

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка програмної системи для математичного моделювання
процесів дифузії у багатошарових зразках з використанням мови програмування

Wolfram Mathematica

Виконав(ла): студент(ка) 4 курсу, групи СП-41

спеціальності 121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

Гикава В. А.

(підпис)

(прізвище та ініціали)

Керівник

Бойко І. В.

(підпис)

(прізвище та ініціали)

Нормоконтроль

Стоянов Ю. М.

(підпис)

(прізвище та ініціали)

Завідувач кафедри

Петрик М. Р.

(підпис)

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Тернопіль

2025

АНОТАЦІЯ

Обсяг роботи: дипломна робота складається з 79 сторінок основного тексту, містить 14 рисунків, 3 додатки та 30 джерела за переліком посилань.

Ключові слова: математичне моделювання, дифузія, цеоліти, скінченно-різницева схема, програмна система, Wolfram Mathematica, візуалізація.

У дипломній роботі досліджено математичне моделювання процесів дифузії в багат шарових мікропористих зразках, зокрема на основі цеолітів типів ZSM-5, ZSM-11, ZSM-12 та силікагелю. Метою є створення універсальної чисельної моделі, яка враховує геометрію, пористість, фізико-хімічні властивості середовищ і систему граничних умов. Побудовано систему рівнянь у частинних похідних, що описує масоперенос у порах і міжкристалітному середовищі, та виконано її апроксимацію скінченно-різницевими схемами. Реалізовано програмну систему у Wolfram Mathematica з алгоритмами прямої/зворотної прогонки, параметричним введенням даних і блоками 2D/3D візуалізації.

Програмна система дозволяє моделювати дифузію у зразках з довільною кількістю шарів (до 1000), змінювати параметри середовища та порівнювати чисельні результати з аналітичними. Проведено тестування на прикладах моно- та багат шарових структур, результати якого підтверджують точність моделей. Візуалізація забезпечує наочний аналіз просторових розподілів, що важливо для інженерів і дослідників.

Розроблений програмний комплекс має універсальне застосування в матеріалознавстві, хімічній технології, енергетиці та нанотехнологіях — для оптимізації пористих матеріалів, мембран, сенсорів і акумуляторів. Робота демонструє ефективне поєднання чисельного моделювання з інтерфейсною реалізацією, що робить систему придатною як для досліджень, так і для освітніх потреб.

ABSTRACT

Scope of the work: the thesis consists of 79 pages of main text, includes 14 figures, 3 appendices, and 30 references.

Keywords: mathematical modeling, diffusion, zeolites, finite difference scheme, software system, Wolfram Mathematica, visualization.

The thesis investigates the mathematical modeling of diffusion processes in multilayer microporous samples, particularly those based on zeolites of types ZSM-5, ZSM-11, ZSM-12, and silica gel. The objective is to develop a universal numerical model that accounts for geometry, porosity, physicochemical properties of media, and a complex system of boundary conditions. A system of partial differential equations describing mass transfer in pores and intercrystalline space has been constructed and approximated using finite difference schemes. A software system was implemented in Wolfram Mathematica, incorporating forward/backward sweep algorithms for solving difference equations, parameter input modules, and 2D/3D visualization blocks.

The software allows simulation of diffusion in samples with any number of layers (up to 1000), modification of environmental parameters, and comparison of numerical results with analytical solutions. Testing was conducted on examples of mono- and multilayer structures, with results confirming the accuracy of the models. Visualization enables clear analysis of spatial concentration distributions, which is valuable for engineers and researchers.

The developed software complex has universal applications in materials science, chemical engineering, energy systems, and nanotechnology—for the optimization of porous materials, membranes, sensors, and batteries. The work demonstrates an effective combination of numerical modeling and user interface implementation, making the system suitable for both research and educational purposes.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Сучасний стан технологій та застосування процесів дифузії в інженерній діяльності	9
1.2 Математичні моделі та програмні системи, що застосовуються до моделювання процесів дифузії	12
1.3. Актуальне програмне забезпечення, яке застосовується до ефективного розв’язування сіткових задач. Сучасний стан проблеми.....	20
РОЗДІЛ 2. РОЗРОБКА ТА ВЕРИФІКАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ ДИФУЗІЇ У БАГАТОШАРОВИХ ЗРАЗКАХ.....	24
2.1 Математична модель дифузії. Рівняння дифузії та граничні вимоги на межах досліджувальних зразків.	24
2.2 Апроксимація математичної моделі дифузії різницевою схемою, лінеаризація вихідних рівнянь та граничних умов.	31
2.3. Постановка задачі із програмної реалізації різницевої схеми математичної моделі дифузії методом прямої та зворотної прогонки.	34
РОЗДІЛ 3. РОЗРОБКА ТА ПРОЄКТУВАННЯ АРХІТЕКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МАТЕМАТИЧНИХ МОДЕЛЕЙ ДИФУЗІЇ В БАГАТОШАРОВИХ ЗРАЗКАХ.....	38
3.1 Вимоги до програмного забезпечення.	38
3.2 Вибір методології розробки.	41
3.3 Архітектура програмної системи.....	43
3.4 Розробка клієнт-машинного інтерфейсу та функціоналу програмної системи. Алгоритм методу. Розробка компонентів програмної системи.	49
3.5 Можливості інтеграції з іншими системами	51
3.6 Тестування роботи з програмною системою. Аналіз дифузії у різнотипних зразках, візуалізація результатів.	54

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	62
4.1 Ергономічні проблеми безпеки життєдіяльності	62
4.2 Значення автоматизації виробничих процесів в питаннях охорони праці..	64
ВИСНОВКИ.....	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	68
ДОДАТКИ	72
Додаток А - Лістинг програми	73
Додаток Б - Тези доповіді на конференції.....	77
Додаток В - Диск з роботою	79

ВСТУП

В останні роки в науці та технологіях спостерігається активне впровадження методологій, розроблених у сфері інформаційних технологій, зокрема інженерії програмного забезпечення. Методи математичного моделювання в поєднанні з об'єктно-орієнтованим програмуванням дають змогу досліджувати нові матеріали, мікро- і наноструктури на якісно новому рівні візуалізації та аналізу. Розробка нанозразків із заданими властивостями та геометрією потребує не лише знання фізико-хімічних характеристик матеріалів, а й використання складних програмних комплексів, які керують цими процесами та зменшують людський вплив.

Важливою складовою застосування ІТ у цій сфері є побудова математичних моделей, розробка числових методів та їх алгоритмізація в середовищах на кшталт MATLAB або Wolfram Mathematica. У даній дипломній роботі поставлено задачу побудови моделей процесів дифузійного переносу у складних багат шарових зразках із мікропористих матеріалів — цеолітів та силікагелю.

Для реалізації моделей ці процеси подаються у вигляді сіткових задач, апроксимованих скінченно-різницеви́ми схемами. З метою зручності користувача створено інтерфейс та архітектуру програмної системи. Розроблена інформаційна система є взаємопов'язаним програмним комплексом, який дозволяє обирати фізичні та геометричні параметри зразків, отримувати чисельні результати моделювання, а головне — інтерактивно візуалізувати їх. Програмна система має широке призначення: як для дослідників, так і для інженерів, з можливістю подальшого розвитку й адаптації її функціоналу.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Сучасний стан технологій та застосування процесів дифузії в інженерній діяльності

Процеси дифузії, що пов'язані з кінетикою масо- та теплоперенесення становлять активно та швидко модернізуючу частину сучасних технологій. Так, функціональні матеріали аморфного кристалічного чи напівпровідникового походження можуть виступати в ролі робочих агентів для транспорту різного типу речовин (молекул вуглеводнів, інертних газів, водяної пари) [1-3] , електронів та квазічастинок [4-6].

Робота з такими функціональними матеріалами є доволі складною та потребують комплексного застосування сучасних методів спектроскопії, матеріалознавства та особливо інформаційних технологій. Зокрема, вирощування мікроструктур даного типу потребує значного контролю над їхніми геометричними та просторовими розмірами, які фактично є визначальним фактором їх властивостей. У зв'язку з цим в інженерії програмного забезпечення виник цілий напрям, який стосується предметної області розробки програмних систем, що дозволяють попередньо варіювати властивостями та розмірами мікросистем [7, 8] . Це є надзвичайно важливим та корисним напрямом, оскільки таким способом за допомогою відповідного програмного забезпечення інженери та дослідники-експериментатори можуть підібрати необхідні для своїх задач фізичні та геометричні параметри мікроструктур та їх систем [9-11]. Також це суттєво заощаджує кошти оскільки вирощування кристалічних структур із наперед заданими параметрами методами молекулярно-променевої епітаксії чи іонного зміщення становить суму порядків кількох мільйонів євро [12-14]. Приклад такої установки для методами молекулярно-променевої епітаксії, що працює в Інституті промислової фізики та хімії у місті Париж подано на Рис.1 .

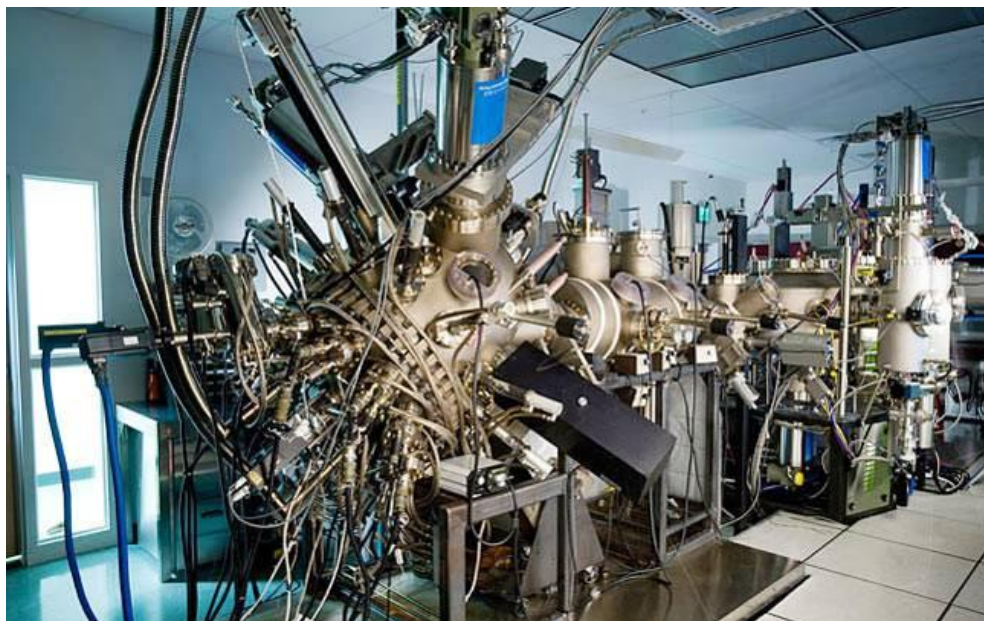


Рис.1.1. – Установка для молекулярно-променевої епітаксії з Інституту промислової фізики та хімії у місті Париж

Дана установка була введена в дію у 2011 році. Вона забезпечує вирощування необхідних мікро- та нано- структур чи їх масивів з точністю по геометричних розмірах в 1 до 3 %. Така висока точність забезпечується моніторингом процесу роботи програмним забезпеченням NanoSoft та багатоядерним процесом із 32 нодами по 128 ядер кожна [15]. Загадане програмне забезпечення є високоспецифікованим і призначення для розв'язування досить широкого колу задач, проте усі вони стосуються одного конкретного напрямку. Однією з проблем, які ставляться перед інженерією програмного забезпечення у цій предметній області це розширення функціоналу та специфікацій набору пакетів програм, що входить у даний програмний комплекс. В результаті над такими програмним забезпеченням одночасно можуть працювати сотні програмістів, які задіяні як у розробці математичних моделей, так і у написанні кінцевого коду, що реалізує моделі [16-17].

Отже, сучасний розвиток технологій дозволяє ефективно використовувати процеси дифузії в різних промислових застосуваннях, зокрема в матеріалознавстві, електроніці, хімічній інженерії та енергетиці.

