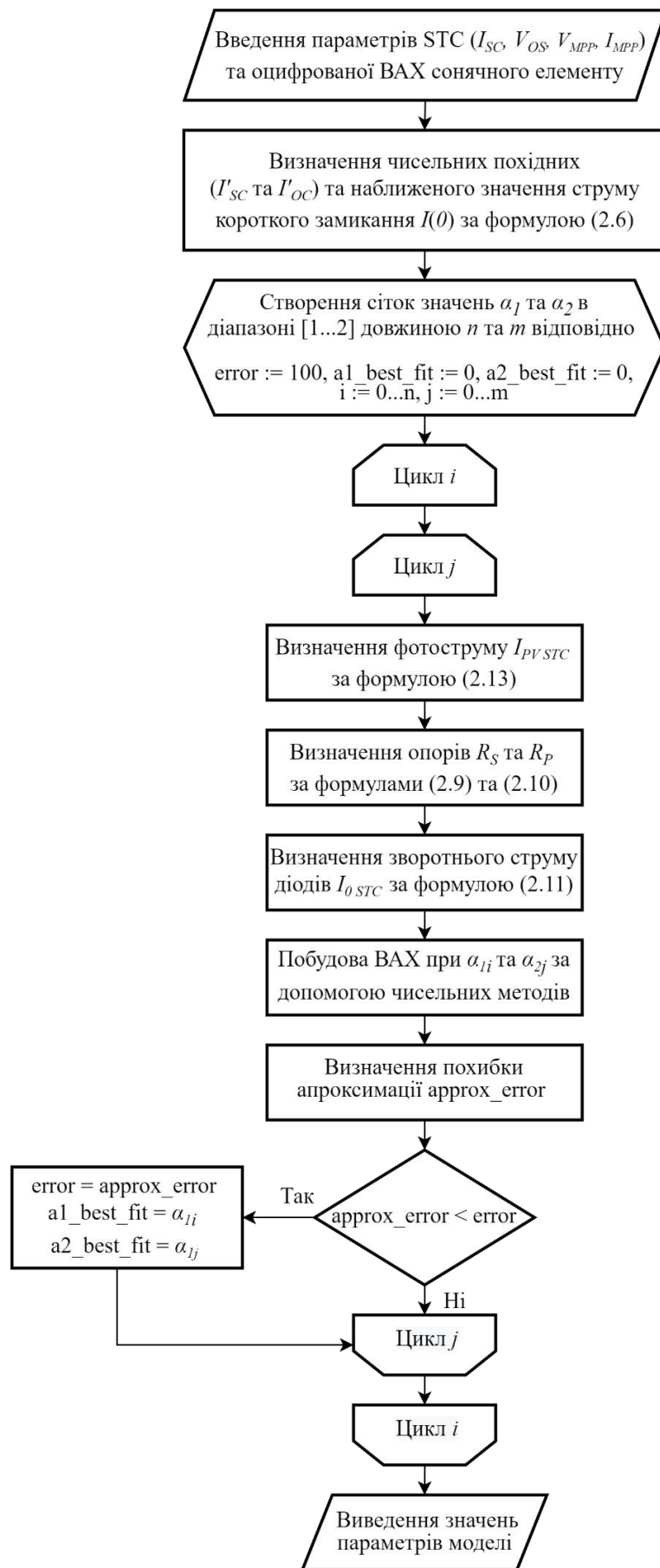


Рис. 2.2. Блок-схема алгоритму ПЗКІ

У якості вихідних даних для алгоритму використовується оцифрована ВАХ, а також табличні дані СЕ або СП. При процесі оцифрування особливу увагу необхідно приділити визначенню значення струму КЗ та напруги ХХ, оскільки дані точки відіграють особливо важливу роль при визначенні значень параметрів моделі.

Наступним кроком є визначення значення похідних в лінійній та спадній областях, проте у процесі оцифрування ВАХ вноситься деяка похибка (рис. 2.3, суцільна крива), яка в кінцевому підсумку впливає на точність визначення значень похідних. Для компенсації даного негативного ефекту необхідно проводити усереднення, яке забезпечується за рахунок дотримання деяких правил визначення числових похідних. Експериментальним шляхом було встановлено наступне:

1. Для знаходження похідної в лінійній області достатньо брати перші 20-25% точок оцифрованої ВАХ і проводити лінійну апроксимацію. На прикладі СП КС200GT поліном визначений даним способом $y = -0.00824x + 8.18787$ (рис. 2.3, штрихова лінія).

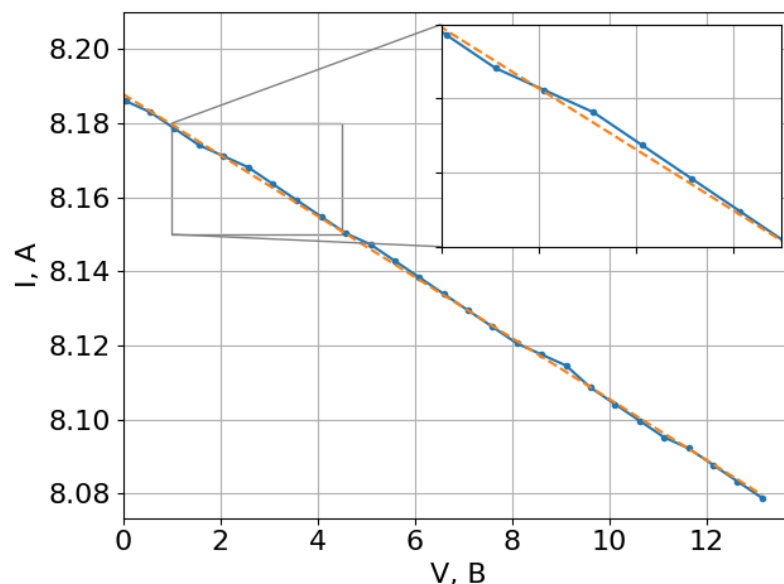


Рис. 2.3. Графік прикладу знаходження похідної в лінійній ділянці ВАХ: оцифрована ВАХ (суцільна лінія), лінійна апроксимація (штрихова)

2. Оскільки ВАХ в спадній області (рис. 2.1, III) характеризується вираженою нелінійністю, для визначення значення похідної доцільно спершу провести апроксимацію поліномом другого порядку, використовуючи останніх 8-15% точок оцифрованої ВАХ. Після чого знаходиться значення похідної від знайденого полінома в точці ХХ.

Приклад знаходження значення чисельної похідної в спадній частині для СП КС200GT наведено на рис. 2.4. Очевидно, що використання апроксимації прямою лінією $y = -2.13062x + 70.50293$ (рис. 2.4, штрихова) призводить до заниженої оцінки значення похідної, тоді як апроксимація поліномом другого порядку $y = -0.08565x^2 + 3.49433x - 21.848071$ (рис. 2.4, штрих-пунктирна) дозволяє провести оцінку кута нахилу більш точно. В обох випадках для отримання регресійного поліному використовувалися останні 10 точок ВАХ, які лежать праворуч від вертикальної лінії (рис. 2.4).

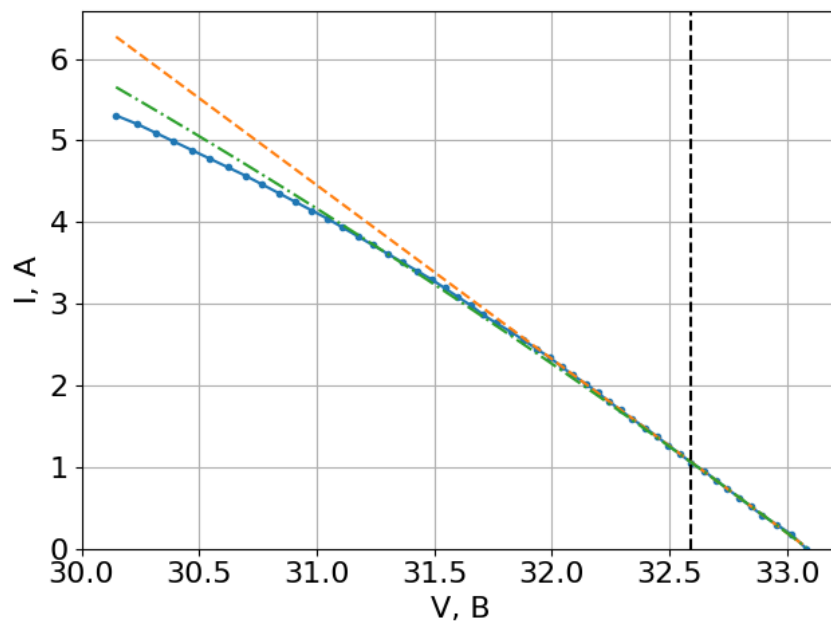


Рис. 2.4. Графік прикладу знаходження похідної в спадній ділянці ВАХ: оцифрована ВАХ (суцільна лінія), лінійна (штрихова) та квадратична апроксимація (штрих-пунктирна)

При ітераційному визначенні оптимальних значень коефіцієнтів ідеальності, які мінімізують похибку апроксимації оцифрованої ВАХ, постає

потреба розв'язку рівняння вихідного струму (2.2) і відповідно побудови ВАХ за значеннями параметрів моделі. Оскільки (2.2) є трансцендентним, то для його вирішення необхідно використовувати чисельні методи.

В даному випадку можуть використовуватися різноманітні чисельні методи, проте ряд досліджень для даної цілі використовують метод Ньютона [69], [108]. Проте надалі буде розглядатися метод простої ітерації, який є одним із найпростіших в комп'ютерній реалізації та який не потребує визначення чисельної похідної на кожній ітерації [103].

Експериментально було встановлено, що при певних значеннях параметрів схеми заміщення чисельний метод може не сходитись поблизу точки ХХ. Для забезпечення збіжності доцільно використовувати метод релаксації, суть якого полягає у введенні наступного рекурентного співвідношення [109], [110]:

$$I_{i+1} = I_i(1 - \varepsilon) + \varepsilon F(I_i, V_j), \quad (2.14)$$

де I_{i+1} – значення струму на наступній ітерації, I_i – значення струму на поточній ітерації, $F(I_i, V)$ – рівняння (2.2) $0 \leq \varepsilon \leq 1$ – коефіцієнт релаксації.

Використання методу релаксації збільшує необхідну кількість ітерацій для побудови ВАХ, що в свою чергу значно зменшує швидкодію алгоритму. Експериментальним шляхом встановлено, що мінімальним значенням, яке одночасно забезпечує збіжність алгоритму та не сильно впливає на швидкодію, є $\varepsilon = 0,2$.

Таким чином, для побудови ВАХ при заданих значеннях параметрів схеми заміщення необхідно для кожного значення сітки напруги V_j проводити ітераційний процес, використовуючи (2.14). Завершенням даного ітераційного процесу є наступна умова:

$$|I_{i+1} - I_i| < \delta,$$

де $\delta < 10^{-6}$ – деяке порогове значення похибки апроксимації.

Блок-схема алгоритму методу простої ітерації з використанням методу релаксації зображена на рис. 2.5.

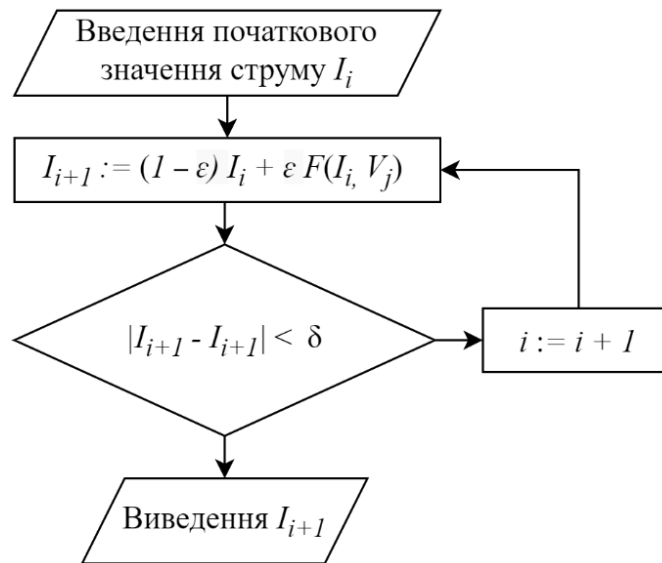


Рис. 2.5. Блок-схема алгоритму методу простої ітерації з використанням методу релаксації

При проведенні оцінки якості апроксимації ВАХ (рис. 2.2) значення похибки може бути рівномірно розподілене по сітці значень напруги. Тобто можлива ситуація, коли найменша похибка апроксимації досягається шляхом мінімізації похибки на лінійних та/або спадних ділянках ВАХ, тоді як апроксимація в робочій області матиме низьку точність. Оскільки робоча точка СП знаходиться саме в робочій області, постає потреба використання вагової функції, яка забезпечуватиме найменшу похибку апроксимації саме на даній ділянці ВАХ.

Існує чимала кількість вагових функцій, проте враховуючи те, що у якості вихідних даних для запропонованого алгоритму використовується оцифрована ВАХ, то доцільно використовувати нормовану криву вихідної потужності досліджуваної СП у ролі вагової функції:

$$w^k = \left(\frac{I_{ei} V_{ei}}{\max(I_{ei} V_{ei})} \right)^k, \quad (2.15)$$

де I_{ei}, V_{ei} – i -те значення струму і напруги оцифрованої ВАХ, $i=1...n$, n – кількість точок ВАХ, $k \geq 1$ – степінь вагової функції.

Приклад вагових функцій для різних значень k наведено на .

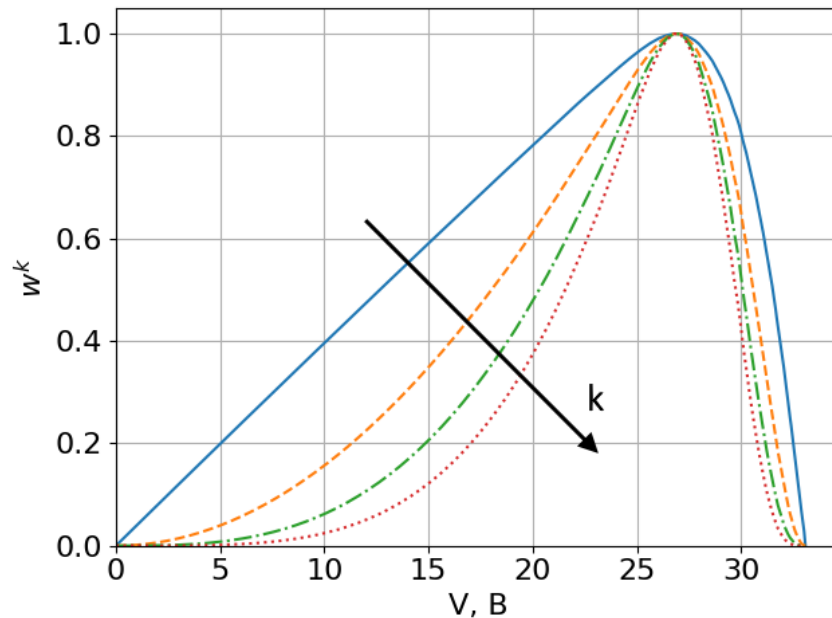


Рис. 2.6. Графік вагової функції при різних значень k : суцільна – 1, штрихова – 2, штрих-пунктирна – 3, пунктирна – 4

Очевидно, що з ростом k зменшується вплив похибки апроксимації в лінійній та спадній областях за рахунок нелінійності вагової функції, що дозволяє покращити точність апроксимації в робочій ділянці ВАХ. Проте при значеннях $k \geq 3$ може спостерігатися зворотній ефект, оскільки в спадній ділянці значення вагової функції зменшується доволі швидко, що може призводити до збільшення похибки поблизу точки ХХ. Результатом даного ефекту є отримання невірних значень коефіцієнтів ідеальності у процесі перебору та хибної апроксимації поблизу точки ХХ.

Приклад використання w^4 при визначенні оптимальних значень коефіцієнтів ідеальності для СП КС200GT зображено на рис. 2.7. Вертикальна лінія відображає поділ ВАХ на робочу і спадну ділянки. Використання даної вагової функції забезпечує хорошу якість апроксимації в робочій області, тоді як в спадній області апроксимація є доволі неточною, а значення струму ХХ не

є нульовим, що є значним недоліком. Причиною даного ефекту є те, що значення вагової функції w^k в спадній ділянці при великих значеннях k близьке до нуля, тобто дана ділянка майже не враховується при визначенні похибки апроксимації ВАХ.

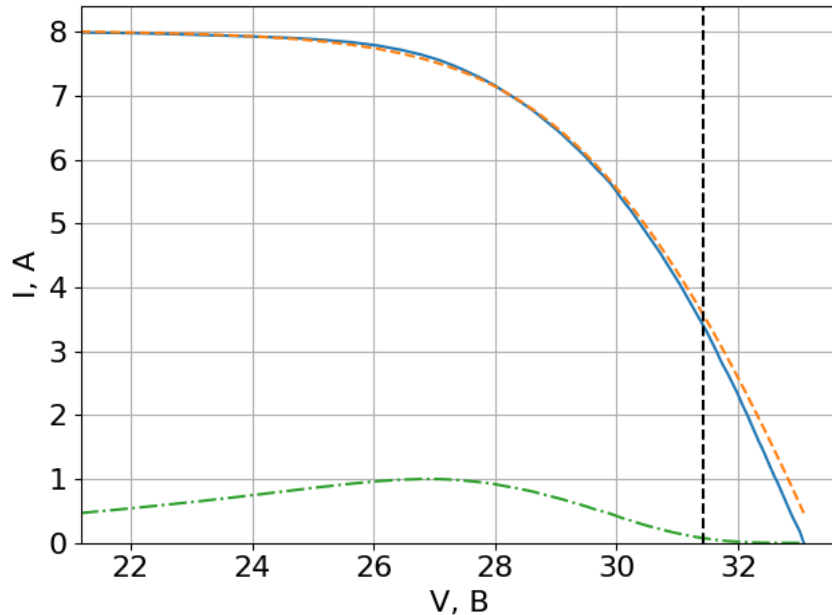


Рис. 2.7. Графік прикладу результатів роботи алгоритму при використанні вагової функції w^4 (штрих-пунктирна лінія): оцифрована (суцільна) та апроксимована (штрихова) ВАХ

Для усунення даного ефекту вагова функція має відповідати наступним вимогам:

1. Повільно спадати поблизу точки ХХ і відповідній області ВАХ.
2. Також бажано, щоб вагова функція не приймала нульове значення в поблизу точок ХХ та КЗ.

Першу умову задовольняє наступна кусково-задана функція:

$$w_{norm}^k = \begin{cases} w^k, & V < V_{MPP} \\ w^1, & V \geq V_{MPP} \end{cases}, \quad (2.16)$$

V_{MPP} – напруга в точці максимальної потужності.

Приклад використання вагової функції w_{norm}^4 зображено на Рис. 2.8. Очевидно, що використання модифікованої вагової функції забезпечує хорошу якість апроксимації як в робочій, так і в спадній області.

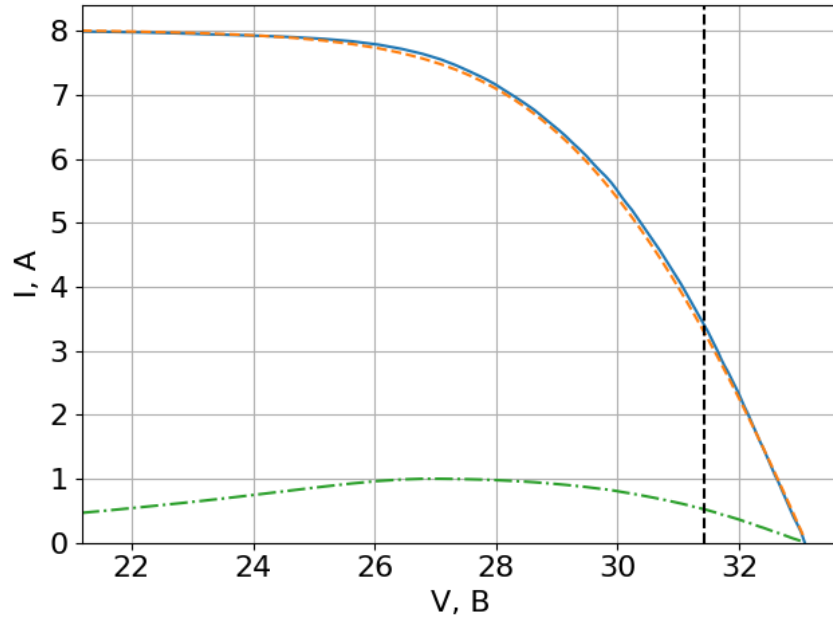


Рис. 2.8. Графік прикладу результатів роботи алгоритму при використанні вагової функції w_{norm}^4 (штрих-пунктирна лінія): оцифрована (суцільна) та апроксимована (штрихова) ВАХ

Окрім зазначених вище вагових функцій, доцільно використовувати Гаусівську віконну функцію, яка задовольняє обидві зазначені вище вимоги до вагових функцій:

$$w_g = e^{-\left(\frac{V-V_{MPP}}{2\sigma V_{MPP}}\right)^2}, \quad (2.17)$$

де $\sigma \leq 0,5$ – коефіцієнт, який визначає ступінь розтягнення гаусіана вздовж сітки значень напруги.

Використання (2.17) дозволяє позбутися всіх недоліків, якими володіють попередні віконні функції. Експериментальним чином було встановлено, що оптимальне значення коефіцієнту в (2.17) лежить в діапазоні $\sigma = 0,05 \dots 0,15$. На Рис. 2.9 наведено приклад використання вагової функції w_g . Візуально результати використання Гаусівської віконної функції співрозмірні з w_{norm}^4 .

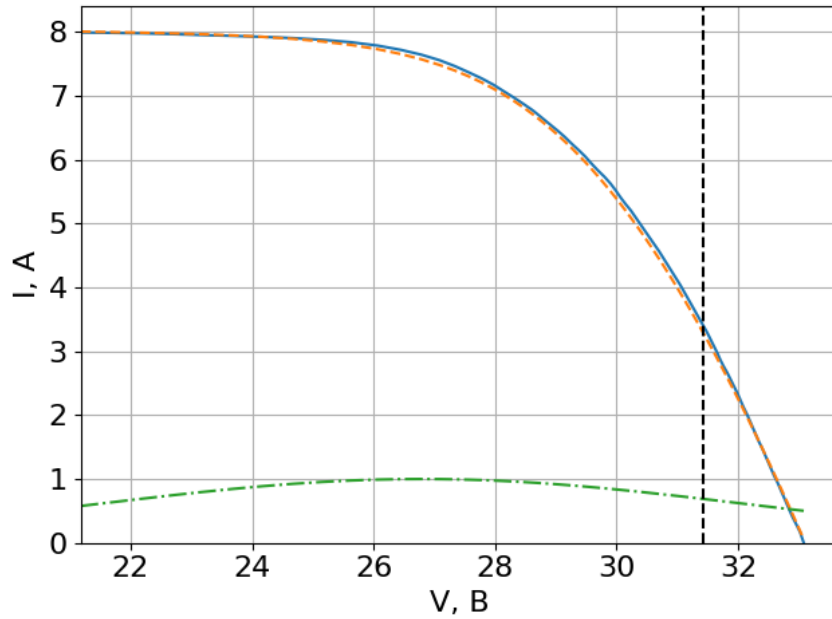


Рис. 2.9. Графік прикладу результатів роботи алгоритму при використанні вагової функції w_g (штрих-пунктирна лінія): оцифрована (суцільна) та апроксимована (штрихова) ВАХ

Проведемо порівняння вагових функцій. У якості метрики для оцінки якості апроксимації використовуються в запропонованому методі використовується корінь з середньоквадратичної похибки:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (2.18)$$

де y_i – i -те значення оцифрованої ВАХ, \hat{y}_i – i -те значення апроксимованої ВАХ.

Проте для можливості порівняння значення похибок при використанні різних вагових функцій необхідно проводити нормалізацію значень похибки відносно значення вагової функції. Враховуючи це, корінь з середньоквадратичної похибки (2.18) з використанням вагової функції та нормування матиме вигляд:

$$wRMSE = \sqrt{\frac{\sum_{i=1}^n w_i (y_i - \hat{y}_i)^2}{\sum_{i=1}^n w_i}} \quad (2.19)$$

Окрім того, у якості додаткового параметру оцінки точності апроксимації використовується середня абсолютна відносна похибка:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|, \quad (2.20)$$

де y_i – i -те значення оцифрованої ВАХ, \hat{y}_i – i -те значення апроксимованої ВАХ.

У якості додаткової метрики також використовуватимемо добуток двох вищезгаданих похибок:

$$M = MAPE \cdot wRMSE \quad (2.21)$$

Порівняння вагових функцій проводитимемо при визначенні параметрів схеми заміщення за допомогою способу ПЗКІ (рис. 2.2) для двохдіодної схеми заміщення на прикладі СП КС200GT, яка є типовим комерційним рішенням першого покоління. За результати порівняння (табл. 2.1) побудовано гістограму значень похибок в залежності від типу використаної вагової функції (рис. 2.10).

Таблиця 2.1. Результати порівняння вагових функцій для ПЗКІ

Вагова функція	w^1	w^4	w_{norm}^4	w_g
$wRMSE$	0,06893	0,05558	0,07808	0,07521
$MAPE$	0,01473	0,09642	0,01473	0,01473
M	0,00102	0,00536	0,00115	0,00111

З таблиці 2.1 видно, що значення похибки $wRMSE$ для вагової функції w^4 на 19% менше, ніж для w^1 , проте візуально апроксимація ВАХ, отримана за допомогою w^4 (рис. 2.7), забезпечує погану апроксимацію поблизу точки ХХ. Даний ефект спостерігається за рахунок того, що вагова функція w^4 приймає значення близькі до нуля в спадній ділянці ВАХ, що нівелює вклад похибки апроксимації в даній області при процесі перебору. Це підтверджується значенням похибки $MAPE$ – w^4 забезпечує на 85% більшу

похибку апроксимації, ніж w^1 . Аналогічна ситуація спостерігається з добутком значень двох похибок M – для w^4 вона на 81% більша, ніж для w^1 . Хоча w^4 і дозволяє збільшити точність апроксимації в робочій ділянці за рахунок зменшення впливу похибки апроксимації в лінійній ділянці, проте одночасно з цим погіршується точність апроксимації в спадній ділянці. Також можна зробити висновок, що $wRMSE$ характеризує похибку апроксимації в робочій області, тоді як $MAPE$ – в спадній, а M – в обох.

У порівнянні з w^1 , вагова функція w_{norm}^4 забезпечує співрозмірні значення похибок – $wRMSE$ та M більші на 12%, тоді як $MAPE$ мають однакові значення. Це дозволяє значно покращити апроксимацію поблизу точки ХХ у спадній ділянці, хоча одночасно з цим дещо зменшується точність апроксимації в робочій області у порівнянні з w^4 .

Використання Гаусівської віконної функції w_g зі значенням $\sigma = 0,1$ забезпечує співрозмірні з w^1 – $wRMSE$ та M мають на 8% більші значення, тоді як $MAPE$ є однаковими.

Таким чином, серед запропонованих вагових функцій найкращу якість апроксимації забезпечують нормована крива вихідної потужності першого порядку w^1 та Гаусівської віконна функція w_g зі значенням $\sigma = 0,1$. При цьому перша забезпечує найменшу похибку апроксимації, але в деяких випадках може призводити до поганого наближення ВАХ поблизу точки ХХ, як у випадку w^4 (рис. 2.7). Тому для в якості вагової функції рекомендовано використовувати саме Гаусівську віконну функцію w_g зі значеннями коефіцієнту $\sigma = 0,05...0,15$, а у якості метрики – добуток нормованої середньоквадратичної та відносної похибок M . Це дозволяє у процесі використання ПЗКІ (рис. 2.2) забезпечити якісну апроксимацію в робочій та спадній ділянці з одночасним зменшенням впливу лінійної ділянки.

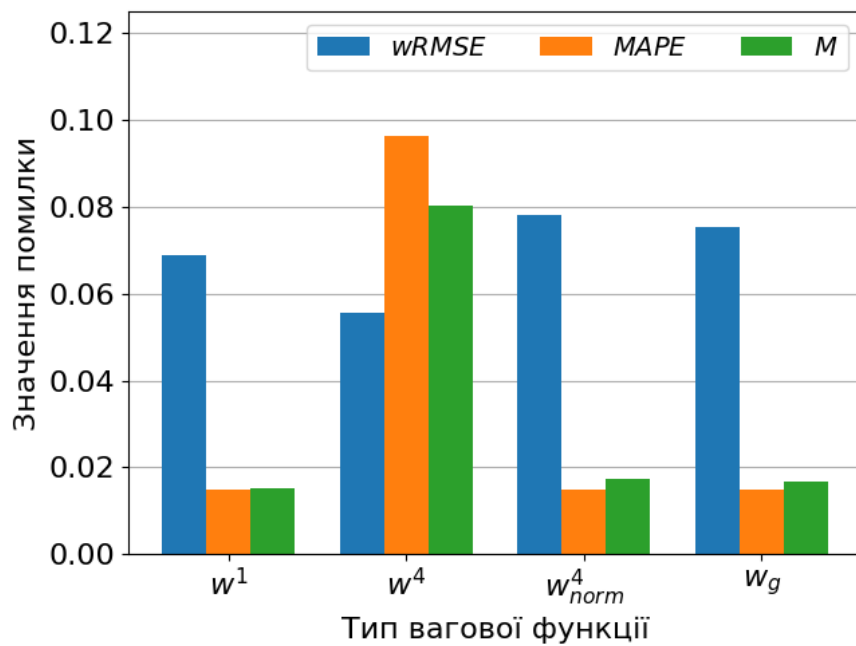


Рис. 2.10. Гістограма залежності значення похибок від типу використаної вагової функції. Значення метрики M збільшене в 15 разів

На кожній ітерації ПЗКІ визначаються значення параметрів моделі за допомогою встановлених раніше залежностей (рис. 2.2), оскільки всі вони залежать від значення коефіцієнтів ідеальності прямо або опосередковано. Проте експериментально було визначено, що значення параметрів проявляють малу мінливість. Для оцінки міри розсіяння відносно середнього значення використовується коефіцієнт варіації [111]:

$$c_v = \frac{\sigma}{\mu} \times 100\%,$$

де σ – стандартне відхилення, μ – середнє значення.

На прикладі СП KC200GT значення коефіцієнту варіації для параметрів моделі схеми заміщення при використанні ПЗКІ наведені в табл. 2.2.

Таблиця 2.2. Статистичні значення параметрів при використанні ПЗКІ

	I_{PV}, A	I_0, mA	R_s, Ω	R_p, Ω	β
μ	8,205	1,368	0,286	121,098	1,0
σ	$9,860 \cdot 10^{-9}$	3,306	$1,455 \cdot 10^{-7}$	$1,455 \cdot 10^{-7}$	$8,336 \cdot 10^{-7}$
c_v	$1,202 \cdot 10^{-7}$	241,496	$5,093 \cdot 10^{-5}$	$1,202 \cdot 10^{-7}$	$8,336 \cdot 10^{-5}$

З таблиці 2.2 видно, що коефіцієнту варіації c_v для більшості параметрів приймає нехтовно малі значення, що свідчить про малий розкид значень даних параметрів відносно середнього значення. В свою чергу, значення коефіцієнту варіації для величин, значення яких близьке до нуля, не є репрезентативним [111], тому попереднє твердження є справедливим і для зворотнього струму I_0 .

Враховуючи вищесказане, визначення значень параметрів (I_{PV}, I_0, R_s, R_p) можна винести поза межі циклів (рис. 2.2). Це значно скорочує кількість необхідних розрахунків, що в значній мірі пришвидшує швидкодію запропонованого способу визначення значень параметрів моделі. Проведемо порівняння якості апроксимації при використанні даної модифікації алгоритму на прикладі СП КС200GT з використанням w_g з $\sigma = 0,1$ (табл. 2.3). За отриманими результатами були побудовані ВАХ робочої та спадної ділянок (рис. 2.11).

Таблиця 2.3. Значення параметрів при різних модифікація ПЗКІ

Метод	I	II
I_{PV} , А	8,205	8.205
I_0 , нА	0,339	0,173
R_s , Ом	0,286	0,286
R_p , Ом	121,098	121,098
α_1	1	1
α_2	2	1
$wRMSE$	0,07521	0,10117
$MAPE$	0,01473	0,01859
M	0,00111	0,00188

Отримані значення параметрів добре узгоджуються між способом з перевизначенням значень параметрів моделі на кожній ітерації (табл. 2.3, I) та одноразовим визначенням поза циклами (табл. 2.3, II). При цьому

спостерігається різниця в 49% в оцінці значення зворотнього струму та коефіцієнту ідеальності α_2 . Перший підхід забезпечує менші значення похибок апроксимації – $wRMSE$ на 26%, $MAPE$ на 21% та M на 41%, що свідчить про кращу якість апроксимації ВАХ (рис. 2.11).

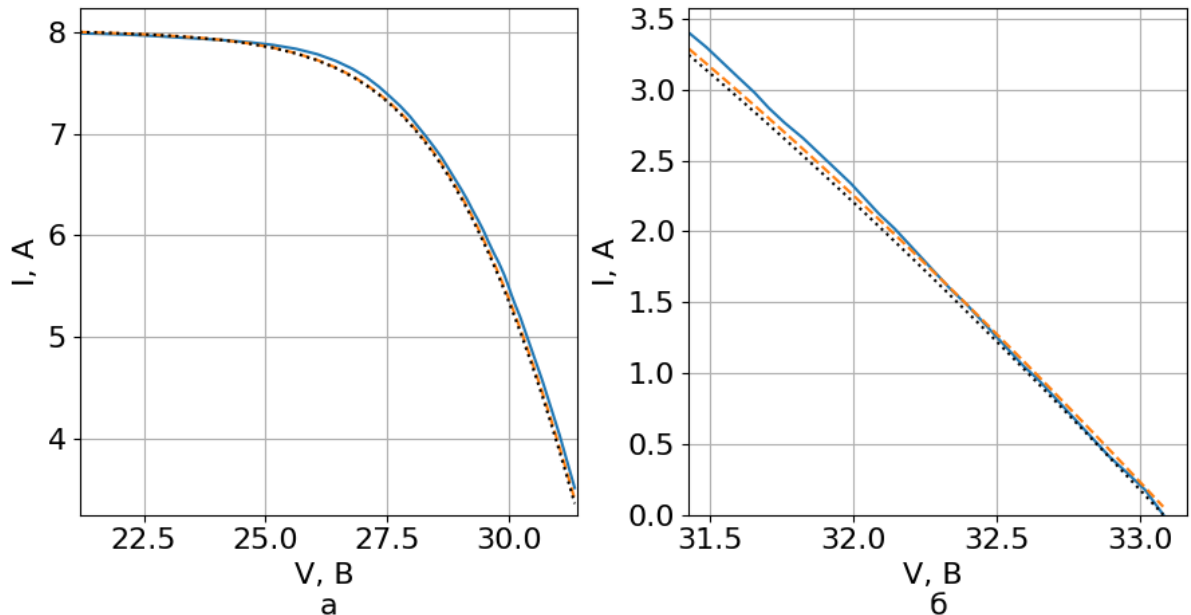


Рис. 2.11. Графік апроксимації ВАХ в робочій (а) та спадній ділянках (б) при використанні ПЗКІ: оцифрована ВАХ (суцільна лінія), апроксимація за допомогою I (штрихова), апроксимація за допомогою II (пунктирна).

Другий підхід забезпечує кращу оцінку струму ХХ. Вищесказане візуально відображається в кращій якості апроксимації в робочій ділянці при використанні першого підходу (рис. 2.11, а), при цьому точність апроксимації залишається співрозмірною в спадній області (рис. 2.11, б). Тобто одноразове визначення параметрів поза циклами підвищує швидкодію, через що дещо зменшується якість наближення ВАХ.

2.3. Спосіб перебору значень опорів

Даний метод у великій мірі є подібним до попереднього – методу перебору коефіцієнтів ідеальності. Блок-схема даного алгоритму зображена на (рис. 2.12).

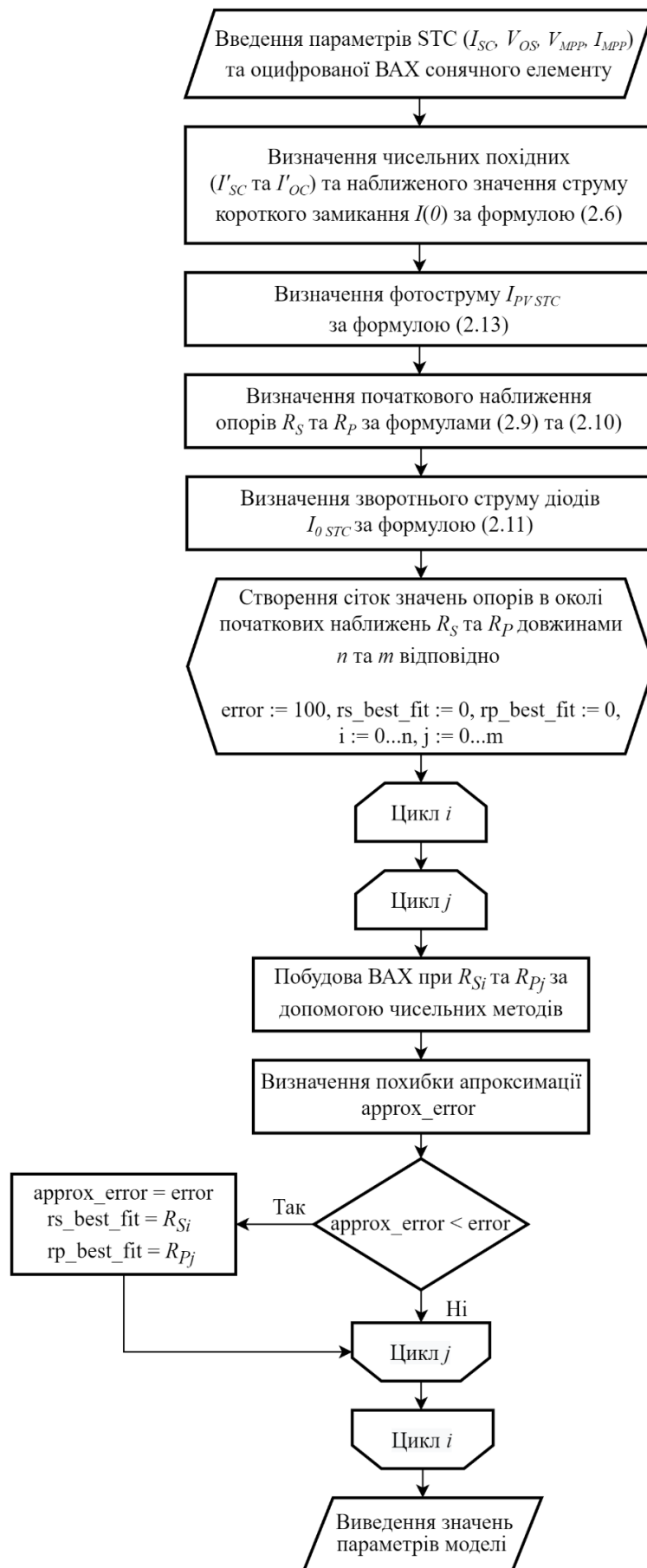


Рис. 2.12. Блок-схема алгоритму ПЗО

Суть даного методу полягає у визначенні початкових наближень значень параметрів схеми заміщення (I_{PV}, I_0, R_s, R_p) за допомогою встановлених раніше залежностей. При цьому, одночасно з цим використовуються типові значення коефіцієнтів, а саме $\alpha_1 = 1$ та $\alpha_2 = 1,2$. Далі відбувається перебір значень опорів R_s та R_p в певному околі для визначення оптимальних значень, які мінімізують похибку апроксимації ВАХ.

При використанні методу ПЗО характеризується наступними особливостями:

1. Правила визначення чисельних похідних залишаються аналогічними, як і для ПЗКІ. Тобто для визначення значення похідної в лінійній ділянці необхідно використовувати 20-25% точок оцифрованої ВАХ (рис. 2.3) і 8-15% у спадній області (рис. 2.4).

2. У якості чисельного методу для вирішення рівняння вихідного струму (2.2) доцільно використовувати метод простої ітерації, а для забезпечення його збіжності – метод релаксації (рис. 2.5).

Проведемо аналіз впливу типу вагової функції на якість апроксимації для даного методу аналогічно ПЗКІ. Значення похибок апроксимації при використанні різних вагових функцій наведені в таблиці 2.4. За отриманими результатами побудовано графік залежності значення похибок від типу вагової функції (рис. 2.13).

Таблиця 2.4. Результати порівняння вагових функцій для ПЗО

Вагова функція	w^1	w^4	w_{norm}^4	w_g
$wRMSE$	0,01794	0,02008	0,02000	0,01996
$MAPE$	0,00582	0,00601	0,00582	0,00580
M	0,00010	0,00012	0,00012	0,00011

З таблиці 2.4 видно, що значення похибки $wRMSE$, $MAPE$ та їх добутку P є співрозмірними, оскільки різниця в значеннях похибок не перевищує 17%.

Через це графіки результатів апроксимації ВАХ з використанням різних вагових функцій не наводяться, оскільки через співрозмірну точність наближення їх не можна візуально диференціювати.

Серед всіх вагових функцій найкращі показники по всім значення похибки апроксимації забезпечує вагова функція w^1 . Проте вона характеризується рядом недоліків і може призводити до поганої якості апроксимації в робочій та спадній ділянці за рахунок великого значення вагової функції в лінійній області ВАХ.

У порівнянні з w^1 вагові функції w^4 та w_{norm}^4 забезпечують більші значення похибок апроксимації. Наприклад, значення параметру M на 17% більші для w^4 та w_{norm}^4 .

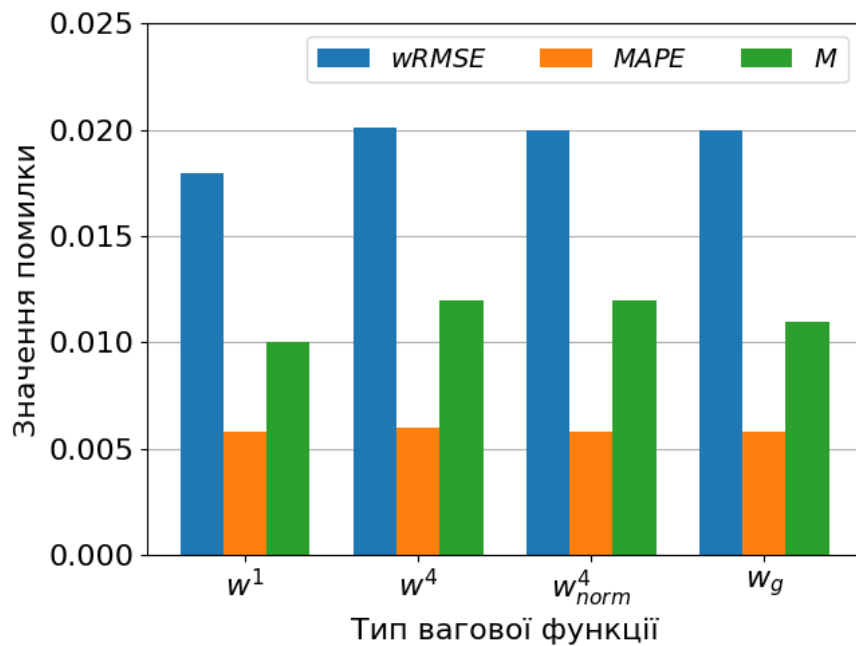


Рис. 2.13. Гістограма залежності значення похибок від типу використаної вагової функції. Значення метрики M збільшене в 100 разів

Гаусівська віконна функція w_g зі значенням $\sigma = 0,1$ забезпечує співрозмірні значення похибок апроксимації у порівнянні з w^1 – $wRMSE$ має на 8% більше значення, M – на 9%, а $MAPE$ на 1% менше. Хоча дана вагова функція і забезпечує гірші значення похибок апроксимації, вона дозволяє покращити якість наближення в робочій ділянці, що є важливим фактором.

Мала варіація значення параметру M свідчить про слабку залежність точності апроксимації від типу використовуваної вагової функції, що є перевагою даного методу. Оптимальною, як і у випадку з ПЗКІ, залишається Гаусівська вагова функція w_g , яка мінімізує похибку апроксимації в робочій і спадній ділянках ВАХ.

2.4. Порівняльний аналіз способів визначення параметрів

Проведемо порівняльний аналіз запропонованих способів визначення параметрів схеми заміщення, а саме ПЗКІ (позначимо I) та ПЗО (позначимо II) на прикладі одно- та двохдіодної схем заміщення. Також проведемо порівняльний аналіз з релевантними наявними методиками [108], [112].

Надалі, в таблицях відсутність значення коефіцієнту ідеальності α_2 відповідає двохдіодній схемі заміщення. При порівнянні використовуються комерційні СП KC200GT та ST40, основні параметри яких наведені в таблиці 2.5 [113], [114].

Таблиця 2.5. Основні параметри досліджуваних СП

	P_{\max} , Вт	V_{MPP} , В	I_{MPP} , А	I_{SC} , А	V_{OC} , В
KC200GT	200	26,3	7,61	8,21	32,9
ST40	40	16,6	2,41	2,68	23,3

В процесі порівняння запропонованих способів визначення параметрів схеми заміщення використовуються наступні параметри:

1. Для мінімізації похибки апроксимації в робочій ділянці використовується Гаусівська віконна функція w_g з $\sigma = 0,1$.
2. Для вирішення трансцендентного рівняння вихідного струму (2.2) використовується метод простої ітерації з $\varepsilon = 0,2$.
3. Для запропонованих способів використовуються описані раніше правила визначення значень похідних. Тобто для визначення значення похідної

в лінійній ділянці застосовано 20-25% точок оцифрованої ВАХ (рис. 2.3) і 8-15% у спадній області (рис. 2.4).

Також оцінимо обчислювальну складність алгоритмів ПЗКІ та ПЗО. Для цього прийнемо, що довжини сіток значень є однаковими для даних алгоритмів і дорівнює n (рис. 2.2 та рис. 2.12). Результати порівняння на прикладі КС200GT в таблиці 2.6. Для інших способів визначення параметрів значення похибки не наводиться, оскільки при їх визначенні використовувалися ВАХ СП, які доволі суттєво відрізняються від використаної для ПЗКІ та ПЗО, що призводить до нерелевантних значень похибок.

Таблиця 2.6. Значення параметрів схеми заміщення СП КС200GT

	ПЗКІ		ПЗО		[108]	[112], мет. І	[112], мет. ІІ
	1	2	3	4	5	6	7
R_s , Ом	0,286	0,286	0,263	0,263	0,320	0,308	0,284
R_p , Ом	121,098	121,098	117,391	120,357	160,5	193,049	157,853
I_{PV} , А	8,205	8,205	8,205	8,205	8,210	8,223	8,210
I_0 , нА	0,346	0,339	0,346	0,339	0,422	2,152	2,195
α_1	1	1	1	1	1	1,076	1,076
α_2	-	2	-	1,2	1,2	-	-
I'_{sc}	-0,00824				-	-	-
I'_{oc}	-2,17256				-	-	-
$wRMSE$	0,07520	0,07521	0,01995	0,01977	-	-	-
$MAPE$	0,01473	0,01473	0,00578	0,00582	-	-	-
M	0,00111	0,00111	0,00012	0,00012	-	-	-
O	$O(n)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	-	-	-

З таблиці 2.6 видно, що значення параметрів для ПЗКІ та ПЗО добре узгоджуються, а значення похибок апроксимації слабко залежить від типу

еквівалентної схеми заміщення (одно- чи двохдіодна) для обох способів (табл. 2.6 стовб. 1-4). Враховуючи це, надалі для кожного зі способів буде будуватися лише один графік ВАХ, який забезпечує найменше значення похибки $wRMSE$. Для ПЗКІ для цього використовується однодіодна схема заміщення (табл. 2.6, стовб. 1), а для ПЗО – двохдіодна (табл. 2.6, стовб. 2). Відповідні графіки наведені на рис. 2.14.

Обидва способи забезпечують однакове значення фотоструму I_{PV} , тоді як значення зворотнього струму I_0 відрізняється для однодіодної (табл. 2.6, стовб. 1, 3) та двохдіодної схем (табл. 2.6, стовб. 2, 4). Це пояснюється наявністю коефіцієнту ідеальності α_2 у двохдіодній схемі заміщення, що в силу експоненційної залежності значення струму I_0 від коефіцієнту ідеальності (2.11) призводить до зміни значення струму I_0 .

Для ПЗКІ значення опорів R_s та R_p не залежать від типу використовуваної схеми заміщення, тоді як для ПЗО спостерігається дана залежність для R_p . Подібна поведінка очікувана для ПЗО, оскільки в даному способі визначення параметрів схеми заміщення відбувається мінімізація похибки апроксимації саме за рахунок перебору значень опорів.

Порівнюючи похибки апроксимації, спосіб ПЗО (табл. 2.6, стовб. 4) забезпечує набагато кращу якість апроксимації у порівнянні ПЗКІ (табл. 2.6, стовб. 1) – значення $wRMSE$ менше на 74%, $MAPE$ на 60%, а параметру M на 89%. Це призводить до значно кращого наближення ВАХ в робочій ділянці, особливо поблизу точки максимальної потужності за рахунок використання вагової функції (рис. 2.14).

Таким чином, серед запропонованих способів визначення параметрів найбільшу точність забезпечує ПЗО для двохдіодної схеми заміщення, при цьому алгоритм даного методу характеризується квадратичною обчислювальною складністю, що є недоліком. На противагу цьому, ПЗКІ забезпечує гіршу якість апроксимації, проте для нього характерна лінійна

складність у випадку використання однодіодної схеми заміщення, що є перевагою даного підходу.

У підсумку, для забезпечення кращої якості апроксимації доцільно використовувати ПЗО для двохдіодної схеми заміщення, а ПЗКІ разом однодіодною моделлю у випадках, де швидкодія є визначальним фактором.

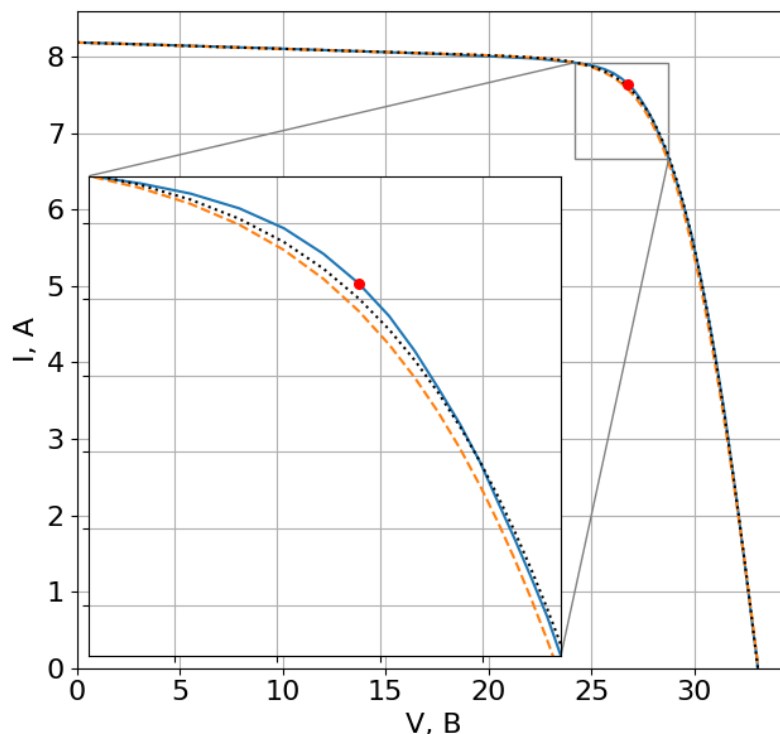


Рис. 2.14. Графік порівняння ПЗКІ (штрихова крива) та ПЗО (точкова) на прикладі СП KC200GT. Суцільна лінія відповідає оцифрованій ВАХ, червона точка – точні максимальної потужності

Проведемо порівняння розробленої моделі з іншими способами визначення параметрів, при цьому будемо використовуватися ПЗО для двохдіодної схеми заміщення. ВАХ для даного порівняння наведені на рис. 2.15.

З таблиці 2.6 видно, що інші способи схильні до завищення значення послідовного R_s та шунтуючого опорів R_p , що відображається у невірному куті нахилу ВАХ у спадній та лінійних ділянках відповідно (рис. 2.15).

Значення фотоструму I_{PV} для всіх способів є добре узгоджуються, тоді як значення зворотнього струму I_0 показують великий розкид і відрізняються в значній мірі.

Візуально інші способи визначення параметрів показують гіршу якість апроксимації в робочій ділянці ВАХ й поблизу ТМП, при цьому якість наближення в лінійній та спідній ділянках є співрозмірними (Рис. 2.15. Графік порівняння ВАХ СП КС200GT: оцифрована (суцільна лінія), ПЗО для двохдіодної схеми (точкова чорна), [108] (штрихова), [112] м. 1 (штрих-пунктирна), [112] м. 2 (точкова, червона)рис. 2.15рис. 2.17).

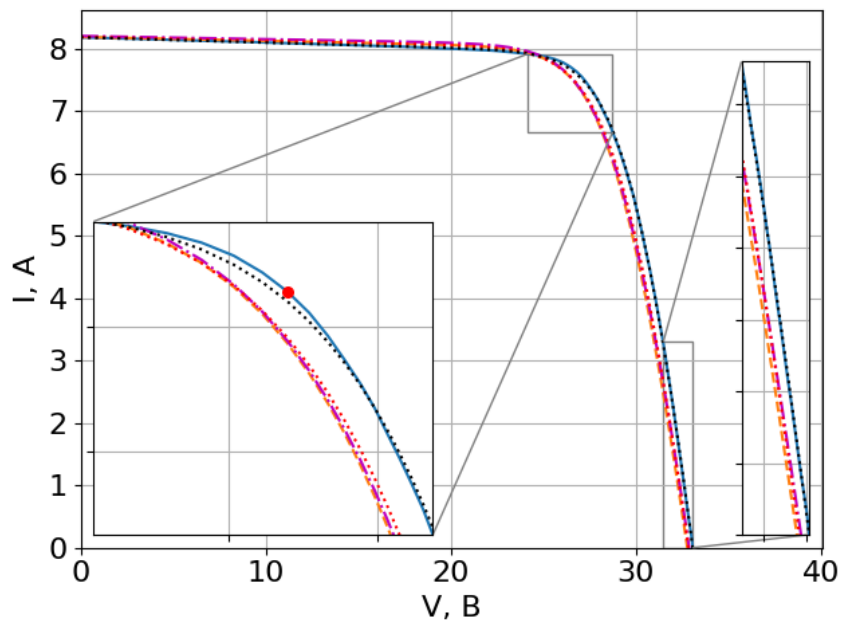


Рис. 2.15. Графік порівняння ВАХ СП КС200GT: оцифрована (суцільна лінія), ПЗО для двохдіодної схеми (точкова чорна), [108] (штрихова), [112] м. 1 (штрих-пунктирна), [112] м. 2 (точкова, червона)

Проведемо аналогічне порівняння запропонованих способів визначення на основі СП ST40 (табл. 2.7). Аналогічно попередньому прикладу (табл. 2.6) для інших способів визначення параметрів значення похибки не наводиться.

З таблиці 2.7 видно, що значення параметрів для ПЗКІ та ПЗО добре узгоджуються, проте на відміну від випадку СП КС200GT, значення похибок

апроксимації залежить від типу еквівалентної схеми заміщення (одно- чи двохдіодна) для обох способів (табл. 2.7 стовб. 1-4).

Аналогічно попередньому прикладу (табл. 2.6), обидва способи забезпечують однакове значення фотоструму I_{PV} та зворотнього струму I_0 . Значення опорів R_s та R_p повністю повторює поведінку для СП KC200GT, тобто для ПЗКІ дані значення не залежать від типу схеми заміщення (табл. 2.7, стовб. 1-2), тоді як для ПЗО спостерігається дана залежність для R_s (табл. 2.7, стовб. 3-4).

Порівнюючи похибки апроксимації, спосіб ПЗО забезпечує набагато кращу якість апроксимації у порівнянні ПЗКІ як для одно-, так для двохдіодної схем заміщення. Залежність похибки апроксимації від способу визначення параметрів та типу схеми показує більш складну залежність, ніж для попереднього порівняння. Тому проведемо детальний аналіз значень похибок.

Таблиця 2.7. Значення параметрів схеми заміщення СП ST40

	ПЗКІ		ПЗО		[108]	[112], мет. I	[112], мет. II
	1	2	3	4	5	6	7
R_s , Ом	1,219	1,219	1,45080	1,436	1,600	1,494	1,372
R_p , Ом	587,520	587,520	411,264	411,264	263,300	257,166	194,664
I_{PV} , А	2,682	2,682	2,682	2,682	2,680	2,696	2,680
I_0 , нА	1,008	1,008	1,008	1,008	1,130	1,414	1,455
α_1	1	1,020	1	1	1	1,135	1,135
α_2	-	1,041	-	1,2	1,2	-	-
I'_{sc}	-0,00170				-	-	-
I'_{oc}	-0,61451				-	-	-
$wRMSE$	0,13183	0,00495	0,00046	0,00048	-	-	-
$MAPE$	0,05521	0,04808	0,01146	0,01951	-	-	-

M	0,00728	0,00024	0,00001	0,00001	-	-	-
O	$O(n)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	-	-	-

У випадку ПЗКІ однодіодна схема заміщення (табл. 2.7, стовб. 1-2) забезпечує кращу якість апроксимації у порівнянні з двохдіодною – значення $wRMSE$ менше на 96%, $MAPE$ на 13%, а параметр M на 97%. Візуально в двохдіодна схема заміщення забезпечує кращу якість апроксимації в робочій області, тоді як в спадній – точність наближення є співрозмірною (рис. 2.16).

Таким чином, якість апроксимації ВАХ способом ПЗКІ доволі суттєво залежить від типу використовуваної схеми заміщення. Також $wRMSE$ характеризує похибку апроксимації в робочій ділянці за рахунок використання вагової функції, тоді як $MAPE$ – в спадній області.

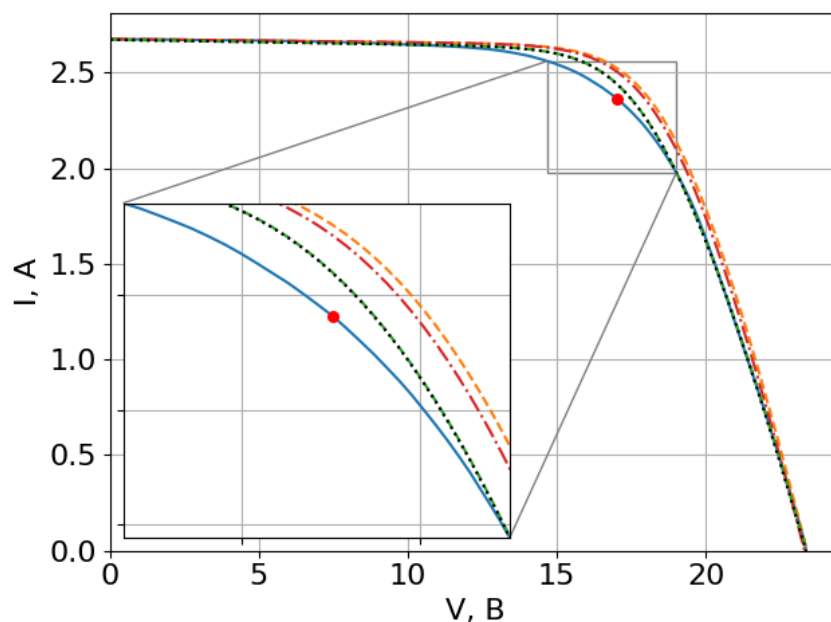


Рис. 2.16. Графік порівняння ПЗКІ та ПЗО на прикладі СП ST40: оцифрована ВАХ (суцільна лінія), ПЗКІ однодіодна схема заміщення (штрихова оранжева), ПЗКІ двохдіодна (штрих-пунктирна), ПЗО однодіодна (штрихова зелена), ПЗО двохдіодна (точкова). Червона точка відображає точку максимальної потужності

Для способу ПЗО значення параметру M є однаковим, що свідчить про однакову якість наближення, проте значення інших похибок відрізняються –

однодіодна схема забезпечує на 4% менше значення $wRMSE$ та на 41% $MAPE$. Це свідчить про однакову якість апроксимації в робочій ділянці та дещо гіршу в спадній у випадку двохдіодної схеми заміщення (рис. 2.16).

Порівнюючи похибки апроксимації, спосіб ПЗО (табл. 2.7, стовб. 4) забезпечує кращу якість апроксимації у порівнянні ПЗКІ (табл. 2.7, стовб. 2) – значення $wRMSE$ менше на 90%, $MAPE$ на 59%, а параметру M на 96%.

Таким чином, серед запропонованих способів визначення параметрів найбільшу точність забезпечує ПЗО для однодіодної схеми заміщення. Проте для узгодження методу порівняння для СП КС200GT та ST40 при порівнянні розробленої моделі з іншими способами будемо використовувати ПЗО для двохдіодної схеми заміщення. ВАХ для даного порівняння наведені на рис. 2.17.

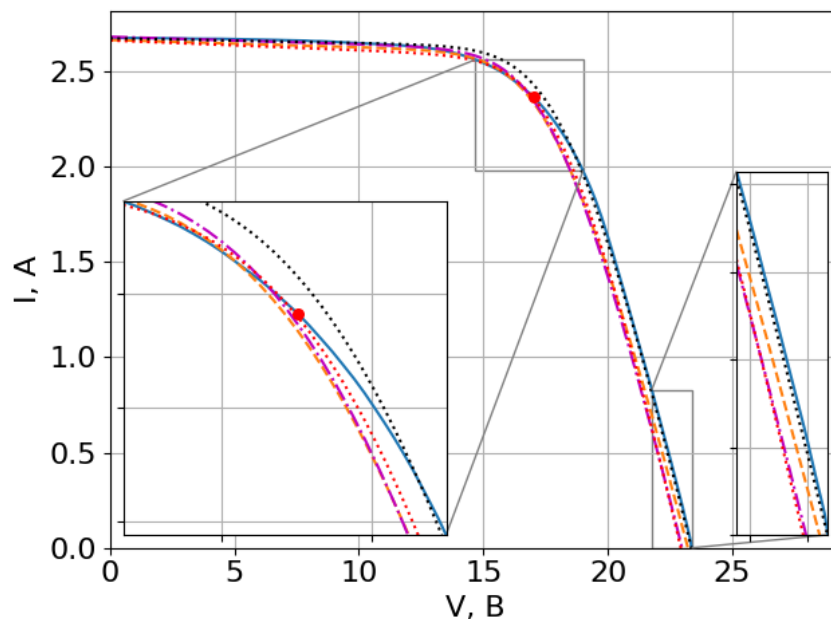


Рис. 2.17. Графік порівняння ВАХ СП ST40: оцифрована (суцільна лінія), ПЗО для двохдіодної схеми (точкова чорна), [108] (штрихова), [112] м. 1 (штрих-пунктирна), [112] м. 2 (точкова, червона)

Таблиця 2.7 демонструє, що інші способи забезпечують занижене значення шунтуючого опору R_p , що відображається у невірному куті нахилу ВАХ у лінійній ділянці (рис. 2.17).

Значення фотоструму I_{PV} для всіх способів є добре узгоджуються, тоді як значення зворотнього струму I_0 відрізняється в значній мірі і показує розкид в 30% у найгіршому випадку.

Візуально інші способи визначення параметрів показують набагато гіршу якість апроксимації в робочій ділянці в цілому, хоча й показують кращу якість наближення поблизу ТМП. Також різниця в значенні I_0 призводить до поганої апроксимації в спадній області, зокрема поблизу точки ХХ (рис. 2.17).

2.5. Залежність ВАХ від параметрів опроміненості і температури

Для можливості відтворення реальних експлуатаційних умов СП математична модель має враховувати температурну залежність і рівень опроміненості. Ряд експериментальних та теоретичних досліджень вказують на залежність параметрів еквівалентної схеми заміщення від опроміненості [115], [116].

Значення фотоструму залежить лінійно від опроміненості [117]:

$$I_{PV}(G) = I_{PV\,STC} \frac{G}{G_{STC}}, \quad (2.22)$$

де $I_{PV\,STC}$ – значення фотоструму за стандартних тестових умов, G_{STC} – значення опроміненості за стандартних тестових умов.

Обернену лінійну залежність проявляє шунтуючий опір:

$$R_P(G) = R_{P\,STC} \frac{G_{STC}}{G}, \quad (2.23)$$

де $R_{P\,STC}$ – значення паралельного опору за стандартних тестових умов.

Експериментально встановлено, що послідовний опір також проявляє дану залежність, проте вона носить нелінійний характер:

$$R_S(G) = R_{S\,STC} \left(\frac{G_{STC}}{G} \right)^k, \quad (2.24)$$

де $k = 0,5 \dots 1$. Оптимальним значенням є $k = 0,5$.

Ряд інших параметрів залежать від рівня опроміненості, проте дана залежність є нелінійною. До таких параметрів відносяться коефіцієнт ідеальності α та напруга ХХ V_{oc} [115]. Запропоновані способи визначення параметрів еквівалентної схеми заміщення забезпечують визначення V_{oc} , а розроблена математична модель припускає, що коефіцієнти ідеальності α не залежать від опроміненості, тому знехтуємо даними залежностями.

Отже, для відтворення ВАХ СП при різних рівнях опроміненості необхідно:

1. Провести визначення значень параметрів еквівалентної схеми заміщення при стандартних тестових умовах, використовуючи ПЗКІ (рис. 2.2) або ПЗО (рис. 2.12).
2. Визначити нові значення залежних від опроміненості параметрів, використовуючи (2.22)-(2.24).
3. Побудувати ВАХ за допомогою чисельного методу (рис. 2.5).

Наведемо приклади визначення ВАХ для типових значень опроміненості, а також проведемо оцінку якості наближення, використовуючи оцифровані ВАХ з паспортних даних СП. Дане порівняння будемо проводити на прикладі СП КС200GT, результати якого наведені в таблиці 2.8. Відповідні ВАХ наведені на рис. 2.18.

Таблиця 2.8. Залежність параметрів СП КС200GT від опроміненості

$G, \text{Вт/м}^2$	1000	800	600	400
$I_{PV}, \text{А}$	8,205	6,562	4,918	3,275
$R_s, \text{Ом}$	0,263	0,299	0,401	0,679
$R_p, \text{Ом}$	120,357	150,446	250,744	626,859
$wRMSE$	0,01977	1,78748	1,49673	1,12237
$MAPE$	0,00582	0,24082	0,33542	1,34172
M	0,00012	0,43047	0,50203	1,50590

З таблиці 2.8 видно, що зі зменшенням рівня опроміненості G , значення фотоструму I_{PV} зменшується, а опорів R_s та R_p росте. Одночасно з цим відбувається ріст всіх похибок апроксимації – $wRMSE$, $MAPE$ та M . Попри це, якість наближення ВАХ залишається задовільною для всіх наведених рівнів опроміненості (рис. 2.18).

Також більшість параметрів еквівалентної схеми заміщення залежать від температури СП.

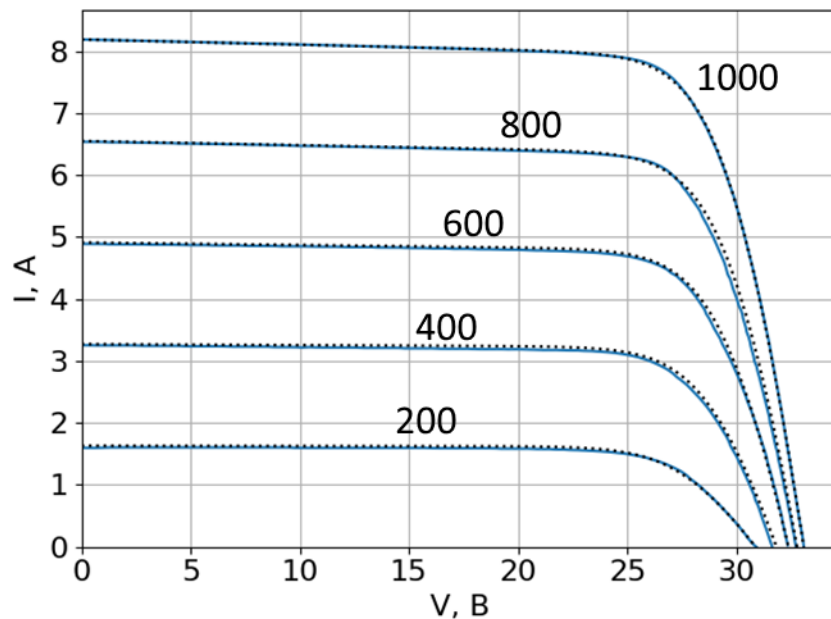


Рис. 2.18. ВАХ СП KC200GT при різних рівнях опроміненості: оцифрована (суцільна) та апроксимована (точкова лінія)

Значення фотоструму залежить лінійно від температури [118]:

$$I_{PV}(T) = I_{PV_{STC}}(1 + K_I(T - T_{STC})) = I_{PV_{STC}}(1 + K_I\Delta T), \quad (2.25)$$

де $I_{PV_{STC}}$ – значення фотоструму за стандартних тестових умов, T_{STC} – температура за стандартних тестових умов, температура за стандартних тестових умов, T – температура СП, K_I – температурний коефіцієнт фотоструму.

Аналогічним чином залежить напруга ХХ:

$$V_{OC}(T) = V_{OC_{STC}}(1 + K_V(T - T_{STC})) = V_{OC_{STC}}(1 + K_V\Delta T), \quad (2.26)$$

де V_{OCSTC} – значення напруги ХХ за стандартних тестових умов, K_V – температурний коефіцієнт напруги ХХ.

З ростом напруги значення шунтуючого опору зменшується нелінійно [119]:

$$R_P(T) = R_{PSTC} \frac{T_{STC}}{T}, \quad (2.27)$$

де R_{PSTC} – значення паралельного опору за стандартних тестових умов, T_{STC} – температура за стандартних тестових умов, T – температура СП.

Послідовний опір також з ростом температури змінюється лінійно [119]:

$$R_S(T) = R_{SSTC} \frac{T}{T_{STC}}, \quad (2.28)$$

де R_{SSTC} значення послідовного опору за стандартних тестових умов.

При дослідженні температурних залежностей важливу роль відіграє зворотній струм. Залежність зворотнього струму у випадку однодіодної схеми заміщення описується наступним чином [117]:

$$I_0(T) = I_{0STC} \left(\frac{T}{T_{STC}} \right)^{\frac{3}{\alpha_1}} \left(e^{\frac{E(T)}{k \left(\frac{1}{T_{STC}} - \frac{1}{T} \right)}} - 1 \right), \quad (2.29)$$

де I_{0STC} – значення зворотного струму за стандартних тестових умов, $E(T)$ – ширина забороненої зони напівпровідникового матеріалу, k – стала Больцмана.

Недоліком (2.29) є те, що дана формула визначена лише для однодіодної схеми заміщення, а також те, що ширини забороненої зони залежить від типу матеріалу та температури. Це призводить до того, що визначити зворотній струм при зміні температури неможливо, використовуючи формулу (2.29).

Щоб визначити залежність $I_0(T)$, проаналізуємо рівняння (2.2) у точці ХХ:

$$0 = I_{PV} - \frac{V_{OC}}{R_p} - I_0 \left[e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2 \right]$$

Звідки зворотній струм визначається наступним чином:

$$I_0 = \frac{I_{PV} - V_{OC}/R_p}{e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2} \quad (2.30)$$

Використовуючи в (2.30) значення фотоструму, напруги ХХ та шунтуючого опору, які знайдені за допомогою (2.25)-(2.28), можна визначити значення зворотнього струму для бажаного значення температури.

Для побудови ВАХ СП при різних значеннях температури необхідно:

1. Визначити значення параметрів еквівалентної схеми заміщення при стандартних тестових умовах, використовуючи ПЗКІ (рис. 2.2) або ПЗО (рис. 2.12).
2. Визначити значення параметрів, які залежать від температури, використовуючи (2.25)-(2.28) та (2.30).
3. Побудувати ВАХ використовуючи чисельні методи (рис. 2.5).

Наведемо приклад визначення ВАХ для різних значень температури, а також порівняємо якість апроксимації, використовуючи оцифровані ВАХ з паспортних даних. Дане порівняння будемо проводити на прикладі СП KC200GT, результати якого наведені в таблиці 2.9. Відповідні ВАХ наведені на

Таблиця 2.9. Залежність параметрів СП KC200GT від температури

$T, ^\circ C$	25	50	75
$R_s, \text{ Ом}$	0,263	0,285	0,333
$R_p, \text{ Ом}$	120,357	111,041	95,087
$I_{PV}, \text{ А}$	8,205	8,285	8,364
$I_0, \text{ нА}$	0,339	3,123	28,583
$wRMSE$	0,01977	0,27464	1,49673
$MAPE$	0,00582	0,49320	0,33542
M	0,00012	0,13545	0,50203

З таблиці 2.9 видно, що з ростом температури T , значення фотоструму I_{PV} зростає, а опорів R_s та R_p росте. Одночасно з цим відбувається ріст значень всіх похибок апроксимації – $wRMSE$, $MAPE$ та M . Попри це, якість наближення ВАХ при різних температурних режимах залишається задовільною (рис. 2.19).

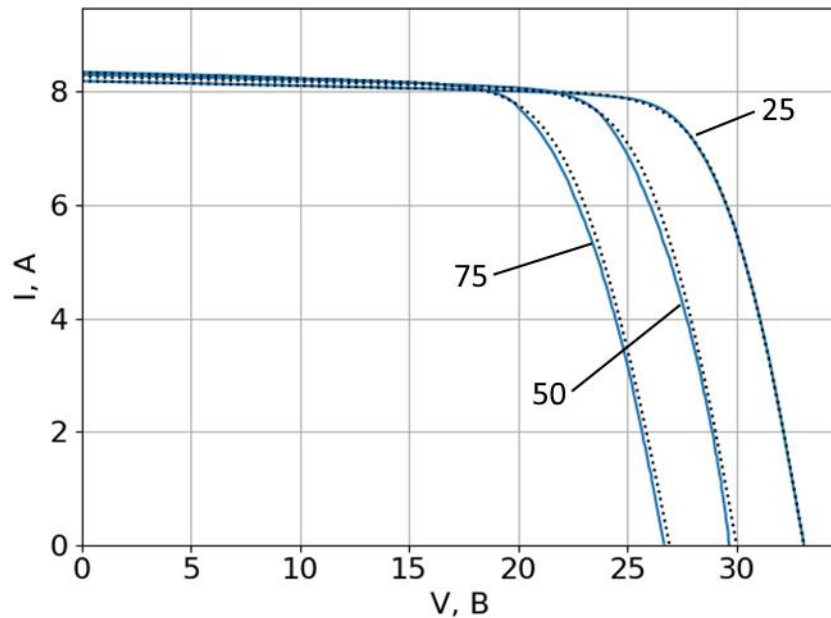


Рис. 2.19. ВАХ СП KC200GT при різних температурах: оцифрована (суцільна) та апроксимована (точкова лінія)

Таким чином, встановлені залежності дозволяють проводити моделювання ВАХ СП при різних умовах температури панелі на опроміненості, тим самим відтворюючи умови експлуатації, які є наближеними до реальних.

Висновки до розділу

1. Створено удосконалено математичну модель сонячної панелі та запропоновано два способи визначення параметрів моделі на основі даних, які надають виробники сонячних панелей. Для обох способів рекомендовано використовувати Гаусівську віконну функцію w_g зі значенням $\sigma = 0,1$, що

дозволяє мінімізувати похибку апроксимації поблизу точки максимальної потужності.

2. Порівняльний аналіз показав, що спосіб перебору значень опорів у порівнянні з перебором значень коефіцієнтів ідеальності забезпечує кращу точність апроксимації ВАХ – на 89% менше значення параметру M для сонячної панелі KC200GT та на 96% для ST40.

3. Створено програмну реалізацію математичної моделі на основі розроблених способів перебору значень опорів та коефіцієнту ідеальності на мові програмування Python, яка дозволяє моделювати ВАХ сонячної панелі при різних умовах опроміненості та температури.

РОЗДІЛ 3. МОДЕЛЬ ХМАРНОГО ПОКРИВУ

3.1. Визначення параметрів моделі хмарного покриву

Модель хмарного покриву відтворює вплив атмосфери на сонячне випромінювання, тим самим відтворюючи значення коефіцієнту пропускання атмосфери (1.2). За основу даної моделі обрано метод спектрального синтезу (СС) на основі швидкого перетворення Фур'є (ШПФ).

Оскільки СС застосовуватиметься для генерації зображень хмарного покриву, проведемо аналіз спектральних характеристик подібного зображення (рис. 3.1). Для генерації зображень використано код моделі (див. додаток в).

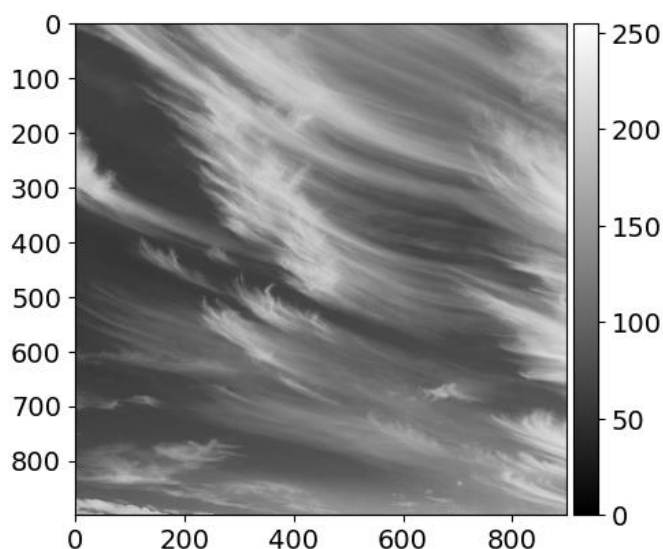


Рис. 3.1. Зображення досліджуваного хмарного покриву

ШПФ передбачає періодичність досліджуваної функції. Однак приклад зображення хмарного покриву (рис. 3.1) не є періодичним, що призводить до вираженого стрибка на межах зображення. Це призводить до явища витікання спектру (рис. 1.32), що суттєво впливає на спектральні характеристики [100]. Щоб зменшити вплив даного ефекту, використовуються різні віконні функції [101]. Визначимо віконну функцію наступним чином [10]:

$$W = \frac{1 - \sqrt{N_x^2 + N_y^2}}{\max(N_x^2 + N_y^2)}, \quad (3.1)$$

де N_x та N_y – номер пікселів вздовж осі абсцис та ординат відповідно на вихідному зображенні відносно його центру.

Також для спрощення аналізу вихідного зображення в частотній області, виключимо його постійну складову:

$$p = W(p_0 - \bar{p}), \quad (3.2)$$

де p_0 – вихідне зображення, \bar{p} – середнє значення пікселів вихідного зображення.

Після застосування віконної функції (3.1) та усунення постійної складової (3.2), вихідне зображення набуває вигляду як показано на рис. 3.2. Візуально це призводить до плавного зменшення яскравості зображення біля його країв, що тим самим мінімізує вплив явища витікання спектру.