В інженерії матеріалів процеси дифузії є ключовими для створення нових сплавів і композитів. Зокрема, в термообробці металів (цементация, азотування, легування) дифузія атомів вуглецю або азоту в металеві поверхні змінює їхні механічні властивості, підвищуючи твердість і зносостійкість, а також температуру обробки. Крім того, сучасні наноматеріали, такі як графен або вуглецеві нанотрубки, вимагають контрольованого процесу дифузії для їх вирощування та покращення й модифікації їхніх характеристик.

У виробництві напівпровідникових пристроїв, таких як мікрочіпи та транзистори, процес дифузії використовується для легування напівпровідникових матеріалів, таких як кремній. Контрольоване введення домішок через процеси дифузії дозволяє змінювати електричні властивості матеріалу, покращуючи функціональну продуктивність і ефективність електронних пристроїв. Також процеси дифузії є важливими у хімічній інженерії, хімічному виробництві та особливо каталізі. У дифузійних процесах, таких як масообмін і реакції в хімічних реакторах, дифузія забезпечує переміщення реагентів до каталізатора, що пришвидшує хімічні реакції. Технології мембранного розділення, які використовуються для очищення води або розділення сумішей газів, також базуються на процесах дифузії через напівпроникні мембрани із застосуванням згаданих вище цеолітів.

В енергетичному секторі процеси дифузії застосовуються в паливних елементах та акумуляторах. Зокрема, літій-іонні батареї, що використовуються в електромобілях та електроніці, залежать від дифузії іонів між катодом і анодом під час зарядки та розрядки. Контрольована дифузія в електроліті батарей дозволяє підвищити ефективність та тривалість роботи пристроїв. Отже, процеси дифузії відіграють критичну роль в сучасних технологіях і промислових процесах, допомагаючи розвивати нові функціональні матеріали, підвищувати ефективність виробництва і вдосконалювати енергетичні системи, й особливо чисту “зелену” енергетику.

1.2 Математичні моделі та програмні системи, що застосовуються до моделювання процесів дифузії

Розробка математичних моделей для процесів дифузії має досить довгу та складну історію. Своїми коренями вона лежить у дослідженнях на початку ХХ століття вченими у галузях математичної фізики та матеріалознавства. За своїм виглядом ці моделі були здебільшого аналітичними й безпосередньо значного впливу на розвиток технологій не мали. Все змінилося в 60-70 рр. минулого століття коли вченим Інституту кібернетики ім. Глушкова вдалося вперше виконати імітаційне та числове моделювання процесів дифузійної кінетики в масивних кристалах. Отримані результати по-перше, добре описували експериментальні результати, а по-друге, розроблене програмне забезпечення навіть у рамках тих часів могло бути швидко модифіковане у необхідності його застосування до систем з іншою специфікою та параметрами [18].

Сучасний стан дослідження даної проблеми часто полягає у двох конкуруючих напрямках. Перший напрямок це вузькоспеціалізоване програмне забезпечення, яке створюється фактично одноразово під якусь конкретну наукову проблему чи задачу. Таке програмне забезпечення ефективно та оптимально працює, проте воно має один значний мінус. Воно фактично не може бути модифікованим для суміжних проблем, а параметри вхідної математичної моделі можуть в такій програмній системі зміцнюватися лише у дуже вузькому діапазоні. Прикладом таких програмних систем є набір програмних пакетів NextNano, який оновлюється кілька разів в рік. При цьому, у набір пакета входить велика кількість програмних систем з яких безпосередньо користувачам буде корисними лише мала кількість. Як показує досвід дослідників спроби одночасного застосування кількох різних програмних систем NextNano можуть давати кардинально різні результати, що суттєво гальмує розвиток цієї предметної області з точки зору сучасних інформаційних технологій.

Якщо виходити із точки зору розробки математичних моделей дифузії, то базовою моделлю є наступна [9-11]. Приймається, що процес дифузії встановився у всьому континуумі досліджуваного зразка, при цьому кінетика дифузії характеризується динамікою процесу лише на межах середовища кристалічного матеріалу цеоліту та мікропор, які в ньому містяться. В такому разі математична модель дифузії виражається самоузгодженою системою рівнянь у частинних похідних, яка має в загальному випадку такий вигляд:

$$\begin{aligned} \frac{\partial c(z,t)}{\partial t} + \frac{\partial a(z,t)}{\partial t} + v \frac{\partial c(z,t)}{\partial z} &= D_{inter} \frac{\partial^2 c(z,t)}{\partial z^2}; \\ \frac{\partial a(z,t)}{\partial t} &= \beta [c(z,t) - c_{eq}(a)]. \end{aligned} \quad (1.1)$$

Перше із рівнянь у системі (1.1) застосовується для опису кінетики дифузії через усе тіло зразка, а друге рівняння процес дифузії летких часток через межу мікропори із середовищем матеріалу довільного цеоліту. Відшукування якихсь, навіть числових розв'язків системи (1.1.) представляє значні труднощі. З цією метою у праці [9] дифузія приймалась рівноважним процесом, що забезпечувалось умовою:

$$c_{eq}(a) = a / b(a_{full} - a), \quad (1.2)$$

де параметри a , b , a_{full} – приймались залежними від поточної концентрації леткого агента, а також в загальній формі вони безумовно були складними нелінійними функціями температури T , згідно так-званою Ленгмюрівської ізотерми, що враховує неідеальність газу, розміри молекул та взаємодію між ними та з кристалічною ґраткою зразка як обмежуючий фактор. Крім того швидкість протікання потоку газу через зразок приймалась адіабатичною, згідно із відомим співвідношенням:

$$v = \sqrt{\frac{C_{mP} R}{C_{mV} T}}. \quad (1.3)$$

У працях [10, 11] було зроблено ряд наближень, як спрощували математичну модуль дифузії (1.1). Зокрема умову рівноваги між дифузійним потоком та середовищем зразка спрощено подавали у такому вигляді:

$$c_{eq}(a) = \gamma a + \varepsilon a^2, \quad (1.4)$$

де $\gamma = 1 / (ba_{full})$ - це постійна величина, що характеризує дифузію у так званому наближенні кінетики Генрі. При цьому, у виразі (1.4) вдалися до розкладу за малим параметром ε , що характеризує фазовий перехід між середовищем мікропори та середовищем зразка. В результаті вираз (1.4) та вихідні рівняння математичної моделі значно спрощуються, є можливість отримати їхні аналітичні вирази шляхом застосування диференціально-інтегральних перетворень Лапласа, Хевісайда. Спрощена математична модель дифузії в такому разі набуває такого вигляду:

$$\begin{aligned} \frac{\partial c(z,t)}{\partial t} + \frac{\partial a(z,t)}{\partial t} + v \frac{\partial c(z,t)}{\partial z} &= D_{inter} \frac{\partial^2 c(z,t)}{\partial z^2}; \\ \frac{\partial a(z,t)}{\partial t} &= \beta [\gamma a(z,t) + \varepsilon a^2(z,t)]. \end{aligned} \quad (1.5)$$

Для реалізації математичної моделі, зокрема числової, приймалося, що модель (1.5) характеризується такими початковими умовами:

$$c(t, z)|_{t=0} = c_s, \quad a(t, z)|_{t=0} = a_s, \quad (1.6)$$

які характеризують початкові концентрації у різних граничних середовищах зразка. Для того щоб можна було застосовувати методи інтегральних перетворень, приймалося також, що математична модель (1.5) характеризується набором простих інтерфейсних граничних умов, значно спрощуючи спектральну задачу:

$$c(t, z)|_{z=0} = \omega_0(t) = \omega_0, \quad \left. \frac{\partial c}{\partial z} \right|_{z=\infty} = 0. \quad (1.7)$$

Розв'язки математичної моделі (1.5) з початковими умовами (1.6) та граничними умовами (1.7) шукались методом послідовних ітерацій з використанням стаціонарної теорії збурень у вигляді таких степеневих рядів – фактичного розкладу у ряд за малою величиною ε :

$$\begin{aligned} c(t, z) &= c_0(t, z) + \varepsilon c_1(t, z) + \varepsilon^2 c_2(t, z) + \dots \\ a(t, z) &= a_0(t, z) + \varepsilon a_1(t, z) + \varepsilon^2 a_2(t, z) + \dots \end{aligned} \quad (1.8)$$

де кожен з доданків у виразах (1.8) міг бути отриманий у аналітичній, але при цьому у доволі складній для програмної реалізації формульній конструкції.

Згадана математична модель, хоча є основною та базовою для опису процесів дифузії у зразках з мікропорами, про те вона має ряд суттєвих недоліків, які не дозволяють застосовувати її до опису процесів масоперенесення у багатошарових зразках. Ці недоліки зокрема такі:

- використання наближення та теорії збурень на ранніх етапах побудови математичної моделі;
- початкові умови виду (1.6) суттєво накладають обмеження на вхідні дані при програмній реалізації математичної моделі;
- збіжність рядів у виразах (1.8) важко безпосередньо встановити, хоча й безпосередні обчислення вказують на можливість досягнення необхідної точності обчислень;
- граничні умови (1.7) є дуже формальними та спрощеними, їх застосування до багатошарових структур мало б тотально не коректні результати.

Для моделювання процесів дифузії можуть бути застосованими різні програмні системи та комплекси, що дозволяють аналітично чи чисельно розв'язувати диференціальні рівняння, які описують такі процеси. Основні інструменти та програмні системи, які часто використовуються спеціалістами для цих задач є такими:

- COMSOL Multiphysics. За своїм призначенням широко застосовується для моделювання складних фізичних процесів, в тому числі і дифузії. Основою системи є окремі програмні модулі. Модулі: Має модуль "Transport of Diluted Species (Транспорт розчинів у зразках)," який використовується для моделювання дифузійних процесів. Основним методом є скінченно-елементний метод (FEM), він широко застосовується для розв'язання диференціальних рівнянь у різних конфігураціях із початковими та граничними умовами.
- ANSYS Fluent. За своїм призначенням це пакет програмних систем для математичного моделювання потоків часток, теплопередачі та дифузії.

Особливості пакету полягають у тому, що він підтримує моделювання дифузійних процесів у рідинах і газах довільної етимології. І методів, які широко застосовуються слід виділити скінченно-об'ємний метод (FVM), що дозволяє досить наочно моделювати конвекційно-дифузійні процеси у різноматніних композитних функціональних матеріалах.

- **MATLAB** (з пакетом розширення PDE Toolbox) За своїм призначенням цей програмний пакет часто застосовується для числового розв'язання диференціальних рівнянь у частинних похідних, в тому числі рівняння дифузії. З якісних особливостей слід виділити те, що MATLAB PDE Toolbox дозволяє створювати інтерактивні 2D і 3D моделі для дифузійних процесів і обчислювати їх основні характеристики у числовому вигляді. Із основних методів, які використовуються для числового розв'язання математичних моделей, то основою MATLAB є метод скінченних елементів (FEM) та доволі широкий набір інших чисельних методів.
- **OpenFOAM**. Система представляє собою відкриту платформу із вільно поширюваним кодом для числового моделювання, в тому числі із процесами масопереносу. Особливості: системи може налаштовуватись під розв'язання різних типів задач дифузії та конвекції. Як і згадані вище програмні числові системи OpenFOAM використовує скінченно-об'ємний метод (FVM) і широко застосовується в дослідженнях з тепломасообміну та є однією з найбільш активно застосовуваних експериментаторами систем в плані адаптації експериментально отриманих результатів.
- **Lattice Boltzmann Method (LBM)** За своєю специфікацією програмна система представляє блок спеціалізованих програмних пакетів, що застосовують метод решітки Больцмана, відомий як так-званий *kr*-метод. Так програмні пакети як Palabos чи LBMMethod можуть безпосередньо застосовуватись для моделювання дифузійних процесів. Загалом дана програмна система добре підходить для моделювання складних граничних умов і мікроструктур, таких як пористі матеріали, зокрема цеоліти. Оскільки даний програмний пакет базується на кінетичних рівняннях Больцмана, то реалізовані в ньому

математичні моделі дозволяють більш ефективно моделювати процеси у мікроструктурах з різноманітною геометричною симетрією.

- Програмна система FEniCS. Платформа має призначення для програмної реалізації числового розв'язання рівнянь у частинних похідних. Підходить особливо для дослідників, що мають потребу в високій гнучкості коду при побудові математичних моделей. Використовує метод скінченних елементів (FEM) для розв'язання задач дифузії у мікропросторових масштабах. Програмні компоненти системи дозволяють ефективно реалізовувати як прості 1D моделі, так і складні 3D моделювання процесів дифузії, включаючи облік конвекції, хімічних реакцій, неоднорідностей та інших ускладнень.