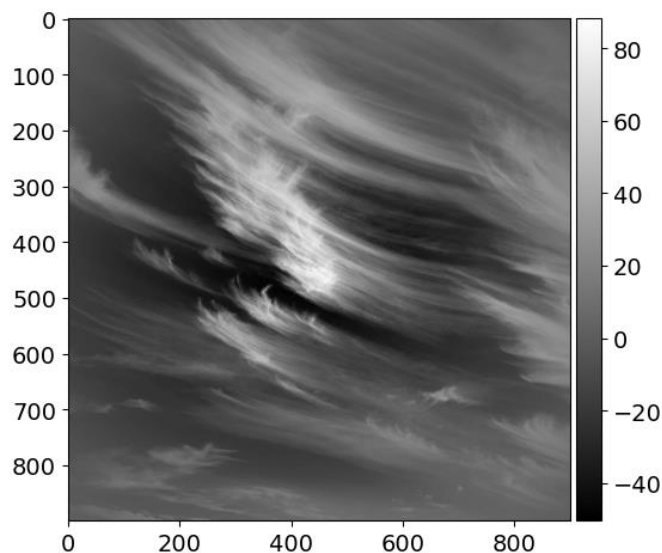


Рис. 3.2. Ілюстрація попередньої обробки вихідного зображення

Наступним кроком є знаходження спектра обробленого зображення за допомогою перетворення ШПФ:

$$P = F[p] = X + jY \quad (3.3)$$

На основі знайденого спектру (3.3) визначимо амплітудно-частотну характеристику (АЧХ):

$$A_p = |P| = |X + jY| \quad (3.4)$$

Та фазо-частотної характеристики (ФЧХ):

$$\varphi_p = \arg P = \arg(X + jY) \quad (3.5)$$

АЧХ та ФЧХ вихідного та обробленого зображення наведені на рис. 3.3.

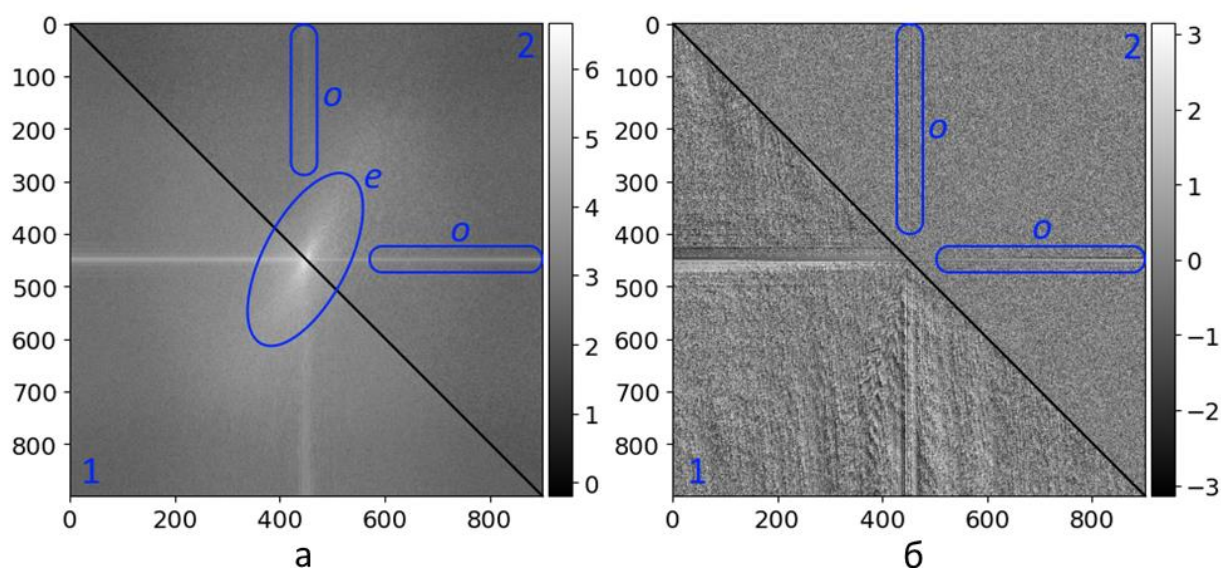


Рис. 3.3. Зображення АЧХ (а) у логарифмічному масштабі вихідного (1) та обробленого (2) зображень: о – осьові складові, е – еліптична складова

Спектри амплітуди та фази мають чітко виражені ділянки, зокрема складові вздовж осей або ж осьові складові (рис. 3.3, о) та еліптичну складову (рис. 3.3, е). При порівнянні частотних характеристик вихідного та обробленого зображення, очевидно, що використання вагової функції W в незначній мірі впливає на АЧХ, зменшуючи прояв осьових складові (рис. 3.3, а). В свою чергу вагова функція W в значній мірі впливає на АЧХ, зменшуючи зашумленість характеристики, що спрощує її аналіз.

Для генерації хмарного покриву за допомогою СС необхідно провести аналіз і відтворити специфічні особливості спектральних характеристик. Спершу проведемо аналіз осьових складових ФЧХ.

Апроксимація ФЧХ є складною задачею, оскільки закон розподілу не є чітко встановленим, на відміну від АЧХ. Тому для визначення впливу осьових складових на процес генерації зображення хмарного покриву доцільно повторно використати СС. Це дозволить відтворити осьові складові ФЧХ, одночасно генеруючи випадкову фазову характеристику.

Для цього спершу знайдемо спектр ФЧХ досліджуваного зображення:

$$\Phi = F[\varphi_p] = X + jY \quad (3.6)$$

А також її АЧХ:

$$A_\Phi = |\Phi| = |X + jY| \quad (3.7)$$

Та ФЧХ:

$$\varphi_\Phi = \arg \Phi = \arg(X + jY) \quad (3.8)$$

ФЧХ в обробці зображень відіграє значно важливішу роль ніж АЧХ, оскільки саме фаза несе в собі інформацію про такі основні особливості зображення (межі об'єктів, текстура та інше) [120], [121].

Враховуючи це, для генерації випадкової фази необхідно зберегти знайдену ФЧХ φ_Φ одночасно змінюючи АЧХ A_Φ . Для цього використаємо СС у показниковій формі:

$$\varphi_{out} = F^{-1} \left[A_\Phi R C_{PSD} \exp(\varphi_\Phi) \right], \quad (3.9)$$

де A_Φ та φ_Φ – АЧХ та ФЧХ досліджуваної фази відповідно, R – випадкова складова, C_{PSD} – коефіцієнт нормалізації.

Маніпуляції з АЧХ призводять до зміни енергії сигналу, що може спричинити до помітної розбіжності в значеннях між оригінальним та модифікованим сигналом. Для усунення даного недоліку необхідно проводити нормалізацію.

Спектральна густина потужності $S(f)$, яка визначає розподіл потужності сигналу по частотним компонентам, залежить від АЧХ наступним чином:

$$S(f) = A^2(f) \quad (3.10)$$

Враховуючи (3.10), коефіцієнт нормалізації (3.9) доцільно визначати як відношення коефіцієнтів СГП:

$$C_{PSD} = \sqrt{\frac{\sum_{i,j} A_{in}^2(i,j)}{\sum_{i,j} A_{out}^2(i,j)}}, \quad (3.11)$$

де $A_{in}(i, j)$ та $A_{out}(i, j)$ – АЧХ оригінального та модифікованого сигналів відповідно з номерами частотних складових (i, j) .

В стандартній процедурі СС R – гаусівський білий шум. Проте при використанні експоненційної форми всі множники пов’язані з АЧХ повинні мати додатні значення, тому у якості випадкової складової доцільно використовувати модуль білого шуму.

Використання модуля білого шуму R під час генерування ФЧХ даним методом призводить до того, що значення можуть виходити за межі типового діапазону $[-\pi; \pi]$. Тим не менш, нормалізація ФЧХ не є необхідною через періодичність базисних функцій у перетворенні Фур’є.

Застосування даного підходу дає наближену фазову характеристику (рис. 3.4, а). Для більшої наочності також наведено нормовану наближену ФЧХ (рис. 3.4, б). Очевидно, що отримана ФЧХ добре відтворює особливості та закон розподілу вихідної фазової характеристики (рис. 3.3, б).

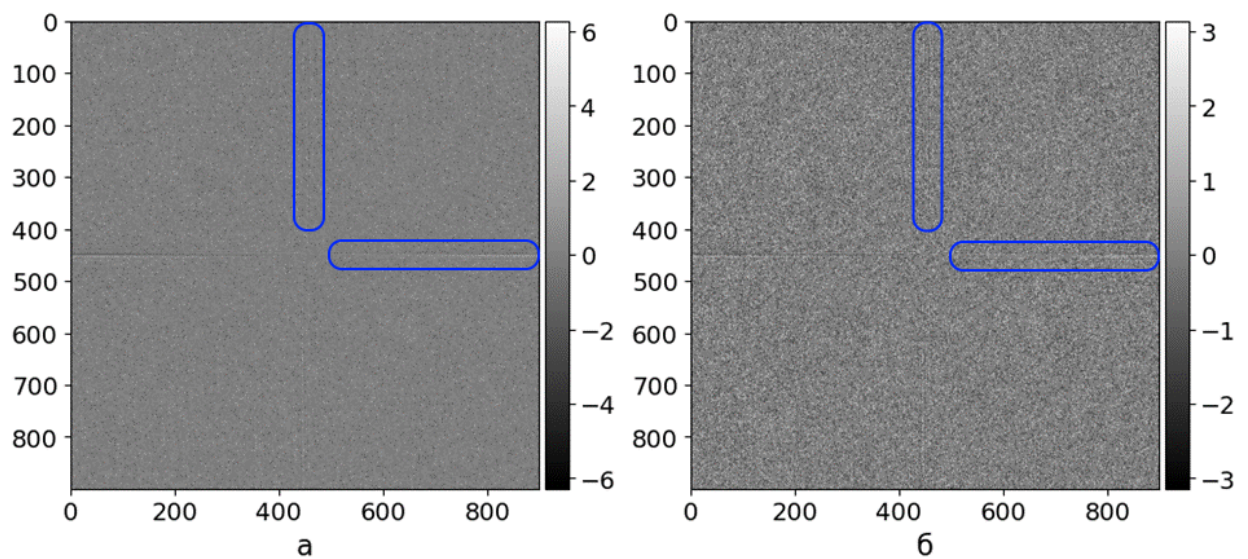


Рис. 3.4. Зображення наближеної ФЧХ: ненормована (а) та нормована (б)

Проведемо порівняння кількісних параметрів вихідної та згенерованої ФЧХ для визначення їх ступеню схожості. Для цього використаємо коефіцієнт якості наближення [122]:

$$GFC = \frac{\left| \sum_{i,j} R_m(i,j) R_e(i,j) \right|}{\sqrt{\sum_{i,j} R_m^2(i,j)} \sqrt{\sum_{i,j} R_e^2(i,j)}}, \quad (3.12)$$

де $R_m(i,j)$ та $R_e(i,j)$ – вихідна та наближена спектральні характеристики відповідно.

Формула (3.12) відповідає коефіцієнту кореляції Пірсона між двома наборами даних, які мають нульову постійну складову. Проте при роботі зі зображеннями їх спектральні характеристики не завжди мають нульове значення постійної складової. Тому для оцінки схожості спектральних характеристик надалі використовуватимемо узагальнення коефіцієнту кореляції Пірсона – нормовану функцію взаємної кореляції [123]:

$$R_{xy}(n) = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y} = \frac{K_{xy}(n)}{\sigma_x \sigma_y}, \quad (3.13)$$

де x та y – досліджувані сигнали; σ_x та σ_y – їх середньоквадратичні відхилення відповідно; $K_{xy}(n)$ – функція взаємної коваріації; n – просторове зміщення (лаг) між сигналами.

За теоремою про взаємну кореляцію [124] функція взаємної коваріації за допомогою ШПФ знаходиться як:

$$K_{xy}(n) = F^{-1}[\bar{X}Y], \quad (3.14)$$

де \bar{X} – комплексно-спряжений спектр x , Y – спектр y .

Відповідно з використанням ШФП, функцію взаємної кореляції знаходиться як:

$$R_{xy}(n) = \frac{F^{-1}[\bar{X}Y]}{\sigma_x \sigma_y} \quad (3.15)$$

А також функції автокореляції, враховуючи теорему Вінера-Хінчіна [125], [126]:

$$\Psi_x(n) = \frac{F^{-1}[A_x^2]}{\sigma_x^2}, \quad (3.16)$$

де A_x – АЧХ сигналу x .

Функція взаємної кореляції для наближеної ненормованої ФЧХ (рис. 3.4, а) та вихідної ФЧХ (рис. 3.3, б) наведена на рис. 3.5.

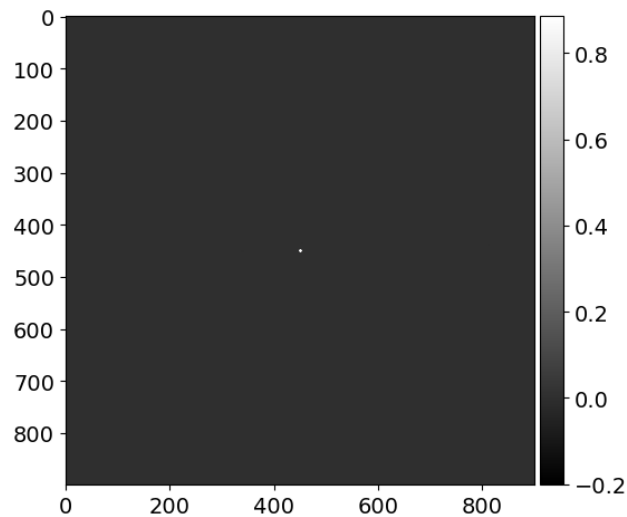


Рис. 3.5. Зображення функції взаємної кореляції вихідної та згенерованої ненормованої ФЧХ

Отримана функція є двовимірною, тому зміщення n визначається як норма частотного вектору (1.4). Пік даної характеристики розташований по центру досліджуваного зображення, що відповідає нульовому зміщенню між сигналами. Дане значення становить $R(n=0)=0,88533$, що свідчить про сильний позитивний лінійний зв'язок між згенерованою та вихідною ФЧХ, тобто згенерована фазова характеристика добре наслідує особливості вихідної ФЧХ.

Аналіз подібних двовимірних характеристик (рис. 3.5) іноді важко проводити, оскільки вони є ненаочними. Тому надалі подібні двовимірні графіки будемо зводити до одновимірних за допомогою радіального усереднення, яке є усередненням у полярних координатах для фіксованого радіусу і всіх значень кута повороту:

$$f_{avg}(n) = \frac{1}{N(n)} \sum_{\|x,y\|=n} f(x,y), \quad (3.17)$$

де $N(n)$ – кількість точок на відстані n від початку координат.

Радіально усереднена (рис. 3.5) за допомогою (3.17) наведена на рис. 3.6.

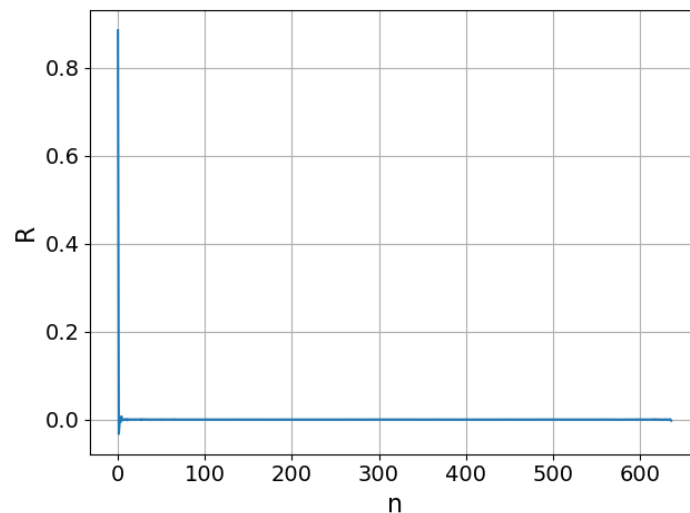


Рис. 3.6. Графік радіально усередненої функції взаємної кореляції вихідної та згенерованої ненормованої ФЧХ

Проаналізуємо функцію автокореляції досліджуваних фазових характеристик. Для вихідної і згенерованої ФЧХ вони однакові і наведені на рис. 3.7. Дана характеристика приймає нульове значення для всіх значень зміщення, окрім нульового, де вона представляє дельта імпульс. Тобто функція корелює сама з собою лише при нульовому зміщенні, що свідчить про те, що вона є білим шумом або ж суперпозицією білих шумів [127], [128].

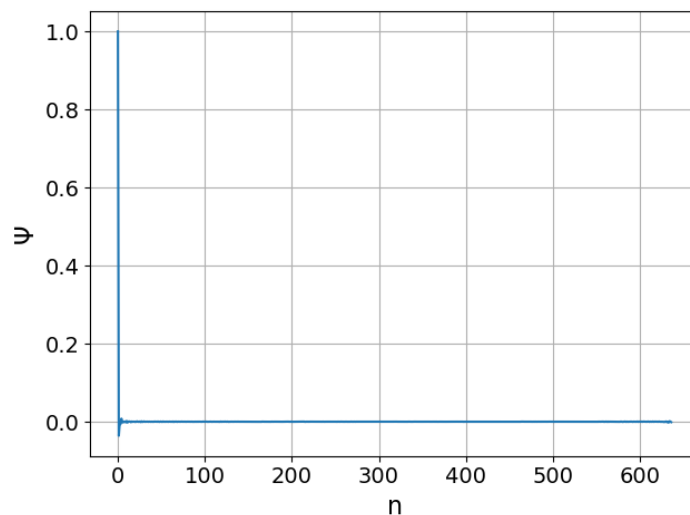


Рис. 3.7. Графік радіально усередненої функції автокореляції вихідної та згенерованої ФЧХ

Також проведемо аналіз статистичних параметрів досліджуваних характеристик, які наведені в таблиці 3.1.

Таблиця 3.1. Статистичні параметри досліджуваних ФЧХ

	Вихідна ФЧХ	Згенерована ФЧХ
μ	$1,1636 \cdot 10^{-5}$	$1,8114 \cdot 10^{-5}$
σ	1,8114	1,8114

З таблиці 3.1 видно, що згенерована ФЧХ, окрім наслідування особливостей вихідної характеристики, добре наближує статистичні параметри, що свідчить також про якісне відтворення закону розподілу вихідної характеристики.

Відновимо зображення хмарного покриву, використовуючи вихідну АЧХ (рис. 3.3, а) та згенеровану ФЧХ (рис. 3.4, а) наступним чином:

$$p_{out} = F^{-1} [A_p \exp(\varphi_{out})], \quad (3.18)$$

де A_p – вихідна АЧХ, φ_{out} – згенерована ФЧХ.

Результати використання (3.18) наведені на рис. 3.8, де обидва зображення (ненормоване та нормоване) побудовані у діапазоні значень вихідного зображення.

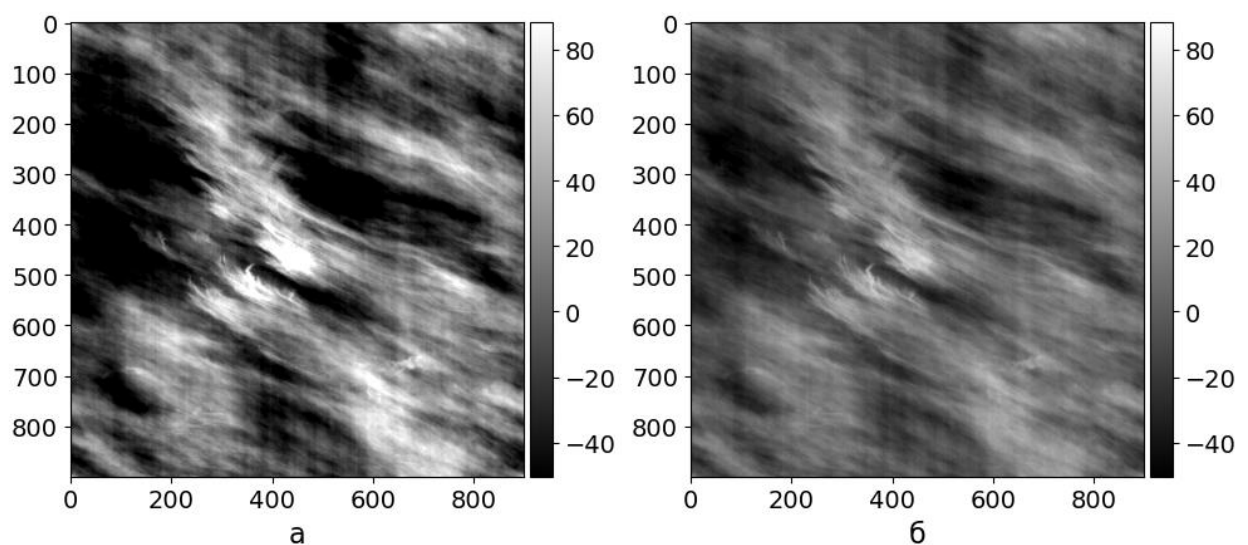


Рис. 3.8. Зображення відновленого зображення: ненормоване (а), нормоване

(б)

Очевидно, що ненормоване (рис. 3.8, а) та вихідне зображення (рис. 3.2) мають різні діапазони, що вказує на те, що нормалізації за допомогою (3.11) недостатньо. При цьому приведення до вихідного діапазону зображення (рис. 3.8, б) показує більшу візуальну схожість з оригінальним зображенням (рис. 3.2).

Експериментально було встановлено, що в певних випадках можливі ситуації при яких $C_{PSD} \approx 1$, що свідчить про рівність енергії вихідного і згенерованого сигналів. Одночасно з цим спостерігається зміна АЧХ або ж ФЧХ, що може призвести до наявності областей у створеному зображенні, які лежать поза діапазоном значень вихідного зображення.

Щоб привести згенероване зображення до діапазону значень вихідного, використовується лінійна регресія [129]:

$$\hat{a} = \frac{\langle p_{out} p \rangle - \langle p_{out} \rangle \langle p \rangle}{\langle p^2 \rangle - \langle p \rangle^2} \quad (3.19)$$

$$\hat{b} = \langle p \rangle - \hat{a} \langle p_{out} \rangle,$$

де p – вихідне зображення, p_{out} – відновлене зображення.

Таким чином, нормоване відтворене зображення (рис. 3.8, б) знаходиться за наступною формулою:

$$p'_{out} = \hat{a} p_{out} + \hat{b} \quad (3.20)$$

Надалі при використанні підходу СС будуть застосовуватися нормалізація АЧХ за допомогою відношення коефіцієнтів СГП (3.11), а також згенерованого зображення за допомогою приведення до діапазону вихідного зображення за рахунок використання лінійної регресії (3.19).

Таким чином, генерація псевдовипадкової ФЧХ з використанням підходу СС дозволяє точно відтворити закон розподілу та особливості вихідної фазової характеристики, а саме – осьові складові. Також за допомогою автокореляційного тесту було встановлено, що вихідна ФЧХ є суперпозицією декількох білих шумів – фоновому шуму, а також осьових складових. Недоліком використання підходу СС є те, що він не дозволяє

керувати ступенем кореляції вихідної та згенерованої фазових характеристик. Також очевидним є те, що відновлене зображення має спотворення у виді вертикальних ліній постійної яскравості (рис. 3.8, б)

Враховуючи, що вихідна ФЧХ є суперпозицією декількох білих шумів, для генерації фазової характеристики доцільно використовувати наступний підхід:

$$\varphi_{out} = \varepsilon \varphi_P + (1 - \varepsilon) \varphi_R, \quad (3.21)$$

де φ_P – фазова характеристика вихідного зображення, φ_R – випадкова фаза, $\varepsilon = 0 \dots 1$ – коефіцієнт пропорційності.

Перевагою використання (3.21) для генерації фазової характеристики є простота контролю ступеню кореляції згенерованої ФЧХ з вихідною за рахунок наявності коефіцієнту пропорційності ε . Також у якості вихідної ФЧХ може використовуватися усереднена фазова характеристика, яка описує певний набір зображень хмарного покриву певного типу.

Наступний крок включає апроксимацію еліптичної складової АЧХ. Для цього необхідно визначити кут повороту еліптичної складової, а також кількісно визначити затухання вздовж її великої та малої півосей.

Для визначення кута повороту необхідно виділити еліптичну складову АЧХ:

$$A_e = \begin{cases} A_p, & A_p \geq th \\ 0, & A_p < th \end{cases}, \quad (3.22)$$

де th – деяке порогове значення, що відповідає межах еліптичної складової АЧХ.

Дане порогове значення визначається експериментально і для АЧХ досліджуваного зображення в логарифмічному масштабі дорівнює $th = 3,908$. Тим не менш, цей підхід не дозволяє повністю відокремити еліптичну складову від осьових (рис. 3.9, а) що призводить до потенційних неточностей у визначенні кута повороту. Щоб зменшити вплив осьових складових, ділянки спектру, які не можна чітко класифікувати, замінюють нульовими значеннями

або найближчими компонентами (горизонтальними чи вертикальними), що продемонстровано на рис. 3.9, б.

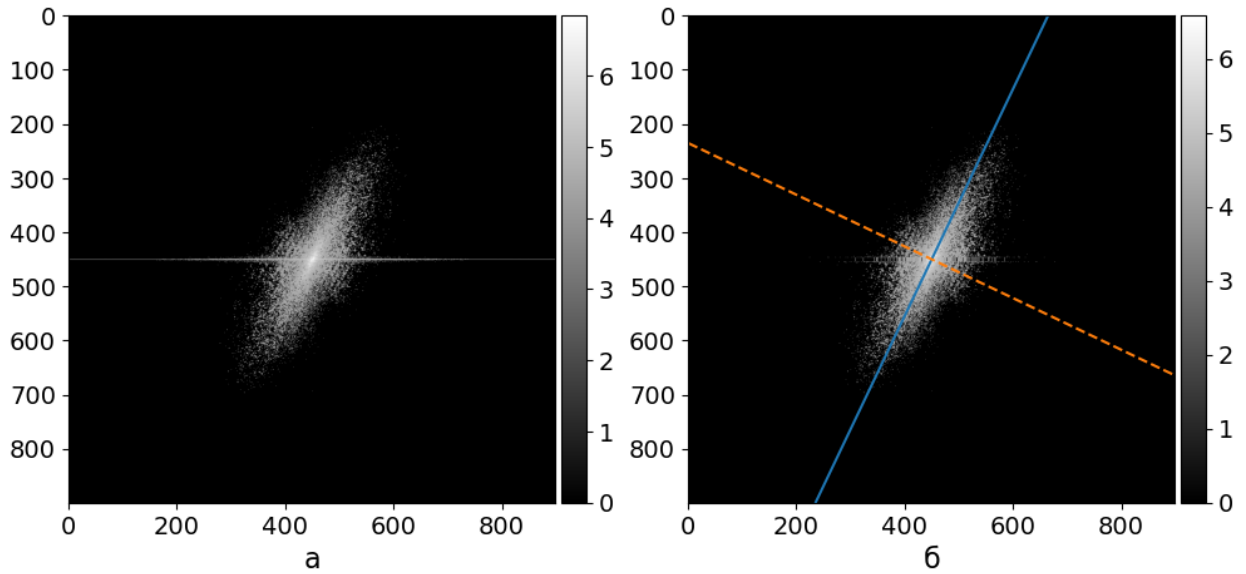


Рис. 3.9. Ілюстрація з прикладом виділення еліптичної складової АЧХ: без (а) та з заміною (б) осьових складових. Велика та мала півосі еліптичної складової показані за допомогою суцільної та штрихової ліній відповідно

Після цього за допомогою лінійної регресії визначається кут повороту еліптичної складової. Для посилення впливу низькочастотних компонентів на результат регресії використовується вагова функція, а саме – СГП досліджуваного зображення. Також еліптична складова АЧХ є симетричною відносно постійної складової спектру, тому в даному випадку постійна складова регресії рівна нулю [130]. Враховуючи це, за допомогою методу найменших квадратів необхідно мінімізувати наступну різницю:

$$SSE = \sum_{i=1}^n \sum_{j=1}^m A_e^2(k_y - \hat{m}k_x), \quad (3.23)$$

де A_e^2 – вагова функція у виді СГП досліджуваного зображення; k_x , k_y – номери частотних компонентів вздовж осі абсцис та ординат відповідно, \hat{m} – кутовий коефіцієнт. Слід зазначити, що всі змінні в рівнянні, окрім кутового коефіцієнту, залежать від i та j , проте ця залежність опущена в позначеннях для спрощення запису.

Далі обчислюється часткова похідна по кутовому коефіцієнту:

$$\frac{\partial}{\partial \hat{m}} SSE = -2 \sum_{i=1}^n \sum_{j=1}^m A_e^2 (k_y - \hat{m} k_x) k_x = 0 \quad (3.24)$$

Звідки кутовий коефіцієнт визначається як:

$$\hat{m} = - \frac{\sum_{i=1}^n \sum_{j=1}^m A_e^2 k_y k_x}{\sum_{i=1}^n \sum_{j=1}^m A_e^2 k_x^2} = - \frac{\langle k_y k_x \rangle}{\langle k_x^2 \rangle} \quad (3.25)$$

Середнє значення $\langle \cdot \rangle$ в (3.25) визначається наступним чином:

$$\langle f(i, j) \rangle = \frac{\sum_{i=1}^n \sum_{j=1}^m A_e^2 f(i, j)}{\sum_{i=1}^n \sum_{j=1}^m A_e^2} \quad (3.26)$$

Регресійний аналіз передбачає наявність залежної змінної та однієї чи кількох незалежних змінних. Рівняння (3.25) передбачає, що незалежною змінною є k_x , але експериментально встановлено, що іноді доцільно вибрати як незалежну змінну k_y . У цьому випадку рівняння (3.25) набуває вигляду:

$$\hat{m} = - \frac{\langle k_y k_x \rangle}{\langle k_y^2 \rangle} \quad (3.27)$$

Приклад використання (3.25) та (3.27) для знаходження кута повороту еліптичної складової АЧХ для досліджуваного зображення наведені в таблиці 3.2. На основі наведених даних побудовані великі півосі еліптичної складової (рис. 3.10).

Таблиця 3.2. Визначення кута повороту за допомогою регресії

Незалежна змінна	\hat{m}	$\gamma, ^\circ$
k_x	-1,032	-27,349
k_y	-0,477	-59,118

Для досліджуваного зображення використання k_y забезпечує кращу оцінку якості кута повороту (рис. 3.10, а). Це пояснюється наявністю осьових

складових вздовж осі абсцис та складністю їх відокремлення від еліптичної складової, що в результаті призводить до неправильної оцінки кута повороту у випадку використання k_x у якості незалежної змінної (рис. 3.10, б).

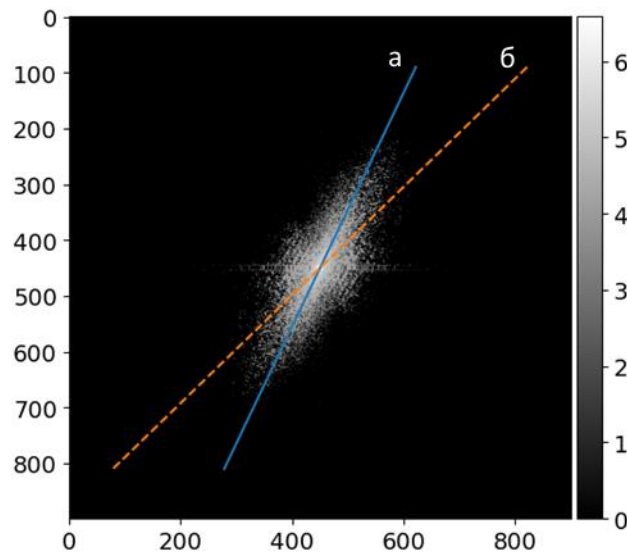


Рис. 3.10. Ілюстрація визначення кута повороту за допомогою регресії:
абсциса (а) та ордината (б) у якості незалежної змінної

Даний недолік можна усунути за рахунок використання методу основних компонент (МОК) – статистичним методом, який дозволяє зменшити розмірність вихідних даних, одночасно зберігаючи при цьому якомога більше інформації про варіативність вихідних даних [131], [132]. Одним з найпоширеніших подібних способів є декомпозиція матриці взаємної коваріації на власні значення та вектори. Дана процедура складається з наступних кроків [133]:

1. Для випадкових величин X та Y визначається матриця взаємної коваріації:

$$\Sigma_{XY} = \text{cov}(X, Y) = E[(X - \mu_X)(Y - \mu_Y)^T], \quad (3.28)$$

де μ_X та μ_Y – середнє значення випадкових величин X та Y відповідно. Варто зауважити, що в контексті МОК великі літери використовуються для позначення випадкових величин та матриць, а не спектру.

2. Розклад матриці взаємної коваріації на власні значення та вектори:

$$\Sigma_{XY} = Q\Lambda Q^{-1}, \quad (3.29)$$

де Q – матриця власних векторів матриці Σ_{XY} , де кожен стовбець визначає власний вектор; Λ – діагональна матриця, де кожен діагональний елемент представляє власне значення матриці Σ_{XY} ; Q^{-1} – обернена матриця.

Власний вектор – ненульовий вектор, для якого виконується нерівність [134]:

$$Aq = \lambda q, \quad (3.30)$$

де q – власний вектор квадратної матриці A , λ – власне значення (скаляр) матриці A .

У багатьох практичних випадках, особливо в контексті коваріаційних матриць, які є симетричними, матриця Q буде ортогональною, тобто $Q^{-1} = Q^T$. Отже, для симетричних матриць, власне розкладання також можна записати як [135]:

$$\Sigma_{XY} = Q\Lambda Q^T \quad (3.31)$$

Власний вектор $q_i \in Q$ та власне значення $\lambda_i \in \Lambda$ з відповідними номерами описують компоненту (вісь) порядку $i = 1 \dots n$. Власний вектор q_i визначає напрям осі, у якому вихідний набір даних характеризується найбільшою дисперсією. В свою чергу, власне значення λ_i описує кількісну міру варіативності вихідних даних у напрямку власного вектора з відповідним номером. Зазвичай при аналізі основних компонентів їх сортують у порядку спадання власних значень з метою спрощення аналізу та виділення найбільш значущих елементів. Існує чимала кількість ітеративних методів розкладання квадратної матриці на власні вектори, проте надалі буде використовуватись QR-алгоритм [136].

3. Нормалізація матриці власних векторів Q . Оскільки Σ_{XY} є ортогональною, власні вектори також будуть ортогональними один одному, але не обов'язково матимуть одиничну довжину, тому зазвичай проводять їх нормалізацію [132]:

$$\hat{q}_i = \frac{q_i}{\|q_i\|}, \quad (3.32)$$

де q_i – власний вектор, $\|q_i\|$ – його норма, $i = 1 \dots n$ – порядковий номер.

Звідки кут повороту i -го власного вектора визначається як:

$$\gamma_i = \arctan\left(\frac{q_{iy}}{q_{ix}}\right), \quad (3.33)$$

де q_{ix} та q_{iy} – складові власного вектора q_i вздовж відповідних осей.

Враховуючи ортогональність основних компонентів, їх розрахунок дозволить визначити напрями великої та малої півосей еліптичної складової АЧХ. Значення основних компонентів та відповідних кутів повороту β , отриманих за допомогою використання МОК до АЧХ досліджуваного зображення, наведені в таблиці 3.3. За даними результатами було побудовано осі еліптичної складової (рис. 3.11). З рис. 3.11 видно, що перша основна компонента відповідає великій півосі еліптичної складової і добре узгоджується з результатами, отриманими за допомогою лінійної регресії з незалежною змінною k_y .

Таблиця 3.3. Визначення кута повороту за допомогою МОК

i	\hat{q}	λ	$\gamma, ^\circ$
1	(-0,863; 0,506)	8783,816	-59,656°
2	(0,506; -0,863)	1265,621	-149,702°

При зміні порядку змінних в коваріаційній матриці, наприклад з Σ_{xy} на Σ_{yx} , власні вектори обчислюються відповідно до нового розташування даних. Проте даний набір буде описувати ті самі вихідні дані і, виходячи з визначення власного вектора (3.30), може мати протилежний напрям, який компенсується за рахунок протилежного знаку матриці власних значень. Таким чином, значення кута повороту не залежить від способу розрахунку коваріаційної

матриці, що є основною перевагою МОК у порівнянні з лінійною регресією при визначенні кута повороту еліптичної складової.

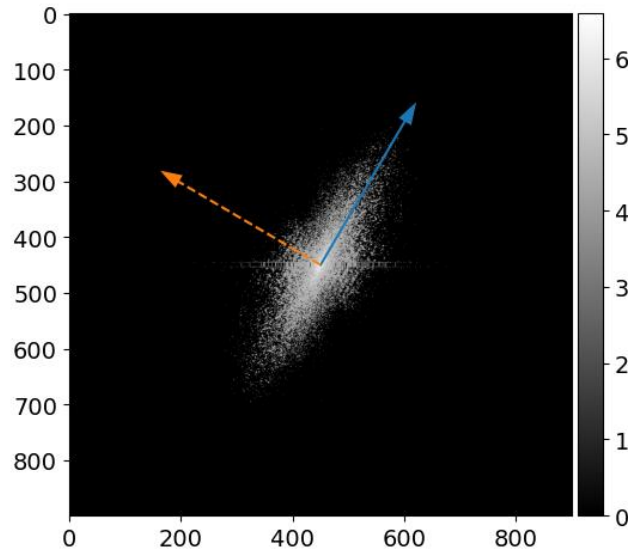


Рис. 3.11. Ілюстрація визначення кута повороту еліптичної складової за допомогою МОК: перша (суцільна) та друга (штрихова) основні компоненти

Наступним кроком є визначення затухання АЧХ вздовж відповідних півосей еліптичної складової. Зображення хмарного покриву описується $1/f$ шумом, тобто його СГП має закон розподілу (1.3). Враховуючи це, закон розподілу АЧХ матиме наступний вигляд:

$$A(f) = \sqrt{S(f)} = \sqrt{\frac{c'}{f^{\alpha'}}} = cf^{-\alpha}, \quad (3.34)$$

де $\alpha = \alpha'/2$ – коефіцієнт затухання, $c = \sqrt{c'}$ – деяка стала.

Використання (3.34) у якості апроксимуючої функції вимагає визначення двох параметрів: коефіцієнта затухання та константи. Застосування методу найменших квадратів у цьому контексті вимагає розв'язування системи двох трансцендентних рівнянь, що потребує застосування чисельних методів [137]. Це, в свою чергу, значно ускладнює алгоритм і знижує швидкість його обчислень.

Тому з метою спрощення доцільно провести нормалізацію АЧХ, що виключить необхідність визначення константи:

$$A'_p = \frac{A_p}{\max(A_p)} \quad (3.35)$$

Використання методу найменших квадратів дає випадок експоненціальної регресії, який спрощується до лінійного випадку [129]. Враховуючи це, необхідно мінімізувати наступну різницю:

$$SSE = \sum_{i=1}^n \sum_{j=1}^m \left[\log A'_p - (\hat{\alpha} \log k) \right]^2, \quad (3.36)$$

де A' – нормована АЧХ, k – норма частотного вектора, $\hat{\alpha}$ ц – коефіцієнт затухання. Як і у випадку (3.23), всі змінні в рівнянні, окрім коефіцієнту затухання, залежать від i та j , проте ця залежність опущена для спрощення запису.

Звідки часткова похідна по коефіцієнту затухання:

$$\frac{\partial}{\partial \hat{\alpha}} SSE = 2 \sum_{i=1}^n \sum_{j=1}^m (\log A'_p + \hat{\alpha} \log k) \log k = 0 \quad (3.37)$$

Вирішивши (3.37) відносно коефіцієнту затухання, отримуємо:

$$\hat{\alpha} = - \frac{\sum_{i=1}^n \sum_{j=1}^m \log k \log A'_p}{\sum_{i=1}^n \sum_{j=1}^m \log^2 k} = - \frac{\langle \log k \log A'_p \rangle}{\langle \log^2 k \rangle}, \quad (3.38)$$

де A' – нормована АЧХ, k – норма частотного вектора вздовж великої та малої півосей еліптичної складової.