Окремо зупинимось на програмній системі Wolfram Mathematica. Wolfram Mathematica є потужним інструментом для моделювання дифузії, особливо завдяки своїм можливостям аналітичного й чисельного розв'язання диференціальних рівнянь. Основні можливості для моделювання процесів дифузії в Wolfram Mathematica такі:

- розв'язання рівняння дифузії - Wolfram Mathematica може аналітично розв'язувати рівняння теплопровідності, яке за структурою аналогічне до рівняння дифузії, для простих випадків та складних випадків, зумовлених граничними умовами. Це дозволяє знайти загальні форми розв'язків для заданих наборів початкових і граничних умов.
- для більш складних випадків є доступними чисельні методи розв'язання з використанням процедури NDSolve. Розширені можливості для чисельного моделювання NDSolve — потужний оператор, що дозволяє розв'язувати крайові задачі для рівнянь у частинних похідних, для прикладу рівняння дифузії. Він підтримує різноманітні чисельні методи, зокрема метод скінченних різниць (FDM) і метод скінченних елементів (FEM).
- Загалом Wolfram Mathematica дозволяє задавати складні початкові та граничні умови, що робить її дуже гнучкою для математичного моделювання

процесів дифузії у мікросистемах зі складною геометрією або неоднорідними матеріалами.

- 3D-моделювання та візуалізація Mathematica може візуалізувати результати моделювання як у 2D, так і в 3D, що корисно для розуміння просторового розподілу концентрацій чи тепла в процесах дифузії. Можливість анімації (модуль Animate) результатів дозволяє створювати часові еволюції процесу дифузії, що може бути дуже інформативним для виконання аналізу динаміки процесу.
- Використання параметричних та символьних розрахунків у Wolfram Mathematica дозволяє проводити моделювання з використанням широких діапазонів вхідних параметрів, а не просто взятих конкретних значень, що зручно для розв'язання оптимізаційних задач або ж при дослідженні впливу різних параметрів на процес дифузії.
- Завдяки можливості легко працювати з системами диференціальних рівнянь, у Wolfram Mathematica можлива ефективна симуляція реакційно-дифузійних систем, зокрема можна моделювати процеси, де дифузія супроводжується хімічними реакціями, що робить її корисною для біологічних та хімічних застосувань.

На Рис. 1.2. подано приклад застосування програмної системи Wolfram Mathematica для числового розв'язання одновимірного рівняння дифузії з однорідними крайовими умовами. В даному прикладі ми чисельно реалізуємо математичну модель процесу дифузії за допомогою модуля NDSolve. Безпосередньо поданий далі код розв'язує 1D-дифузійне рівняння на нормованому відрізку $[0,1] \times [0,1]$ з нульовими граничними умовами і початковим профілем температурним профілем системи: $u(x, 0) = \sin \pi x$.

Слід зауважити, що Wolfram Mathematica містить зручні та ефективні інструменти для моделювання та аналізу дифузійних процесів, особливо для тих випадків, коли аналітичні розв'язки не можливо отримати й потрібно застосувати ефективні чисельні методи.

```

= solution = NDSolve[ {
    |чисельні розв'язки диференціальних рівнянь
    D[u[x, t], t] == D[u[x, t], {x, 2}],
    |диференціювати |диференціювати
    u[0, t] == 0,
    u[1, t] == 0,
    u[x, 0] == Sin[Pi x]], u, {x, 0, 1}, {t, 0, 1}];
    |си... |число пі

(*Анімація еволюції профілю u(x,t) у часі*)
Animate[Plot[Evaluate[u[x, t] /. solution], {x, 0, 1}, PlotRange -> {{0, 1}, {-1, 1}},
|анімувати |гра... |обчислити |діапазон значень на графіку
    PlotLabel -> StringTemplate["Time: `time` seconds"] [ <| "time" -> t |> ],
    |позначка гр... |рядковий шаблон
    AxesLabel -> {"x", "u(x, t)"}, {t, 0, 1} ]
    |позначення на осях

```

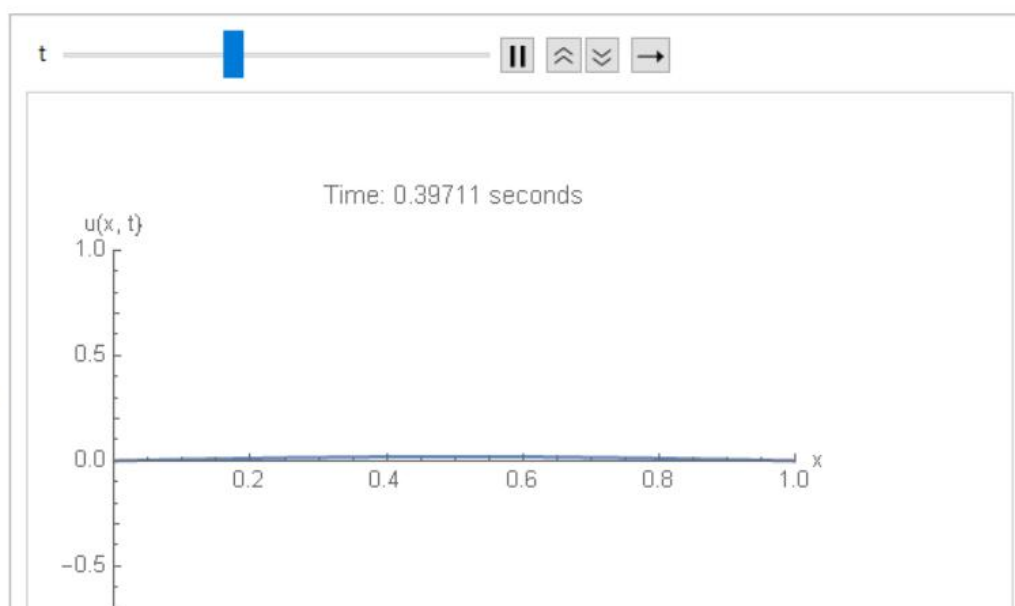


Рис. 1.2. Приклад реалізації числових розв'язків одомірного рівняння дифузії у Wolfram Mathematica з використанням модуля Animate

Підсумовуючи, слід зробити висновок, що для успішного розв'язання проблеми із побудови математичної моделі масопереносу у багатошарових зразках слід значною мірою відкоригувати та покращити вже існуючу математичну модель. Для цього слід виконати наступні речі:

- увести більш загальні початкові умови для концентрації у зразку та його мікропорах ;

- слід повністю, в іншому підході підійти до виводу граничних умов на межах шарів зразка, що відноситься до різного типу матеріалів з метою не більш реалістичного їх опису ;
- для отриманої математичної моделі слід провести її заміну ефективною різницевою схемою, яка не найбільш оптимально підходить для розробки програмної системи в тому числі на основі розподілених та розпаралелених обчислень;
- для програмної системи, реалізуючи математичну модель необхідно чітко продумати її архітектуру та інтерфейс взаємодії з користувачем.

Згадані проблеми далі послідовно вирішуються у даній дипломній роботі.

1.3. Актуальне програмне забезпечення, яке застосовується до ефективного розв'язування сіткових задач. Сучасний стан проблеми.

Для опису сучасного стану програмного забезпечення, яке застосовується для розв'язання сіткових задач, можна висвітлити кілька ключових аспектів: види сіткових задач, найпопулярніші програмні інструменти, які сьогодні використовують, та актуальні тенденції й виклики в цій галузі. Сіткові задачі застосовуються для чисельного розв'язання диференціальних рівнянь, що описують процеси переносу, теплопередачі, механіки рідин, еластичності, електромагнітних полів тощо. Вони є актуальні у фізиці, інженерії, біології, економіці та інших науках, де важливо моделювати поведінку складних систем за допомогою спеціалізованого програмного забезпечення. Для розв'язання таких задач широко застосовуються методи дискретизації, зокрема, метод кінцевих елементів (FEM), метод кінцевих різниць (FDM), метод кінцевих об'ємів (FVM) та метод решітки Больцмана (LBM). Кожен із цих методів має свої особливості й сфери застосування у предметній області, але всі вони потребують застосування або спеціалізованої розробки потужного програмного забезпечення, здатного працювати з великими

обсягами даних та високою обчислювальною ефективністю, зокрема в останній час зростає необхідність реалізації парадигм паралельного програмування, розподілених обчислень.

З ростом обсягів обчислень, зумовленим складністю розвинених математичних моделей і кількістю точок нерівномірної сітки, зростає потреба в обчисленнях на кластерах та графічних процесорах (GPU). Більшість сучасних програмних систем, таких як ANSYS, OpenFOAM і FEniCS, мають вбудовану підтримку паралельних чи розподілених потокових обчислень, що дозволяє ефективно використовувати ресурси цих процесорів для прискорення реалізації кінцевих числових симуляцій. Зокрема, інтеграція обчислень на GPU дозволяє значно підвищити продуктивність і швидкість роботи з числовими математичними моделями.

Адаптивна сітка та автоматична оптимізація дискретної сітки дозволяє автоматично змінювати густоту сітки залежно від характеру моделювання (наприклад, в областях з великою зміною градієнта концентрації або швидкості ламінарного потоку). Програмні системи COMSOL та ANSYS або ж Wolfram Mathematica мають засоби імплементації адаптивної сітки, що підвищує точність обчислень і знижує втрати потужності на обчислення за рахунок зменшення обчислювальної складності. Це є важливий напрямок у розвитку спеціалізованого програмного забезпечення для сіткових задач.

Приклад програмної реалізації сіткового методу розв'язання крайової задачі для математичної моделі дифузії подано далі безпосередньо методом NSolve та з використанням алгоритму прямої прогонки подано далі.

метод NSolve:

```
D = 0.1;
solution = NDSolve[
  {
    D[u[x, t], t] == D D[u[x, t], {x, 2}],
    u[0, t] == 0,
    u[1, t] == 0,
    u[x, 0] == Sin[Pi x]
  },
  u, {x, 0, 1}, {t, 0, 0.1}
];

Plot3D[Evaluate[u[x, t] /. solution], {x, 0, 1}, {t, 0, 0.1},
  PlotRange -> All, AxesLabel -> {"x", "t", "u(x, t)"},
  PlotLabel -> "Розв'язок рівняння дифузії"]
```

числовий метод прямої прогонки:

```
(* Параметри *)
D = 0.1;
dx = 0.01;
dt = 0.0005;
nx = 101;
nt = 1000;
alpha = D dt / dx^2;

(* Початкові умови *)
u = Table[Sin[Pi x], {x, 0, 1, dx}];
uNew = ConstantArray[0, nx];

(* Основний цикл *)
Do[
  Do[
    uNew[[i]] = u[[i]] + alpha (u[[i + 1]] - 2 u[[i]] + u[[i - 1]]),
    {i, 2, nx - 1}
  ];
  u = uNew;
  ,
  {n, nt}
];

(* Виведення результатів *)
ListLinePlot[u, DataRange -> {0, 1}, PlotRange -> All,
  AxesLabel -> {"x", "u"}, PlotLabel -> "Концентрація після nt кроків"]
```

Рис. 1.3. Реалізація числового розв'язку математичної моделі дифузії у Wolfram Mathematica за допомогою методу NSolve та прямого числового методу прогонки

В останнє десятиліття активно досліджується та впроваджується можливість застосування методів машинного навчання для прискорення і оптимізації розв'язання сіткових задач. Слід зауважити, що нейронні мережі застосовуються для інтерполяції дискретних полів, прогнозування динаміки поведінки систем і вибору оптимальних параметрів дискретної просторової сітки. Такі технології, як Physics-Informed Neural Networks (PINNs), є системами інтегрують фізичні закони в архітектуру нейронних мереж, також є перспективним напрямком, оскільки можуть суттєво знижувати вимоги до обчислювальних ресурсів.

Багато сучасних програмних систем для сіткових задач, таких як COMSOL, мають зручні інтерфейси, що дозволяють легко будувати, налаштовувати та візуалізувати кінцеві математичні моделі. Водночас такі програмні системи, як MATLAB і FEniCS, дозволяють реалізовувати самі численні процеси через блочне і об'єктно-орієнтоване програмування, що важливо для довготривалих і комплексних досліджень.

Використання програмних системі комплексів із відкритим кодом, таких як OpenFOAM і FEniCS, стає доволі популярним через їх доступність, можливість модифікації та забезпечення прозорості й реплікабельності досліджень, а також вільне поширення пакетів і кінцевих результатів. Це дозволяє вченим і інженерам по всьому світу використовувати і розвивати сучасні інструменти для сіткових задач. Отже, сучасне програмне забезпечення для реалізації сіткових задач розвивається в напрямку збільшення продуктивності, гнучкості та інтеграції з новітніми технологіями, такими як штучний інтелект та наука про дані. Високопродуктивні обчислення та адаптивні сітки залишаються критичними проблемами для забезпечення точності й ефективності при розв'язуванні складних наукових задач. Водночас, використання відкритого програмного забезпечення стимулює прозорість і співпрацю, відкриваючи нові можливості для розвитку інструментів і методів у галузі сіткових задач.

РОЗДІЛ 2. РОЗРОБКА ТА ВЕРИФІКАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ ДИФУЗІЇ У БАГАТОШАРОВИХ ЗРАЗКАХ

2.1 Математична модель дифузії. Рівняння дифузії та граничні вимоги на межах досліджувальних зразків.

Досліджуватимемо кінетику дифузійних процесів у сферичному багат шаровому зразку цеоліту, поперечний переріз якого наведено на Рис. 2.1. Точкою 0 позначено центр зразка, а координатами r_0, r_1, r_2, r_3, r_4 – межі середовищ з різною пористістю.

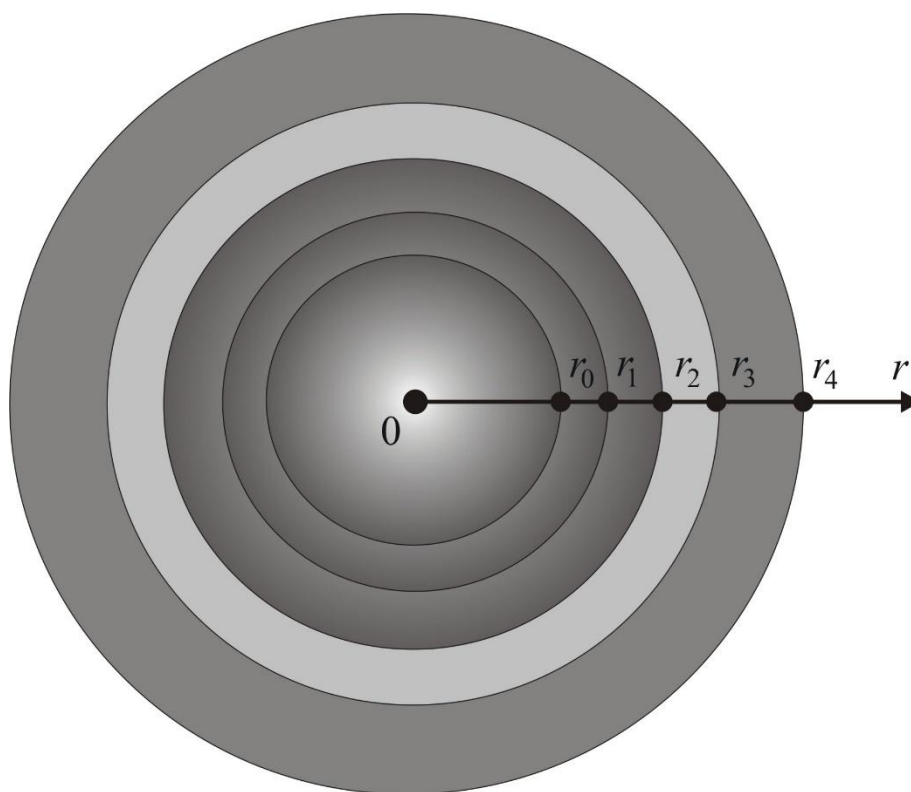


Рис. 2.1. Поперечний переріз зразка багат шарового цеоліту сферичної форми.

З фізичної точки зору процес дифузії в зразку складний і полягає не тільки в проникненні дифундуючого агента в середовище цеоліту. По суті, цей процес складається з двох частин: дифузії в міжкристалітному середовищі зразка та дифузії в порах цього зразка. Відповідно залежні від часу концентрації розсіяної речовини

в міжкристалітному середовищі і в порах позначимо через $c(r,t)$ і $a(r,X,t)$, де $0 \leq X \leq \bar{X}$ і передбачаємо, що пори в цеоліті мають сферичну форму і середній радіус $\bar{X}$. Також будемо враховувати, що дифузія відбувається для суміші з i компонентів, які не взаємодіють між собою. Отже, така дифузія відбувається як всередині об'єму, так $4\pi r_4^3/3$ і об'єму $4\pi \bar{X}^3/3$ найбільш повно описується системою самоузгоджених рівнянь [11, 13]:

$$\frac{\partial C_i(r,t)}{\partial t} = \frac{D_i^{(c)}}{r_4^2} \frac{\partial^2 C_i(r,t)}{\partial r^2} - e_i \frac{D_i^{(p)}}{\bar{X}^2} \frac{\partial a_j(r,X,t)}{\partial X}, \quad (2.1)$$

$$\frac{\partial a_i(r,X,t)}{\partial t} = \frac{D_i^{(p)}}{R^2} \left[\frac{\partial^2 a_i(r,X,t)}{\partial X^2} + \frac{2}{X} \frac{\partial a_i(r,X,t)}{\partial X} \right], \quad (2.2)$$

де $D_i^{(c)}$ і $D_i^{(p)}$ - це коефіцієнти дифузії молекул i -го типу всередині матеріалу цеоліту та середовища пор відповідно. Величина e_i характеризує поправку на те, що пори в цеоліті виконують роль активних центрів адсорбції. Він визначається наступним чином:

$$e_i = \varepsilon_{\text{por}} / (1 - \varepsilon_{\text{por}}) K_i^H, \quad (2.3)$$

де безрозмірна величина ε_{por} визначає різну в кожному шарі зразка пористість цеолітового середовища K_i^H – постійна Генрі.

Приймається до уваги, що для концентрацій поза і всередині пор виконуються такі початкові умови:

$$c_i(r,t)|_{t=0} = 0; \quad a_i(r,X,t)|_{t=0} = 0, \quad X \in (0, \bar{X}), \quad i = \overline{1, n}. \quad (2.4)$$

Загалом цеолітна система з порами характеризується трьома типами граничних умов. Перша умова характеризує постійність концентрації на межі пор:

$$\left. \frac{\partial a_i(r,X,t)}{\partial X} \right|_{X=\bar{X}} = 0. \quad (2.5)$$

Друга умова задає адсорбційну рівновагу за аналогією з умовою Ленгмюра [13]:

$$a_i(r, X, t) \Big|_{X=\bar{X}} = \frac{K_i c_i(r, t)}{1 + K_1 c_1(r, t) + K_2 c_2(r, t) + \dots + K_n c_n(r, t)}. \quad (2.6)$$

Третя група граничних умов описує рівновагу концентрацій і потоків частинок всередині шарів зразка цеоліту. Ці граничні умови мають такий вигляд:

$$\begin{cases} c_i^{(j)}(r, t) \Big|_{r=r_j} = c_i^{(j+1)}(r, t) \Big|_{r=r_j} ; j = 0, 1, 2, 3 \\ D_i^{(j)} \frac{\partial c_i^{(j)}(r, t)}{\partial r} \Big|_{r=r_j} = D_i^{(j+1)} \frac{\partial c_i^{(j+1)}(r, t)}{\partial r} \Big|_{r=r_j} \end{cases} \quad (2.7)$$

Методологія розв'язання системи рівнянь (2.1) і (2.2) з граничними умовами (2.4)-(2.7) базується на тому, що адсорбцію молекул порами цеоліту можна розглядати як фазовий перехід [4, 11, 12]. Це дозволяє нам виконати функціональну декомпозицію шуканого розв'язку:

$$\varphi_i(c_1, c_2, \dots, c_n) = K_i c_i(r, t) / \left[1 + \sum_{i=1}^n K_i c_i(r, t) \right], \quad i = \overline{1, n}. \quad (2.8)$$

що відповідає граничній умові (8) підряд в околі точок набуття нульової концентрації компонентів, що беруть участь у процесі адсорбції. Наприклад, у випадку адсорбції трикомпонентної суміші з точністю до значень другого порядку розклад в ряд Маклорена дає такий вираз:

$$\varphi_i^0(C_1, C_2, C_3) = \varphi_i^0 + \left(\frac{\partial \varphi_i^0}{\partial C_1} C_1 + \frac{\partial \varphi_i^0}{\partial C_2} C_2 + \frac{\partial \varphi_i^0}{\partial C_3} C_3 \right)$$

$$\begin{aligned}
& + \frac{1}{2} \left(\frac{\partial^2 \phi_i^0}{\partial C_1^2} C_1^2 + \frac{\partial^2 \phi_i^0}{\partial C_2^2} C_2^2 + \frac{\partial^2 \phi_i^0}{\partial C_3^2} C_3^2 \right) \\
& + \left(\frac{\partial^2 \phi_i^0}{\partial C_1 \partial C_2} C_1 C_2 + \frac{\partial^2 \phi_i^0}{\partial C_1 \partial C_3} C_1 C_3 + \frac{\partial^2 \phi_i^0}{\partial C_2 \partial C_3} C_2 C_3 \right) + \dots; \\
& a_1(r, X, t) \Big|_{X=\bar{X}} = K_1 (C_1 - K_1 C_1^2 - K_2 C_1 C_2 - K_3 C_1 C_3); \\
& a_2(r, X, t) \Big|_{X=\bar{X}} = K_2 (C_2 - K_2 C_2^2 - K_1 C_1 C_2 - K_3 C_2 C_3); \\
& a_3(r, X, t) \Big|_{X=\bar{X}} = K_3 (C_3 - K_3 C_3^2 - K_1 C_1 C_3 - K_2 C_2 C_3).
\end{aligned} \tag{2.9}$$

Далі скористаємося фактом наявності в розкладах (2.9) малого параметра, який визначається наступним чином:

$$\varepsilon = K^2 \ll 1; \quad K = \max \{ K_i, \quad K_i < 1 \}_{i=1}^n, \quad i = \overline{1, n}. \tag{2.10}$$

Це дозволяє шукати розв'язки рівнянь (2.1), (2.2) у вигляді асимптотичних розкладів:

$$\begin{aligned}
c_i(r, t) &= C_{i0}(r, t) + \varepsilon C_{i1}(r, t) + \varepsilon^2 C_{i2}(r, t) + \dots, \quad i = \overline{1, 3} \\
a_j(r, X, t) &= a_{j0}(r, X, t) + \varepsilon a_{j1}(r, X, t) + \varepsilon^2 a_{j2}(r, X, t) + \dots
\end{aligned} \tag{2.11}$$

причому точністю таких обчислень і збіжністю ряду (2.11) зручно керувати такими умовами:

$$\begin{aligned}
\nu_c^{(m)} &= \ln \left| \frac{c^{(m)}(r, t) - c^{(m-1)}(r, t)}{c^{(m)}(r, t)} \right|; \\
\nu_a^{(m)} &= \ln \left| \frac{a^{(m)}(r, \bar{X}, t) - a^{(m-1)}(r, \bar{X}, t)}{a^{(m)}(r, \bar{X}, t)} \right|,
\end{aligned} \tag{2.12}$$

де m – порядок наближення.