Значення коефіцієнтів затухання, отримані за допомогою (3.38), складають для великої півосі $\alpha_{maj} = 1,447$ та $\alpha_{min} = 1,591$ для малої. При цьому Приклади апроксимації нормованої АЧХ вздовж осей еліптичної складової наведені на рис. 3.12.

На основі отриманих значень коефіцієнтів затухання визначається коефіцієнт асиметрії, який визначає наскільки затухання вздовж малої півосі більше по відношенню до затухання вздовж великої півосі:

$$\eta = \sqrt{\frac{\alpha_{min}}{\alpha_{maj}} - 1} \quad (3.39)$$

З урахуванням асиметрії, еліптична складова АЧХ апроксимується наступною функцією розподілу

$$A_{out} = \sqrt{(1-\eta)M^2 + \eta N^2}, \quad (3.40)$$

де η – коефіцієнт асиметрії, який визначає ступінь розтягнення симетричної складової $M = 1/k^\alpha$ вздовж великої півосі еліптичної складової $N = k_y - \hat{m}k_x$; k – норма частотного вектору; k_x та k_y – складові частотного вектору вздовж відповідних осей; \hat{m} – кутовий коефіцієнт, який відповідає куту повороту еліптичної складової.

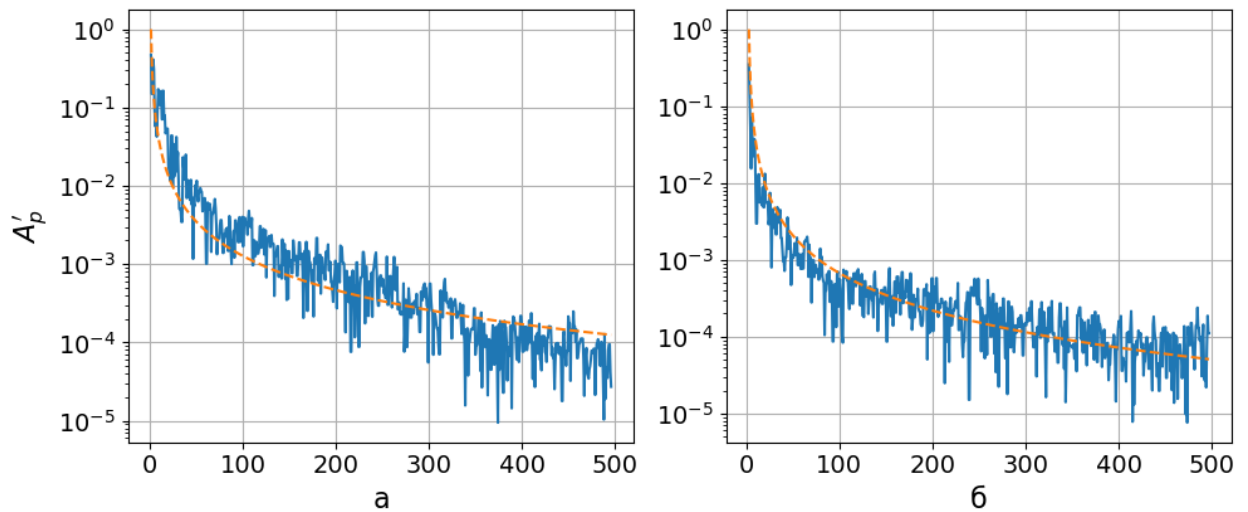


Рис. 3.12. Графік прикладу апроксимації нормованої АЧХ вздовж: великої (а) та малої (б) півосей

Для оцінки впливу еліптичної складової АЧХ, проведемо відновлення досліджуваного зображення хмарного покритву, використовуючи наближену АЧХ (3.40) та вихідну ФЧХ (рис. 3.3, б) наступним чином:

$$p' = F^{-1} \left[A_{out} C_{PSD} \exp(\varphi_P) \right], \quad (3.41)$$

де A_{out} – наближена АЧХ, φ_P – вихідна ФЧХ, C_{PSD} – коефіцієнт нормалізації.

Значення параметрів, необхідних для наближення еліптичної складової АЧХ, наведені в таблиці 3.4. На основі даних значень побудовано апроксимація еліптичної складової АЧХ (рис. 3.13, а) та відновлене зображення хмарного

покриву (рис. 3.13, б). До отриманого зображення застосовано всі описані раніше способи нормалізації.

На рисунку рис. 3.13 ілюструється приклад відтворення зображення хмарного покриву за допомогою апроксимованої еліптичної складової АЧХ разом з оригінальною ФЧХ. Отримане зображення демонструє збереження основних характеристик початкового зображення, забезпечуючи високу достовірність представлення.

Таблиця 3.4. Значення параметрів апроксимації еліптичної складової

$\gamma, ^\circ$	-59,656
α_{maj}	1,447
α_{min}	1,591
η	0,315

Експериментально встановлено, що як і у випадку ФЧХ, осьові складові АЧХ досліджуваного зображення мають доволі складний закон розподілу, який важко відтворити. Також їх вплив на згенеровані зображення є досить малим, тому ними можна знехтувати.

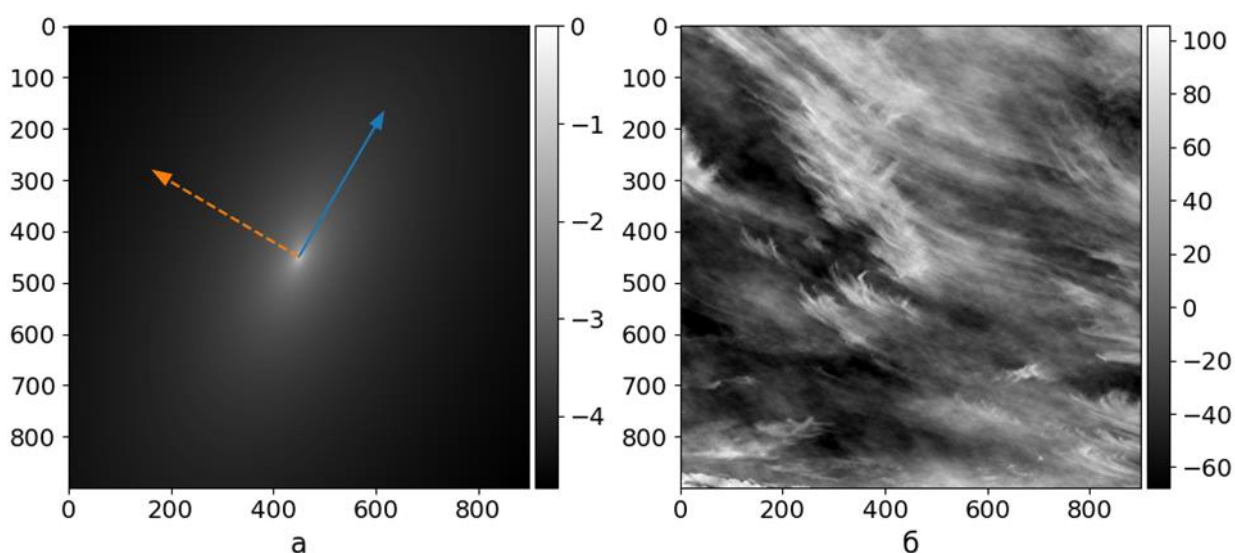


Рис. 3.13. Приклад наближеної АЧХ (а) та відновленого зображення (б)

Важливим кроком моделювання хмарного покриву є визначення його фізичних розмірів. Вихідне зображення для визначення параметрів моделі має

певні фізичні розміри, тому всі згенеровані зображення (незалежно від їх розміру у пікселях) матимуть фізичні розміри, що й вихідне. Приклад подібної поведінки наведено на рис. 3.14, де на основі наведених в таблиці 3.4 параметрів та φ_{out} при $\varepsilon = 0$ згенеровано зображення хмарного покриву для різних розмірів зображення у пікселях. Очевидно, що обидва зображення мають однакові фізичні розміри.

Для генерації зображення хмарного покриву з особливостями досліджуваного зображення, а також бажаними фізичними розмірами, необхідно проводити його масштабування [138]. Для цього спершу визначаються фізичні розміри вихідного та згенерованого зображення наступним чином:

$$\begin{aligned} P_1 &= S_1 N_1 \\ P_2 &= S_2 N_2 \end{aligned} \quad (3.42)$$

де S_1 та S_2 – масштаб вихідного та згенерованого зображення, вимірюється в одиницях довжини на піксель; N_1 та N_2 – розміри у пікселях відповідних зображень; P_1 та P_2 – фізичні розміри вихідного та згенерованого зображення в одиницях довжини.

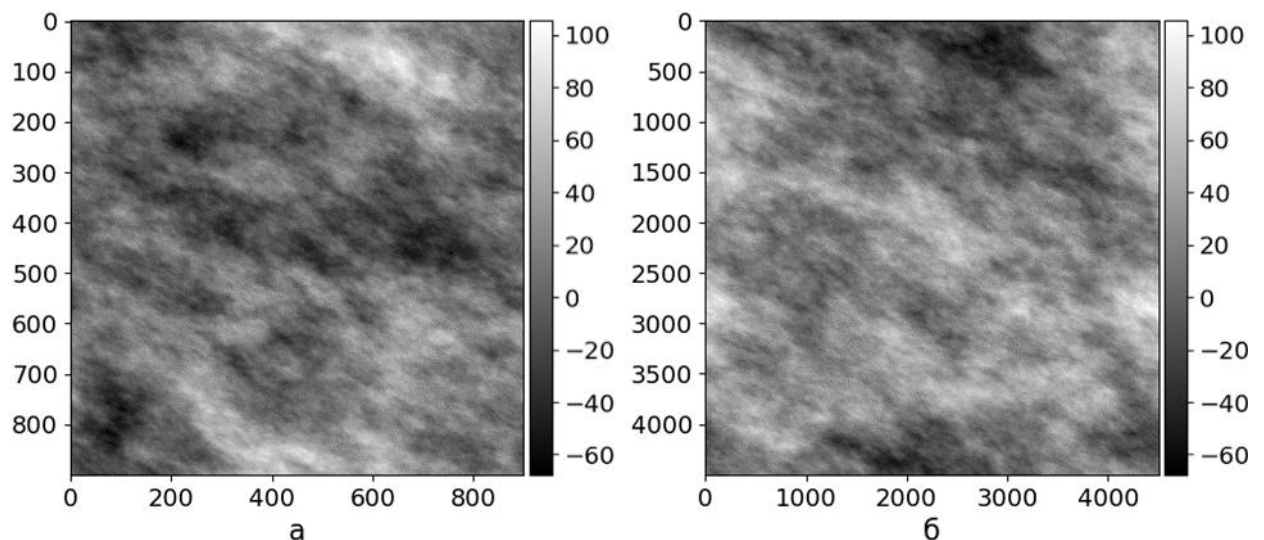


Рис. 3.14. Приклад зображень з однаковим фізичним розміром при різних розмірах в пікселях: розмір вихідного зображення (а), п'ятикратне збільшення (б)

Незалежно від розміру зображення у пікселях, фізичний розмір залишається незмінним, тобто:

$$S_1 N_1 = S_2 N_2 \quad (3.43)$$

Вирішивши (3.43) відносно розміру у пікселях вихідного зображення, маємо:

$$N_2 = \frac{S_1 N_1}{S_2} = r N_1, \quad (3.44)$$

де $r = S_1/S_2$ – коефіцієнт масштабування, який визначає наскільки необхідно збільшити зображення в пікселях, щоб досягти бажаного фізичного масштабу.

Знайдений розмір зображення N_2 визначає кількість пікселів, яку має мати нове зображення, щоб зберегти той самий фізичний розмір, але з іншим масштабом. Для зміни фізичного розміру зі згенерованого зображення з розміром N_2 виділяється зображення з вихідним розміром N_1 [139]. Приклад використання подібного масштабування з $r = 5$ наведено на рис. 3.15.

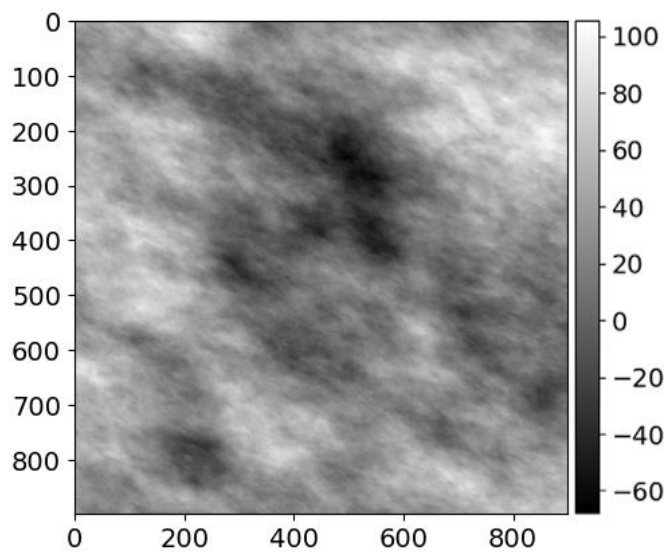


Рис. 3.15. Ілюстрація зміни фізичного масштабу при $r = 5$

Таким чином, для генерування зображень хмарного покриву певного типу необхідно знайти чотири параметри: коефіцієнт затухання вздовж великої α_{maj} та малої α_{min} півосей еліптичної складової, кут повороту еліптичної

складової γ та ФЧХ φ_{out} (вихідна або псевдовипадкова). Тобто розроблена модель хмарного покриву містить чотири параметри.

Блок-схема алгоритму визначення параметрів запропонованої моделі хмарного покриву зображено на рис. 3.16.



Рис. 3.16. Блок-схема алгоритму визначення параметрів запропонованої моделі хмарного покриву

Для визначення параметрів моделі і генерації хмарного покриву у якості вихідних даних може використовуватися як одичне зображення, так і їх певний набір. У другому випадку значення параметрів моделі визначаються для кожного зі зображень, а для генерації використовується усереднене значення знайдених параметрів або їх лінійні комбінації.

При використанні формули (3.41), окрім нормалізації за коефіцієнтами СГП (3.11), необхідно використовувати нормалізацію за допомогою лінійної регресії (3.19), що дозволяє привести згенероване зображення до діапазону

значень вихідного зображення. При необхідності також відбувається зміна фізичного масштабу вихідного зображення за формулою (3.44).

3.2. Порівняльний аналіз

Проведемо порівняльний аналіз запропонованої моделі хмарного покриву у порівнянні зі стандартним підходом СС на основі ШПФ [90]. Порівняння моделей виконуватиметься на основі точності наближення вихідного зображення (рис. 3.1). Модель хмарного покриву у випадку стандартного СС складається з двох характеристик, а саме АЧХ та ФЧХ. Оскільки порівнюватиметься точність наближення вихідного зображення, обидві моделі використовуватимуть ФЧХ вихідного зображення (рис. 3.3, б, складова 1).

У випадку стандартного СС АЧХ є фігурою обертання, тобто є симетричною. Тому для апроксимації амплітудної характеристики необхідно визначити лише один параметр – коефіцієнт затухання. Для цього використаємо знайдену регресійну формулу для коефіцієнту затухання (3.38). У якості вихідних даних для даної формули використовуються нормована АЧХ та норма частотного вектору, а не їх зрізи як у випадку запропонованої моделі.

Для порівняння моделей використовуватиметься використовується корінь з середньоквадратичної похибки (2.18), яка дозволить оцінити абсолютну похибку наближення зображення.

Також у якості додаткової метрики використаємо індекс структурної подібності [140]. Значення даного параметру лежать у діапазоні 0...1, що дозволяє за його допомогою оцінити ступінь візуальної схожості двох зображень у спосіб, який притаманний людському сприйняттю [141]. Даний індекс визначається наступним чином:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\text{cov}(x, y) + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (3.45)$$

де x та y – масив значень пікселів порівнюваних зображень; μ_x та μ_y – середні значення x та y ; σ_x та σ_y – середньоквадратичне відхилення x та y ; $\text{cov}(x, y)$ – коваріація x та y ; $c_1 = (k_1 L)^2$ та $c_2 = (k_2 L)^2$ – константи, які залежать від динамічного діапазону порівнюваних зображень $L = 255$; $k_1 = 0,01$ та $k_2 = 0,03$.

Для зручності візуального порівняння повторно наведемо досліджуване зображення (рис. 3.17). Також для порівняння за допомогою методу перебору визначимо оптимальні значення параметрів запропонованої моделі. У якості метрики в процесі перебору використовуватиметься похибка $RMSE$. Визначатимуться лише значення параметрів коефіцієнту затухання вздовж малої півосі α_{min} (діапазон перебору 1...2) та коефіцієнту асиметрії η (діапазон перебору 0...1).

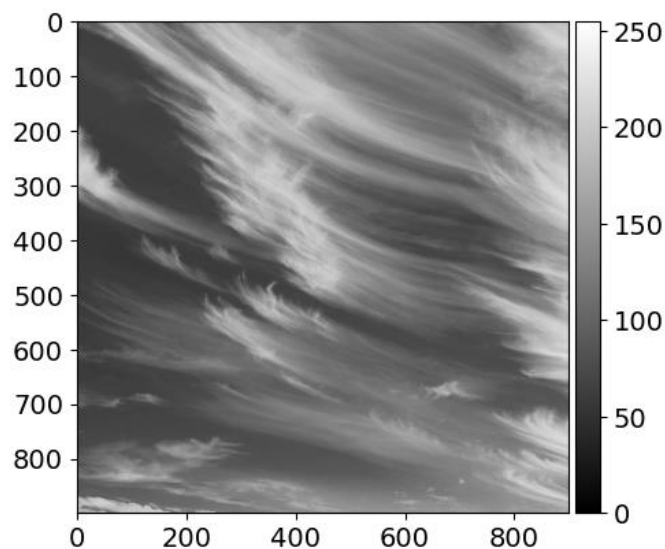


Рис. 3.17. Зображення досліджуваного хмарного покриву

Значення параметрів моделей, а також похибки апроксимації вихідного зображення (рис. 3.17) наведені в таблиці 3.5. На основі даних значень побудовані зображення хмарного покриву для кожної з моделей (рис. 3.18).

Таблиця 3.5. Параметри для порівняння моделей

	Запропонована модель	Метод перебору	Стандартний метод СС
$\gamma, ^\circ$	-59,656	-59,656	-
α_{maj}	1,447	-	-
α_{min}	1,591	1,551	1,572
η	0,315	0,327	-
<i>RMSE</i>	22,401	22,372	28,981
<i>SSIM</i>	0,534	0,531	0,416

При візуальному порівнянні апроксимованих зображень (рис. 3.18) очевидно, що запропонована модель забезпечує набагато кращу апроксимацію вихідного зображення (рис. 3.17). Запропонована модель за рахунок відтворення асиметрії еліптичної складової АЧХ передає більше високочастотних складових, що значно збільшує чіткість та контрастність згенерованого зображення.

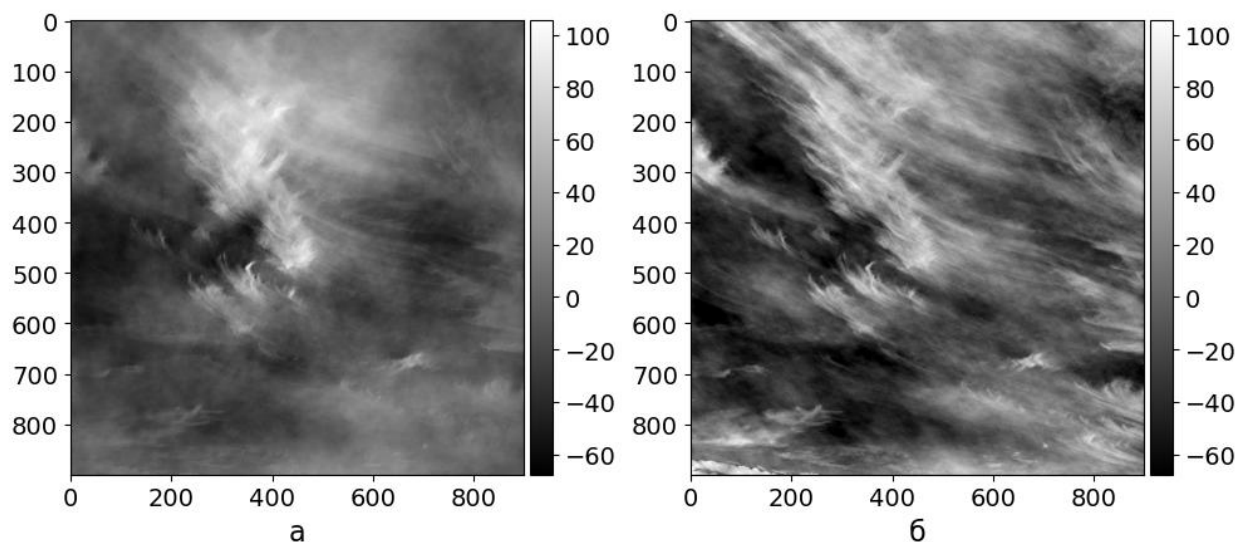


Рис. 3.18. Приклад апроксимації вихідного зображення: стандартний метод СС (а) та запропонований спосіб (б)

З таблиці 3.5 видно, що значення коефіцієнту затухання добре узгоджуються для всіх моделей, аналогічна ситуація спостерігається і для

коефіцієнту асиметрії. Запропонована модель забезпечує на 22,65% краще значення похибки апроксимації $RMSE$ та на 22,10% краще значення індексу структурної подібності $SSIM$ у порівнянні зі стандартним підходом.

При цьому метод перебору забезпечує дещо краще значення похибки $RMSE$ та гірше значення індексу структурної подібності $SSIM$. Дана різниця становить менше 1%, тому можна стверджувати, що значення параметрів запропонованої моделі є близькими до оптимальних. Ілюстрація методу перебору наведена на рис. 3.19.

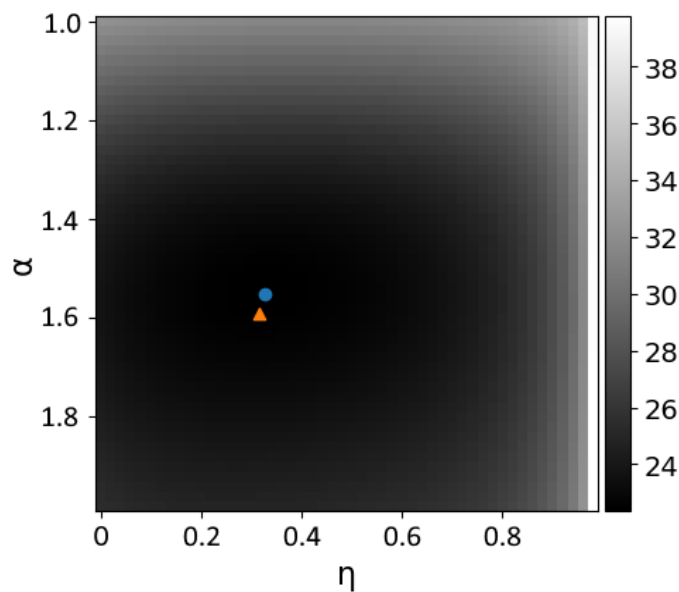


Рис. 3.19. Ілюстрація результату методу перебору: оптимальне значення параметрів (кругла), значення параметрів запропонованої моделі (трикутна точка). Кольорова шкала відображає значення похибки

Одночасно з цим, за рахунок нехтування осьовими складовими АЧХ та ряду інших спрощень при аналізі спектральних характеристик досліджуваного зображення, значення $SSIM$ є доволі малим. Проте для потреб моделювання часткового затінення масиву СП дана точність є задовільною.

Висновки до розділу

1. Створено математичну модель хмарного покриву, засновану на швидкому перетворенні Фур'є зображення хмар та використанні його спектральних характеристик.
2. Запропоновано використання вагової функції для мінімізації явища витікання спектру, що дозволяє знизити спотворення спектральних характеристик та спростити їх аналіз.
3. Запропоновано генерацію зображення хмарного покриву на основі використання псевдовипадкової фази, що дозволяє за знайденими параметрами відтворювати основні особливості вихідного зображення. Встановлено, що запропонована модель забезпечує на 22,65% краще значення похибки *RMSE* та на 22,10% краще значення індексу структурної подібності *SSIM* у порівнянні зі поширеним підходом спектрального синтезу.
4. Розроблено програмну реалізацію математичної моделі на основі мови програмування Python та його дистрибутиву Anaconda, яка дозволяє проводити генерацію хмарного покриву з відтворенням особливостей, притаманних для вихідних даних.

РОЗДІЛ 4. КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ

Визначення оптимальної топології масиву СП в умовах часткового затінення вимагає моделювання роботи СП, системи комутації, а також відтворення умов нерівномірної освітленості. Для цього використовується програмний пакет Matlab R2022b Simulink, який надає дозволяє проводити імітаційне моделювання нелінійних динамічних систем з використанням графічного інтерфейсу користувача [142].

Максимальна потужність СЕС для живлення приватних господарств є відносно малою (до декількох кіловат), тому при проведенні комп'ютерного моделювання використаємо масив СП, який складається з чотирьох панелей. Одночасно це дозволить спростити систему комутації, оскільки складні топології масиву СП потребують більшої кількості панелей [143]. Таким чином при проведенні моделювання врахуватимуться лише послідовне, паралельне та послідовно-паралельне увімкнення.

У якості досліджуваних СП використаємо раніше дослідженні KC200GT та ST40 (див. таблицю 2.6). При проведенні моделювання використовуватимуться значення параметрів еквівалентної схеми заміщення, знайдені за допомогою способу перебору значень опорів (див. таблиці 2.6 та 2.7).

4.1. Моделювання масиву сонячних панелей

Проведемо моделювання одиничної СП для побудови її ВАХ та характеристики вихідної потужності. Схема моделі СП в середовищі Simulink наведена на рис. 4.1.

Дана модель складається з наступних блоків:

1. Сонячної панелі *PV*. Даний блок дозволяє обрати тип еквівалентної схеми заміщення СП, а також визначити її параметри. Налаштування даного блоку для моделювання панелі KC200GT наведені на

рис. 4.2. Для еквівалентної схеми заміщення СП послідовний опір визначається наступним чином [144]:

$$R_s = \frac{N_s}{N_p} R'_s, \quad (4.1)$$

де N_s – кількість послідовно з'єднаних СЕ в СП, N_p – кількість паралельно з'єднаних СЕ в СП, R'_s – послідовний опір СЕ.

Аналогічно для шунтуючого опору:

$$R_p = \frac{N_s}{N_p} R'_p, \quad (4.2)$$

де R'_p – шунтуючий опір СЕ.

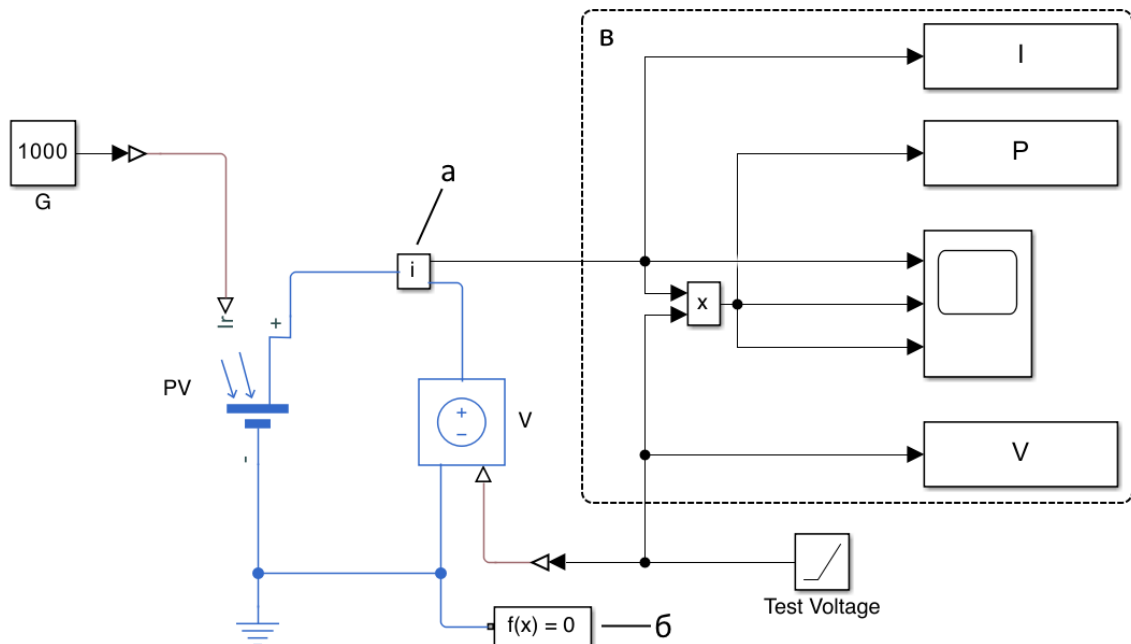


Рис. 4.1. Схема моделі СП в середовищі Simulink

Виходячи з офіційної документації виробника [113], СП KC200GT складається з 54 послідовно з'єднаних СЕ, тобто $N_s = 54$ та $N_p = 1$. Таким чином, на основі (4.1) та (4.2) визначаються значення опорів СЕ, які необхідні для визначення блоку PV . Значення даних опорів для досліджуваних СП наведені в таблиці 4.1.

Таблиця 4.1. Значення опорів СЕ для досліджуваних СП

СП	N_s	R'_s	R'_p
KC200GT	54	0,00487	2,22883
ST40	42	0,03419	9,79200

2. Блоку константи G , який визначає рівень опроміненості відповідної СП на час моделювання.

3. Керованого джерела напруги V , яке виступає у ролі навантаження. Напруга на виходах лінійно залежить від значення керуючого сигналу.

4. Блоку лінійного сигналу *Test Voltage* – слугує опорним сигналом, для керування вихідною напругою джерела V .

5. Блоку виміру струму (рис. 4.1, а), який вимірює вихідний струм досліджуваної СП.

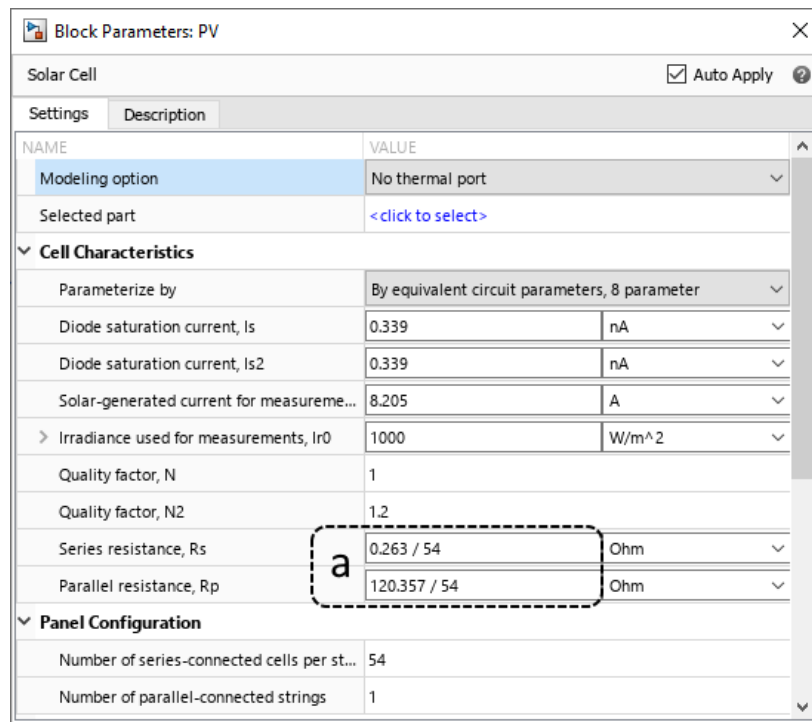


Рис. 4.2. Зображення параметрів блоку PV для СП KC200GT: визначення значень опорів СЕ (а)

6. Блоку визначення параметрів моделювання (рис. 4.1, б). Даний блок відповідає за визначення кроку моделювання, способу вирішення диференціальних рівнянь та іншого.

7. Блоків для побудови графіків ВАХ та характеристики вихідної потужності (рис. 4.1, б).

За допомогою створеної моделі СП (рис. 4.1) побудовано ВАХ досліджуваних СП для різних рівнів опроміненості (рис. 4.3). Окрім моделі Simulink для побудови графіків та автоматизації процесу моделювання використовувалися додаткові Matlab скрипти (додаток г).

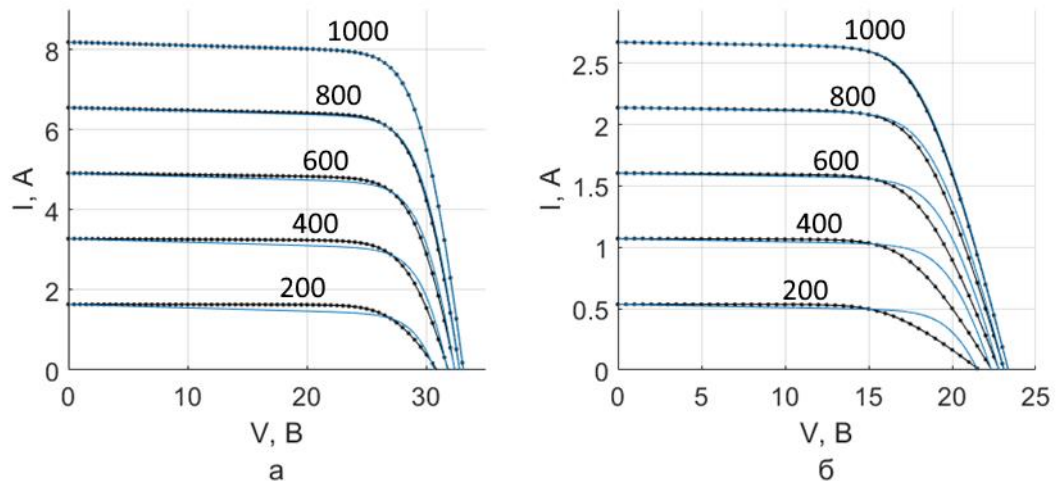


Рис. 4.3. Графіки результатів моделювання ВАХ СП: KC200GT (а) та ST40 (б). Суцільна крива – моделі Simulink, точкова – запропонована модель (розділ 2)

Для обох досліджуваних СП отримані ВАХ добре узгоджуються для рівнів опроміненості 1000 та 800 Вт/м² (рис. 4.3). Проте з подальшим зменшенням рівня опроміненості спостерігається збільшення різниці між ВАХ моделей. Особливо виражено це проявляється для тонкоплівкової панелі ST40 (рис. 4.3, б), для яких залежність параметрів еквівалентної схеми заміщення є більш вираженою у порівнянні з полікристалічними СП, як у випадку KC200GT (рис. 4.3, а).

Ріст різниці між ВАХ зі зменшенням опроміненості пояснюється тим, що стандартна модель Simulink відтворює лише зміну значення фотоструму при варіації значення опроміненості [145], тоді як запропонована модель враховує також зміну значень опорів (див. розділ 2.5), що дозволяє більш точно відтворювати ВАХ СП.

Надалі для проведення моделювання масиву СП використовуватиметься панель KC200GT. Схема моделі масиву чотирьох СП в середовищі Simulink наведена на рис. 4.4.

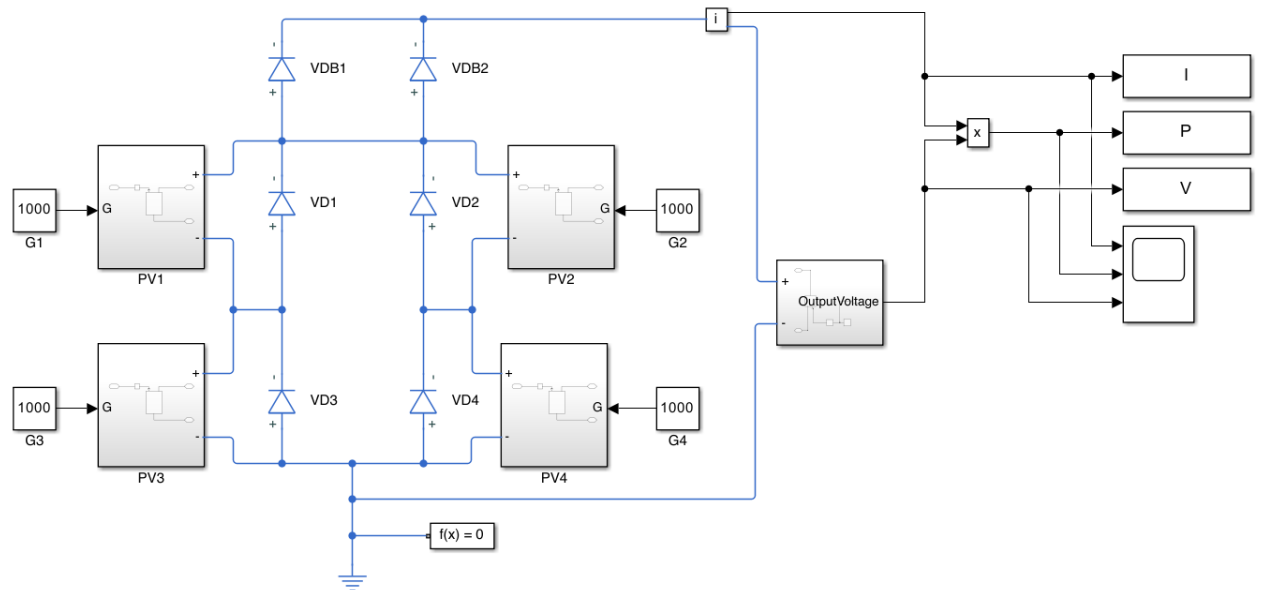


Рис. 4.4. Схема моделі масиву СП в середовищі Simulink

Дана схема складається з наступних елементів:

1. Масиву користувацьких блоків СП ($PV1$, $PV2$, $PV3$, $PV4$), кожен з яких складається з блоку PV (рис. 4.1) та конвертора типу сигналу. Даний масив складається з двох паралельних віток, кожна з яких складається з двох послідовно з'єднаних СП. Використання користувацьких блоків дозволяє візуально спростити отриману модель.
2. Шунтуючих діодів ($VD1$, $VD2$, $VD3$, $VD4$), які слугують для захисту панелей від негативного впливу при виникненні явища часткового затінення.
3. Блокуючих діодів ($VDB1$, $VDB2$), які використовуються для захисту паралельних віток масиву СП від протікання зворотнього струму, що може виникати у випадку нерівномірної опроміненості панелей.
4. Керованого навантаження, яке складається з блоків V та $Test Voltage$ (рис. 4.1) і потрібне для побудови ВАХ у визначеному діапазоні напруги.

5. Всі інші блоки взяті з моделі СП (рис. 4.1) і дублюють їх функції.

Дослідимо ВАХ масиву СП за різних умов опроміненості, значення яких наведені в таблиці 4.2. На основі даних рівнів опроміненості проведено моделювання, результати якого у виді ВАХ та характеристики вихідної потужності наведені на рис. 4.5. Значення вихідного струму для всіх ВАХ збільшено в 20 разів для збереження масштабу.

Таблиця 4.2. Досліджувані умови опроміненості

$G1$	$G2$	$G3$	$G4$
1000	1000	1000	1000
1000	1000	1000	500
1000	800	1000	600
1000	800	600	400

При рівномірній опроміненості (рис. 4.5, а) крива вихідної потужності та ВАХ мають одну точку максимуму, тоді як при умовах часткового затінення (рис. 4.5, б-г) на даних характеристиках формується локальний пік, який ускладнює роботу ВТМП. Причиною їх появи є шунтуючі діоди, які використовуються для захисту СП від перегріву і потенційного виходу з ладу в умовах часткового затінення.

Зі зменшенням рівня опроміненості спостерігається зменшення значення струму КЗ за рахунок зменшення значення фотоструму, а також зсув ТМП в сторону точки ХХ (рис. 4.5, а-г).

На всіх характеристиках вихідної потужності в умовах часткового затінення (рис. 4.5, б-г) спостерігається наявність одного локального піку. При нижчих рівнях опроміненості даний пік є більш вираженим (рис. 4.5, г), що ускладнює роботу пристроїв ВТМП.