Отже, для довільного порядку апроксимації m слід знайти розв'язок такої системи диференціальних рівнянь:

$$\begin{aligned}
\frac{\partial c_{i_m}(r,t)}{\partial t} &= \frac{D_i^{(c)}}{r_4^2} \frac{\partial^2 c_{i_m}(r,t)}{\partial r^2} - e_i \frac{D_i^{(p)}}{\bar{X}^2} \left(\frac{\partial N_{i_m}(r,X,t)}{\partial X} - N_{i_m}(r,X,t) \right); \\
\frac{\partial N_{i_m}(r,X,t)}{\partial t} &= \frac{D_i^{(p)}}{\bar{X}^2} \frac{\partial^2 N_{i_m}(r,X,t)}{\partial X^2}; \\
N_{i_m}(r,X,t) &= X a_{i_m}(r,X,t).
\end{aligned} \tag{2.13}$$

з початковими умовами, гомологічними до умов (2.4), (2.5):

$$c_{i_m}(r,t)\big|_{t=0} = 0; \quad a_{i_m}(r,X,t)\big|_{t=0} = 0, \quad X \in (0, \bar{X}), \quad i = \overline{1, n}. \tag{2.14}$$

і граничними умовами, що аналогічні умовам (2.6) і (2.7):

$$\begin{cases}
N_{i_m}(r,X,t)\big|_{X=0} = 0; \\
N_{i_m}(r,X,t)\big|_{X=\bar{X}} = K_i c_{i_m}(r,t) - F_{i_m}(r,t); \\
c_{i_m}^{(j)}(r,t)\big|_{r=r_j} = c_{i_m}^{(j+1)}(r,t)\big|_{r=r_j}; \\
D_{i_m}^{(j)} \frac{\partial c_{i_m}^{(j)}(r,t)}{\partial r}\bigg|_{r=r_j} = D_{i_m}^{(j+1)} \frac{\partial c_{i_m}^{(j+1)}(r,t)}{\partial r}\bigg|_{r=r_j}.
\end{cases} \tag{2.15}$$

$$\text{де } F_{i_m}(t, Z) = \sum_{s=0}^{m-1} \sum_{k=1}^n \frac{K_i K_k}{K_1^2} c_{i_s}(r, r) c_{k_{m-1-s}}(r, t).$$

Знаходження розв'язку зв'язаної системи рівнянь (2.13) базується на застосуванні операційного методу Хевісайда [14]. Якщо при цьому врахувати, що шукані функції є оригіналами Лапласа, то кожна із задач (2.13) еквівалентна наступному рівнянню:

$$\begin{aligned}
\frac{\partial^2 c_{i_m}^*}{\partial r^2} - \frac{r_4^2}{D_i^{(c)}} p c_{i_m}^* - \frac{3}{e_i} \frac{r_4^2}{\bar{X}^2} \frac{D_i^{(c)}}{D_i^{(p)}} \left(\frac{\partial N_{i_m}^*}{\partial X} - N_{i_m}^* \right) &= 0; \\
\frac{\partial^2 N_{i_m}^*}{\partial X^2} - \frac{\bar{X}^2}{D_i^{(i)}} p N_{i_m}^* &= 0.
\end{aligned} \tag{2.16}$$

де в рівняннях (2.16):

$$\begin{aligned}
c_{i_m}^*(r, p) &= \int_0^\infty c_{i_m}(r, t) e^{-pt} dt; \\
N_{i_m}^*(r, X, p) &= \int_0^\infty N_{i_m}^*(r, X, t) e^{-pt} dt.
\end{aligned} \tag{2.17}$$

Після спрощень граничні умови набувають такого вигляду:

$$\begin{cases}
N_{i_m}^*(r, X, p)|_{X=0} = 0; \\
N_{i_m}^*(r, X, p)|_{X=\bar{X}} = K_i c_{i_m}^*(r, p) - F_{i_m}^*(r, p); \\
c_{i_m}^{(j)}(r, p)|_{r=r_j} = c_{i_m}^{(j+1)}(r, p)|_{r=r_j}; \\
\left. \frac{\partial c_{i_m}^{(j)*}(r, p)}{\partial r} \right|_{r=r_j} - \frac{D_{i_m}^{(j+1)}}{D_{i_m}^{(j)}} \left. \frac{\partial c_{i_m}^{(j+1)*}(r, p)}{\partial r} \right|_{r=r_j} = 0.
\end{cases} \tag{2.18}$$

Тепер із рівняння (2.16) отримуємо:

$$N_{i_m}^*(r, X, p) = K_i c_{i_m}^*(r, p) \operatorname{sh} \left(\bar{X} \sqrt{\frac{p}{D_i^{(c)}}} X \right) / \operatorname{sh} \left(\bar{X} \sqrt{\frac{p}{D_i^{(c)}}} \right), \tag{2.19}$$

що дає наступне рівняння:

$$\frac{\partial^2 c_{i_m}^*(r, p)}{\partial r^2} - \gamma_i^2(p) c_{i_m}^*(r, p) = 0, \tag{2.20}$$

де

$$\begin{aligned}
\gamma_i^2(p) &= \Gamma_i K_i \left(\frac{e_i}{3K_i} \frac{r_4^2}{D_i^{(c)}} p + R \sqrt{\frac{p}{D_i^{(c)}}} \operatorname{cth} \left(R \sqrt{\frac{p}{D_i^{(c)}}} \right) - 1 \right); \\
\Gamma_i &= \frac{3}{e_i} \left(\frac{r_4}{\bar{X}} \right)^2 \frac{D_i^{(c)}}{D_i^{(p)}}.
\end{aligned} \tag{2.21}$$

Тепер розв'язки рівняння (2.20) набувають такого вигляду:

$$c_{i_m}^*(r, p) = c_i^{in} \frac{1}{p} \frac{\operatorname{ch}[\gamma_i(p)r]}{\operatorname{ch}[\gamma_i(p)]} = c_i^{in} \frac{1}{p} \frac{\cos[\gamma_i(p)r]}{\cos[\gamma_i(p)]}, \tag{2.22}$$

де ми застосували перші дві граничні умови (2.18).

Щоб здійснити зворотний перехід від зображення Лапласа до оригіналів, ми скористаємося теоремою Хевісайда про розкладання раціонального комплексного виразу в збіжний ряд за коренями його знаменника [14]. Вносячи зміни до змінної:

$$p = -\frac{D_i^{(c)} \beta_i^2}{\bar{X}^2} \quad (2.23)$$

де β_i – корені трансцендентного дисперсійного рівняння (випливаючи з двох останніх граничних умов (2.18):

$$\beta_i \operatorname{ctg}(\beta_i) - \frac{e_i}{3K_i} (\beta_i)^2 = 1 - \frac{1}{\Gamma_i K_i} \left(\frac{2k-1}{2} \pi \right)^2, \quad k = \overline{1, \infty}, \quad (2.24)$$

отримуємо вираз для зворотного переходу від зображення Лапласа до вихідного:

$$c_{i_m}(r, t) = c_i^{in} \left(1 + \sum_{s=1}^{\infty} \sum_{k=1}^{\infty} \frac{\operatorname{ch}[\gamma_j(p)r] \exp(p_k t)}{p_{ks} \frac{d}{dp} \operatorname{ch}[\gamma_j(p)] \Big|_{p=p_{ks} = -\frac{D_i^{(c)} \beta_i^2}{\bar{X}^2}}} \right). \quad (2.25)$$

Після громіздких, але простих перетворень ми отримуємо:

$$c_{i_m}(r, t) = c_i^{in} \left(1 + 2 \left(\frac{\bar{X}}{r_4} \right)^2 \frac{D_i^{(p)}}{D_i^{(c)}} \times \sum_{s=1}^{\infty} \sum_{k=1}^{\infty} \frac{(2k-1)\pi \cos\left(\frac{2k-1}{2}\pi r\right) \exp\left(-\frac{D_i^{(c)}}{\bar{X}}(\beta_s)^2 t\right)}{(-1)^s (\beta_s)^2 \left[\frac{3K_i}{e_i} \left(\frac{1}{\sin^2(\beta_s)} - \frac{\operatorname{ctg}(\beta_s)}{\beta_s} \right) + 2 \right]} \right). \quad (2.26)$$

Далі перетворюємо вираз (2.19) до аналітичного вигляду наступним чином:

$$N_{i_m}^*(r, X, p) = c_{i_m}^*(r, p) = \frac{c_i^{in}}{p} \frac{\operatorname{ch}[\gamma_j(p)r]}{\operatorname{ch}[\gamma_j(p)]}, \quad (2.27)$$

$$i\beta = \bar{X} \sqrt{\frac{p}{D_i^{(c)}}}.$$

Знову застосовуючи теорему Хевісайда, ми отримаємо вираз для оберненого перетворення для функції $N_{i_m}(r, X, t)$:

$$N_{i_m}(r, X, t) = 1 + \sum_{n=1}^{\infty} \sum_{k=1}^{\infty} \frac{\sin(\beta_i X) \operatorname{ch}[\gamma_{s_1}(p_{kn})r] \exp(p_{kn_1} t)}{p_{kn_1} \sin(\beta_i) \frac{d}{dp} \operatorname{ch}[\gamma_{s_1}(p)r_4] \Big|_{p=-\frac{D_i^{(c)} \beta_i^2}{R^2}}}. \quad (2.28)$$

В результаті перетворень співвідношення, яке згідно (2.28) дає вираз, що описує концентрацію речовини всередині пор зразка цеоліту, визначається наступним чином:

$$a_{i_m}(r, r) = c_i^{in} \left(1 + 2 \left(\frac{R}{l} \right)^2 \frac{D_i^{(p)}}{D_i^{(c)}} \times \sum_{s=1}^{\infty} \sum_{k=1}^{\infty} \frac{(2k-1)\pi \sin(\beta_i X) \cos\left(\frac{2k-1}{2}\pi r\right) \exp\left(-\frac{D_i^{(c)}}{r_4^2}(\beta_i)^2 t\right)}{(-1)^n (\beta_i)^2 X \sin(\beta_i) \left[\frac{3K_i}{e_i} \left(\frac{1}{\sin^2(\beta_i)} - \frac{\operatorname{ctg}(\beta_{ks}^j)}{\beta_{ks}^j} \right) + 2 \right]} \right). \quad (2.29)$$

Отримані розв'язки практично не можливо застосувати до багат шарових зразків, оскільки граничні умови (2.7) на межах кожного шару приведуть до дисперсійного рівняння, яке розв'язується лише на кластері або з використанням хмарних обчислень. Проте ми будемо використовувати результати аналітичного методу при порівнянні результатів, що їх дає метод на основі сіткових схем для моношарових зразків.

2.2 Апроксимація математичної моделі дифузії різницевою схемою, лінеаризація вихідних рівнянь та граничних умов.

Виконаємо заміну математичної моделі дифузії, яка задається рівняннями, такого виду:

$$\begin{aligned}\frac{\partial c(z,t)}{\partial t} + \frac{\partial a(z,t)}{\partial t} + v \frac{\partial c(z,t)}{\partial z} &= D_{inter} \frac{\partial^2 c(z,t)}{\partial z^2}; \\ \frac{\partial a(z,t)}{\partial t} &= \beta [\gamma a(z,t) + \varepsilon a^2(z,t)].\end{aligned}\quad (2.30)$$

Для цього замінімо область визначення системи рівнянь (2.30). дискретною областю – двовимірною сіткою за змінними z та t . В результаті дискретизації областю визначення за кожною змінною отримується сітка із вузлами, що задаються наступним чином:

$$\Omega(T, E) \subset R^2; \Omega = \{ T_0 Z ; T_i = (0, t_{max}); \quad (2.31)$$

$$Z_s = (0, Z_{max}); \Delta t = i \Delta t; \Delta z = j \Delta z \}; i \in Z^+$$

Далі нам треба отримати скінчено-різницеву формулу системи рівнянь (2.30), що задають досліджувану математичну модель. Для цього скористаємося апроксимацією першою та другою похідних у вигляді із апроксимованою(правою) першою похідною та триточковою схемою другою похідної. Для довільної функції апроксимація має такий вигляд:

$$\frac{\partial \varphi(z,t)}{\partial z} \approx \frac{\varphi_{i+1} - \varphi_i}{\Delta z}; \frac{\partial^2 \varphi(z,t)}{\partial z^2} \approx \frac{\varphi_{i-1} - 2\varphi_i + \varphi_{i+1}}{\Delta z^2} \quad (2.32)$$

В результаті математична модель (2.30) дискретизується й замінюється різницевою схемою для вихідних рівнянь у частинних похідних:

$$\begin{aligned}\frac{c_{i+1,j,m} - c_{i,j,m}}{\Delta t} + \frac{a_{i+1,j,m} - a_{i,j,m}}{\Delta t} + v \frac{c_{i,j+1,m} - c_{i,j,m}}{\Delta z} &= D_{inter} \frac{c_{i,j-1,m} - 2c_{i,j,m} + c_{i,j+1,m}}{\Delta z^2}; \\ \frac{a_{i+1,j,m} - a_{i,j,m}}{\Delta t} &= \beta (\gamma a_{i,j,m} + \varepsilon a_{i,j,m}^2).\end{aligned}\quad (2.33)$$