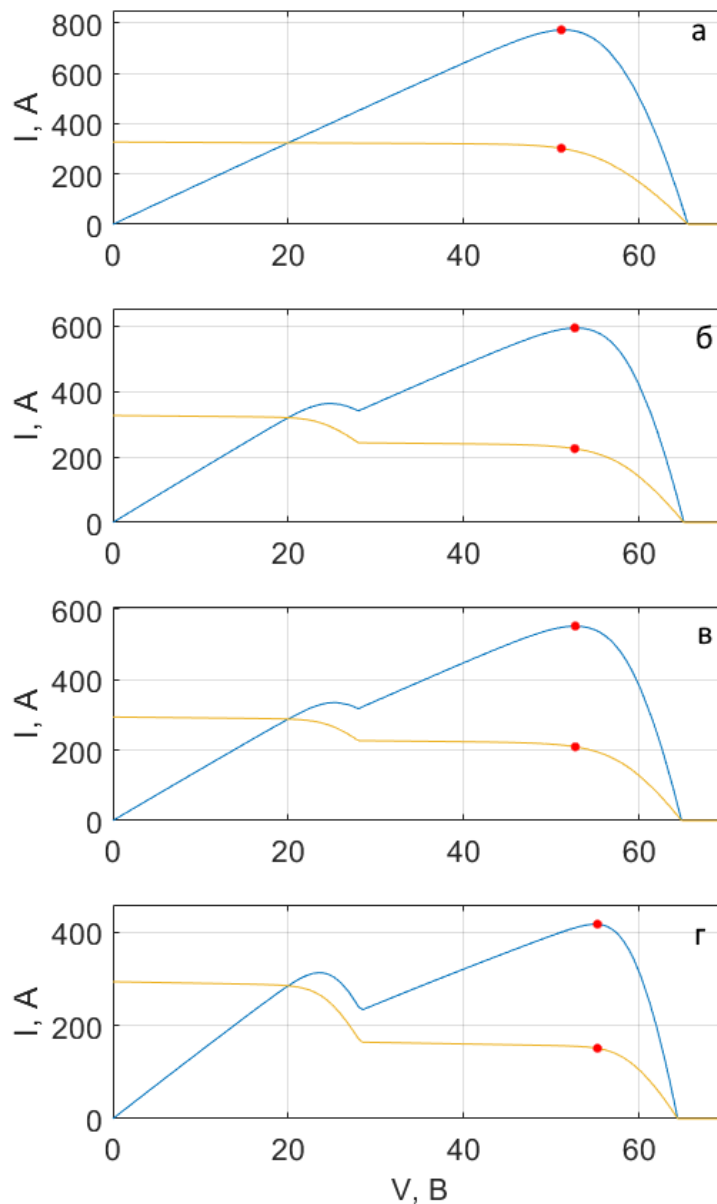


Рис. 4.5. Графіки результатів комп'ютерного моделювання для різних рівнів опроміненості: нумерація (а-г) відповідає рядками таблиці 4.2. ТМПП позначена червоною точкою

Запропонована модель СП добре узгоджується з моделлю середовища Simulink при рівнях освітленості, близьких до стандартних тестових умов. З подальшим зменшенням рівня освітленості розроблена модель показує кращу якість наближення ВАХ, що пояснюється врахуванням більшої кількості залежностей розробленої математичної моделі у порівнянні з середовищем Simulink. Числове порівняння похибок є ускладненим через відмінність сіток напруги розглянутих моделей.

4.2. Моделювання комутованого масиву сонячних панелей

Проведемо моделювання комутованого масиву СП для побудови її ВАХ та характеристики вихідної потужності. Схема даної моделі в середовищі Simulink наведена на рис. 4.6.

Дана схема складається з наступних елементів:

1. Блоків сигналів керування пристроїв комутації верхнього плеча або ж позитивного полюсу ($C11$, $C21$, $C31$, $C41$) та нижнього плеча ($C12$, $C22$, $C32$, $C42$). Дані блоки реалізовані за допомогою блоків констант, які генерують сигнали постійного рівня на час моделювання. Блоки керування верхнього плеча відповідають за комутацію позитивного полюсу СП, а нижнє плече – за сигнали керування для пристроїв комутації негативного полюсу.

Значення сигналів керування та відповідні вихідні стани пристроїв комутації наведені в таблиці 4.3. Варто зауважити, що сигнали для блоків керування першої та останньої СП масиву є фіксованим і їх значення становлять $C11=1$ та $C42=0$.

Таблиця 4.3. Значення сигналів керування та їх функції

Сигнал	Значення	Шина
$CX1$ (верхнє плече)	0	М
	1	+
$CX2$ (нижнє плече)	0	-
	1	М

2. Комутаційних пристроїв позитивного полюсу ($S11$, $S21$, $S31$, $S41$) та негативного полюсу ($S12$, $S22$, $S32$, $S42$), які виконують підключення відповідних виводів СП до позитивної (+), негативної (-) або проміжної шини (М). Проміжна шина використовується для можливості реалізації послідовного увімкнення всіх СП в масиві [136]. При цьому позитивний полюс першої панелі $PV1$ підключається лише до позитивного полюсу навантаження,

а негативний полюс останньої панелі $PV4$ – до негативного полюсу навантаження.

Дані блоки можуть реалізовуватися за допомогою напівпровідникових пристроїв (тиристори, транзистори) або ж за допомогою реле, які використовуються в даній моделі.

3. Всі інші блоки взяті з моделі масиву СП (рис. 4.4) і дублюють їх функції.

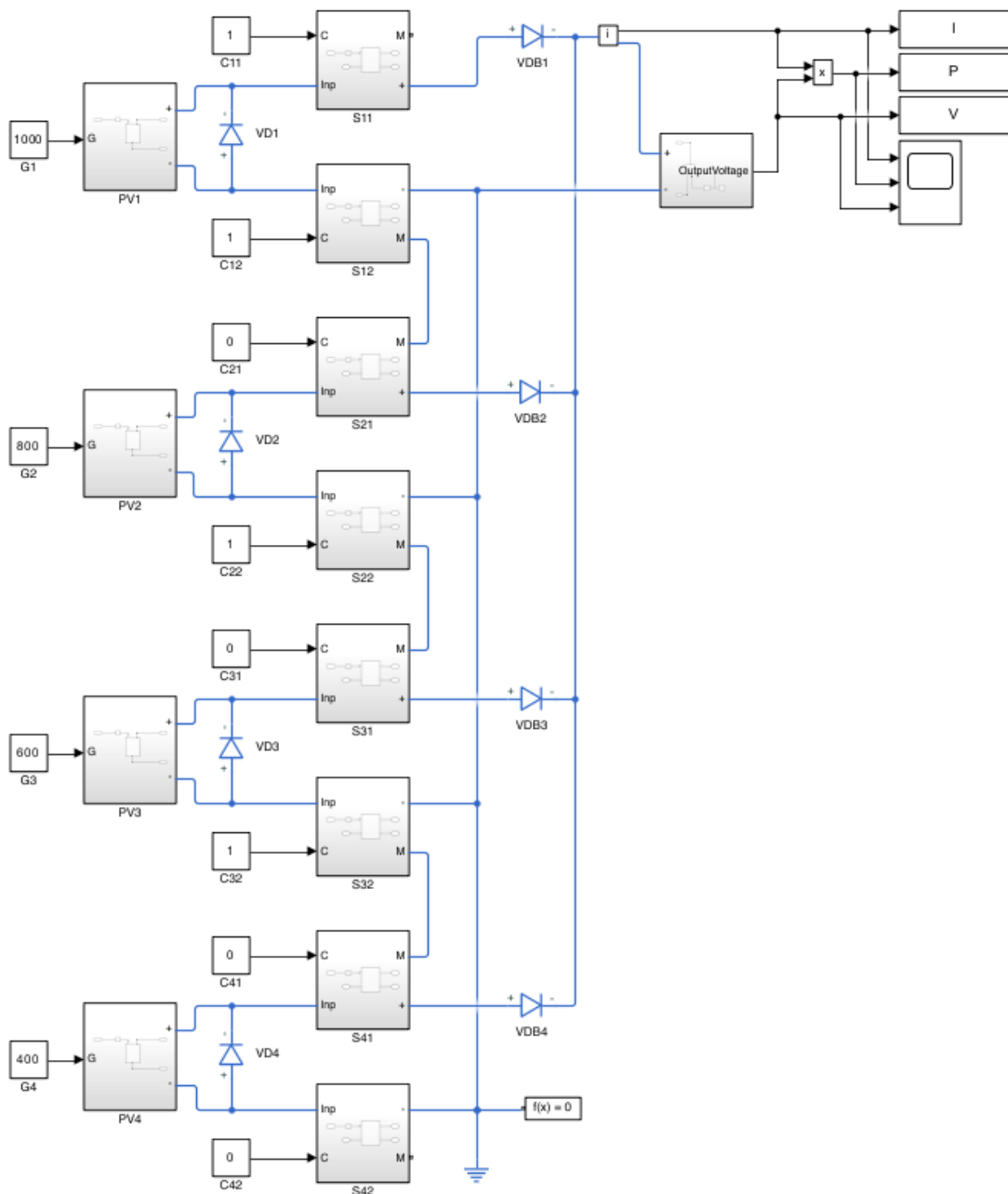


Рис. 4.6. Схема моделі комутованого масиву СП в середовищі Simulink

Проведемо моделювання комутованого масиву СП для різних значень сигналів керування в умовах часткового затінення для значень опроміненості $G1=1000$, $G2=800$, $G3=600$ та $G4=400$. Значення сигналів комутації наведені в таблиці 4.4, схеми увімкнення, які відповідають даним сигналам комутації – рис. 4.7, результати моделювання – рис. 4.8.

Таблиця 4.4. Сигнали комутації для деяких схем увімкнення масиву СП

№	а	б	в	г
<i>C12</i>	1	1	1	0
<i>C21</i>	0	0	0	1
<i>C22</i>	1	1	0	0
<i>C31</i>	0	0	1	1
<i>C32</i>	1	0	0	0
<i>C41</i>	0	1	1	1

Очевидно, що за отриманими результатами моделювання (рис. 4.8) найбільшу вихідну потужність забезпечує паралельне увімкнення СП (рис. 4.8, а). При такій схемі увімкнення не виникають локальні піки на ВАХ, а різниця у вихідній напрузі паралельних віток компенсується наявністю блокуючих діодів.

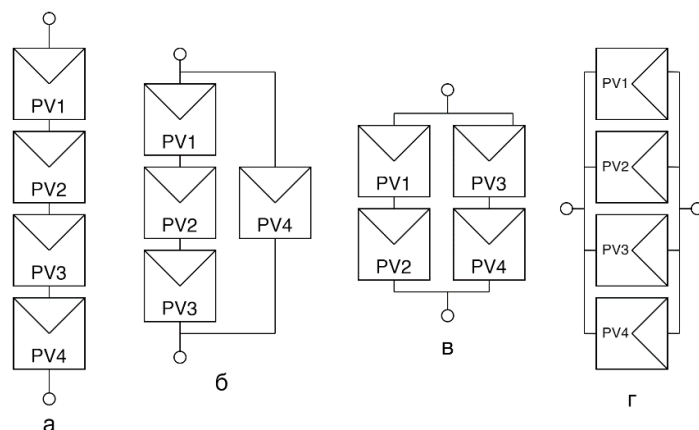


Рис. 4.7. Схеми увімкнення, які відповідають досліджуваним сигналам комутації

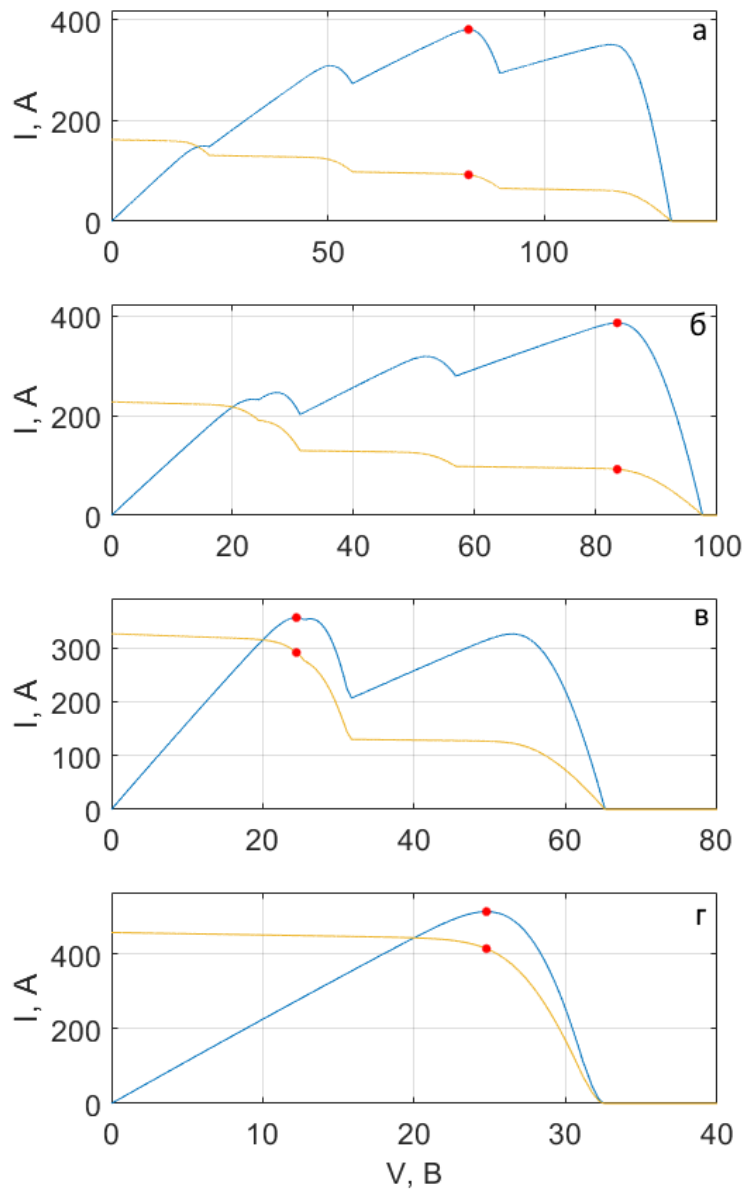


Рис. 4.8. Графіки результатів комп'ютерного моделювання для різних схем увімкнення опроміненості: нумерація (а-г) відповідає рядками таблиці 4.4.

Точка максимальної потужності позначена червоною точкою

Тим не менш, ТМП для розглянутих схем увімкнення лежить у великому діапазоні вихідної напруги, що призводить до ускладнення перетворювальної частини. Для забезпечення ефективності роботи перетворювальної у всьому діапазоні вихідної напруги прийемо його рівним $V = 40...80$ В.

Дане обмеження призводить до того, що вихідна напруга таких схем увімкнення як послідовне та паралельне лежить поза заданим діапазоном. Відповідно забезпечити необхідний діапазон може лише послідовно-

паралельне з'єднання масиву СП. Для реалізації комутованого масиву СП з послідовно-паралельним з'єднанням використовується модель, схема якої зображена на рис. 4.9.

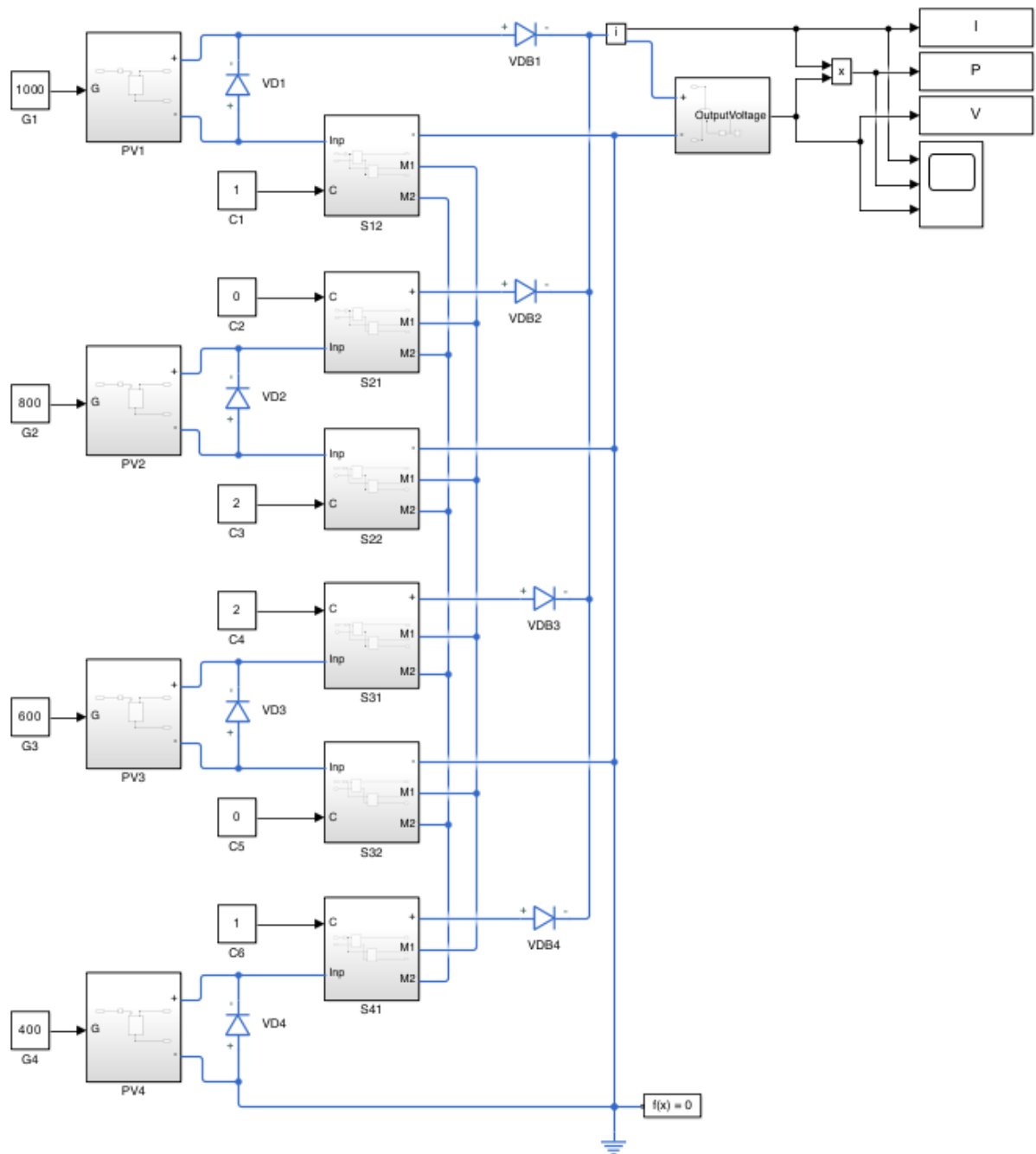


Рис. 4.9. Схема моделі комутованого масиву СП в середовищі Simulink для реалізації послідовно-паралельного з'єднання

Дана модель, на відміну від попередньої моделі (рис. 4.6), містить дві проміжні шини $M1$ та $M2$, що надає можливість реалізувати всі можливі варіанти послідовно-паралельного увімкнення масиву СП з чотирьох панелей.

Рівні керуючих сигналів та виходи комутаційного пристрою, які їх відповідають, наведені в таблиці 4.5 (х позначає заборонений стан для комутуючого пристрою).

Таблиця 4.5. Вихідні шини та відповідні рівні комутуючих сигналів

Шина	Позитивний полюс	Негативний полюс
+	0	х
M1	1	1
M2	2	2
-	х	0

Для спрощення моделі було видалено комутаційні пристрої та блоки сигналів керування для позитивного полюсу першої *PV1* та негативного полюсу останньої *PV4* панелей, оскільки дані СП завжди підключені до позитивного полюсу та негативного полюсів навантаження відповідно.

4.3. Порівняльний аналіз

Проведемо порівняльний аналіз вихідної потужності масиву СП без та з комутацією. Генерацію значень опроміненості проводитимемо за допомогою запропонованої моделі хмарного покриву. Для цього використаємо значення параметрів досліджуваного хмарного покриву (див таблицю 3.4) з використанням псевдовипадкової фази та коефіцієнту масштабування $r = 5$, що забезпечить хмарного покриву розміром приблизно 400 метрів. Згенероване зображення хмарного покриву та опроміненості наведене на рис. 4.10. Щільність хмарного покриву пропорційна яскравості пікселя, відповідно зображення опроміненості (рис. 4.10, б) знаходиться наступним чином:

$$G(i, j) = 1000 \cdot [1 - P'(i, j)], \quad (4.3)$$

де $P'(i, j)$ – значення яскравості пікселя хмарного покриву, приведенного до діапазону $[0...1]$.

Схематичний підхід до моделювання руху хмарного покриву наведено на рис. 4.10, б. Стрілка відображає напрямок руху хмарного покриву, числові значення – положення масиву СП на зображенні хмарного покриву на відповідній ітерації. Швидкість руху хмарного покриву змінюється за рахунок зміни відстані між положеннями масиву СП на сусідніх ітераціях.

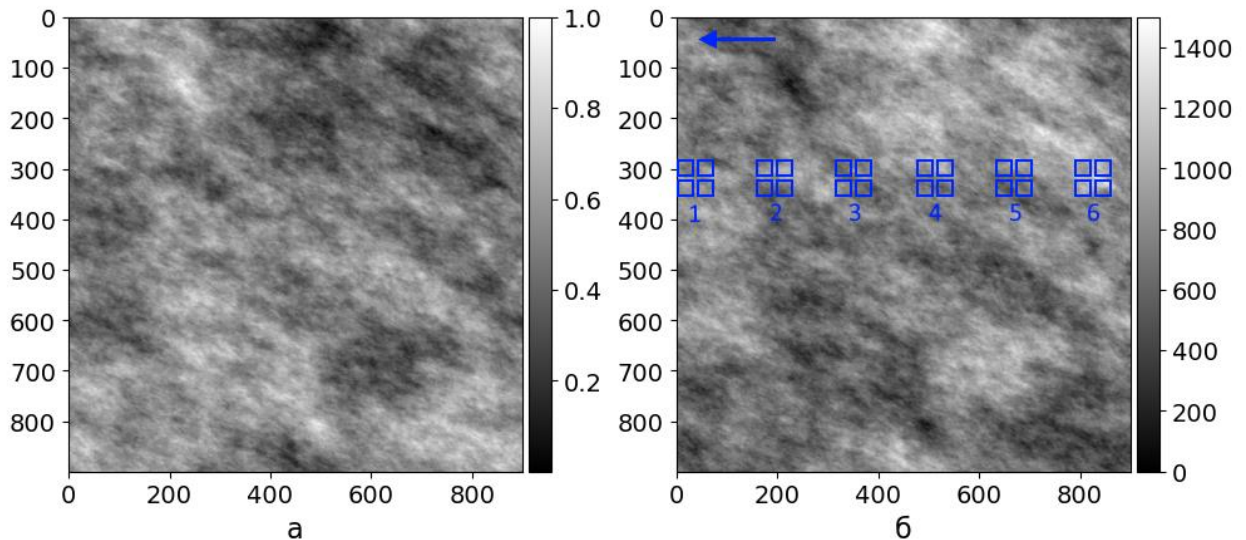


Рис. 4.10. Згенеровані зображення хмарного покриву (а) та опроміненості (б).

Масив СП умовно позначений чотирма квадратами, числові значення визначають ітерацію моделювання

Поширеним алгоритмом визначення оптимальної схеми увімкнення при наявності часткового затінення є вирівнювання рівня опромінення для горизонтальних ліній масиву СП [146], що еквівалентно вирівнюванню мінімізації різниці струмів паралельно увімкнених СП. Даний підхід застосовується для схеми увімкнення ТСТ, для послідовно-паралельної схеми використаємо його варіацію, яка полягає у вирівнюванні рівня опромінення паралельних віток масиву СП. На основі даного алгоритму було отримано відповідні сигнали комутації (рис. 4.13) для наявних рівнів освітленості СП (рис. 4.12) отримано характеристику вихідної потужності (рис. 4.11). Дана характеристика описує положення ТМП масиву СП при наявному рівні опроміненості. Також для наочності наведено графік, який представляє різницю між вихідною потужністю масиву СП з та без комутацією (рис. 4.14).

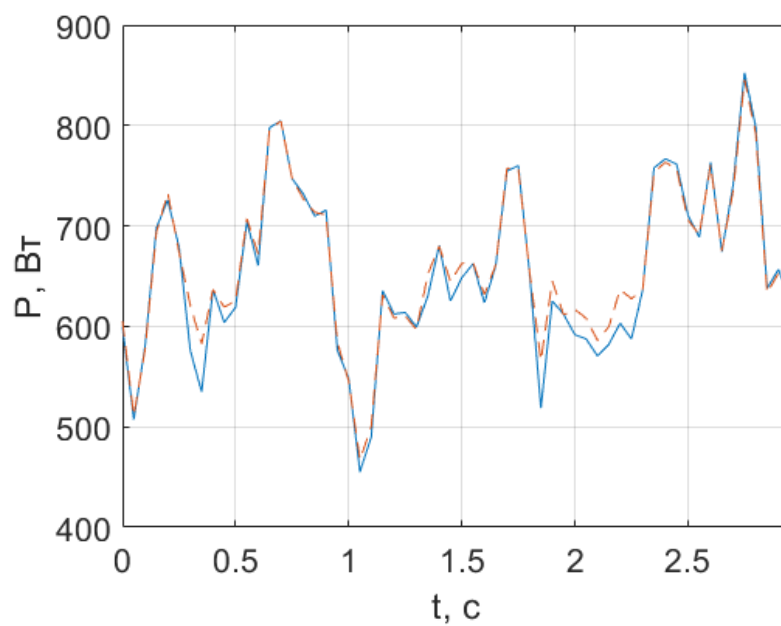


Рис. 4.11. Графік часової залежності ТМП масиву СП: без комутації (суцільна лінія) та з комутацією (штрихова)

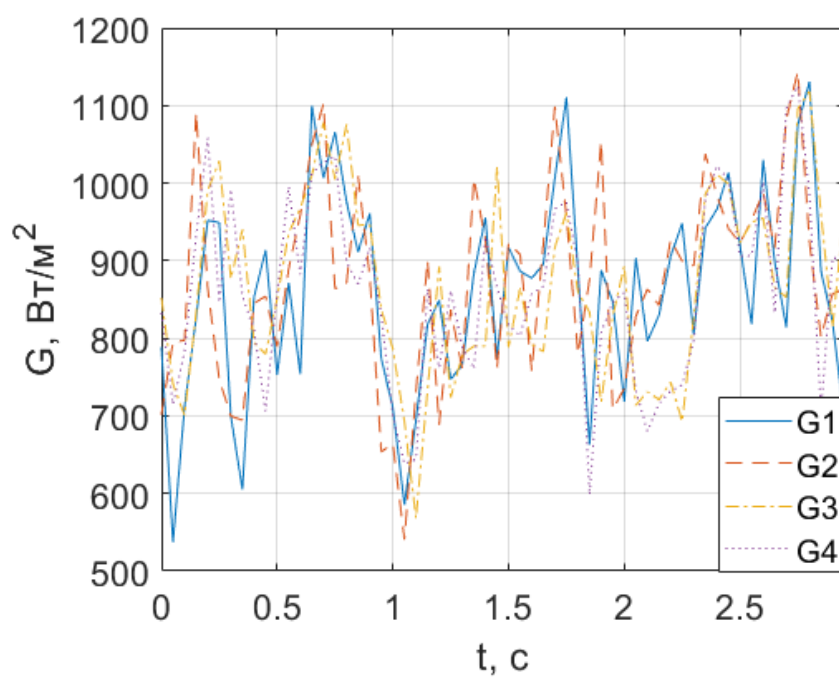


Рис. 4.12. Графік часової залежності рівнів опроміненості СП масиву

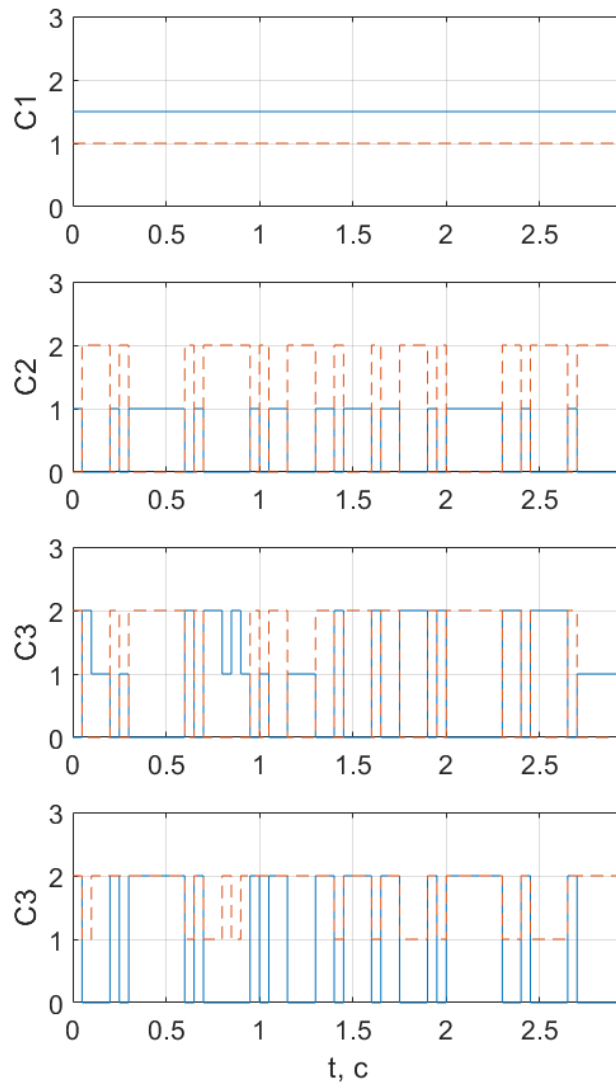


Рис. 4.13. Графік сигналів комутації СП масиву позитивного (суцільна) та негативного (штрихова лінія) полюсів. Для *PVI* сигнал позитивного полюсу збільшено в 1,5 разів

З рис. 4.11 видно, за допомогою комутації масиву СП забезпечується більше значення потужності в ТМП. З графіку різниці (рис. 4.14) видно, що для окремих точок приріст становить 50,6694 Вт ($t=0,35$), що для номінальної потужності масиву 800 Вт дає приріст в 6,33% у вихідній потужності.

Також існують проміжки, на яких різниця має негативне значення, тобто масив СП з комутацією має меншу вихідну потужність. Найбільше подібне

значення становить 3,9895 Вт ($t = 2,75$), що складає 0,5% у порівнянні з номінальною потужністю. Наявність подібних втрат пояснюється наявністю додаткових втрат за рахунок ключів та додаткових блокуючих діодів.

Загальний приріст вихідної потужності для проведеного моделювання складає 568,1125 Вт, що по відношенню до потужності масиву без комутації 39211 Вт, дає приріст в 1,5%. Застосування даного підходу до масиву з більшою кількістю СП та іншими схемами увімкнення (ТСТ) дозволить отримати більше значення приросту вихідної потужності.

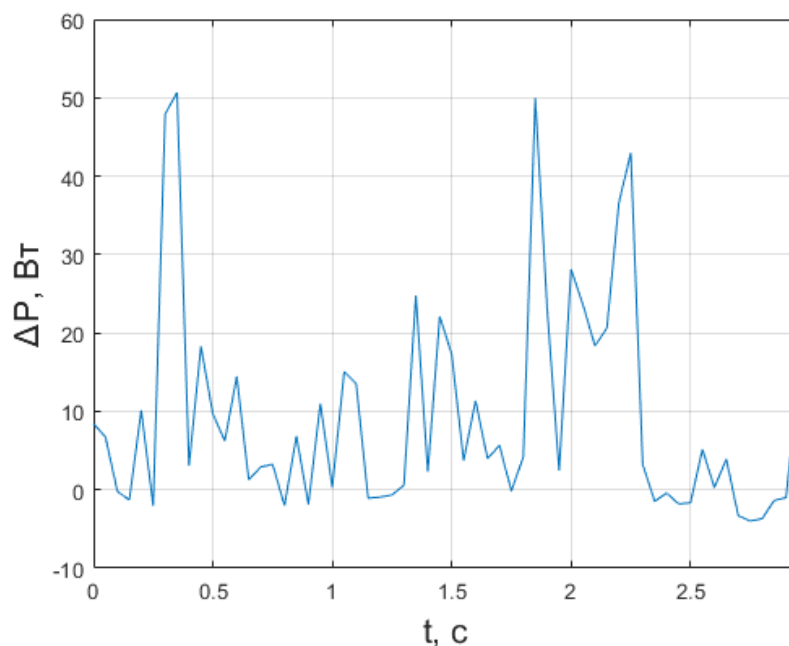


Рис. 4.14. Графік різниці вихідної потужності масиву СП з комутацією та без

Таким чином, на основі проведеного моделювання отримано наступну структурну схему масиву СП з комутацією рис. 4.15.

Дана схема складається з системи наступних блоків:

1. Системи керування (СК), яка на основі інформації про рівень опроміненості кожної (рис. 4.12). СП масиву визначає оптимальну структуру масиву та генерує відповідні сигнали керування (рис. 4.13)
2. Масиву СП ($PV1, PV2, PV3, PV4$), який працює на інвертор у якості навантаження.

3. Шунтуючих ($VD1, VD2, VD3, VD4$) та блокуючих ($VDB1, VDB2, VDB3, VDB4$) діодів, які слугують для захисту СП від негативного впливу, який може виникнути за умов часткового затінення

4. Комутаційних пристроїв позитивного ($C21, C31, C41$) та негативного ($C12, C22, C32$) полюсів, які підключають відповідні полюси СП до необхідної шини.

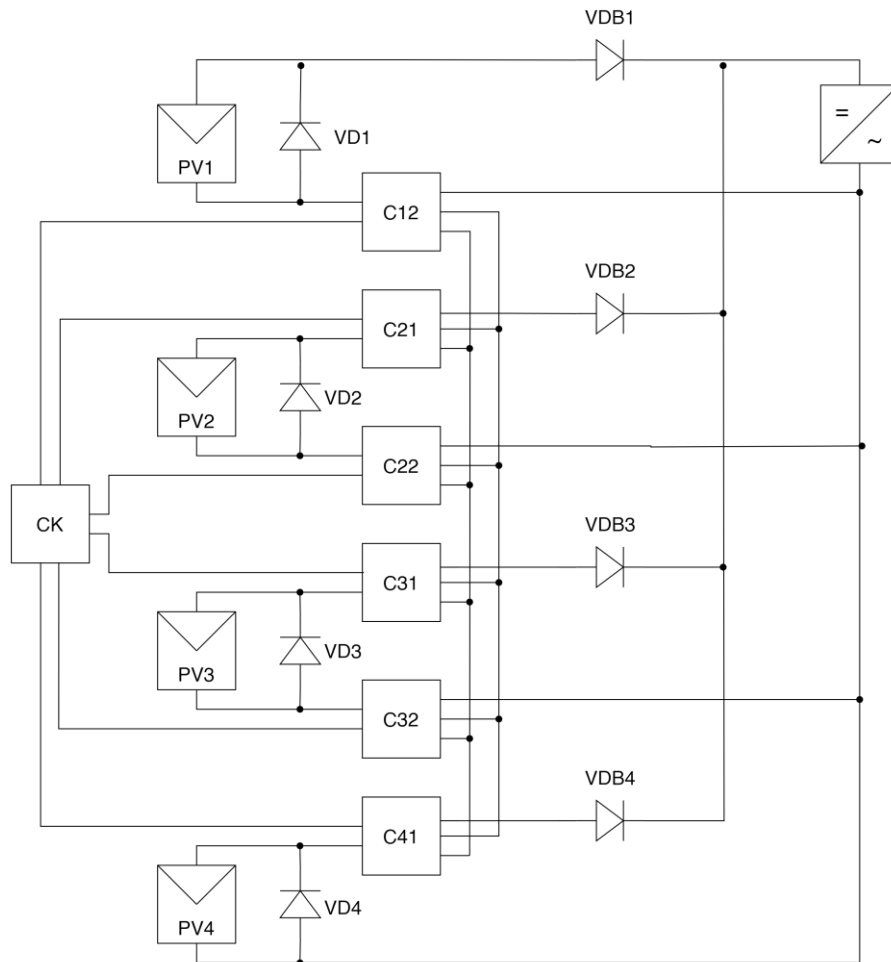


Рис. 4.15. Структурна схема

Таким чином реалізація динамічної комутації масиву СП в умовах часткового затінення дозволяє в середньому підвищити на 1,5% рівень вихідної потужності масиву, який складається з чотирьох панелей.

Висновки до розділу

1. Проведено комп'ютерне моделювання ВАХ в середовищі Matlab Simulink, яке показало, що для сонячних панелей KC200GT та ST40 отримані ВАХ добре узгоджуються зі вбудованою моделлю, що свідчить про правочинність використання запропонованих моделей та системи динамічної комутації.
2. Проведено моделювання роботи комутованого масиву сонячних панелей в умовах часткового затінення, яке реалізовано за допомогою використання розробленої моделі хмарного покриву. Було встановлено, що для номінальної потужності масиву 800 Вт приріст потужності складає 50,669 Вт для окремих точок, що дає приріст у 6,33% у вихідній потужності.
3. Моделювання виявило, що на всьому часовому проміжку роботи загальний приріст вихідної потужності становить 568,113 Вт, що по відношенню до потужності масиву без комутації 39211 Вт, дає приріст в 1,5%. Застосування даного системи до масиву з більшою кількістю СП та іншими схемами увімкнення (ТСТ) дозволить отримати більше значення приросту вихідної потужності.

ЗАГАЛЬНІ ВИСНОВКИ

В роботі вирішена задача подальшого розвитку теорії моделювання хмарного покриву для створення системи динамічної комутації масиву сонячних панелей з урахуванням нелінійності ВАХ та підвищенням ефективності використання сонячної енергії.

1. Створено удосконалено математичну модель сонячної панелі та запропоновано два способи визначення параметрів моделі на основі даних, які надають виробники сонячних панелей. Для обох способів рекомендовано використовувати Гаусівську віконну функцію w_g зі значенням $\sigma = 0,1$, що дозволяє мінімізувати похибку апроксимації поблизу точки максимальної потужності. Порівняльний аналіз показав, що спосіб перебору значень опорів у порівнянні з перебором значень коефіцієнтів ідеальності забезпечує кращу точність апроксимації ВАХ – на 89% менше значення параметру M для сонячної панелі KC200GT та на 96% для ST40.

2. Створено математичну модель хмарного покриву, засновану на швидкому перетворенні Фур'є зображення хмар та використанні його спектральних характеристик. Запропоновано використання вагової функції для мінімізації явища витікання спектру, що дозволяє знизити спотворення спектральних характеристик та спростити їх аналіз.

3. Запропоновано генерацію зображення хмарного покриву на основі використання псевдовипадкової фази, що дозволяє за знайденими параметрами відтворювати основні особливості вихідного зображення. Встановлено, що запропонована модель забезпечує на 22,65% краще значення похибки $RMSE$ та на 22,10% краще значення індексу структурної подібності $SSIM$ у порівнянні зі поширеним підходом спектрального синтезу.

4. Розроблено програмну реалізацію математичної моделі сонячної панелі та хмарного покриву на основі мови програмування Python та його дистрибутиву Anaconda, яка дозволяє моделювати ВАХ панелі при різних

умовах опроміненості та температури, а також проводити генерацію хмарного покриву з відтворенням особливостей, притаманних для вихідних даних.

5. Проведено комп'ютерне моделювання ВАХ в середовищі Matlab Simulink, яке показало, що для сонячних панелей KC200GT та ST40 отримані ВАХ добре узгоджуються зі вбудованою моделлю, що свідчить про правочинність використання запропонованих моделей та системи динамічної комутації.

6. Проведено моделювання роботи комутованого масиву сонячних панелей в умовах часткового затінення, яке реалізовано за допомогою використання розробленої моделі хмарного покриву. Було встановлено, що для номінальної потужності масиву 800 Вт приріст потужності складає 50,669 Вт для окремих точок, що дає приріст у 6,33% у вихідній потужності.

7. Моделювання виявило, що на всьому часовому проміжку роботи загальний приріст вихідної потужності становить 568,113 Вт, що по відношенню до потужності масиву без комутації 39211 Вт, дає приріст в 1,5%. Застосування даного системи до масиву з більшою кількістю СП та іншими схемами увімкнення (ТСТ) дозволить отримати більше значення приросту вихідної потужності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Y. Ziabina and T. Pimonenko, “UKRAINIAN ENERGY POLICY: EFFICIENCY AND WAYS FOR IMPROVEMENT,” *Econ. scope*, 2020.
- [2] “Стратегія сталого розвитку України до 2030 року | United Nations Development Programme,” 2017. [Online]. Available: <https://www.undp.org/uk/ukraine/publications/стратегія-сталого-розвитку-україни-до-2030-року>. [Accessed: 16-Feb-2024].
- [3] “Енергетична стратегія | Міністерство енергетики України,” 2022. [Online]. Available: <https://www.mev.gov.ua/reforma/enerhetychna-stratehiya>. [Accessed: 16-Feb-2024].
- [4] “Перехід України на відновлювану енергетику до 2050 р. | Heinrich Böll Stiftung | Київ – Україна.” [Online]. Available: <https://ua.boell.org/uk/2017/10/24/perehid-ukrayini-na-vidnovlyuvanu-energetiku-do-2050-r>. [Accessed: 27-Sep-2023].
- [5] В.І. Мартинюк і К.С. Клен, «Лінійні функції на базі функцій Уолша», *Мікросистеми, Електроніка та Акустика*, т. 24, вип. 1, с. 29-34, Лютий 2019, doi:10.20535/2523-4455.2019.24.1.160960.
- [6] K. Klen, V. Martynyuk, and M. Yaremenko, «Prediction of the wind speed change function by linear regression method», *Computational Problems of Electrical Engineering*, vol. 9, no. 2, pp. 28–33, Nov 2019, doi: 10.23939/jcpee2019.02.028.
- [7] K. Klen, V. Martynyuk, and M. Yaremenko, «Prediction of primary energy flow considering its stochastic nature», *Maszyny Elektryczne. – Zeszyty Problemowe*, no. 1(125), pp. 107–110, 2021
- [8] В.І. Мартинюк, К.С. Клен, і В. Я. Жуйков, «Визначення параметрів схем заміщення сонячних панелей за експериментальними даними», *Мікросистеми, Електроніка та Акустика*, т. 26, вип. 2, с. 237316-1, Серпень 2021, doi:10.20535/2523-4455.me.237316.
- [9] V. Martyniuk and K. Klen, «Estimation of Parameters of Photovoltaic

- Modules Based on Ideality Factor Variation», *2022 IEEE 41st International Conference on Electronics and Nanotechnology (ELNANO)*, IEEE, Oct. 2022, pp. 692-696, doi: 10.1109/ELNANO54667.2022.9927063.
- [10] V. I. Martyniuk and V. Y. Zhuikov, «Generation of Anisotropic Cloud Cover», *Microsystems, Electronics and Acoustics*, vol. 28, no. 3, pp. 291803.1–291803.10, Dec 2023, doi: 10.20535/2523-4455.mea.291803.
- [11] D. Gielen *et al.*, “Global Energy Transformation: A Roadmap to 2050,” Apr. 2019.
- [12] I. Khan, F. Hou, and H. P. Le, “The impact of natural resources, energy consumption, and population growth on environmental quality: Fresh evidence from the United States of America,” *Sci. Total Environ.*, vol. 754, p. 142222, Feb. 2021.
- [13] E. Uherek *et al.*, “Transport impacts on atmosphere and climate: Land transport,” *Atmos. Environ.*, vol. 44, no. 37, pp. 4772–4816, Dec. 2010.
- [14] S. Bilgen, “Structure and environmental impact of global energy consumption,” *Renew. Sustain. Energy Rev.*, vol. 38, pp. 890–902, Oct. 2014.
- [15] X. Wu, D. Freese, A. Cabrera, and W. A. Kitch, “Electric vehicles’ energy consumption measurement and estimation,” *Transp. Res. Part D Transp. Environ.*, vol. 34, pp. 52–67, Jan. 2015.
- [16] IEA, *Electricity Market Report*. 2020.
- [17] Energy Institute, “The Energy Institute Statistical Review of World Energy 2023,” 2023.
- [18] A. Zaharia, M. C. Diaconeasa, L. Brad, G.-R. Lădaru, and C. Ioanăș, “Factors Influencing Energy Consumption in the Context of Sustainable Development,” *Sustainability*, vol. 11, no. 15, p. 4147, Aug. 2019.
- [19] M. S. Nazir, M. Bilal, H. M. Sohail, B. Liu, W. Chen, and H. M. N. Iqbal, “Impacts of renewable energy atlas: Reaping the benefits of renewables and biodiversity threats,” *Int. J. Hydrogen Energy*, vol. 45, no. 41, pp. 22113–22124, Aug. 2020.
- [20] P. A. Østergaard, N. Duic, Y. Noorollahi, and S. Kalogirou, “Renewable

- energy for sustainable development,” *Renew. Energy*, vol. 199, pp. 1145–1152, Nov. 2022.
- [21] D. Maradin, “ADVANTAGES AND DISADVANTAGES OF RENEWABLE ENERGY SOURCES UTILIZATION,” *Int. J. Energy Econ. Policy*, vol. 11, no. 3, pp. 176–183, Apr. 2021.
- [22] P. G. V. Sampaio and M. O. A. González, “Photovoltaic solar energy: Conceptual framework,” *Renew. Sustain. Energy Rev.*, vol. 74, pp. 590–601, Jul. 2017.
- [23] I. Renewable Energy Agency, “RENEWABLE CAPACITY STATISTICS 2023 STATISTIQUES DE CAPACITÉ RENOUVELABLE 2023 ESTADÍSTICAS DE CAPACIDAD RENOVABLE 2023 About IRENA,” 2023.
- [24] Ember, “Share of electricity production by source, World.” [Online]. Available: <https://ourworldindata.org/grapher/share-elec-by-source>. [Accessed: 27-Dec-2023].
- [25] F. Blaabjerg, Y. Yang, D. Yang, and X. Wang, “Distributed Power-Generation Systems and Protection,” *Proc. IEEE*, vol. 105, no. 7, pp. 1311–1331, Jul. 2017.
- [26] M. F. Akorede, H. Hizam, and E. Pouresmaeil, “Distributed energy resources and benefits to the environment,” *Renew. Sustain. Energy Rev.*, vol. 14, no. 2, pp. 724–734, Feb. 2010.
- [27] R. W. Boyd, *Radiometry and the Detection of Optical Radiation*. .
- [28] G. Kopp and J. L. Lean, “A new, lower value of total solar irradiance: Evidence and climate significance,” *Geophys. Res. Lett.*, vol. 38, no. 1, p. n/a-n/a, Jan. 2011.
- [29] S. Dewitte and N. Clerbaux, “Measurement of the Earth Radiation Budget at the Top of the Atmosphere—A Review,” *Remote Sens. 2017, Vol. 9, Page 1143*, vol. 9, no. 11, p. 1143, Nov. 2017.
- [30] C. A. Gueymard, “A reevaluation of the solar constant based on a 42-year total solar irradiance time series and a reconciliation of spaceborne observations,”

- Sol. Energy*, vol. 168, pp. 2–9, Jul. 2018.
- [31] “E490 Standard Solar Constant and Zero Air Mass Solar Spectral Irradiance Tables.” [Online]. Available: <https://www.astm.org/e0490-00ar19.html>. [Accessed: 19-Aug-2023].
 - [32] C. A. Gueymard, “The sun’s total and spectral irradiance for solar energy applications and solar radiation models,” *Sol. Energy*, vol. 76, no. 4, pp. 423–453, Apr. 2004.
 - [33] C. Eddy, “Aerosol Direct Radiative Forcing: A Review,” in *Atmospheric Aerosols - Regional Characteristics - Chemistry and Physics*, InTech, 2012.
 - [34] J. E. Penner *et al.*, “Aerosols, their Direct and Indirect Effects,” pp. 289–348, 2001.
 - [35] D. L. Hartmann, *Global physical climatology*. .
 - [36] H. D. Kambezidis, “The Solar Resource,” *Compr. Renew. Energy*, pp. 27–84, Jan. 2012.
 - [37] Solargis, “Solar resource maps of Ukraine,” *World Bank Group*, 2021. [Online]. Available: <https://solargis.com/maps-and-gis-data/download/ukraine>. [Accessed: 21-Aug-2023].
 - [38] P. Würfel, *Physics of Solar Cells*. Wiley, 2005.
 - [39] C. A. Gueymard, D. Myers, and K. Emery, “Proposed reference irradiance spectra for solar energy systems testing,” *Sol. Energy*, vol. 73, no. 6, pp. 443–467, Dec. 2002.
 - [40] ASTM, “Standard Tables for Reference Solar Spectral Irradiances : Direct Normal and Hemispherical on 37° Tilted Surface,” *ASTM*, vol. 03, no. Reapproved, pp. 1–21, 2013.
 - [41] S. M. Sze and K. K. Ng, *Physics of Semiconductor Devices*. Wiley, 2006.
 - [42] J. Pastuszak and P. Węgierek, “Photovoltaic Cell Generations and Current Research Directions for Their Development,” *Materials (Basel)*., vol. 15, no. 16, p. 5542, Aug. 2022.
 - [43] R. A. Marques Lameirinhas, J. P. N. Torres, and J. P. de Melo Cunha, “A Photovoltaic Technology Review: History, Fundamentals and Applications,”

- [44] K. Ranabhat, L. Patrikeev, A. Antal'evna-Revina, K. Andrianov, V. Lapshinsky, and E. Sofronova, "An introduction to solar cell technology," *Istraz. i Proj. za privredu*, vol. 14, no. 4, pp. 481–491, 2016.
- [45] S. Rühle, "Tabulated values of the Shockley–Queisser limit for single junction solar cells," *Sol. Energy*, vol. 130, pp. 139–147, Jun. 2016.
- [46] F. Meillaud, A. Shah, C. Droz, E. Vallat-Sauvain, and C. Miazza, "Efficiency limits for single-junction and tandem solar cells," *Sol. Energy Mater. Sol. Cells*, vol. 90, no. 18–19, pp. 2952–2959, Nov. 2006.
- [47] N. N. Lal, Y. Dkhissi, W. Li, Q. Hou, Y. Cheng, and U. Bach, "Perovskite Tandem Solar Cells," *Adv. Energy Mater.*, vol. 7, no. 18, Sep. 2017.
- [48] "Best Research-Cell Efficiency Chart - NREL." [Online]. Available: <https://www.nrel.gov/pv/cell-efficiency.html>. [Accessed: 22-Aug-2023].
- [49] A. DJALAB, N. BESSOUS, M. M. REZAOUI, and I. MERZOUK, "Study of the Effects of Partial Shading on PV Array," in *2018 International Conference on Communications and Electrical Engineering (ICCEE)*, 2018, pp. 1–5.
- [50] T. J. Silverman, L. Mansfield, I. Repins, and S. Kurtz, "Damage in Monolithic Thin-Film Photovoltaic Modules Due to Partial Shade," *IEEE J. Photovoltaics*, vol. 6, no. 5, pp. 1333–1338, Sep. 2016.
- [51] S. K. Das, D. Verma, S. Nema, and R. K. Nema, "Shading mitigation techniques: State-of-the-art in photovoltaic applications," *Renew. Sustain. Energy Rev.*, vol. 78, pp. 369–390, Oct. 2017.
- [52] M. A. Mohamed, A. A. Zaki Diab, and H. Rezk, "Partial shading mitigation of PV systems via different meta-heuristic techniques," *Renew. Energy*, vol. 130, pp. 1159–1175, Jan. 2019.
- [53] E. V. Verbytskii and V. Y. Romashko, "Search features of solar battery maximum power point by methods open and short circuits," *Electron. Commun.*, vol. 18, no. 5, pp. 13–18, Dec. 2013.
- [54] R. Vieira, F. de Araújo, M. Dhimish, and M. Guerra, "A Comprehensive Review on Bypass Diode Application on Photovoltaic Modules," *Energies*,

vol. 13, no. 10, p. 2472, May 2020.

- [55] S. Silvestre, A. Boronat, and A. Chouder, "Study of bypass diodes configuration on PV modules," *Appl. Energy*, vol. 86, no. 9, pp. 1632–1640, Sep. 2009.
- [56] O. Shavolkin, I. Shvedchykova, and J. M. J. Jasim, "Improved control of energy consumption by a photovoltaic system equipped with a storage device to meet the needs of a local facility," *Eastern-European J. Enterp. Technol.*, vol. 2, no. 8 (110), pp. 6–15, Apr. 2021.
- [57] D. Prince Winston, S. Kumaravel, B. Praveen Kumar, and S. Devakirubakaran, "Performance improvement of solar PV array topologies during various partial shading conditions," *Sol. Energy*, vol. 196, pp. 228–242, Jan. 2020.
- [58] S. R. Pendem and S. Mikkili, "Modelling and performance assessment of PV array topologies under partial shading conditions to mitigate the mismatching power losses," *Sol. Energy*, vol. 160, pp. 303–321, Jan. 2018.
- [59] K. A. Bapurao, S. Mikkili, and P. K. Bonthagorla, "A Review on Static Reconfiguration Techniques of Solar PV to Mitigate Mismatch Loss and Minimize Partial Shading Effect," *IETE J. Res.*, vol. 69, no. 9, pp. 6356–6386, Sep. 2023.
- [60] M. Wolf, "New developments in silicon photovoltaic devices," *J. Br. Inst. Radio Eng.*, vol. 18, no. 10, pp. 583-594(11), 1958.
- [61] H. ~Y. Tada and J. Carter J.~R., *Solar cell radiation handbook*. 1977.
- [62] D. Bickler, "The simulation of solar radiation," *Sol. Energy*, vol. 6, no. 2, pp. 64–68, 1962.
- [63] H. S. Rauschenbach, "Electrical output of shadowed solar arrays," *IEEE Trans. Electron Devices*, vol. 18, no. 8, pp. 483–490, Aug. 1971.
- [64] B. S. Murthy, P. Tiku, and N. K. Sharma, "Some Comments on the Evaluation of Electrical Parameters of a Solar Cell," *IETE J. Res.*, vol. 21, no. 7, pp. 359–363, Jul. 1975.
- [65] J.-P. Charles, G. Bordure, A. Khoury, and P. Mialhe, "Consistency of the double exponential model with physical mechanisms of conduction for a solar

- cell under illumination,” *J. Phys. D. Appl. Phys.*, vol. 18, no. 11, pp. 2261–2268, 1985.
- [66] A. Polman, W. G. J. H. M. Van Sark, W. Sinke, and F. W. Saris, “A new method for the evaluation of solar cell parameters,” *Sol. Cells*, vol. 17, no. 2, pp. 241–251, 1986.
- [67] D. S. H. Chan and J. C. H. Phang, “Analytical methods for the extraction of solar-cell single- and double-diode model parameters from I-V characteristics,” *IEEE Trans. Electron Devices*, vol. 34, no. 2, pp. 286–293, 1987.
- [68] S. Lineykin, M. Averbukh, and A. Kuperman, “Five-parameter model of photovoltaic cell based on STC data and dimensionless,” in *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel*, 2012, pp. 1–5.
- [69] V. Tamrakar, S. C. Gupta, and Y. Sawle, “Single-Diode Pv Cell Modeling And Study Of Characteristics Of Single And Two-Diode Equivalent Circuit,” *Electr. Electron. Eng. An Int. J.*, vol. 4, pp. 13–24, 2015.
- [70] Maouhoub Nouredine, “Analytical Identification Method for the Single Diode Model Parameters of a Photovoltaic Panel using Datasheet Values,” *Журнал нано- та електронної фізики (Journal nano- Electron. physics)*, 2017.
- [71] N. Yildiran and E. Tacer, “Identification of photovoltaic cell single diode discrete model parameters based on datasheet values,” *Sol. Energy*, vol. 127, pp. 175–183, 2016.
- [72] S. Pindado and J. Cubas, “Simple mathematical approach to solar cell/panel behavior based on datasheet information,” *Renew. Energy*, vol. 103, pp. 729–738, 2017.
- [73] V. Khanna, B. K. Das, D. Bisht, Vandana, and P. K. Singh, “A three diode model for industrial solar cells and estimation of solar cell parameters using PSO algorithm,” *Renew. Energy*, vol. 78, pp. 105–113, 2015.
- [74] T. Ahmad, S. Sobhan, and F. Nayan, “Comparative Analysis between Single Diode and Double Diode Model of PV Cell: Concentrate Different Parameters

- Effect on Its Efficiency,” *J. Power Energy Eng.*, vol. 04, pp. 31–46, 2016.
- [75] L. Duan *et al.*, “Relationship Between the Diode Ideality Factor and the Carrier Recombination Resistance in Organic Solar Cells,” *IEEE J. Photovoltaics*, vol. 8, pp. 1701–1709, 2018.
- [76] H. Bayhan and M. Bayhan, “A simple approach to determine the solar cell diode ideality factor under illumination,” *Sol. Energy - Sol. ENERG*, vol. 85, pp. 769–775, 2011.
- [77] A. Fathy and H. Rezk, “Parameter estimation of photovoltaic system using imperialist competitive algorithm,” *Renew. Energy*, vol. 111, pp. 307–320, Oct. 2017.
- [78] A. Jain, S. Sharma, and A. Kapoor, “Solar cell array parameters using Lambert W-function,” *Sol. Energy Mater. Sol. Cells*, vol. 90, no. 1, pp. 25–31, Jan. 2006.
- [79] E. L. Meyer, “Extraction of Saturation Current and Ideality Factor from Measuring V_{oc} and I_{sc} of Photovoltaic Modules,” *Int. J. Photoenergy*, vol. 2017, pp. 1–9, 2017.
- [80] J. Cubas, S. Pindado, and C. de Manuel, “Explicit Expressions for Solar Panel Equivalent Circuit Parameters Based on Analytical Formulation and the Lambert W-Function,” *Energies*, vol. 7, no. 7, pp. 4098–4115, Jun. 2014.
- [81] S. Kato *et al.*, “A comparison of the aerosol thickness derived from ground-based and airborne measurements,” *J. Geophys. Res. Atmos.*, vol. 105, no. D11, pp. 14701–14717, Jun. 2000.
- [82] M. Bradley and M. Gasseller, “Measurement of Aerosols Optical Thickness of the Atmosphere using the GLOBE Handheld Sun Photometer,” *J. Vis. Exp.*, no. 147, May 2019.
- [83] M. Chin *et al.*, “Tropospheric Aerosol Optical Thickness from the GOCART Model and Comparisons with Satellite and Sun Photometer Measurements,” *J. Atmos. Sci.*, vol. 59, no. 3, pp. 461–483, Feb. 2002.
- [84] D. R. Myers, “Solar Radiation Resource Assessment for Renewable Energy Conversion,” in *Comprehensive Renewable Energy*, Elsevier, 2012, pp. 213–

- [85] D. Serrano, M. J. Marín, M. Núñez, S. Gandía, M. P. Utrillas, and J. A. Martínez-Lozano, “Relationship between the effective cloud optical depth and different atmospheric transmission factors,” *Atmos. Res.*, vol. 160, pp. 50–58, Jun. 2015.
- [86] P. Bak, C. Tang, and K. Wiesenfeld, “Self-organized criticality: An explanation of the $1/f$ noise,” *Phys. Rev. Lett.*, vol. 59, no. 4, pp. 381–384, Jul. 1987.
- [87] F. N. Hooge, T. G. M. Kleinpenning, and L. K. J. Vandamme, “Experimental studies on $1/f$ noise,” *Reports Prog. Phys.*, vol. 44, no. 5, pp. 479–532, May 1981.
- [88] D. L. Gilden, T. Thornton, and M. W. Mallon, “ $1/f$ Noise in Human Cognition,” *Science (80-.)*, vol. 267, no. 5205, pp. 1837–1839, Mar. 1995.
- [89] B. Kaulakys and T. Meškauskas, “Modeling $1/f$ noise,” *Phys. Rev. E*, vol. 58, no. 6, pp. 7013–7019, Dec. 1998.
- [90] M. Lagae, Ares & Lefebvre, Sylvain & Cook, Rob & Derosé, T. & Drettakis, George & Ebert, David & Lewis, J.P. & Perlin, Ken & Zwicker, “State of the Art in Procedural Noise Functions,” 2010.
- [91] E. Bourque and G. Dudek, “Procedural Texture Matching and Transformation,” *Comput. Graph. Forum*, vol. 23, no. 3, pp. 461–468, Sep. 2004.
- [92] J. R. Frisvad and G. Wyvill, “Fast high-quality noise,” in *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, 2007, pp. 243–248.
- [93] F. J. Harris, “On the use of windows for harmonic analysis with the discrete Fourier transform,” *Proc. IEEE*, vol. 66, no. 1, pp. 51–83, 1978.
- [94] J. Dong *et al.*, “Survey of Procedural Methods for Two-Dimensional Texture Generation,” *Sensors*, vol. 20, no. 4, p. 1135, Feb. 2020.
- [95] K. Perlin, “An image synthesizer,” *ACM SIGGRAPH Comput. Graph.*, vol. 19, no. 3, pp. 287–296, Jul. 1985.

- [96] S. Gustavson, “Simplex noise demystified,” 2005.
- [97] J. P. Lewis, “Algorithms for solid noise synthesis,” in *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, 1989, pp. 263–270.
- [98] Andrew S. Glassner, *Principles of Digital Image Synthesis*, vol. 1. San Francisco, California: Morgan Kaufmann Publishers, Inc., 1995.
- [99] Y. S. Yamnenko, V. V. Tielieha, and K. S. Niemchinova, “The use of Haar and OB wavelets in the signals analysis,” *Electron. Commun.*, vol. 22, no. 4, pp. 51–58, Aug. 2017.
- [100] D. Lyon, “The Discrete Fourier Transform, Part 4: Spectral Leakage,” *J. Object Technol.*, vol. 8, no. 7, p. 23, 2009.
- [101] D.-J. Jwo, W.-Y. Chang, and I.-H. Wu, “Windowing Techniques, the Welch Method for Improvement of Power Spectrum Estimation,” *Comput. Mater. Contin.*, vol. 67, no. 3, pp. 3983–4003, 2021.
- [102] A. Goldberg, M. Zwicker, and F. Durand, “Anisotropic noise,” *ACM Trans. Graph.*, vol. 27, no. 3, pp. 1–8, Aug. 2008.
- [103] J. D. Hoffman, J. D. Hoffman, and S. Frankel, *Numerical Methods for Engineers and Scientists*. CRC Press, 2018.
- [104] J. Ahmad, A. Ciocia, S. Fichera, A. F. Murtaza, and F. Spertino, “Detection of Typical Defects in Silicon Photovoltaic Modules and Application for Plants with Distributed MPPT Configuration,” *Energies*, vol. 12, no. 23, p. 4547, Nov. 2019.
- [105] A. D. Dhass, E. Natarajan, and L. Ponnusamy, “Influence of shunt resistance on the performance of solar photovoltaic cell,” in *2012 International Conference on Emerging Trends in Electrical Engineering and Energy Management (ICETEEEM)*, 2012, pp. 382–386.
- [106] S. Karthick, S. Velumani, and J. Bouclé, “Experimental and SCAPS simulated formamidinium perovskite solar cells: A comparison of device performance,” *Sol. Energy*, vol. 205, pp. 349–357, Jul. 2020.
- [107] N. Santakrus Singh, A. Jain, and A. Kapoor, “Determination of the solar cell

- junction ideality factor using special trans function theory (STFT),” *Sol. Energy Mater. Sol. Cells*, vol. 93, no. 8, pp. 1423–1426, Aug. 2009.
- [108] K. Ishaque, Z. Salam, and Syafaruddin, “A comprehensive MATLAB Simulink PV system simulator with partial shading capability based on two-diode model,” *Sol. Energy*, vol. 85, no. 9, pp. 2217–2227, 2011.
- [109] P. Salinas, D. Pavlidis, Z. Xie, A. Adam, C. C. Pain, and M. D. Jackson, “Improving the convergence behaviour of a fixed-point-iteration solver for multiphase flow in porous media,” *Int. J. Numer. Methods Fluids*, vol. 84, no. 8, pp. 466–476, 2017.
- [110] S. Butenko and P. M. Pardalos, *Numerical Methods and Optimization*. Chapman and Hall/CRC, 2014.
- [111] C. Pélabon, C. H. Hilde, S. Einum, and M. Gamelon, “On the use of the coefficient of variation to quantify and compare trait variation,” *Evol. Lett.*, vol. 4, no. 3, pp. 180–188, Jun. 2020.
- [112] I. Nassar-eddine, A. Obbadi, Y. Errami, A. El fajri, and M. Agunaou, “Parameter estimation of photovoltaic modules using iterative method and the Lambert W function: A comparative study,” *Energy Convers. Manag.*, vol. 119, pp. 37–48, 2016.
- [113] “KC200GT Specifications.” [Online]. Available: <https://www.energymatters.com.au/images/kyocera/KC200GT.pdf>.
- [114] “Merbenit ST40. Technical data sheet.” [Online]. Available: https://www.ulbrich.cz/chemical-technical-products/TDS_MERBENIT_ST_40_eng.pdf.
- [115] E. Cuce, P. M. Cuce, and T. Bali, “An experimental analysis of illumination intensity and temperature dependency of photovoltaic cell parameters,” *Appl. Energy*, vol. 111, pp. 374–382, Nov. 2013.
- [116] Suman, P. Sharma, and P. Goyal, “Analysing the effects of solar insolation and temperature on PV cell characteristics,” *Mater. Today Proc.*, vol. 45, pp. 5539–5543, 2021.
- [117] A. Zielinska, M. Skowron, and A. Bien, “Modelling of photovoltaic cells in

- variable conditions of temperature and intensity of solar insolation as a method of mapping the operation of the installation in real conditions,” in *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, 2018, pp. 200–204.
- [118] M. Barukčić, V. Ćorluka, and K. Miklošević, “The irradiance and temperature dependent mathematical model for estimation of photovoltaic panel performances,” *Energy Convers. Manag.*, vol. 101, pp. 229–238, Sep. 2015.
- [119] A. D. Dhass, P. Lakshmi, and E. Natarajan, “Investigation of Performance Parameters of Different Photovoltaic Cell Materials using the Lambert-W Function,” *Energy Procedia*, vol. 90, pp. 566–573, Dec. 2016.
- [120] E. Pavelyeva, “Image processing and analysis based on the use of phase information,” *Comput. Opt.*, vol. 42, no. 6, pp. 1022–1034, 2018.
- [121] A. V. Oppenheim and J. S. Lim, “The importance of phase in signals,” *Proc. IEEE*, vol. 69, no. 5, pp. 529–541, 1981.
- [122] F. H. Imai, M. R. Rosen, and R. S. Berns, “Comparative Study of Metrics for Spectral Match Quality,” *Conf. Colour Graph. Imaging, Vis.*, vol. 1, no. 1, pp. 492–496, Jan. 2002.
- [123] J. Benesty, J. Chen, Y. Huang, and I. Cohen, “Pearson Correlation Coefficient,” 2009, pp. 1–4.
- [124] Weisstein; Eric W., “Cross-Correlation Theorem.” [Online]. Available: <https://mathworld.wolfram.com/Cross-CorrelationTheorem.html>. [Accessed: 24-Jan-2024].
- [125] E. W. Weisstein, “Wiener-Khinchin Theorem .” [Online]. Available: <https://mathworld.wolfram.com/Wiener-KhinchinTheorem.html>. [Accessed: 25-Jan-2024].
- [126] L. Cohen, “The generalization of the Wiener-Khinchin theorem,” in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 3, pp. 1577–1580.
- [127] A. Oppenheim, A. Willsky, and S. Nawab, *Signals and Systems*, 2nd ed. Pearson, 1996.

- [128]D. E. Newland, *An Introduction to Random Vibrations, Spectral & Wavelet Analysis*, 3rd ed. Dover Publications, 2005.
- [129]M. Kutner, C. Nachtsheim, and J. Neter, *Applied Linear Regression Models-4th Edition with Student CD (McGraw Hill/Irwin Series: Operations and Decision Sciences)*, 4th ed. McGraw Hill, 2004.
- [130]J. G. Eisenhauer, “Regression through the Origin,” *Teach. Stat.*, vol. 25, no. 3, pp. 76–80, 2003.
- [131]I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 374, no. 2065, p. 20150202, Apr. 2016.
- [132]H. Abdi and L. J. Williams, “Principal component analysis,” *WIREs Comput. Stat.*, vol. 2, no. 4, pp. 433–459, Jul. 2010.
- [133]*Principal Component Analysis*. New York: Springer-Verlag, 2002.
- [134]L. David, L. Steven, and M. Judi, *Linear Algebra and Its Applications*, 5th ed. Pearson, 2014.
- [135]F. Pedro García Márquez, Ed., *Advances in Principal Component Analysis*. IntechOpen, 2022.
- [136]L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, 1st ed. SIAM: Society for Industrial and Applied Mathematics, 1997.
- [137]S. Kaur and S. K. Sharma, “An Efficient Iterative Methods for Solving Transcendental Equations,” 2023, pp. 191–203.
- [138]“Algorithm and Architecture Design of High-Quality Video Upscaling Using Database-Free Texture Synthesis,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 7, pp. 1221–1234, Jul. 2014.
- [139]P. Cignoni, C. Montani, C. Rocchini, R. Scopigno, and M. Tarini, “Preserving attribute values on simplified meshes by resampling detail textures,” *Vis. Comput.*, vol. 15, no. 10, pp. 519–539, Dec. 1999.
- [140]Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, pp. 1398–1402.

- [141] A. Loza, L. Mihaylova, N. Canagarajah, and D. Bull, "Structural Similarity-Based Object Tracking in Video Sequences," in *2006 9th International Conference on Information Fusion*, 2006, pp. 1–6.
- [142] K. Bennett, Ed., *MATLAB Applications for the Practical Engineer*. InTech, 2014.
- [143] D. La Manna, V. Li Vigni, E. Riva Sanseverino, V. Di Dio, and P. Romano, "Reconfigurable electrical interconnection strategies for photovoltaic arrays: A review," *Renew. Sustain. Energy Rev.*, vol. 33, pp. 412–426, May 2014.
- [144] X. H. Nguyen and M. P. Nguyen, "Mathematical modeling of photovoltaic cell/module/arrays with tags in Matlab/Simulink," *Environ. Syst. Res.*, vol. 4, no. 1, p. 24, Dec. 2015.
- [145] "Photovoltaic solar cell - MATLAB." [Online]. Available: <https://www.mathworks.com/help/sps/ref/solarcell.html>. [Accessed: 01-Feb-2024].
- [146] Z. Zhu, M. Hou, L. Ding, G. Zhu, and Z. Jin, "Optimal Photovoltaic Array Dynamic Reconfiguration Strategy Based on Direct Power Evaluation," *IEEE Access*, vol. 8, pp. 210267–210276, 2020.

ДОДАТОК А. Рівняння математичної моделі сонячної панелі

При виведенні рівнянь математичної моделі СП використовується рівняння вихідного струму з рядом спрощень:

$$I = I_{PV} - \frac{V + IR_S}{R_P} - I_0 \left[e^{\frac{V + IR_S}{\alpha_1 V_T}} + e^{\frac{V + IR_S}{\alpha_2 V_T}} - 2 \right] \quad (5.1)$$

Також приймемо, що значення струму КЗ на основі значень опорів R_S та R_P визначається наступним чином:

$$I(0) = \frac{I_{PV}}{1 + R_S / R_P} \quad (5.2)$$

Поблизу точки ХХ рівняння вихідного струму приймає вигляд:

$$0 = I_{PV} - \frac{V_{OC}}{R_P} - I_0 \left[e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2 \right] \quad (5.3)$$

Звідки отримаємо вираз для зворотного струму діодів VD1, VD2:

$$I_0 = \frac{I_{PV} - \frac{V_{OC}}{R_P}}{e^{\frac{V_{OC}}{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2} \quad (5.4)$$

Проведемо диференціювання рівняння вихідного струму СП (5.1). Для цього приймемо:

$$U = V + IR_S \quad (5.5)$$

Звідси, за правилом диференціювання складеної функції маємо:

$$I' = \frac{dI}{dV} = \frac{dI}{dU} \frac{dU}{dV} \quad (5.6)$$

Враховуючи заміну змінної (5.5) та правило диференціювання складеної функції (5.7), маємо:

$$I' = 0 - \frac{1}{R_P} (1 + I'R_S) - I_0 \left[\frac{1 + I'R_S}{\alpha_1 V_T} e^{\frac{U}{\alpha_1 V_T}} + \frac{1 + I'R_S}{\alpha_2 V_T} e^{\frac{U}{\alpha_2 V_T}} \right] \quad (5.8)$$

Спрощуючи (5.8), отримуємо вираз для похідної вихідного струму СП:

$$\frac{I'}{1 + I'R_s} = -\frac{1}{R_p} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V+IR_s}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V+IR_s}{\alpha_2 V_T}} \right] \quad (5.9)$$

Підставивши значення напруги ХХ V_{OC} в отриманий вираз (5.9), знайдемо похідну виразу від вихідного струму поблизу точки ХХ:

$$\frac{I'_{OC}}{1 + I'_{OC}R_s} = -\frac{1}{R_p} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V_{OC}}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V_{OC}}{\alpha_2 V_T}} \right] \quad (5.10)$$

На основі рівнянь похідної від вихідного струму поблизу точки ХХ (5.10) та значення струму КЗ (5.2) маємо наступну систему рівнянь:

$$\begin{cases} I(0) = \frac{I_{PV}}{1 + R_s/R_p}, \\ I'_{SC} = -\frac{1}{R_s + R_p}. \end{cases} \quad (5.11)$$

Вирішимо дану систему відносно опорів R_s та R_p . Для цього спершу знайдемо вираз для послідовного опору R_s , використовуючи друге рівняння:

$$R_s = -\frac{1 + I'_{SC}R_p}{I'_{SC}} \quad (5.12)$$

Підставивши (5.12) в перше рівняння системи (5.11), отримаємо наступне:

$$I(0) = \frac{I_{PV}R_p}{R_p - \frac{1 + I'_{SC}R_p}{I'_{SC}}} = -I'_{SC}I_{PV}R_p \quad (5.13)$$

Звідки вираз для шунтуючого опору:

$$R_p = -\frac{I(0)}{I_{PV}I'_{SC}} \quad (5.14)$$

Підставивши (5.14) в (5.12), отримаємо вираз для послідовного опору R_s , який залежить лише від значень струму та похідної:

$$R_s = -\frac{1 - I(0)/I_{PV}}{I'_{SC}} = -\frac{1}{I'_{SC}} \left(1 - \frac{I(0)}{I_{PV}} \right) \quad (5.15)$$

Для отримання виразу для фотоструму спершу підставимо вираз для шунтуючого опору R_p (5.14) у вираз для зворотнього струму I_0 (5.4):

$$I_0 = \frac{I_{PV} + \frac{I'_{SC} I_{PV} V_{OC}}{I(0)}}{\frac{V_{OC}}{e^{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2} = \frac{I_{PV} I(0) + I'_{SC} I_{PV} V_{OC}}{I(0)} \frac{1}{\frac{V_{OC}}{e^{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2} \quad (5.16)$$

Далі підставимо вирази для опорів R_s та R_p в вираз для похідної від вихідного струму поблизу точки ХХ (5.10):

$$\frac{\frac{I'_{OC}}{1 - \frac{I'_{OC}}{I'_{SC}} \left(1 - \frac{I(0)}{I_{PV}}\right)}} = \frac{I_{PV} I'_{SC}}{I(0)} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V_{OC}}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V_{OC}}{\alpha_2 V_T}} \right] \quad (5.17)$$

Спростивши (5.17), отримаємо:

$$\frac{I'_{OC} I'_{SC} I_{PV}}{I'_{SC} I_{PV} - I'_{OC} I_{PV} + I'_{OC} I(0)} = \frac{I'_{SC} I_{PV}}{I(0)} - \frac{I_0}{V_T} \left[\frac{1}{\alpha_1} e^{\frac{V_{OC}}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V_{OC}}{\alpha_2 V_T}} \right] \quad (5.18)$$

Далі, підставивши вираз для зворотнього струму (5.16) в отриманий вираз, маємо:

$$\frac{I'_{OC} I'_{SC} I_{PV}}{I'_{SC} I_{PV} - I'_{OC} I_{PV} + I'_{OC} I(0)} = \frac{I'_{SC} I_{PV}}{I(0)} - \frac{I_{PV} I(0) + I_{PV} I'_{SC} V_{OC}}{I(0) V_T} \beta, \quad (5.19)$$

$$\text{де } \beta = \frac{\frac{1}{\alpha_1} e^{\frac{V_{OC}}{\alpha_1 V_T}} + \frac{1}{\alpha_2} e^{\frac{V_{OC}}{\alpha_2 V_T}}}{\frac{V_{OC}}{e^{\alpha_1 V_T}} + e^{\frac{V_{OC}}{\alpha_2 V_T}} - 2}$$

Приведемо до спільного знаменника та скоротимо I_{PV} в обох частинах рівняння (5.19):

$$\frac{I'_{OC} I'_{SC} I_{PV}}{I'_{SC} I_{PV} - I'_{OC} I_{PV} + I'_{OC} I(0)} = \frac{I'_{SC} I_{PV} V_T - I_{PV} I(0) \beta + I_{PV} I'_{SC} V_{OC} \beta}{I(0) V_T} \quad (5.20)$$

$$\frac{I'_{OC} I'_{SC}}{I'_{SC} I_{PV} - I'_{OC} I_{PV} + I'_{OC} I(0)} = \frac{I'_{SC} V_T - I(0) \beta + I'_{SC} V_{OC} \beta}{I(0) V_T}$$

Проведемо подальше спрощення.

$$\begin{aligned}
\frac{I'_{oc}I'_{sc}I(0)V_T}{I'_{sc}I_{PV} - I'_{oc}I_{PV} + I'_{oc}I(0)} &= I'_{sc}V_T - I(0)\beta + I'_{sc}V_{oc}\beta \\
I'_{sc}I_{PV} - I'_{oc}I_{PV} + I'_{oc}I(0) &= \frac{I'_{oc}I'_{sc}I(0)V_T}{I'_{sc}V_T - I(0)\beta + I'_{sc}V_{oc}\beta} \\
I_{PV}(I'_{sc} - I'_{oc}) + I'_{oc}I(0) &= \frac{I'_{oc}I'_{sc}I(0)V_T}{I'_{sc}V_T - I(0)\beta + I'_{sc}V_{oc}\beta} \\
I_{PV}(I'_{sc} - I'_{oc}) &= \frac{I'_{oc}I'_{sc}I(0)V_T}{I'_{sc}V_T - I(0)\beta + I'_{sc}V_{oc}\beta} - I'_{oc}I(0)
\end{aligned}$$

Звідки вираз для фотоструму:

$$I_{PV} = \frac{I'_{oc}I'_{sc}I(0)V_T}{[I'_{sc}V_T - I(0)\beta + I'_{sc}V_{oc}\beta][I'_{sc} - I'_{oc}]} - \frac{I'_{oc}I(0)}{I'_{sc} - I'_{oc}} \quad (5.21)$$

ДОДАТОК Б. Програмна реалізація ВАХ сонячної панелі

Для роботи зі запропонованою моделлю СП використовується мова програмування Python. Структурно код програми розподілений на дві частини:

1. Модуль з функціями загального призначення – `common.py`. Дані функції використовуються для вирішення таких завдань як обробка вихідних даних, перетворення типів даних, зчитування та запис даних у файли.