Далі також апроксимуються граничні умови (2.5)-(2.7):

$$\begin{aligned}
a_{i,j,0} &= 0; \\
a_{i,0,m} &= K_i c_{i,0,m} (1 - c_{i,0,m}); \\
c_{i,j,m} &= c_{i,j-1,m}; \quad j = 0, 1, 2, 3 \\
D_i^{(j)} \frac{c_{i,j,m} - c_{i,j-1,m}}{\Delta r} &= D_i^{(j+1)} \frac{c_{i,j+1,m} - c_{i,j,m}}{\Delta r}.
\end{aligned} \tag{2.34}$$

В результаті остаточно отримується різницева схема, що поєднує рівняння (2.33) і (2.34):

$$\left\{ \begin{aligned}
&\frac{c_{i+1,j,m} - c_{i,j,m}}{\Delta t} + \frac{a_{i+1,j,m} - a_{i,j,m}}{\Delta t} + v \frac{c_{i,j+1,m} - c_{i,j,m}}{\Delta z} - D_{inter} \frac{c_{i,j-1,m} - 2c_{i,j,m} + c_{i,j+1,m}}{\Delta z^2} = 0; \\
&\frac{a_{i+1,j,m} - a_{i,j,m}}{\Delta t} - \beta (\gamma a_{i,j,m} + \varepsilon a_{i,j,m}^2) = 0; \\
&a_{i,j,0} = 0; \\
&a_{i,0,m} - K_i c_{i,0,m} (1 - c_{i,0,m}) = 0; \\
&c_{i,j,m} - c_{i,j-1,m} = 0; \\
&D_i^{(j)} c_{i,j-1,m} - (D_i^{(j)} + D_i^{(j+1)}) c_{i,j,m} + D_i^{(j+1)} c_{i,j+1,m} = 0.
\end{aligned} \right. \tag{2.35}$$

$j = 0, 1, 2, 3$

За різницевою схемою (2.35) проводилась алгоритмізація математичної моделі та її реалізація як різницевої схеми методами прямої та зворотної прогонки.

Для сферичних зразків необхідно побудувати різницеву схем, виходячи із рівнянь (2.1), (2.2) та граничних умов (2.5)-(2.7). Використовуючи апроксимацію (2.32) ми будемо мати:

$$\begin{aligned}
\frac{C_{i+1,j,k} - C_{i,j,k}}{\Delta t} &= \frac{D_i^{(c)}}{r_4^2} \frac{C_{i-1,j,k} - 2C_{i,j,k} + C_{i+1,j,k}}{\Delta r^2} - e_i \frac{D_i^{(p)}}{\bar{X}^2} \frac{a_{i,j+1,k} - a_{i,j,k}}{\Delta x} \\
\frac{a_{i+1,j,k} - a_{i,j,k}}{\Delta t} &= \frac{D_i^{(p)}}{R^2} \left[\frac{a_{i,j-1,k} - 2a_{i,j,k} + a_{i,j+1,k}}{\Delta x^2} + \frac{2}{\bar{X}} \frac{a_{i,j+1,k} - a_{i,j,k}}{\Delta x} \right],
\end{aligned} \tag{2.36}$$

Тепер, разом із апроксимованими граничними умовами (2.5)-(2.7) отримується сіткова різницева схема, як має такий вигляд:

$$\begin{aligned}
& \frac{C_{i+1,j,k} - C_{i,j,k}}{\Delta t} - \frac{D_i^{(c)}}{r_4^2} \frac{C_{i-1,j,k} - 2C_{i,j,k} + C_{i+1,j,k}}{\Delta r^2} + e_i \frac{D_i^{(p)}}{\bar{X}^2} \frac{a_{i,j+1,k} - a_{i,j,k}}{\Delta x} = 0 \\
& \frac{a_{i+1,j,k} - a_{i,j,k}}{\Delta t} - \frac{D_i^{(p)}}{R^2} \left[\frac{a_{i,j-1,k} - 2a_{i,j,k} + a_{i,j+1,k}}{\Delta x^2} + \frac{2}{\bar{X}} \frac{a_{i,j+1,k} - a_{i,j,k}}{\Delta x} \right] = 0 \\
& a_{i,j,0} = 0; \\
& a_{i,0,m} - K_i c_{i,0,m} (1 - c_{i,0,m}) = 0; \\
& c_{i,j,m} - c_{i,j-1,m} = 0; \\
& D_i^{(j)} c_{i,j-1,m} - (D_i^{(j)} + D_i^{(j+1)}) c_{i,j,m} + D_i^{(j+1)} c_{i,j+1,m} = 0
\end{aligned} \tag{2.37}$$

Дана різницєва може бути застосована в довільній просторовій області:

$\Omega_{i,j,k} = \{i\Delta t; j\Delta x, k\Delta r, i=1..N, j=1..M, k=1..L\}$, де

$\Delta t = T / N; \Delta x = \bar{X} / M; \Delta r = R / L$.

2.3. Постановка задачі із програмної реалізації різницевої схеми математичної моделі дифузії методом прямої та зворотної прогонки.

Для алгоритмізації розвинених сіткових математичних моделей ми застосовували пряму і зворотну прогонки — це числові методи для розв'язку лінійних систем рівнянь, які виникають у різницєвих схемах, зокрема в нашому випадку при моделюванні дифузії Пряма прогонка (Forward Sweep) нами використовується для знаходження проміжних коефіцієнтів у тридіагональній системі рівнянь. Для цього друге рівняння системи (2.35) подавалось у лінеаризованій формі:

$$a_{i+1,j,m} - a_{i,j,m} - (\beta\Delta t + \varepsilon)\gamma a_{i,j,m} = 0. \tag{2.36}$$

На першому етапі система рівнянь (2.35) приводиться до верхньої трикутної форми, що значно спрощує обчислення. В цьому процесі змінюються коефіцієнти матриці та вектора правої частини, щоб підготувати систему до розв'язання методом підстановки.

Зворотна прогонка (Backward Sweep). В нашому випадку це є другий етап алгоритму, де після прямої прогонки відбувається послідовне обчислення шуканих значень змінних.

Оскільки система приведена до трикутного вигляду, рішення можна знайти шляхом підстановки від останнього рівняння до першого.

Зворотна прогонка дозволяє ефективно отримати розв'язок системи, виходячи із початкових та граничних умов. В даному випадку пряма прогонка — це підготовчий етап, на якому зводять систему до трикутного вигляду. Зворотна прогонка — це етап безпосереднього знаходження розв'язку, використовуючи отримані на першому етапі коефіцієнти. У розвиненому нами методі моделювання процесу дифузії ці два етапи працюють разом: спочатку знаходяться проміжні значення (пряма прогонка), а потім — остаточне розв'язання (зворотна прогонка). Також слід зауважити, що розвинена нами дана методика побудови різницевих схем математичних моделей та їх алгоритмізації ефективна при розв'язку рівнянь теплопровідності та дифузії, особливо при використанні неявних схем, таких як метод Крана-Ніколсон.

Нами виконувалась програмна реалізація блоку різницевої схеми (2.35), (2.36). Вона будувалась з використанням модуля Manipulate, що дозволяє довільно міняти розміри зразка та часовий інтервал в якому відбувається дифузійний процес. Загальна схема і програмний код, що реалізує алгоритм прямої і зворотної прогонки є така, як подано далі.

```

Manipulate[L = 1; T = 0.1;
[маніпулювати
  nx = nxVal; nt = ntVal; dx = L / nx;
  dt = T / nt;
  alpha = 0.01; \r = alpha dt / dx^2;
  u = Table[0, {i, 0, nx}, {j, 0, nt}];
  [таблиця значень
  Do[u[[i+1, 1]] = Sin[Pi i dx], {i, 0, nx}];
  [оператор циклу [с... [число пі
  Do[u[[1, j+1]] = 0; u[[nx+1, j+1]] = 0, {j, 1, nt}];
  [оператор циклу
  (*Пряма прогонка*) A = Table[0, {i, 1, nx-1}, {j, 1, nx-1}];
  [таблиця значень
  B = Table[0, {i, 1, nx-1}];
  [таблиця значень
  Do[Do[A[[i, i]] = 1 + 2 r; If[i > 1, A[[i, i-1]] = -r];
  [... [оператор циклу [умовний оператор
    If[i < nx-1, A[[i, i+1]] = -r], {i, 1, nx-1}];
  [умовний оператор
  Do[B[[i]] = u[[i+1, j]], {i, 1, nx-1}];
  [оператор циклу
  sol = LinearSolve[A, B];
  [розв'язати систему лінійних рівнянь
  Do[u[[i+1, j+1]] = sol[[i]], {i, 1, nx-1}, {j, 1, nt}];
  [оператор циклу
  (*Зворотна прогонка*) Do[sol = LinearSolve[A, B];
  [операто... [розв'язати систему лінійних рівнянь
  Do[u[[i+1, j+1]] = sol[[i]], {i, 1, nx-1}, {j, nt, 1, -1}];
  [оператор циклу
  (*Візуалізація результату*) ListPlot3D[u, Mesh -> All,
  [тримірна діагр... [сітка [все
  ColorFunction -> "TemperatureMap",
  [функція забарвлення
  AxesLabel -> {"x", "t", "u"}], {{nxVal, 10, "nx"}, 5, 50, 1}, {{ntVal, 100, "nt"}, 50, 500, 10}]

```

Рис. 2.1. Загальна схема реалізації алгоритму прямої і зворотної прогонки при реалізації різницевої схеми дифузії.

Далі на Рис. 2.3. подано приклад моделювання просторового розподілу у зразку цеоліту, а також переріз цієї залежності. Для забезпечення можливості побудови перерізу просторової залежності використовувалась модифікація програмного коду наступним чином:

```
(*Візуалізація результату*)
Grid[{ {ListPlot3D[u, Mesh → All, ColorFunction → "TemperatureMap", AxesLabel → {"x", "t", "u"}],
таблиця тримірна діагр... сітка все функція забарвлювання позначення на осях
ListLinePlot[Transpose[u][[pereriz]], Mesh → All, PlotStyle → Thick, AxesLabel → {"x", "u"}],
лінійний графік... транспозиція сітка все стиль графіка жирни... позначення на осях
ContourPlot[u[Round[x nx] + 1, Round[t nt] + 1], {x, 0, 1}, {t, 0, T},
контурний графік округлити округлити
Contours → 20, ColorFunction → "TemperatureMap", AxesLabel → {"x", "t"}]}],
контури функція забарвлювання позначення на осях
|{{nxVal, 10, "nx"}, 5, 50, 1}, {{ntVal, 100, "nt"}, 50, 500, 10},
|{{pereriz, 10, "t-slice"}, 1, nt, 1}]
```

Рис. 2.2. Модифікація програмного коду для 2D-візуалізації просторової залежності дифузії у вигляді перерізів

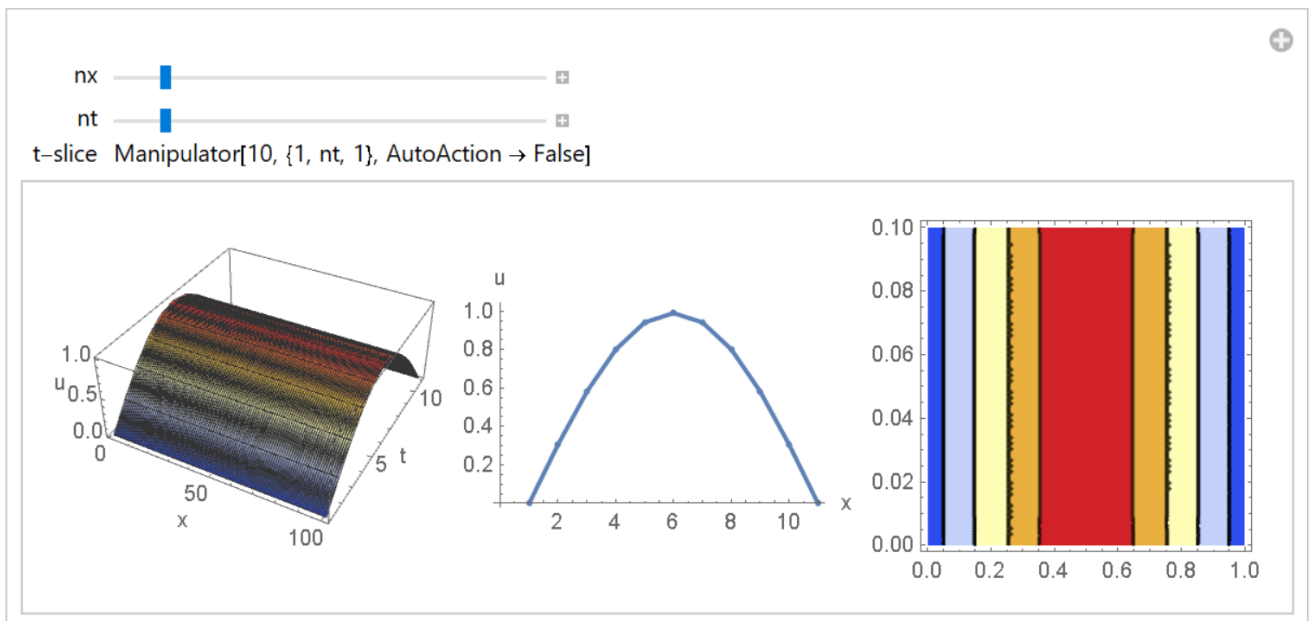


Рис. 2.3. Результат роботи блоку програмної системи із моделювання просторової залежності концентраційного розподілу та його перерізів за допомогою директиви Manipulate

РОЗДІЛ 3. РОЗРОБКА ТА ПРОЄКТУВАННЯ АРХІТЕКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МАТЕМАТИЧНИХ МОДЕЛЕЙ ДИФУЗІЇ В БАГАТОШАРОВИХ ЗРАЗКАХ.