2. Класу `SolarPanel`, який знаходиться в однойменному модулі. Даний клас представляє собою інкапсуляцію даних та методів роботи з ними, які реалізують запропоновану математичну модель СП.

Використання коду передбачає встановлення наступних поширених програмних пакетів:

1. `NumPy` – популярний програмний пакет Python для виконання операцій з матрицями, а також ШПФ у одновимірному та двовимірному випадках.

2. `Matplotlib` – це комплексна бібліотека для створення статичних, анімованих та інтерактивних графіків на мові програмування Python.

Також використовуються наступні стандартні пакети, які входять в стандартну бібліотеку і поширюються разом з інтерпретатором Python:

1. `Typing` – програмний пакет для анотації типів даних функцій та методів, що спрощує написання та відлагодження коду.

2. `Math` – програмний пакет з реалізацією математичних функцій

3. `Os` – пакет, який реалізовує інтерфейс для роботи з файловою системою операційної системи.

Лістинг коду файлу `common.py`:

```
import numpy as np

def replace_str(initial_str: str, str_to_replace: str,
               replace_with: str) -> str:
    while str_to_replace in initial_str:
        initial_str = initial_str.replace(str_to_replace, replace_with)
    return initial_str

def get_data(file):
```

```

x_coord, y_coord = [], []
for line in file:
    buff = list(map(float, line.strip(" ").split()))
    x_coord.append(buff[0])
    y_coord.append(buff[1])
return x_coord, y_coord

def write_in_file(file, *iterables):
    for x_, y_ in zip(*iterables):
        file.write(f"{x_:.20f}\t\t{y_:.20f}\n")
    file.write(f"\n\n")

def read_file(filename, sep=',', outp_type=float):
    with open(filename) as file:
        lines = file.readlines()
        res = []
        for line in lines:
            res += [outp_type(i) for i in line.split(sep) if line]

    return res

def write_file(filename, data, sep=' '):
    with open(filename, 'w') as file:
        for line in data:
            str_line = str(line) + '\n' # sep.join(line) + '\n'
            file.write(str_line)

def rmse(arr1, arr2):
    diff = (arr1 - arr2) ** 2
    mse = np.sum(diff) / arr1.size
    rmse = np.sqrt(mse)
    return rmse

def minimize(init_cur, eta_min=0.5, eta_max=3, steps=500):
    min_error = 10 ** 10

    for eta in np.linspace(eta_min, eta_max, steps):
        approx = init_cur * np.exp(v / v_t / eta - 1)
        error = rmse(i, approx)
        errors.append(error)

        if error < min_error:
            min_error = error
            min_eta = eta

    return min_eta, min_error

def print_progress_bar(iteration, total, prefix='', suffix='', decimals=1,
                        length=100, fill='█', print_end='\r'):
    percent = ("{0:." + str(decimals) + "f}").format(100 * (iteration /
float(total)))
    filled_length = int(length * iteration // total)
    bar = fill * filled_length + '-' * (length - filled_length)
    print(f'\r{prefix} |{bar}| {percent}% {suffix}', end=print_end)
    # Print New Line on Complete
    if iteration == total:
        print()

```

Лістинг коду файлу SolarPanel.py:

```
from typing import Union, Sequence
from math import exp, floor, log
import numpy as np

class SolarPanel:
    def __init__(self,
                  cell_num: int = 36,
                  mesh_points_num: int = 1000,

                  Voc: float = 22.0,
                  Isc: float = 3.45,

                  Pmpp: float = 55,
                  Impp: float = 3.15,
                  Vmpp: float = 17.4,

                  experimental_V: Sequence[Union[float, int]] = (),
                  experimental_I: Sequence[Union[float, int]] = (),

                  Ku: Union[float, int] = 1.0,
                  Ki: Union[float, int] = 1.0,
                  Kp: Union[float, int] = 1.0,

                  T: Union[float, int] = 25,
                  Tstc: Union[float, int] = 25,

                  G: Union[float, int] = 1000,
                  Gstc: Union[float, int] = 1000,

                  custom_thermal_voltage_func=None,
                  custom_photovoltaic_current_func=None,
                  custom_backward_current_func=None,

                  thermal_coefficient_type="relative",
                  is_experimental_data_provided=False,

                  a1=1.0, a2=1.2,

                  I0=None, Ipv=None,
                  start_slope=None,
                  end_slope=None,
                  Vt=None,

                  fixed_point_method_tolerance=10 * 10 ** -12,
                  brute_force_steps=50,
                  mode="overall",
                  approx_range_minimization=None
                  ):

        if thermal_coefficient_type in ("relative", "absolute"):
            self.thermal_coefficient_type = thermal_coefficient_type
        else:
            raise ValueError("Unknown type of 'thermal_coefficient_type'
parameter")

        # Custom functions
        self.custom_thermal_voltage_func = custom_thermal_voltage_func
        self.custom_photovoltaic_current_func =
custom_photovoltaic_current_func
```

```

self.custom_backward_current_func = custom_backward_current_func

self.cell_num = cell_num

# Constants
self.k = 1.3806503 * 10 ** -23
self.q = 1.60217646 * 10 ** -19

# STC parameters
self.Gstc = Gstc
self.Tstc = Tstc + 273

# Environmental conditions
self.G = G
self.T = T + 273

# Experimental data
self.e_current = experimental_I
self.e_voltage = experimental_V

# Solar cell parameters
self.Voc = Voc
self.Isc = Isc
self.Pmpp = Pmpp
self.Impp = Impp
self.Vmpp = Vmpp

self.Ku = Ku
self.Ki = Ki
self.Kp = Kp
self.dT = self.T - self.Tstc

# Model parameters
self.I0 = I0
self.Ipv = Ipv
self.start_slope = start_slope
self.end_slope = end_slope
self.Rs = None
self.Rp = None
self.Vt = self.find_thermal_voltage() if Vt is None else Vt

# Impurity coefficients
self.a1 = a1
self.a2 = a2

# Algorithm's variables
self.is_experimental_data_provided = is_experimental_data_provided
self.fixed_point_method_tolerance = fixed_point_method_tolerance
self.mesh_points_num = mesh_points_num
self.brute_force_steps = brute_force_steps
self.brute_force_range = 0.3
self.mode = mode
self.eta = 0.2
self.approx_range_minimization = (floor(len(self.e_voltage) / 10) if
    approx_range_minimization is None
    else approx_range_minimization)

# Approximation
self.I_0 = self.e_current[0]
self.V_0 = self.e_voltage[-1]

if is_experimental_data_provided:
    if 0 in self.e_voltage:

```

```

        self.voltage = self.e_voltage
    else:
        self.voltage = [0.0] + self.e_voltage
    else:
        self.create_voltage_mesh()

    self.start_fit_points_number = floor(len(self.voltage) * 0.22)
    self.end_fit_points_number = floor(len(self.voltage) * 0.085)

    self.area1_2 = int(0.55 * len(self.voltage))
    self.area2_3 = int(0.9 * len(self.voltage))

    if self.start_slope is None:
        self.start_slope = self.find_slope("start")
    if self.end_slope is None:
        self.end_slope = self.find_slope("end")

    self.beta = self.find_beta()

    # Algorithm parameters
    self.current = None
    self.approx_error = None
    if self.Ipv is None:
        self.Ipv = self.find_photovoltaic_current()

    if self.I0 is None:
        self.Rs, self.Rp, self.I0 = self.find_parameters()

    def __str__(self):
        return (f'Rs = {self.Rs}' + f'\nRp = {self.Rp}' + f'\na1 = {self.a1}'
+
                f'\na2 = {self.a2}' + f'\nIpv = {self.Ipv}' + f'\nI0 =
{self.I0}'
                f'\nSlope1 = {self.start_slope}' + f'\nSlope2 =
{self.end_slope}')

    def recalc(self, G=1000, T=25, alpha=0.5, voltage=None):
        self.G = G
        self.T = T + 273

        self.Ipv = self.find_photovoltaic_current()
        self.Rp = self.Rp * self.Gstc / G * (self.Tstc / self.T)
        self.Rs = self.Rs * (self.Gstc / G) ** alpha * self.T / self.Tstc

        if self.T != self.Tstc:
            Voc = self.V_0 + self.Ku * (self.T - self.Tstc)
            self.I0 = (self.Ipv - Voc / self.Rp) / (np.exp(Voc / self.a1 /
self.Vt) +
                                                    np.exp(Voc / self.a2 /
self.Vt) - 2)

        self.current = self.fixed_point_method(self.Rs, self.Rp, voltage)
        return self.current

    def res_enum(self, weight_arr=None):
        self.current, self.Rs, self.Rp, self.approx_error =
self.find_best_fit(weight_arr)
        mape = self.find_mape(self.e_current, self.current)
        print(f"\nRMSE = {self.approx_error:.5f}")
        print(f"MAPE = {mape:.5f}")
        print(f"Prod = {mape * self.approx_error:.10f}")

```

```

def alpha_enum(self, weight_arr=None, use_one_diode_model=True,
recalculate=False):
    self.current, *_ , self.approx_error =
self.find_best_fit_1(weight_arr, use_one_diode_model,
recalculate)
    mape = self.find_mape(self.e_current, self.current)
    print(f"\nRMSE = {self.approx_error:.5f}")
    print(f"MAPE = {mape:.5f}")
    print(f"Prod = {mape * self.approx_error:.10f}")

def create_voltage_mesh(self) -> Sequence[Union[float, int]]:
    mesh_center_index = self.mesh_points_num // 2
    return np.concatenate(
        (
            np.linspace(0, self.Vmpp, mesh_center_index, endpoint=False),
            np.linspace(self.Vmpp, self.Voc, self.mesh_points_num -
mesh_center_index)
        )
    )

def find_thermal_voltage(self) -> Union[float, int]:
    if self.custom_thermal_voltage_func is None:
        return self.cell_num * self.k * self.T / self.q
    else:
        return self.custom_thermal_voltage_func()

def find_slope(self, slope_point='start') -> Union[float, int]:
    if slope_point == 'start':
        k, b = np.polyfit(self.voltage[:self.start_fit_points_number:],
self.e_current[:self.start_fit_points_number:],
1)
        return k
    if slope_point == 'end':
        a, b, c = np.polyfit(self.voltage[-self.end_fit_points_number::],
self.e_current[-
self.end_fit_points_number::], 2)
        return a * self.V_0 * 2 + b

def find_beta(self):
    self.beta = (exp(self.V_0 / self.Vt / self.a1) +
exp(self.V_0 / self.Vt / self.a2)) / \
(exp(self.V_0 / self.Vt / self.a1) +
exp(self.V_0 / self.Vt / self.a2) - 2)
    return self.beta

def find_photovoltaic_current(self) -> Union[float, int]:
    self.Ipv = ((self.end_slope * self.start_slope * self.I_0 * self.Vt)
/ (
        (self.start_slope * self.Vt - self.beta * self.I_0 -
self.beta * self.V_0 * self.start_slope) *
(self.start_slope - self.end_slope)) -
self.end_slope * self.I_0 / (self.start_slope -
self.end_slope))

    self.Ipv = self.Ipv + self.Ki * (self.T - self.Tstc)
    self.Ipv = self.Ipv * self.G / self.Gstc
    return self.Ipv

def find_parameters(self):
    Isc_g = self.I_0 * self.G / self.Gstc
    self.Rs = (Isc_g - self.Ipv) / self.Ipv / self.start_slope
    self.Rp = -Isc_g / self.start_slope / self.Ipv

```

```

        self.I0 = (self.Ipv - self.V_0 / self.Rp) / \
            (exp(self.V_0 / self.Vt / self.a1) + exp(self.V_0 / self.Vt
/ self.a2) - 2)
        return self.Rs, self.Rp, self.I0

    def find_backward_current(self):
        return ((self.Ipv - self.V_0 / self.Rp) /
            (exp(self.V_0 / self.Vt / self.a1) + exp(self.V_0 / self.Vt /
self.a2) - 2))

    def fixed_point_method(self, Rs, Rp, voltage=None):
        current = [self.Ipv - Rs * self.Ipv / Rp]
        if voltage is None:
            voltage = self.voltage[1::] if 0 in self.voltage else
self.voltage
        else:
            voltage = voltage[1:]

        lIb = log(self.I0)

        for V in voltage:
            prev_current = current[-1]
            while True:
                arg = (V + prev_current * Rs) / self.Vt
                new_current = self.Ipv - (V + prev_current * Rs) / Rp -
exp(arg / self.a1 + lIb) - \
                    exp(arg / self.a2 + lIb) + 2 * self.I0
                prev_current = prev_current * (1 - self.eta) + self.eta *
new_current

                if abs(prev_current - new_current) <
self.fixed_point_method_tolerance:
                    break
            current.append(prev_current)
        return current

    def fixed_point_method_1(self, a1, a2, voltage=None):
        current = [self.Ipv - self.Rs * self.Ipv / self.Rp]
        if voltage is None:
            voltage = self.voltage[1::] if 0 in self.voltage else
self.voltage
        else:
            voltage = voltage[1:]

        lIb = log(self.I0)

        for V in voltage:
            prev_current = current[-1]
            while True:
                arg = (V + prev_current * self.Rs) / self.Vt
                new_current = self.Ipv - (V + prev_current * self.Rs) /
self.Rp - \
                    exp(arg / a1 + lIb) - \
                    exp(arg / a2 + lIb) + 2 * self.I0
                prev_current = prev_current * (1 - self.eta) + self.eta *
new_current

                if abs(prev_current - new_current) <
self.fixed_point_method_tolerance:
                    break
            current.append(prev_current)
        return current

```

```

@staticmethod
def find_mape(arr1, arr2, weight_arr=None, ignore_shape_mismatch=False):
    arr1 = np.array(arr1)
    arr2 = np.array(arr2)

    if not ignore_shape_mismatch:
        if arr1.shape != arr2.shape:
            raise ValueError("Shape mismatch between input arrays.")

    weight_arr = np.ones_like(arr1) if weight_arr is None else weight_arr
    if arr1.shape != weight_arr.shape:
        raise ValueError("Shape mismatch between input arrays and weight
array.")

    error = 0
    for exper_val, approx_val, weight in zip(arr1, arr2, weight_arr):
        if exper_val == 0:
            continue
        error += weight * abs((exper_val - approx_val) / exper_val)

    mare = error / np.sum(weight_arr)
    return mare

@staticmethod
def find_rmse(arr1, arr2, weight_arr=None, ignore_shape_mismatch=False):
    arr1 = np.array(arr1)
    arr2 = np.array(arr2)

    if not ignore_shape_mismatch:
        if arr1.shape != arr2.shape:
            print(f"arr1 size {len(arr1)}")
            print(f"arr2 size {len(arr2)}")
            raise ValueError("Shape mismatch between input arrays.")

    weight_arr = np.ones_like(arr1) if weight_arr is None else weight_arr
    if arr1.shape != weight_arr.shape:
        raise ValueError("Shape mismatch between input arrays and weight
array.")

    wmse = 0
    for val_1, val_2, w in zip(arr1, arr2, weight_arr):
        wmse += w * (val_1 - val_2) ** 2
    wmse = wmse / np.sum(weight_arr)
    return np.sqrt(wmse)

@staticmethod
def progress_bar(idx, iter_num):
    idx += 1
    print('\rCompleted {}% |{:<20}|'.format(int(idx / iter_num * 100),
                                             '█' * int(idx / iter_num *
20)), end='')

def find_best_fit(self, weight_arr=None):
    c = self.brute_force_range
    best_fit_error = 100
    best_fit_current = None
    best_fit_rs = None
    best_fit_rp = None

    for idx, Rs in enumerate(np.linspace(self.Rs * (1 - c), self.Rs * (1
+ c),
                                     self.brute_force_steps)):
        self.progress_bar(idx, self.brute_force_steps)

```

```

        for Rp in np.linspace(self.Rp * (1 - c), self.Rp * (1 + c),
self.brute_force_steps):
            current = self.fixed_point_method(Rs, Rp)

            rmse = self.find_rmse(self.e_current, current, weight_arr)
            mape = self.find_mape(self.e_current, current)
            # approx_error = rmse * mape
            approx_error = rmse

            if approx_error < best_fit_error:
                best_fit_error = approx_error
                best_fit_current = current
                best_fit_rs = Rs
                best_fit_rp = Rp

            self.approx_error = best_fit_error
            self.current = best_fit_current
            self.Rs = best_fit_rs
            self.Rp = best_fit_rp
            return best_fit_current, best_fit_rs, best_fit_rp, best_fit_error

    def find_best_fit_1(self, weight_arr=None, use_one_diode_model=True,
recalc_on_each_step=False):
        best_fit_error = 100
        best_fit_current = None
        best_fit_params = None

        betas = []
        photocur = []
        rs = []
        rp = []
        backward_cur = []

        for idx, a1 in enumerate(np.linspace(1, 2, self.brute_force_steps,
endpoint=True)):
            self.progress_bar(idx, self.brute_force_steps)

            if not use_one_diode_model:
                for a2 in np.linspace(1, 2, self.brute_force_steps,
endpoint=True):
                    self.a1, self.a2 = a1, a2

                    if recalc_on_each_step:
                        self.find_beta()
                        self.find_photovoltaic_current()
                        self.find_parameters()

                    betas.append(self.beta)
                    photocur.append(self.Ipv)
                    rs.append(self.Rs)
                    rp.append(self.Rp)
                    backward_cur.append(self.I0)

                    current = self.fixed_point_method_1(a1, a2)
                    rmse = self.find_rmse(self.e_current, current,
weight_arr)

                    mape = self.find_mape(self.e_current, current)
                    # approx_error = rmse * mape
                    approx_error = rmse

                    if approx_error < best_fit_error:
                        best_fit_error = approx_error
                        best_fit_current = current

```

```

        best_fit_params = (self.a1, self.a2, self.Rs,
self.Rp, self.Ipv, self.I0)
    else:
        current = self.fixed_point_method_1(a1, 10 ** 10)
        approx_error = self.find_rmse(self.e_current, current,
weight_arr)

        if approx_error < best_fit_error:
            best_fit_error = approx_error
            best_fit_current = current
            best_fit_params = (self.a1, 10 ** 10, self.Rs, self.Rp,
self.Ipv, self.I0)

        self.approx_error = best_fit_error
        self.current = best_fit_current
        self.a1 = best_fit_params[0]
        self.a2 = best_fit_params[1]
        self.Rs = best_fit_params[2]
        self.Rp = best_fit_params[3]
        self.Ipv = best_fit_params[4]
        self.I0 = best_fit_params[5]
        return best_fit_current, *best_fit_params, best_fit_error

def find_best_fit_2(self, weight_arr=None):
    best_fit_error = 100
    best_fit_current = None
    best_fit_params = None

    Rs_vals = np.linspace(0.8 * self.Rs, 1.2 * self.Rs,
self.brute_force_steps, endpoint=True)
    Rp_vals = np.linspace(0.8 * self.Rp, 1.2 * self.Rp,
self.brute_force_steps, endpoint=True)

    for idx, a1 in enumerate(np.linspace(1, 2, self.brute_force_steps,
endpoint=True)):
        self.progress_bar(idx, self.brute_force_steps)
        self.a1 = a1
        for a2 in np.linspace(1, 2, self.brute_force_steps,
endpoint=True):
            self.a2 = a2
            for Rs in Rs_vals:
                for Rp in Rp_vals:
                    current = self.fixed_point_method(Rs, Rp)
                    rmse = self.find_rmse(self.e_current, current,
weight_arr)

                    mape = self.find_mape(self.e_current, current)
                    approx_error = rmse * mape

                    if approx_error < best_fit_error:
                        best_fit_error = approx_error
                        best_fit_current = current
                        best_fit_params = (self.a1, self.a2, self.Rs,
self.Rp,

                                                self.Ipv, self.I0)

            return best_fit_current, *best_fit_params, best_fit_error

    @staticmethod
    def find_power(voltage, current):
        return [V * I for V, I in zip(voltage, current)]

    @staticmethod
    def find_max_power_index(power):

```

```
max_power_index = 0
max_power = 0
for index, power in enumerate(power):
    if power > max_power:
        max_power_index = index
        max_power = power
return max_power_index, max_power
```

ДОДАТОК В. Програмна реалізація моделі хмарного покриву

Для роботи зі створеною моделлю хмарного покриву використовується мова спеціальний дистрибутив мови програмування Python – Anaconda. Даний дистрибутив включає ряд необхідних програмний пакетів, які спрощують роботу з графічними даними.

Структурно код програми розділено на дві частини:

1. Модуль з функціями загального призначення – `utils.py`. Дані функції використовуються для вирішення таких завдань як обробка вихідних даних, перетворення типів даних, зчитування та запис даних у файли, перетворення Фур'є та побудови графіків.

2. Документу Jupyter Notebook – `simulus.ipynb`. Даний файл виконує роль головного скрипту і містить код, який реалізовує на основі функцій загального призначення (`utils.py`) створену модель хмарного покриву.

Використання коду передбачає встановлення наступних програмних пакетів:

1. PIL – пакет для роботи і обробки зображень.
2. Scikit-image – пакет для визначення значення структурної схожості двох зображень.
3. Пакетів, перерахованих в додаток б.

Лістинг коду файлу `utils.py`:

```
import matplotlib.animation as animation
import matplotlib.pyplot as plt
import numpy as np
import plotly.graph_objects as go

from mpl_toolkits.axes_grid1 import make_axes_locatable
from PIL import Image

#####
#                                     Array utils                               #
#####

def get_slice(arr, rows, cols):
    if len(rows) != len(cols):
        raise ValueError('Row and column indexes arrays have different
sizes')

    arr_cols, arr_rows = arr.shape
```

```

sliced_arr = np.zeros_like(rows)

for index, (row, col) in enumerate(zip(rows, cols)):
    c_l, c_r = int(col), int(col) + 1
    r_t, r_b = int(row), int(row) + 1
    eta_col, eta_row = (col - c_l), (row - r_t)

    if c_r >= arr_cols:
        c_r = c_r - 1

    if r_b >= arr_rows:
        r_b = r_b - 1

    top = (1 - eta_row) * arr[r_t, c_l] + eta_row * arr[r_t, c_r]
    bottom = (1 - eta_row) * arr[r_b, c_l] + eta_row * arr[r_b, c_r]
    total = (1 - eta_col) * top + eta_col * bottom
    sliced_arr[index] = total
return sliced_arr

def normalize(arr, zero_padding=False, offset_coef=0.001):
    min_val = np.min(arr)
    max_val = np.max(arr)

    if zero_padding:
        new_arr = (arr - min_val) / (max_val - min_val)
    else:
        offset = offset_coef * abs(min_val)
        new_arr = (arr - min_val + offset) / (max_val - min_val + offset)
    return new_arr

def normalize_img(arr):
    arr_min = np.min(arr)
    arr_max = np.max(arr)

    if arr_min >= 0 and arr_max <= 255:
        return arr
    else:
        arr_norm = 255 * (arr - arr_min) / (arr_max - arr_min)
        return arr_norm

def threshold_arr_1d(arr, th_val, use_abs=False):
    th_val = abs(th_val) if use_abs else th_val
    for i in range(len(arr)):
        if arr[i] < th_val:
            arr[i] = th_val
    return arr

def threshold_arr_2d(arr, th_val, use_abs=False):
    for i in range(arr.shape[0]):
        arr[i, :] = threshold_arr_1d(arr[i, :], th_val, use_abs)
    return arr

def clip_graph(x_arr, y_arr, x_min=0, x_max=100, y_min=0, y_max=100):
    x_mask = (x_arr >= x_min) & (x_arr <= x_max)
    y_mask = (y_arr >= y_min) & (y_arr <= y_max)

    x_arr = x_arr[x_mask & y_mask]
    y_arr = y_arr[x_mask & y_mask]

```

```

    return x_arr, y_arr

#####
#                                     FFT utils                                     #
#####

def find_ft_1d(arr):
    ft = np.fft.fft(arr)
    return np.fft.fftshift(ft)

def find_ifft_1d(arr):
    ift = np.fft.ifftshift(arr)
    return np.fft.ifft(ift).real

def adjust_spectrum_1d(img):
    if len(img) % 2:
        return img
    else:
        return np.append(img, abs(img[0]))

def find_ft_2d(arr):
    ft = np.fft.fft2(arr)
    return np.fft.fftshift(ft)

def find_ifft_2d(arr):
    ift = np.fft.ifftshift(arr)
    return np.fft.ifft2(ift).real

def freq_numbers_1d(size):
    if size % 2:
        return np.arange(-(size // 2), size // 2 + 1, 1)
    else:
        return np.arange(-(size // 2), size // 2, 1)

def freq_arr_1d(size, step=1):
    freq = freq_numbers_1d(size)
    return freq / step / size

def freq_numbers_2d(shape):
    y_size, x_size = shape
    y_freq_numbers = freq_numbers_1d(y_size)
    x_freq_numbers = freq_numbers_1d(x_size)
    return x_freq_numbers, y_freq_numbers

def freq_arr_2d(shape, x_step=1, y_step=1):
    y_size, x_size = shape
    x_freq_numbers, y_freq_numbers = freq_numbers_2d(shape)
    x_freq = x_freq_numbers / x_step / x_size
    y_freq = y_freq_numbers / y_step / y_size
    return x_freq, y_freq

def freq_mesh_2d(shape):
    x_freq_numbers, y_freq_numbers = freq_numbers_2d(shape)
    x_mesh, y_mesh = np.meshgrid(x_freq_numbers, y_freq_numbers)

```

```

    return x_mesh, y_mesh

#####
#                                     Frequency Domain Filters                                     #
#####

def freq_pink_filter_1d(x_freq, factor=0.5, no_mean=False):
    x_freq = np.abs(x_freq)
    f = 1 / np.where(x_freq <= 1, 1, x_freq)
    f = f ** factor
    if no_mean:
        f_mask = x_freq < 1
        f[f_mask] = 0
    return f

def freq_pink_filter_1d_alt(x_freq, factor=0.5, no_mean=False):
    x_freq = np.abs(x_freq)
    f = 1 / (1 + x_freq)
    f = f ** factor
    if no_mean:
        f_mask = x_freq < 1
        f[f_mask] = 0
    return f

def freq_filter_2d(x_freq, y_freq):
    x, y = np.meshgrid(x_freq, y_freq)
    f = np.hypot(x, y)
    return f

def freq_pink_filter_2d(x_freq, y_freq, factor=1, x_stretch=1, y_stretch=1,
no_mean=False):
    x_freq, y_freq = np.abs(x_freq), np.abs(y_freq)
    fr = freq_filter_2d(x_freq / x_stretch, y_freq / y_stretch)
    f = 1 / np.where(fr <= 1, 1, fr)
    f = f ** factor
    if no_mean:
        f_mask = np.abs(fr) < 1
        f[f_mask] = 0
    return f

def freq_pink_filter_2d_alt(x_freq, y_freq, factor=1, x_stretch=1,
y_stretch=1, no_mean=False):
    f = freq_filter_2d(x_freq / x_stretch, y_freq / y_stretch)
    f = 1 / (1 + f)
    f = f ** factor
    f = np.where(f == 1, 0, f) if no_mean else f
    return f

def freq_sharp_round_filter_2d(x_freq, y_freq, radius, low_pass_filter=True):
    check = np.less if low_pass_filter else np.greater
    f = freq_filter_2d(x_freq, y_freq)
    f = np.where(check(f, radius), 1, 0)
    return f

def freq_sharp_square_filter_2d(x_freq, y_freq, width, angle=0,
low_pass_filter=True):
    check = np.less if low_pass_filter else np.greater
    angle_radians = np.radians(angle)

    x, y = np.meshgrid(x_freq, y_freq)
    rotated_x = x * np.cos(angle_radians) - y * np.sin(angle_radians)
    rotated_x = np.where(check(np.abs(rotated_x), width), 1, 0)

```

```

rotated_y = x * np.sin(angle_radians) + y * np.cos(angle_radians)
rotated_y = np.where(check(np.abs(rotated_y), width), 1, 0)

f = rotated_x * rotated_y
return f

#####
#                               Spatial Domain Utils                               #
#####

def spatial_smooth_filter(x_size, y_size, depth, horiz=True):
    values = np.linspace(0, 1, depth)
    values = 1 - fifth_order_interp(values)

    if horiz:
        kernel = np.tile(values, (y_size, 1))
    else:
        kernel = values[:, np.newaxis] * np.ones((1, x_size))
    return kernel

def make_img_transition_x(img, depth, is_dx_pos=True, outer_smooth=False):
    y_size, x_size = img.shape
    add_img = gen_cloud(x_size + depth, y_size)
    transition_kernel = spatial_smooth_filter(x_size, y_size, depth)

    img_copy = np.copy(img)
    if is_dx_pos:
        img_copy[:, -depth:] = (img[:, -depth:] * transition_kernel +
                                add_img[:, :depth] * (1 - transition_kernel))
        img_copy = np.concatenate((img_copy, add_img[:, depth:]), axis=1)
    else:
        transition_kernel = np.fliplr(transition_kernel)
        img_copy[:, :depth] = (img[:, :depth] * transition_kernel +
                                add_img[:, -depth:] * (1 - transition_kernel))
        img_copy = np.concatenate((add_img[:, 0:-depth], img_copy), axis=1)

    if outer_smooth:
        add_img = gen_cloud(2 * depth, y_size)
        transition_kernel = spatial_smooth_filter(x_size, y_size, depth)
        transition_kernel = np.fliplr(transition_kernel)
        img_copy[:, :depth] = (img_copy[:, :depth] * transition_kernel +
                                add_img[:, :depth] * (1 - transition_kernel))
        transition_kernel = np.fliplr(transition_kernel)
        img_copy[:, -depth:] = (img_copy[:, -depth:] * transition_kernel +
                                add_img[:, -depth:] * (1 -
transition_kernel))
    return img_copy

def make_img_transition_y(img, depth, is_dy_pos=True, outer_smooth=False):
    y_size, x_size = img.shape
    add_img = gen_cloud(x_size, y_size + depth)
    transition_kernel = spatial_smooth_filter(x_size, y_size, depth,
horiz=False)

    img_copy = np.copy(img)
    if is_dy_pos:
        img_copy[-depth:, :] = (img[-depth:, :] * transition_kernel + \
                                add_img[:depth, :] * (1 - transition_kernel))
        img_copy = np.concatenate((img_copy, add_img[depth:, :]), axis=0)
    else:
        transition_kernel = np.flipud(transition_kernel)
        img_copy[:depth, :] = (img[:depth, :] * transition_kernel +

```

```

        add_img[-depth:, :] * (1 - transition_kernel))
    img_copy = np.concatenate((add_img[:-depth, :], img_copy), axis=0)

    if outer_smooth:
        add_img = gen_cloud(x_size, 2 * depth)
        transition_kernel = spatial_smooth_filter(x_size, y_size, depth,
horiz=False)
        transition_kernel = np.flipud(transition_kernel)
        img_copy[:depth, :] = (img_copy[:depth, :] * transition_kernel +
                                add_img[:depth, :] * (1 - transition_kernel))
        transition_kernel = np.flipud(transition_kernel)
        img_copy[-depth:, :] = (img_copy[-depth:, :] * transition_kernel +
                                add_img[-depth:, :] * (1 -
transition_kernel))
    return img_copy

def make_img_transition_xy(img, depth, is_dx_pos=True, is_dy_pos=True,
outer_smooth=False):
    new_img = make_img_transition_x(img, depth, is_dx_pos=is_dx_pos,
outer_smooth=outer_smooth)
    new_img = make_img_transition_y(new_img, depth, is_dy_pos=is_dy_pos,
outer_smooth=outer_smooth)
    return new_img

def shift_img_x(img, dx, is_dx_pos=True):
    _, width = img.shape

    if is_dx_pos:
        c1 = img[:, -dx:]
        c2 = img[:, :-dx]
    else:
        c1 = img[:, dx:]
        c2 = img[:, :dx]
    return np.concatenate((c1, c2), axis=1)

def shift_img_y(img, dy, is_dy_pos=True):
    height, _ = img.shape

    if is_dy_pos:
        c1 = img[-dy:, :]
        c2 = img[:-dy, :]
    else:
        c1 = img[dy:, :]
        c2 = img[:dy, :]
    return np.concatenate((c1, c2), axis=0)

def shift_img_xy(img, dx, dy, is_dx_pos=True, is_dy_pos=True):
    shifted_img = shift_img_x(img, dx, is_dx_pos=is_dx

#####
#                                Other                                #
#####

def normalize_psd(original_magn, modified_magn):
    original_psd_sum = np.sum(original_magn ** 2)
    modified_psd_sum = np.sum(modified_magn ** 2)
    psd_coef = original_psd_sum / modified_psd_sum
    return np.sqrt(psd_coef)

def show_images(*images, vrange=None, x_label=None, figsize=(10, 10),
cmap='gray', aspect='equal', graphs_per_row=2):
    total_images = len(images)

```

```

    row_num = total_images // graphs_per_row + bool(total_images %
graphs_per_row)
    col_num = total_images // row_num + bool(total_images % row_num)
    vrange_individual = True if isinstance(vrange, dict) else False

    f, axes = plt.subplots(row_num, col_num, figsize=figsize)
    axes = np.array(axes).reshape(-1)

    for index, ax in enumerate(axes):
        if index < total_images:
            display_params = {'cmap': cmap, 'aspect': aspect}
            if vrange:
                if vrange_individual:
                    try:
                        display_params.update({'vmin': vrange[index][0],
'vmax': vrange[index][1]})
                    except KeyError:
                        pass
                else:
                    display_params.update({'vmin': vrange[0], 'vmax':
vrange[1]})

            if x_label:
                ax.set_xlabel(x_label[index])

            im = ax.imshow(images[index], **display_params)

            divider = make_axes_locatable(ax)
            cax = divider.append_axes('right', size='5%', pad=0.05)
            plt.colorbar(im, cax=cax)
        else:
            ax.axis('off')

    plt.tight_layout()
    return f, axes

def show_surfaces(*surfaces, axes=None, vrange=None, colorscale='Thermal',
showscale=True):
    if vrange:
        cmin, cmax = vrange
    else:
        z_data = surfaces if axes else [surf[-1] for surf in surfaces]
        cmin, cmax = np.min(z_data), np.max(z_data)

    fig = go.Figure()
    for surf in surfaces:
        x, y, z = (*axes, surf) if axes else surf
        fig.add_trace(go.Surface(x=x, y=y, z=z, cmin=cmin, cmax=cmax,
colorscale=colorscale, showscale=showscale))
        showscale = False
    return fig

def plot_graphs(*graphs, ax=None, figsize=None, grid=False, xlabel='',
ylabel='', title=''):
    if ax is None:
        fig, ax = plt.subplots()

    if figsize:
        fig.set_figheight(figsize[1])
        fig.set_figwidth(figsize[0])

    for graph in graphs:

```

```

        ax.plot(*graph) if isinstance(graph, (list, tuple)) else
ax.plot(graph)

    ax.grid(grid)
    ax.set_xlabel(xlabel)
    ax.set_ylabel(ylabel)
    ax.set_title(title)
    return ax

def open_img(filename, no_mean=True, grayscale=True):
    img = Image.open(filename)
    img_array = np.array(img, dtype=float)

    if grayscale:
        img_array = img.convert('L')

    if no_mean:
        img_array -= np.mean(img_array)
    return np.array(img_array, dtype=float)

def rmse(arr1, arr2, weight_arr=None, normalize=False):
    if arr1.shape != arr2.shape:
        raise ValueError("Shape mismatch between input arrays.")

    if weight_arr is None:
        weight_arr = np.ones_like(arr1)
    elif arr1.shape != weight_arr.shape:
        raise ValueError("Shape mismatch between input arrays and weight
array.")

    diff = weight_arr * (arr1 - arr2) ** 2
    mse = np.sum(diff) / arr1.size
    rmse = np.sqrt(mse)
    rmse = rmse / np.mean(weight_arr) if normalize else rmse
    return rmse

def print_progress_bar(iteration, total, prefix='', suffix='', decimals=1,
length=100, fill='█', print_end='\r'):
    percent = ("{0:." + str(decimals) + "f}").format(100 * (iteration /
float(total)))
    filled_length = int(length * iteration // total)
    bar = fill * filled_length + '-' * (length - filled_length)
    print(f'\r{prefix} |{bar}| {percent}% {suffix}', end=print_end)
    if iteration == total:
        print()

def find_index_by_val(arr, target_val, find_last=True):
    idx = -1
    for index, val in enumerate(arr):
        if val == target_val:
            idx = index

def is_symmetric(arr):
    res = np.full_like(arr, 1)

    for i in range(y_size):
        for j in range(x_size):
            row = y_size - i
            col = x_size - i

            if i == 0 or j == 0:
                continue

```

```

        if not np.isclose(magn[i, j], magn[y_size - i, x_size - j]):
            res[i, j] = 0

    return res

def find_radial_average(arr):
    avg_psd = dict()
    y_size, x_size = arr.shape
    y_shift, x_shift = y_size // 2, x_size // 2
    r_shift = (x_shift ** 2 + y_shift ** 2) ** 0.5
    r_arr = np.full_like(arr, 1)

    for row in range(y_size):
        for col in range(x_size):
            y = row - y_shift
            x = col - x_shift
            r = int(abs((x ** 2 + y ** 2) ** 0.5))
            r_arr[row, col] = r

            if r in avg_psd:
                avg_psd[r]['sum'] += arr[row, col]
                avg_psd[r]['count'] += 1

            else:
                avg_psd[r] = dict()
                avg_psd[r]['sum'] = arr[row, col]
                avg_psd[r]['count'] = 1

    for radius in avg_psd:
        avg_psd[radius]['avg'] = (avg_psd[radius]['sum'] /
                                   avg_psd[radius]['count'])

    rad = []
    avg = []
    for radius in sorted(avg_psd):
        rad.append(radius)
        avg.append(avg_psd[radius]['avg'])

    return np.array(rad), np.array(avg)

def autocorrelation_test(arr):
    arr_fft = find_ft_2d(arr)
    magn = np.abs(arr_fft)
    freq, magn_avg = radial_average_spectrum(magn)
    magn_avg = np.array(magn_avg)
    return freq, find_ifft_1d(magn_avg * magn_avg)

def gfc(arr1, arr2):
    if arr1.shape != arr2.shape:
        raise ValueError('Different array size')

    rows, cols = arr1.shape
    num = np.abs(np.sum(arr1 * arr2))
    denum = np.sqrt(np.sum(arr1 * arr1)) * np.sqrt(np.sum(arr2 * arr2))
    return num / denum

def find_autocorrelation_2d(arr):
    auto_cor = find_cross_correlation_2d(arr, arr)
    return auto_cor

```

```

def find_cross_correlation_2d(arr_1, arr_2):
    ft_1 = find_ft_2d(arr_1)
    ft_1_conj = np.conjugate(ft_1)
    ft_2 = find_ft_2d(arr_2)

    cross_cor_ft = ft_1_conj * ft_2
    cross_cor = find_ifft_2d(cross_cor_ft)
    cross_cor = np.fft.ifftshift(cross_cor)
    return cross_cor / arr_1.size / np.std(arr_1) / np.std(arr_2)
#####
#                               Lattice Noise Utils                               #
#####

def graph_mesh(x_min, x_max, x_step):
    mesh = np.arange(x_min, x_max + x_step, x_step)
    return mesh

def qubic_interp(x):
    return x * x * (3 - 2 * x)

def fifth_order_interp(x):
    return 6 * x ** 5 - 15 * x ** 4 + 10 * x ** 3

def gen_value_noise_1d(outp_noise_size, iterm_mesh_size, octaves_num,
persistence, func=None):
    random_values = []
    harmonics = []
    amplitude = persistence

    for _ in range(octaves_num):
        rv = amplitude * np.random.rand(outp_noise_size // iterm_mesh_size +
1)
        octave = gen_single_octave_1d(rv, outp_noise_size, iterm_mesh_size,
func=func)

        random_values.append(rv)
        harmonics.append(octave)
        iterm_mesh_size //= 2
        amplitude *= persistence
    return harmonics, random_values

def gen_single_octave_1d(random_values, size, grid_size, func=None):
    noise_map = np.zeros(size + 1)

    for x in range(size):
        cell_x = x // grid_size
        local_x = (x % grid_size) / grid_size
        left = random_values[cell_x]
        right = random_values[cell_x + 1]
        smooth_x = func(local_x) if func else local_x
        interp = left * (1 - smooth_x) + right * smooth_x
        noise_map[x] += interp

    noise_map[-1] = random_values[-1]
    return noise_map

```

```

def gen_value_noise_2d(outp_noise_shape, iterm_mesh_shape, octave_nums,
persistence, func=None):
    harmonics = []
    random_values = []
    h, w = outp_noise_shape
    gr_h, gr_w = iterm_mesh_shape
    amplitude = persistence

    for _ in range(octave_nums):
        rv = amplitude * np.random.rand(h // gr_h + 1, w // gr_w + 1)
        octave = gen_single_octave_2d(rv, outp_noise_shape, (gr_h, gr_w),
func=func)

        random_values.append(rv)
        harmonics.append(octave)
        gr_h, gr_w = gr_h // 2, gr_w // 2
        amplitude *= persistence

    # noise_map = np.sum(harmonics, axis=0)
    # noise_map = (noise_map - np.min(noise_map)) / (np.max(noise_map) -
np.min(noise_map))
    return harmonics

def gen_single_octave_2d(random_values, outp_noise_shape, iterm_mesh_shape,
func=None):
    h, w = outp_noise_shape
    gr_h, gr_w = iterm_mesh_shape
    noise_map = np.zeros((h + 1, w + 1))

    for y in range(h):
        for x in range(w):
            cell_x, cell_y = x // gr_w, y // gr_h
            local_x, local_y = (x % gr_w) / gr_w, (y % gr_h) / gr_h

            top_left = random_values[cell_y, cell_x]
            top_right = random_values[cell_y, cell_x + 1]
            bottom_left = random_values[cell_y + 1, cell_x]
            bottom_right = random_values[cell_y + 1, cell_x + 1]

            smooth_x = func(local_x) if func else local_x
            smooth_y = func(local_y) if func else local_y

            interp_top = top_left * (1 - smooth_x) + top_right * smooth_x
            interp_bottom = bottom_left * (1 - smooth_x) + bottom_right *
smooth_x
            noise_map[y, x] = interp_top * (1 - smooth_y) + interp_bottom *
smooth_y
            noise_map[-1, x] = bottom_left * (1 - smooth_x) + bottom_right *
smooth_x
            noise_map[y, -1] = top_right * (1 - smooth_y) + bottom_right *
smooth_y
            noise_map[-1, -1] = random_values[-1, -1]
    return noise_map

#####
#                               Other (Deprecation candidates)                               #
#####

def fit_clement(x_freq, y_freq, alpha1, alpha2, eta=0.1, angle=3):
    x, y = np.meshgrid(x_freq, y_freq)
    f = np.hypot(x, y)

```

```

fp = abs(angle * x + y)
f = np.sqrt((1 - eta) * f ** 2 + eta * fp ** 2)
f = (1 + abs(f)) ** (-alpha1) + 0.02 * (1 + abs(f)) ** (-alpha2)
return f

def fit_clement_new(x_freq, y_freq, alpha1, eta=0.1, slope=1):
    x, y = np.meshgrid(x_freq, y_freq)
    f = np.hypot(x, y)
    fp = abs(y - slope * x)
    f = np.sqrt((1 - eta) * f ** 2 + eta * fp ** 2)
    f = 1 / ((1 + abs(f)) ** alpha1)
    return f

def surrogates(x, ns, tol_pc=5., verbose=True, maxiter=1E6,
sorttype="quicksort"):
    # as per the steps given in Lancaster et al., Phys. Rep (2018)
    nx = x.shape[0]
    xs = np.zeros((ns, nx))
    maxiter = 10000
    ii = np.arange(nx)

    # get the fft of the original array
    x_amp = np.abs(np.fft.fft(x))
    x_srt = np.sort(x)
    r_orig = np.argsort(x)

    # loop over surrogate number
    pb_fmt = "{desc:<5.5}{percentage:3.0f}%|{bar:30}{r_bar}"
    pb_desc = "Estimating IAAFT surrogates ..."
    for k in range(ns):
        # 1) Generate random shuffle of the data
        count = 0
        r_prev = np.random.permutation(ii)
        r_curr = r_orig
        z_n = x[r_prev]
        percent_unequal = 100.

        # core iterative loop
        while (percent_unequal > tol_pc) and (count < maxiter):
            r_prev = r_curr

            # 2) FFT current iteration yk, and then invert it but while
            # replacing the amplitudes with the original amplitudes but
            # keeping the angles from the FFT-ed version of the random
            y_prev = z_n
            fft_prev = np.fft.fft(y_prev)
            phi_prev = np.angle(fft_prev)
            e_i_phi = np.exp(phi_prev * 1j)
            z_n = np.fft.ifft(x_amp * e_i_phi)

            # 3) rescale zk to the original distribution of x
            r_curr = np.argsort(z_n, kind=sorttype)
            z_n[r_curr] = x_srt.copy()
            percent_unequal = ((r_curr != r_prev).sum() * 100.) / nx

            # 4) repeat until number of unequal entries between r_curr and
            # r_prev is less than tol_pc percent
            count += 1

        if count >= (maxiter - 1):
            print("maximum number of iterations reached!")