3.1 Вимоги до програмного забезпечення.

У процесі створення програмної системи надзвичайно важливо чітко сформулювати вимоги до програмного забезпечення. Це дозволяє не лише забезпечити правильне функціонування програмної системи, але й гарантує її відповідність очікуванням кінцевого користувача, підтримку подальшої модифікації, адаптацію до інших задач і надійність експлуатації. Вимоги до програмного забезпечення поділяються на:

- функціональні;
- нефункціональні.

Функціональні вимоги описують конкретні можливості, які повинна реалізовувати програмна система. Вони базуються на задачах, які стоять перед користувачем — дослідником, що працює з пористими матеріалами. До функціональних вимог системи належать наступні аспекти:

1. Введення параметрів моделі:

- Система повинна дозволяти користувачеві задавати кількість шарів зразка (від 1 до 1000);
- Для кожного шару має бути можливість задання індивідуальних параметрів: пористість, коефіцієнт дифузії, товщина шару, матеріал (цеоліти типів ZSM-5, ZSM-11, ZSM-12, силікагель);
- Має бути забезпечений вибір дифундуючого агента із списку доступних (метан, водяна пара, інертні гази тощо);
- Необхідна підтримка вводу температури та інших фізико-хімічних параметрів середовища.

2. Побудова математичної моделі:

- Система повинна реалізовувати математичну модель дифузії з підтримкою апроксимації методом скінченних різниць;
- Необхідна реалізація схеми Кранка-Ніколсона для неявної апроксимації диференціальних рівнянь у частинних похідних;
- Повинна бути реалізована система обчислень за допомогою методу прямої та зворотної прогонки для тридіагональних систем.

3. Обробка результатів та візуалізація:

- Побудова графіків розподілу концентрацій у часі й просторі для заданої геометрії зразка;
- Створення 2D-графіків (перерізи) та 3D-візуалізацій (ізометрична проекція, топологія дифузії);
- Підтримка динамічної візуалізації (анімації) еволюції концентраційного профілю у часі;
- Можливість масштабування графіків, накладання кількох залежностей, збереження графічних результатів.

4. Інтерфейс користувача:

- Інтерфейс повинен бути реалізований із використанням елементів керування (слайдери, випадаючі меню, поля вводу);
- Передбачена перевірка введених значень на допустимість (наприклад, $0 < \text{пористість} < 1$);
- Можливість збереження й завантаження конфігурацій моделей.

5. Експорт та імпорт даних:

- Підтримка експорту числових результатів у формати CSV, TXT для подальшої обробки в інших системах;
- Можливість збереження графіків у форматах PNG, SVG або PDF;
- Опціональна підтримка автоматичного створення звітів (у форматі LaTeX, PDF).

Нефункціональні вимоги визначають якісні характеристики програмної системи, її обмеження, продуктивність, зручність та інші важливі аспекти.

1. Продуктивність:

- Система повинна працювати без збоїв із моделями, що містять до 1000 шарів, та забезпечувати час обчислення результатів не більше 10 секунд на стандартному ПК (Intel i5, 8 ГБ оперативної пам'яті);
- Побудова графіків та візуалізацій не повинна перевищувати 2–3 секунд при зміні параметрів моделі;
- Використання кешування проміжних результатів для зменшення навантаження на систему при повторних обчисленнях.

2. Надійність:

- Програмна система повинна мати механізми обробки помилок: у випадку некоректного введення система повинна видавати відповідне повідомлення, не допускаючи аварійного завершення виконання;
- Результати чисельного моделювання мають бути перевірені на стабільність і збіжність з аналітичними розв'язками для базових випадків.

3. Зручність використання:

- Інтерфейс користувача повинен бути інтуїтивно зрозумілим та не вимагати від користувача спеціальної підготовки в області програмування або чисельного аналізу;
- Забезпечення підтримки контекстної допомоги до основних функціональних блоків;
- Наявність короткої довідки (help) або інтерактивної документації до інтерфейсу.

4. Розширюваність та масштабованість:

- Програмна система повинна бути побудована за модульним принципом, що дозволяє легко додавати нові моделі, змінювати алгоритми розрахунку або додавати нові матеріали;
- Архітектура повинна дозволяти адаптацію системи до нових умов (наприклад, конвекційно-дифузійні задачі або реакційно-дифузійні системи).

5. Сумісність:

- Програмна система повинна бути повністю сумісною з Wolfram Mathematica версії 13.0 і вище;
- Коректна робота має забезпечуватись на основних операційних системах: Windows 10/11, Linux (через Wolfram Kernel), macOS;
- Підтримка експорту в універсальні формати для інтеграції з іншими науковими інструментами (MATLAB, Excel, LaTeX).

6. Документованість:

- Усі основні компоненти програмної системи мають бути супроводжені коментарями в коді з описом функціоналу;
- Повинен бути створений файл документації користувача із прикладами використання системи, описом інтерфейсу та параметрів моделювання;
- Для розробників має бути сформований внутрішній технічний опис (специфікація модулів, структура програми, схема обробки даних).

Таким чином, формалізація вимог до програмного забезпечення є основою для забезпечення стабільності, ефективності, масштабованості та зручності використання розробленої системи. Правильна постановка вимог дозволяє системі залишатися актуальною не лише для вирішення поточних задач, а й у перспективі її розширення для нових видів мікропористих матеріалів або більш складних фізико-хімічних процесів.

3.2 Вибір методології розробки.

У процесі створення програмної системи для моделювання процесів дифузії було прийнято рішення використовувати раціональний підхід до вибору методології, орієнтований на поєднання науково-обґрунтованих чисельних методів із принципами модульного програмування. Основними критеріями при виборі методології стали:

- необхідність реалізації складної математичної моделі, що включає систему рівнянь у частинних похідних з гнучкими граничними умовами;
- потреба у високій точності чисельного розв’язання з можливістю контролю похибки;
- підтримка гнучкої взаємодії з користувачем, включаючи зміну параметрів та вибір алгоритмів;
- забезпечення масштабованості та можливості модифікації системи для інших типів матеріалів або розширення фізико-хімічної моделі.

З огляду на ці вимоги, як основу обрано адаптивну інкрементну методологію розробки з елементами прототипування. Це дозволило на ранніх етапах створити функціональний каркас системи (мінімальний життєздатний продукт – MVP) і поступово розширювати функціональність, перевіряючи точність реалізації на базових тестових випадках.

Програмна реалізація базується на використанні сіткових чисельних методів (метод скінченних різниць) із подальшою реалізацією методу прямої та зворотної прогонки, що дозволяє ефективно розв’язувати тридіагональні системи рівнянь. Цей метод було обрано з огляду на його високу обчислювальну ефективність, простоту реалізації та стабільність при роботі з великими об’ємами даних.

Для забезпечення інтерактивності та зручного доступу до функціоналу обрано Wolfram Mathematica, яка поєднує аналітичні та чисельні можливості з візуалізацією і елементами GUI через модулі Manipulate, NDSolve, Plot3D тощо.

Загалом обрана методологія дозволяє підтримувати:

- гнучку структуру архітектури, побудовану на модульному принципі;
- сумісність з подальшими науковими розробками в суміжних галузях;
- можливість адаптації до нових математичних моделей (наприклад, реакційно-дифузійних або конвекційних систем);
- використання адаптивних сценаріїв тестування для перевірки точності чисельних результатів.

Таким чином, вибрана методологія розробки є оптимальною для поставлених задач, забезпечує баланс між точністю математичних обчислень, гнучкістю архітектури та зручністю кінцевого користувача.

3.3 Архітектура програмної системи.

У відповідності до скінченно-різницевої форми, що реалізує математичні моделі кінетики дифузії, виконувалась розробка програмного забезпечення, що реалізує згадані моделі. На першому етапі розроблялась архітектура програмного забезпечення, яка забезпечує найбільш зрозумілу інтерактивну взаємодію користувача з програмною системою. Безпосередньою метою було забезпечення варіативності у можливості користувача змінювати вхідні параметри математичної моделі, її фізичні параметри та геометричний конфаймент реальних досліджуваних систем.

Отримана архітектура програмної системи, яка виражається у діаграмі варіантів використання, подана на Рис. 3.1. Вона відображає основні взаємодії користувача із системою, визначаючи ключові сценарії її використання. Згідно з розробленою архітектурою, користувачеві не обов'язково безпосередньо ознайомлюватися із суттю математичної реалізації абстрактних моделей, оскільки система автоматизує більшість процесів розрахунку та обробки даних. Проте, на діапазон змінних вхідних параметрів наперед накладалися обмеження, що виражають технічну сторону проблеми та властивості досліджуваних матеріалів. Ці обмеження визначені на основі емпіричних даних і аналітичних досліджень, що дозволяє зменшити похибку обчислень та підвищити достовірність отриманих результатів. Окрім цього, архітектура передбачає модульність системи, що дає змогу її подальшої адаптації та вдосконалення. Основні компоненти системи розподілені таким чином, щоб забезпечити ефективність обчислювальних процесів, а також зручність для кінцевого користувача. Важливим аспектом є можливість

інтеграції з іншими програмними комплексами, що розширює сферу застосування розробленої системи.



Рис. 3.1. Діаграма варіантів використання

В результаті реалізація архітектури згідно діаграми варіантів використання користувач програмної системи може виконувати такі дії:

Таблиця 3.1 – Сценарії взаємодії користувача з програмною системою

№	Функція інтерфейсу	Опис дії користувача
1	Введення типу матеріалу	Користувач обирає тип матеріалу шару з доступного списку: ZSM-5, ZSM-11, ZSM-12, силікагель.
2	Введення кількості шарів зразка	Користувач задає кількість шарів, з яких складається зразок (наприклад, від 1 до 1000).
3	Введення розмірів зразків шарів	Задаються геометричні параметри кожного шару: товщина, радіус, положення шару.

Продовження таблиці 3.1

4	Введення фізичних параметрів зразка	Вводяться значення коефіцієнтів дифузії, пористості, температури, розміру пор тощо.
5	Вибір моделі: Кранк-Ніколсон	Користувач обирає неявну схему Кранка-Ніколсона для апроксимації диференціальних рівнянь.
6	Вибір моделі: Різницева схема прямої прогонки	Вибирається метод розв'язання тридіагональної системи — пряма та зворотна прогонка.
7	Вибір 2D візуалізації	Користувач обирає побудову 2D-графіка розподілу концентрації (переріз, залежність від координати).
8	Вибір 3D візуалізації	Вибирається режим побудови тривимірної моделі або ізоповерхонь концентрації.