```

```

        xs[k] = np.real(z_n)
    return xs

def gen_cloud(x_size, y_size, factor=2.4):
    xx = np.linspace(-x_size / 2, x_size / 2, x_size)
    yy = np.linspace(-y_size / 2, y_size / 2, y_size)
    whitenoise = np.random.normal(0, 1, (y_size, x_size))
    cloud_freq = find_ft_2d(whitenoise)
    kernel = freq_pink_filter_2d(xx, yy, factor=factor)
    cloud_freq_filtered = cloud_freq * kernel
    cloud_spatial = find_ifft_2d(cloud_freq_filtered).real
    return normalize_img(cloud_spatial)

def lin_phase(start, end, size):
    pos_freq = np.linspace(start, end, size // 2)
    neg_freq = -pos_freq[::-1]
    if size % 2:
        neg_freq = np.append(neg_freq, [0])
        freq = np.append(neg_freq, pos_freq)
    else:
        freq = np.append(neg_freq, pos_freq)
    return freq

def gaussian_new(x_vals, y_vals, std):
    exp = np.exp(-(x_vals ** 2 + y_vals ** 2) / (2 * std ** 2))
    return exp

def fit(x_freq, y_freq, alpha1, eta=0.1, angle=1):
    x, y = np.meshgrid(x_freq, y_freq)
    f = np.hypot(x, y)
    fp = abs(y - angle * x)
    f = np.sqrt((1 - eta) * f ** 2 + eta * fp ** 2)
    f = 1 / ((1 + abs(f)) ** alpha1)
    return f

def read_file(filename):
    with open(filename, 'rb') as file:
        res = np.load(file)
    return res

def write_file(filename, var):
    with open(filename, 'wb') as file:
        np.save(file, var)

```

Написання коду з використанням Jupyter Notebook передбачає використання комірок, код в яких виконуються послідовно та окремо один від одного. Початок кожної комірки в лістингу позначено наступним чином – # In[]. Лістинг коду документу `cumulus.ipynb`:

```

# In[ ]:
import random

```

```

from os import getcwd, listdir
from common.utils import *
from skimage.metrics import structural_similarity as ssim

figsize = (10, 8.5)
figsize_small = (5, 8.5)
figsize_default = (6.4, 4.8)

input_dir = getcwd() + '/img/input_samples/'
output_dir = getcwd() + '/files/'

fit_3par_error_fname = 'fit_three_par_error.txt'
fit_3par_vals_fname = 'fit_three_par_vals.txt'
fit_2par_error_fname = 'fit_two_par_error.txt'
fit_2par_vals_fname = 'fit_two_par_vals.txt'

SMALL_SIZE = 8
MEDIUM_SIZE = 14
BIGGER_SIZE = 16
MARKERSIZE = 100

plt.rc('font', size=SMALL_SIZE)
plt.rc('axes', titlesize=BIGGER_SIZE)
plt.rc('axes', labelsiz=MARKERSIZE)
plt.rc('xtick', labelsiz=MARKERSIZE)
plt.rc('ytick', labelsiz=MARKERSIZE)
plt.rc('legend', fontsize=MARKERSIZE)
plt.rc('figure', titlesize=SMALL_SIZE)

get_ipython().run_line_magic('matplotlib', 'inline')
get_ipython().run_line_magic('load_ext', 'pycodestyle_magic')

# In[ ]:
img = open_img(input_dir + 'image.jpg')
img_fr = find_ft_2d(img)

img_shape = img.shape
y_size, x_size = img_shape
x_shift, y_shift = x_size // 2, y_size // 2
xx, yy = freq_numbers_2d(img_shape)
x, y = freq_mesh_2d(img_shape)

magn_raw = np.abs(img_fr)
magn = normalize(magn_raw)
magn_factor = normalize_psd(magn_raw, magn)
phase = np.angle(img_fr)
restored_img = find_ifft_2d(magn_factor * magn * np.exp(1j * phase))

f, ax = show_images(img, np.log10(magn), phase, restored_img,
figsize=figsize)
plt.show()

# In[ ]:
x_arr, y_arr = np.meshgrid(xx, yy)
window = normalize(1 - np.sqrt(x_arr ** 2 + y_arr ** 2))
window = window

windowed_img = window * img
window_ft = find_ft_2d(window)
windowed_img_ft = find_ft_2d(windowed_img)

```

```

vmin = np.min(windowed_img)
vmax = np.max(windowed_img)

magn_windowed = np.abs(windowed_img_ft)
magn_n = normalize(magn_windowed)
magn_l = np.log10(magn_n)
phase_windowed = np.angle(windowed_img_ft)
restored_img = find_ifft_2d(magn_windowed * np.exp(1j * phase_windowed)).real

f, ax = show_images(windowed_img, np.log10(magn_windowed), phase_windowed,
                    restored_img, figsize=figsize)
plt.show()

# In[ ]:
weight_exp = 2
threshold = 3.90865
width_x = 3
width_y = 0

# Removing lines along X-axis to get better angle estimation
min_magn = np.min(magn_windowed)
fit_magn = np.log10(magn_windowed / min_magn)
fit_no_clip = fit_magn

for w in range(-width_y, width_y):
    fit_magn[:, y_shift + w] = fit_magn[:, y_shift + width_y] * 0

for w in range(-width_x, width_x):
    fit_magn[x_shift + w, :] = fit_magn[x_shift + width_x, :]

# Clipping magnitude
fit_magn = np.where(fit_magn < threshold, 0, fit_magn)
fit_magn_max = np.exp(np.max(fit_magn))
# fit_magn = fit_magn / np.mean(fit_magn)

# Defining slope using regression
avg_xy = -np.sum(x * y * fit_magn ** weight_exp)
avg_y2 = np.sum(y * y * fit_magn ** weight_exp)
avg_x2 = np.sum(x * x * fit_magn ** weight_exp)
avg_z2 = np.sum(fit_magn ** weight_exp)
a = avg_xy / avg_y2
b = avg_xy / avg_x2
angle = np.arctan(-1 / a) * 180 / np.pi

# Defining STD along the lines with the found slopes
lperp2 = np.sum((x + a * y) ** 2 * fit_magn ** weight_exp) / avg_z2
lpara2 = np.sum((a * x - y) ** 2 * fit_magn ** weight_exp) / avg_z2
lperp = lperp2 ** 0.5
lpara = lpara2 ** 0.5

# Outputting results
print(f'A = {a}')
print(f'Angle = {angle}')
print(f'Slope = {1 / a}')
print(f'l_perp = {lperp}')
print(f'l_para = {lpara}')
print(f'l_perp corrected = {lperp2 ** 0.5}')
print(f'X indep: {-b}')
print(f'Y indep: {-a}')

# In[ ]:

```

```

max_line_len = int(max(img_shape) * 2 ** 0.5) + 1

# Parallel line
para_slope = -1 / a
x_line_para = np.linspace(0, x_size, x_size, endpoint=False)
y_line_para = para_slope * (x_line_para - x_size // 2) + y_size // 2

# Clipping arrays
x_line_para, y_line_para = clip_graph(x_line_para, y_line_para, x_max=x_size,
y_max=y_size)

# Making arrays to have the same shape as original array
x_line_min, x_line_max = int(np.min(x_line_para)), int(np.max(x_line_para))
x_line_para = np.linspace(x_line_min, x_line_max, max_line_len,
endpoint=False)
y_line_para = para_slope * (x_line_para - x_size // 2) + y_size // 2

# Perpendicular line
perp_slope = -1 / para_slope
x_line_perp = np.linspace(0, x_size, x_size, endpoint=False)
y_line_perp = perp_slope * (x_line_perp - x_size // 2) + y_size // 2

# Clipping arrays
x_line_perp, y_line_perp = clip_graph(x_line_perp, y_line_perp, x_max=x_size,
y_max=y_size)
x_line_min, x_line_max = int(np.min(x_line_perp)) + 1,
int(np.max(x_line_perp))
x_line_perp = np.linspace(x_line_min, x_line_max, max_line_len,
endpoint=False)
y_line_perp = perp_slope * (x_line_perp - x_size // 2) + y_size // 2

# Plotting
f, ax = show_images(magn_l, cmap='gray', figsize=figsize_small)
ax[0].plot(x_line_para, y_line_para)
ax[0].plot(x_line_perp, y_line_perp)

# In[ ]:
#1D regression along para line
line_mid = x_line_para.size // 2
x_indexes_para_n = np.around(x_line_para[line_mid:]) #.astype(int)
y_indexes_para_n = np.around(y_line_para[line_mid:]) #.astype(int)

x_vals_para_n = np.hypot(x_line_para - x_shift, y_line_para - y_shift)
x_vals_para_n = x_vals_para_n[line_mid:]
y_vals_para_n = get_slice(magn_n, y_indexes_para_n, x_indexes_para_n) #
magn_n[y_indexes_para_n, x_indexes_para_n]

max_val = np.max(y_vals_para_n)
mid = find_index_by_val(y_vals_para_n, max_val)
x_vals_para_n = x_vals_para_n[mid:]
y_vals_para_n = y_vals_para_n[mid:]

# Regression variables
x_log_para_n = np.log(1 + x_vals_para_n)
y_log_para_n = np.log(y_vals_para_n)
alpha_para = -2 * np.sum(y_log_para_n * x_log_para_n) / np.sum(x_log_para_n
** 2)
alpha_para = alpha_para / 2

para_approx = (1 + np.linspace(0, np.max(x_vals_para_n), x_vals_para_n.size))
** (-alpha_para)
# para_approx /= np.max(para_approx)

```

```

plt.plot(x_vals_para_n, y_vals_para_n)
plt.plot(x_vals_para_n, para_approx)
plt.yscale('log')

print(f'Mid = {mid}')
print(f'Fade para = {alpha_para}')
print(f'Arr max = {max_val}')

# In[ ]:
line_mid = x_line_perp.size // 2
x_indexes_perp_n = np.around(x_line_perp[line_mid:])
y_indexes_perp_n = np.around(y_line_perp[line_mid:])

x_vals_perp_n = np.hypot(x_line_perp - x_shift, y_line_perp - y_shift)
x_vals_perp_n = x_vals_perp_n[line_mid:]
y_vals_perp_n = get_slice(magn_n, y_indexes_perp_n, x_indexes_perp_n)

max_val = np.max(y_vals_perp_n)
mid = find_index_by_val(y_vals_perp_n, max_val)
x_vals_perp_n = x_vals_perp_n[mid:]
y_vals_perp_n = y_vals_perp_n[mid:]

# Regression variables
x_log_perp = np.log(1 + x_vals_perp_n)
y_log_perp = np.log(y_vals_perp_n)
alpha_perp = -2 * np.sum(y_log_perp * x_log_perp) / np.sum(x_log_perp ** 2)
alpha_perp = alpha_perp / 2

perp_approx = (1 + np.linspace(0, np.max(x_vals_perp_n), x_vals_perp_n.size))
** (-alpha_perp)
# perp_approx /= np.max(perp_approx)

plt.plot(x_vals_perp_n, y_vals_perp_n)
plt.plot(x_vals_perp_n, perp_approx)
plt.yscale('log')

print(f'Mid = {mid}')
print(f'Fade para = {alpha_perp}')
print(f'Arr max = {max_val}')

# In[ ]:
fig, ax = plt.subplots(1, 2, figsize=(10, 8.5 / 2))
ax1, ax2 = np.array(ax).reshape(-1)

ax1.plot(x_vals_para_n, y_vals_para_n)
ax1.plot(x_vals_para_n, para_approx, '--')
ax1.set_yscale('log')
ax1.grid()
ax1.set_ylabel('$A_p^{\prime}$')
ax1.set_xlabel('a')

ax2.plot(x_vals_perp_n, y_vals_perp_n)
ax2.plot(x_vals_perp_n, perp_approx, '--')
ax2.set_yscale('log')
ax2.grid()
ax2.set_xlabel('σ')

plt.tight_layout()

```

```

# In[ ]:
# Two parameters fit
min_error = 10 ** 10
alpha_steps = 50
eta_steps = 50

alphas = list(np.linspace(1, 2, alpha_steps))
etas = list(np.linspace(0, 1, eta_steps))
errors = np.zeros((alpha_steps, eta_steps))
images = []

power = 1
g = np.ones_like(img) # 0.5 * gaussian_new(x, y, max(lpara, lperp)) + 0.5
#gaussian(x, para_slope * y, lpara2_new ** 0.5) * gaussian(para_slope * x, -
y, lperp2_new ** 0.5)
weight = g ** power

for i, alpha in enumerate(alphas):
    print_progress_bar(i + 1, len(alphas), prefix = 'Progress:', suffix =
'Complete', length=50)
    for j, eta in enumerate(etas):
        my_fit = fit_clement_new(xx, yy, alpha, eta=eta, slope=para_slope)

        psd_factor = normalize_psd(magn_raw, my_fit)
        restored_img = find_ifft_2d(psd_factor * my_fit * np.exp(1j * phase))
        a, b = lin_regression(restored_img, img)
        restored_img = a * restored_img + b
        restored_img = np.clip(restored_img, vmin, vmax)

        rmse_error = rmse(img, restored_img)
        # rmse2 = rmse(np.log10(my_fit), magn_l, weight, normalize=True)
        error = rmse_error
        errors[i, j] = rmse_error
        images.append(restored_img)

        if min_error > error:
            min_error = error
            min_params = np.array([alpha, eta])
            best_fit = my_fit

write_file(output_dir + fit_2par_error_fname, errors)
write_file(output_dir + fit_2par_vals_fname, min_params)
print(*min_params, min_error)

# In[ ]:
# Visual results of two parameters enumeration
stretch_factor = np.sqrt(1 - alpha_para / alpha_perp)
stretch_factor = np.sqrt(alpha_perp / alpha_para - 1)
print(stretch_factor_1, stretch_factor_2)

if fit_2par_error_fname in listdir(output_dir):
    errors = read_file(output_dir + fit_2par_error_fname)
    min_params = read_file(output_dir + fit_2par_vals_fname)

f, ax = show_images(errors, figsize=figsize_small)
ax[-1].scatter([min_params[1] * eta_steps], [(min_params[0] - 1) *
alpha_steps])
ax[-1].scatter([stretch_factor * eta_steps], [(alpha_perp - 1) *
alpha_steps], marker='^')
ax[-1].set_xlabel('η')
ax[-1].set_ylabel('α')

```

```

plt.tight_layout()

# Approximation of original image
# Comparison between proposed approach and 2D regression with symetric
# magnitude kernel and original phase

# 2D regression
y_vals = magn
x_vals = freq_filter_2d(xx, yy)
y_log = np.log10(y_vals)
x_log = np.log10(1 + x_vals)
alpha = -np.sum(y_log * x_log) / np.sum(x_log ** 2)

tr_kernel = freq_pink_filter_2d(xx, yy, factor=alpha)
tr_kernel_factor = normalize_psd(magn_raw, tr_kernel)
tr_image = find_ifft_2d(tr_kernel_factor * tr_kernel * np.exp(1j *
phase_windowed))
a, b = lin_regression(tr_image, img)
tr_image = a * tr_image + b
tr_image = np.clip(tr_image, vmin, vmax)

# Proposed method approximation
r_kernel = fit_clement_new(xx, yy, alpha_perp, slope=para_slope,
eta=stretch_factor_2)
r_kernel_factor = normalize_psd(magn_raw, r_kernel)
r_image = find_ifft_2d(r_kernel_factor * r_kernel * np.exp(1j * phase))
a, b = lin_regression(r_image, img)
r_image = a * r_image + b
r_image = np.clip(r_image, vmin, vmax)

print(f'Traditional approach alpha = {alpha}')
print(r_err := rmse(img, r_image))
print(tr_err := rmse(img, tr_image))
print(abs(r_err - tr_err) / tr_err * 100)

f, ax = show_images(tr_image, r_image, vrange=(vmin, vmax), figsize=figsize,
x_label=['a', 'b'])
plt.show()

```

ДОДАТОК Г. Програмна реалізація комп'ютерного моделювання

При комп'ютерному моделюванні використовується програмний пакет Matlab Simulink. Окрім моделі Simulink використовуються скрипти-функції, короткий опис яких і лістинг яких наведено далі:

1. Скрипт-функція `get_model_param.m` – повертає значення вказаного параметру для вказаного блоку Simulink моделі. Лістинг даного файлу:

```
function [param_val] = get_model_param(model_name, block_name, control_param_name)
    block_path=strcat(model_name, '/', block_name);
    param_val = get_param(block_path, control_param_name);
end
```

2. Скрипт-функція `set_model_param.m` – встановлює вказане значення вказаного параметру для блоку Simulink моделі. Лістинг даного файлу:

```
function [] = set_model_param(model_name, block_name, control_param_name,
control_param_value)
    block_path=strcat(model_name, '/', block_name);
    set_param(block_path, control_param_name, num2str(control_param_value));
end
```

3. Скрипт-функція `read_data_from_file.m` – зчитує та повертає дані з файлу з вказаним іменем і форматом даних. Лістинг даного файлу:

```
function [var_values] = read_data_from_file(filename, format)
    fileID = fopen(filename, 'r');
    var_values = fscanf(fileID, format);
end
```

4. Скрипт-функція `run_simulation_scalar.m` – запускає симуляцію вказаної Simulink моделі та повертає результати моделювання у вигляді одного числового значення або ж скаляру. Лістинг файлу:

```
function [var_values] = run_simulation_scalar(model_name, model_time, var_names)
    if nargin < 2
        model_time = -1;
    end

    var_values = [];
    num_variables = size(var_names);
    num_variables = num_variables(1);
    simulink_control_warnings('off');

    for j=1:num_variables
        cur_var_name = var_names(j);
```

```

        eval_cmd = strcat("var_values.", cur_var_name, "=zeros(1,",
num2str(length(var_names)), ");");
        eval(eval_cmd);
    end

    for i=1:length(var_names)
        if (model_time ~= -1)
            sim(model_name, model_time);
        else
            sim(model_name);
        end

        for j=1:num_variables
            cur_var_name = var_names(i);
            cur_var_value = eval(strcat(cur_var_name, ".Data"));
            eval_cmd = strcat("var_values.", cur_var_name, "=",
num2str(max(cur_var_value)), ';');
            eval(eval_cmd);
        end
    end

    simulink_control_warnings("on");
end

```

5. Скрипт-функція `run_simulation_vector.m` – запускає симуляцію вказаної Simulink моделі та повертає результати моделювання у вигляді масиву числових значень або ж вектору. Лістинг файлу:

```

function [var_values] = run_simulation_vector(model_name, model_time, var_names)
    if nargin < 2
        model_time = -1;
    end

    var_values = [];
    num_variables = size(var_names);
    num_variables = num_variables(1);
    simulink_control_warnings('off');

    for j=1:num_variables
        cur_var_name = var_names(j);
        eval_cmd = strcat('var_values.', cur_var_name, '=zeros(1,',
num2str(length(var_names)), ');');
        eval(eval_cmd);
    end

    for i=1:length(var_names)
        if (model_time ~= -1)
            sim(model_name, model_time);
        else
            sim(model_name);
        end

        for j=1:num_variables
            cur_var_name = var_names(i);
            cur_var_value = eval(strcat(cur_var_name, '.Data'));
            eval_cmd = strcat('var_values.', cur_var_name, '=[');

            for k=1:length(cur_var_value)
                eval_cmd = strcat(eval_cmd, num2str(cur_var_value(k)), ');');
            end
        end
    end
end

```

```

        end

        eval_cmd = strcat(eval_cmd, '];');
        eval(eval_cmd);
    end
end

simulink_control_warnings('on');
end

```

6. Скрипт-функція `simulink_control_warnings.m` – використовується для заборони виводу некритичних попереджень в процесі моделювання. Лістинг файлу:

```

function simulink_control_warnings(cmd)

%w = warning('query','last')

%cmd - 'on' or 'off'
warning(cmd, 'Simulink:Engine:LineWithoutDst');
warning(cmd, 'Simulink:Engine:OutputNotConnected');
warning(cmd, 'Simulink:blocks:MatchingFromNotFound');
warning(cmd, 'Simulink:blocks:MatchingGotoNotFound');
warning(cmd, 'Simulink:Engine:UsingDefaultMaxStepSize');

end

```

Також використовуються наступні скрипти:

1. Скрипт `SP_cuves.m` – використовується для побудови ВАХ досліджуваних СП. Лістинг файлу:

```

clear
clc

g = [1000, 800, 600, 400, 200];

initial_param_value = get_model_param("SP_KC200GT", "G", "Value");
voltage_my = load_data_from_file("chapter_four_KC200GT_voltage.txt");
[var_values]=simulink_get_vector_data('SP_KC200GT', 35, 'G', 'Value', 1000, ["V"]);
voltage = var_values.V;
max_ = 0;

figure
box on;

subplot(1,2,1)
hold on;

for i=1:length(g)
    [res]=simulink_get_vector_data('SP_KC200GT', 35, 'G', 'Value', g(i), ["I"]);
    current = res.I;
    current_my = load_data_from_file(strcat("chapter_four_KC200GT_current_",
num2str(g(i)), ".txt"));
    current_max = max(current_my);
    max_ = max(current_max, max_);
end

```

```

        plot(voltage_my, current_my, "Marker", ".", "Color", "black")
        plot(voltage, current, "Color", "#0072BD")
    end
    ymax = max_ * 1.1;

    grid()
    xlim([0, 35])
    ylim([0, ymax])
    xlabel(["V, B", "a"], "FontSize", 16);
    ylabel("I, A", "FontSize", 16)
    ax = gca;
    ax.FontSize = 16;

    set_model_param("SP_KC200GT", "G", "Value", initial_param_value);

    subplot(1,2,1)
    initial_param_value = get_model_param("SP_ST40", "G", "Value");
    voltage_my = load_data_from_file("chapter_four_ST40_voltage.txt");
    [var_values]=simulink_get_vector_data('SP_ST40', 25, 'G', 'Value', 1000, ["V"]);
    voltage = var_values.V;
    max_ = 0;

    subplot(1,2,2)
    hold on;

    for i=1:length(g)
        [res]=simulink_get_vector_data('SP_ST40', 25, 'G', 'Value', g(i), ["I"]);
        current = res.I;
        current_my = load_data_from_file(strcat("chapter_four_ST40_current_",
num2str(g(i)), ".txt"));
        current_max = max(current_my);
        max_ = max(current_max, max_);

        plot(voltage_my, current_my, "Marker", ".", "Color", "black")
        plot(voltage, current, "Color", "#0072BD")
    end
    ymax = max_ * 1.1;

    grid()
    xlim([0, 25])
    ylim([0, ymax])
    xlabel(["V, B", "6"], "FontSize", 16);
    ylabel("I, A", "FontSize", 16)
    ax = gca;
    ax.FontSize = 16;

    set_model_param("SP_ST40", "G", "Value", initial_param_value);

```

2. Скрипт SP_KC200GT_array_no_com.m – використовується для побудови ВАХ масиву СП KC200GT без комутації для умов часткового затінення. Лістинг файлу:

```

clear
clc

g = vertcat( ...
    [1000, 1000, 1000, 1000], ...
    [1000, 1000, 1000, 500], ...
    [1000, 800, 1000, 400], ...

```

```

        [1000, 800, 600, 400]...
    );
    [ii, jj] = size(g);
    initial_g = zeros(jj, 1);
    figure

    for j=1:jj
        eval_str = strcat("G", num2str(j));
        initial_g(j) = str2double(get_model_param("SP_KC200GT_array", eval_str,
"Value"));
    end

    for i=1:ii
        for j=1:jj
            eval_str = strcat("G", num2str(j));
            set_model_param("SP_KC200GT_array", eval_str, "Value", num2str(g(i,j)));
        end

        [res] = run_simulation_vector("SP_KC200GT_array", 70, ["I", "V", "P"]);

        V = res.V;
        I = res.I;
        P = res.P;
        [el, idx] = max(P);

        subplot(ii, 1, i)
        hold on;
        plot(V, P)
        plot(V(idx), P(idx), "Color", "Red", "Marker", ".", "MarkerSize", 15)
        plot(V, 20 * I)
        plot(V(idx), 20 * I(idx), "Color", "Red", "Marker", ".", "MarkerSize", 15)
        grid()
        xlim([0, 70])
        ylim([0, max(P) * 1.1])
        ylabel("I, A", "FontSize", 16)
        ax = gca;
        ax.FontSize = 16;
        box on;

    end

    for j=jj
        eval_str = strcat("G", num2str(j));
        set_model_param("SP_KC200GT_array", eval_str, "Value", num2str(initial_g(j,1)));
    end

    xlabel("V, B", "FontSize", 16);

```

3. Скрипт SP_KC200GT_array_no_com_cloud.m – використовується для побудови ВАХ масиву СП KC200GT без комутації для умов часткового затінення, які реалізуються за допомогою запропонованої моделі хмарного покритву. Для кожної з СП масиву використовується відповідний файл з даними про опроміненість. Лістинг файлу:

```

format = '%F';
g1 = read_data_from_file("G1.txt", format);

```

```

g2 = read_data_from_file("G2.txt", format);
g3 = read_data_from_file("G3.txt", format);
g4 = read_data_from_file("G4.txt", format);

ii = size(g1);
power_ = zeros(ii);
current = zeros(ii);
voltage = run_simulation_vector("SP_KC200GT_array", 70, ["V"]);

for i=1:ii
    for j=1:4
        eval_str = strcat("G", num2str(j));
        g = eval(strcat("g", num2str(j), "(" , num2str(i), ");"));
        set_model_param("SP_KC200GT_array", eval_str, "Value", num2str(g));
    end

    [res] = run_simulation_scalar("SP_KC200GT_array", 70, ["I", "P"]);

    current(i) = res.I;
    power_(i) = res.P;
end

```

4. Скрипт SP_KC200GT_array_com.m – використовується для побудови ВАХ масиву СП KC200GT з комутацією для умов часткового затінення. Лістинг файлу:

```

clear
clc

c = vertcat( ...
    [1, 1, 0, 1, 0, 1, 0, 0], ...
    [1, 1, 0, 1, 0, 0, 1, 0], ...
    [1, 1, 0, 0, 1, 0, 1, 0], ...
    [1, 0, 1, 0, 1, 0, 1, 0]...
);
[ii, jj] = size(c);
initial_c = zeros(jj, 1);
figure

for j=1:jj
    eval_str = strcat("C", num2str(j));
    initial_c(j) = str2double(get_model_param("SP_KC200GT_array_com", eval_str, "Value"));
end

for i=1:ii
    for j=1:jj
        eval_str = strcat("C", num2str(j));
        set_model_param("SP_KC200GT_array_com", eval_str, "Value", num2str(c(i,j)));
    end

    [res] = run_simulation_vector("SP_KC200GT_array_com", 140, ["I", "V", "P"]);

    V = res.V;
    I = res.I;
    P = res.P;
    [el, idx] = max(P);

    subplot(ii, 1, i)

```

```

hold on;
plot(V, P)
plot(V(idx), P(idx), "Color", "Red", "Marker", ".", "MarkerSize", 15)
plot(V, 20 * I)
plot(V(idx), 20 * I(idx), "Color", "Red", "Marker", ".", "MarkerSize", 15)
grid()
xlim([0, 140])
ylim([0, max(P) * 1.1])
ylabel("I, A", "FontSize", 16)
ax = gca;
ax.FontSize = 16;
box on;

end

for j=jj
    eval_str = strcat("C", num2str(j));
    set_model_param("SP_KC200GT_array_com", eval_str, "Value",
num2str(initial_c(j,1)));
end

xlabel("V, B", "FontSize", 16);

```

5. Скрипт SP_KC200GT_array_com_cloud.m – використовується для побудови ВАХ масиву СП KC200GT з комутацією для умов часткового затінення, які реалізуються за допомогою запропонованої моделі хмарного покриття. Для кожної СП масиву використовується відповідний файл з даними про опроміненість та сигналами комутації. Лістинг файлу:

```

format = '%f';

g1 = read_data_from_file("G1.txt", format);
g2 = read_data_from_file("G2.txt", format);
g3 = read_data_from_file("G3.txt", format);
g4 = read_data_from_file("G4.txt", format);

c1 = read_data_from_file("C1.txt", format);
c2 = read_data_from_file("C2.txt", format);
c3 = read_data_from_file("C3.txt", format);
c4 = read_data_from_file("C4.txt", format);
c5 = read_data_from_file("C5.txt", format);
c6 = read_data_from_file("C6.txt", format);

ii = size(g1);
power_ = zeros(ii);
current = zeros(ii);
voltage = run_simulation_vector("SP_KC200GT_array_com_new", 70, ["V"]);

for i=1:ii
    for j=1:4
        eval_str = strcat("G", num2str(j));
        g = eval(strcat("g", num2str(j), "(", num2str(i), ");"));
        set_model_param("SP_KC200GT_array_com_new", eval_str, "Value", num2str(g));
    end

    for j=1:6

```

```

        eval_str = strcat("C", num2str(j));
        c = eval(strcat("c", num2str(j), "(", num2str(i), ");"));
        set_model_param("SP_KC200GT_array_com_new", eval_str, "Value", num2str(c));
    end

    [res] = run_simulation_scalar("SP_KC200GT_array_com_new", 70, ["I", "P"]);

    current(i) = res.I;
    power_(i) = res.P;
end

```

ДОДАТОК Д. Акт впровадження

«ЗАТВЕРДЖУЮ»
Проректор з навчальної
роботи КПІ ім. Ігоря Сікорського
Автоматично сформовано
Вихідний № _____
від «__» __/____

АКТ
про впровадження результатів дисертаційної роботи
Мартинюка Вадима Ігоровича
у навчальний процес Національного технічного університету України
«Київський політехнічний інститут імені Ігоря Сікорського»

Ми, що нижче підписалися, перший заступник декана факультету електроніки к.т.н., доц. Гармаш О.В., зав. кафедрою електронних пристроїв та систем д.т.н., доц. Вербицький Є.В., доцент кафедри електронних пристроїв та систем к.т.н., доц. Батрак Л.М., склали Акт впровадження результатів дисертаційної роботи аспіранта кафедри електронних пристроїв та систем Мартинюка В.І. у навчальний процес КПІ ім. Ігоря Сікорського.

Матеріали дисертаційної роботи Мартинюка В.І. використані у навчальному процесі кафедри електронних пристроїв та систем при підготовці бакалаврів за освітньою програмою «Електронні компоненти і системи», а саме, у рамках дисципліни «Мережі постійного і змінного струму з альтернативними джерелами енергії - І» додано лекцію, в якій:

- 1) Наведено детальну інформацію щодо структури, вихідних характеристик сонячних елементів, а також розглянуто основні проблеми, які виникають в процесі експлуатації реальних перетворювачів сонячної енергії та способи їх подолання.
- 2) Розглянуто класифікацію методів зменшення впливу ефекту часткового затінення.
- 3) Наведено алгоритм визначення параметрів еквівалентних схем заміщення сонячних панелей, який, завдяки введенню додаткового вихідного параметру (оцифрованої ВАХ сонячної панелі), дозволяє підвищити точність апроксимації у порівнянні з наявними методами.
- 4) Описано основні підходи до імітаційного моделювання сонячних панелей (за допомогою одно- та двохдіючої еквівалентних схем заміщення сонячного елемента) та створення моделі хмарного покриття (на основі використання шумів та на основі одно- та двовимірною перетворення Фур'є), які використовуються при дослідженні роботи сонячних панелей в умовах часткового затінення.

Перший заступник декана факультету електроніки, к.т.н., доц.	Оксана ГАРМАШ
Завідувач кафедри електронних пристроїв та систем, д.т.н., доц.	Євген ВЕРБИЦЬКИЙ
Доцент кафедри електронних пристроїв та систем, к.т.н., доц.	Лариса БАТРАК

ДОДАТОК Е. Заявка на патент на винахід

МІНЕКОНОМІКИ
НАЦІОНАЛЬНИЙ ОРГАН ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ
ДЕРЖАВНА ОРГАНІЗАЦІЯ
«УКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ОФІС
ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ ТА ІННОВАЦІЙ»
(УКРНОІВІ)

вул. Глазунова, 1, Київ, 01601, тел.: +380 44 209-27-06, +380 67 501-05-95
e-mail: office@nipo.gov.ua, http://www.nipo.gov.ua, код згідно з ЄДРПОУ 44673629

Розписка про одержання електронної заявки на винахід

Вх. № 367556 Дата одержання 26.07.2023 11:08:48

Номер заявки **а 2023 03604** (в подальшому обов'язково посилається на цей номер)
Відправник Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"
Назва винаходу Спосіб керування накопичувачем енергії

Подані матеріали

Вх-28200/2023 Заява про видачу патенту на винахід (КМ)
Заява_Сп-бКеруванНакопичувачем_25-07-23.index.xml, Заява_Сп-бКеруванНакопичувачем_25-07-23.pdf.1.p7s, Заява_Сп-бКеруванНакопичувачем_25-07-23.pdf, Заява_Сп-бКеруванНакопичувачем_25-07-23.bib1.xml
Лист
Лист.pdf.1.p7s, Лист.pdf, Лист.index.xml
Довіреність (документ, що підтверджує повноваження представника)
Довіреність (документ, що підтверджує повноваження представника)_доручення Пасічник_2023.pdf, Довіреність (документ, що підтверджує повноваження представника)_доручення Пасічник_2023.pdf.1.p7s, Довіреність (документ, що підтверджує повноваження представника)_доручення Пасічник_2023.index.xml
Реферат
Реферат_Сп-бКеруванНакопичувачем_25-07-23.pdf, Реферат_Сп-бКеруванНакопичувачем_25-07-23.pdf.1.p7s, Реферат_Сп-бКеруванНакопичувачем_25-07-23.index.xml
Креслення
Креслення_Сп-бКеруванНакопичувачем_25-07-23.pdf, Креслення_Сп-бКеруванНакопичувачем_25-07-23.pdf.1.p7s, Креслення_Сп-бКеруванНакопичувачем_25-07-23.index.xml
Лист
Лист_довідка неприбутковості.pdf, Лист_довідка неприбутковості.pdf.1.p7s, Лист_довідка неприбутковості.index.xml
Формула винаходу (КМ)
Формула_Сп-бКеруванНакопичувачем_25-07-23.index.xml, Формула_Сп-бКеруванНакопичувачем_25-07-23.pdf.1.p7s, Формула_Сп-бКеруванНакопичувачем_25-07-23.pdf
Опис винаходу (КМ)
Опис_Сп-бКеруванНакопичувачем_25-07-23.pdf, Опис_Сп-бКеруванНакопичувачем_25-07-23.pdf.1.p7s, Опис_Сп-бКеруванНакопичувачем_25-07-23.index.xml

Примітка:

Заявки, які подані в неробочий час, будуть опрацьовані наступного робочого дня.

Якщо подання заявки відбувається у неробочий час, у вихідний або святковий день, датою одержання заявки буде дата наступного робочого дня.

Подана електронна заявка вважається оригіналом. Надсилання паперової копії заявки не вимагається!