Такий підхід дозволяє формалізувати функціональність на ранньому етапі розробки, забезпечити повноту реалізації вимог, а також слугує зручним інструментом для комунікації між розробниками, користувачами та замовниками системи.

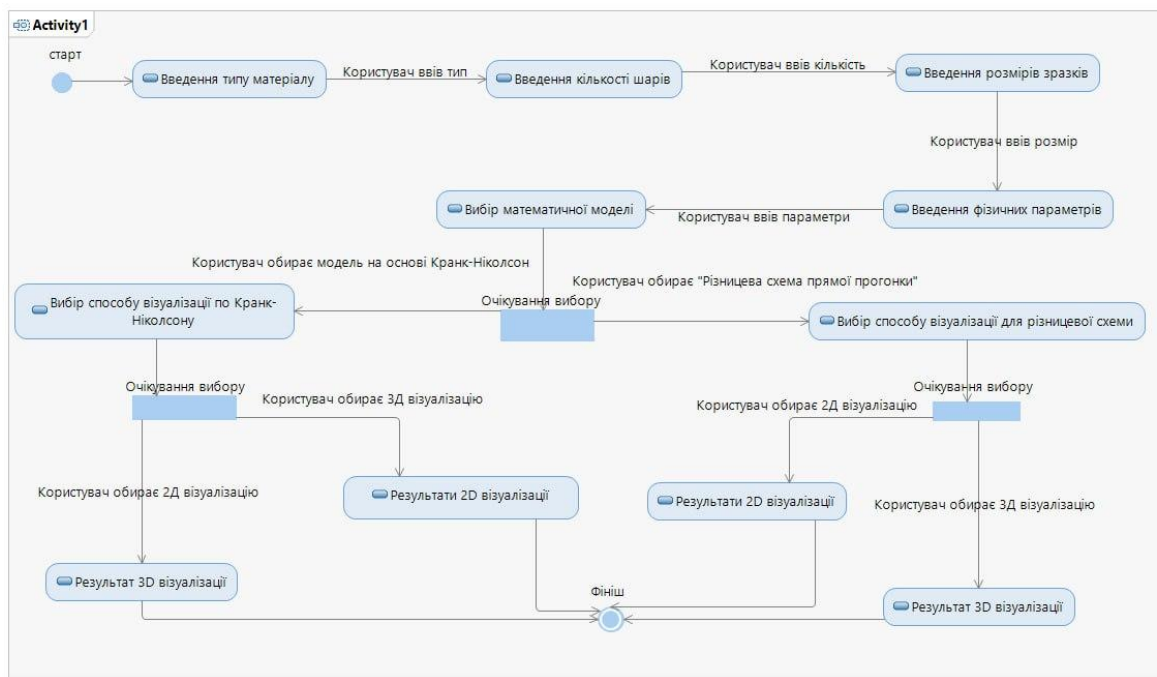


Рис. 3.2. Діаграма активності

На Рис. 3.2 подано діаграму активності розробленої програмної системи, що узагальнює діаграму варіантів використання подану вище. З діаграми видно детальну послідовність дій користувача в процесі використання програмної системи для моделювання фізичних процесів у багатошарових матеріалах. Весь процес починається зі стартової точки, після чого користувач вводить базові дані: тип матеріалу, кількість шарів та розміри зразків. Це дозволяє системі зібрати необхідну початкову інформацію для подальших обчислень. Далі користувач вводить фізичні параметри матеріалу, що є критично важливими для точності математичного моделювання.

Після введення параметрів користувач обирає математичну модель, за якою буде здійснюватися розрахунок. У діаграмі представлені два варіанти: модель на основі методу Кранк-Ніколсона та різницева схема прямої прогонки. Кожен з цих методів передбачає власний підхід до обчислення та візуалізації даних, тому після вибору моделі користувач має окремо обрати спосіб візуалізації результатів саме для цієї моделі.

У наступному кроці користувач обирає формат відображення результатів: 2D або 3D візуалізація. У залежності від обраного формату система генерує відповідне

зображення результатів обчислень. Якщо обрано 2D-візуалізацію, користувач отримує плоске графічне представлення даних, а при виборі 3D – тривимірну модель розподілу параметрів у зразку.

У завершальній фазі користувач отримує результати у вибраному форматі, після чого процес моделювання вважається завершеним. Таким чином, діаграма чітко демонструє логіку взаємодії користувача із системою, можливість вибору між різними математичними моделями та форматами візуалізації, а також гнучкість та адаптивність програми до потреб користувача.

Однією з ключових складових архітектури розробленої програмної системи є модульна архітектура, представлена у вигляді UML-діаграми (рис. 3.3). Дана діаграма відображає основні компоненти системи, їхні функції та взаємозв'язки, що дозволяє забезпечити модульність, масштабованість і розширюваність програмного продукту.

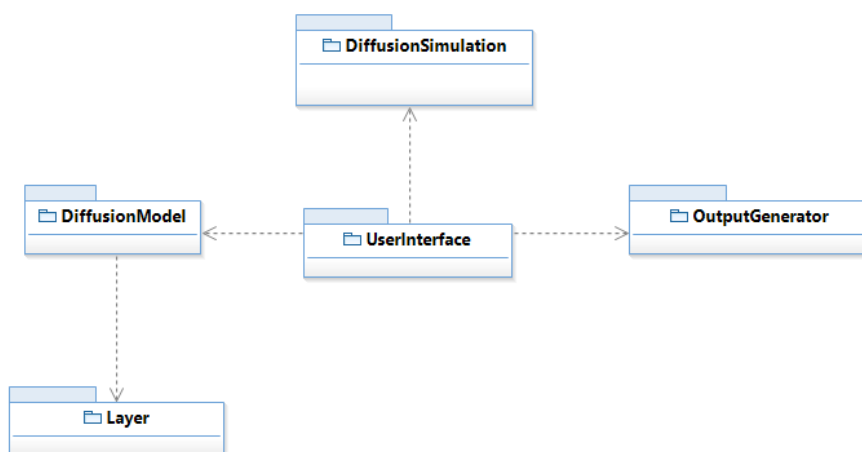


Рис. 3.3. Модульна архітектура програмної системи

У центрі архітектури знаходиться клас DiffusionSimulation, який виконує роль координатора процесу моделювання. Він взаємодіє з іншими компонентами, ініціює розрахунки, встановлює параметри та формує результати. До нього

підключається клас `DiffusionModel`, який реалізує фізико-математичну модель дифузії. У складі моделі використовується композиційний зв'язок з класом `Layer`, що описує окремі шари матеріалу за такими параметрами, як товщина, коефіцієнт дифузії та початкова концентрація речовини.

Клас `UserInterface` відповідає за взаємодію з користувачем. Він забезпечує виведення графічного інтерфейсу, обробку введених параметрів та оновлення результатів у зручному для користувача форматі. Таким чином, взаємозв'язок між класами `DiffusionSimulation` та `UserInterface` дозволяє адаптувати поведінку системи до введених даних та візуально демонструвати результати моделювання.

Окремий функціональний блок — `OutputGenerator` — відповідає за формування звітів та експорт результатів. Він реалізує функції виводу графіків, таблиць, а також збереження числових даних у форматі CSV чи PDF. Така декомпозиція дозволяє чітко розмежувати функціональність між підсистемами.

Загалом представлена архітектура реалізує принципи розділення відповідальності, інкапсуляції та низької зв'язаності, що забезпечує гнучкість при модернізації або інтеграції програмної системи в інші програмні середовища. UML-діаграма класів наочно демонструє структуру проекту та слугує основою для подальшої розробки, тестування та супроводу системи.

Згідно розробленої архітектури програмна система за своєю практичною реалізацією складається з двох окремих блоків, кожен з яких містить чотири підпрограми, які відповідають різним типам досліджуваних зразків. Таким чином система дозволяє забезпечити потреби широкого кола дослідників. Зокрема, вона дозволяє бути зручною для використання вченими та інженерами, які працюють безпосередньо в експерименті і потребують здебільшого конкретних числових даних для порівняння. Крім того, застосовані у Wolfram Mathematica гнучкі засоби для візуалізації будуть корисними для дослідників, які займаються виключно аспектами прикладного математичного моделювання таких систем та зразків.

Таким чином, розроблена архітектура програмної системи забезпечує необхідну функціональність, зручність користування та гнучкість для подальшого розвитку. Вона поєднує обчислювальну ефективність із високою інформативністю

візуальних результатів, що є критично важливим для наукових та прикладних досліджень.

3.4 Розробка клієнт-машинного інтерфейсу та функціоналу програмної системи. Алгоритм методу. Розробка компонентів програмної системи.

Інтерфейс користувача відіграє ключову роль у взаємодії дослідника з програмною системою. Його основне завдання — забезпечити інтуїтивно зрозумілий, зручний і ефективний спосіб введення параметрів математичної моделі, запуску обчислень та візуалізації результатів. У межах даного проєкту було розроблено інтерактивний інтерфейс із використанням інструментів середовища Wolfram Mathematica, зокрема функції Manipulate, що дозволяють створювати інтерактивні елементи керування.

Реалізація інтерфейсу ґрунтується на принципах доступності, модульності та зворотного зв'язку. Користувач отримує можливість змінювати всі ключові параметри обчислень у режимі реального часу, а система — оперативно реагувати на введені значення шляхом оновлення результатів візуалізації чи повідомлень про помилки. Це дозволяє проводити численні обчислювальні експерименти без необхідності перезапуску програми або редагування коду вручну.

Проаналізуємо принципи роботи з розробленою програмною системою та її компонентами більш детально. На Рис. 3.2 подано приклад введення вхідних параметрів цеоліту користувачем та вибір необхідної для його роботи складової програмної системи.

Лістинг 3.1 – Реалізація інтерфейсу:

```

Manipulate[Module[{result, poreVolume, surfaceArea, adsorptionCapacity,
[маніпулювати] [програмний модуль]
    thermalConductivity}, poreVolume = porosity * unitCellVolume;
    surfaceArea = 2 length width + 2 width height + 2 length height;
    adsorptionCapacity = surfaceArea * adsorptionFactor;
    thermalConductivity = baseThermalConductivity * (1 - porosity);
    result = Switch[component, "Структурні властивості",
[перемикуючий елемент]
        "Цеоліт ZSM-5: Структурні характеристики\n" <>
        "Розміри -> Довжина: " <> ToString[length] <> ",
[перетворити в стрічку]
        Ширина: " <> ToString[width] <> ", Висота: " <> ToString[height] <> "\n" <>
[перетворити в стрічку] [перетворити в стрічку]
        "Пористість: " <> ToString[porosity] <> ",
[перетворити в стрічку]
        Об'єм пор: " <> ToString[poreVolume], "Сорбційні характеристики",
[перетворити в стрічку]
        "Цеоліт ZSM-5: Сорбційні параметри\n" <> "Площа поверхні:
        " <> ToString[surfaceArea] <> "\n" <> "Фактор адсорбції: " <> ToString[adsorptionFactor] <> "\n" <>
[перетворити в стрічку] [перетворити в стрічку]
        "Ємність адсорбції: " <> ToString[adsorptionCapacity],
[перетворити в стрічку]
        "Теплопровідність", "Цеоліт ZSM-5: Теплофізичні параметри\n" <> "Базова теплопровідність: "
        <> ToString[baseThermalConductivity] <> "\n" <>
[перетворити в стрічку]
        "Теплопровідність з урахуванням пористості: " <> ToString[thermalConductivity]];
[перетворити в стрічку]

    result], (*Геометричні параметри*){{length, 1, "Довжина (нм)"}, {0.1, 10}
, {{width, 1, "Ширина (нм)"}, {0.1, 10}, {{height, 1, "Висота (нм)"}, {0.1, 10},
{{porosity, 0.3, "Пористість"}, {0.1, 0.9},
{{unitCellVolume, 0.5, "Об'єм елементарної комірки (нм³)"}, {0.1, 2.0},
{{adsorptionFactor, 10, "Фактор адсорбції"}, {1, 50},
{{baseThermalConductivity, 0.2, "Базова теплопровідність (Вт/м·К)"}, {0.1, 1.0},
|{component, {"Структурні властивості", "Сорбційні характеристики", "Теплопровідність"},
    ControlType -> PopupMenu}]]
[тип елемента у...] [випадаюче меню]

```

В результаті отримується інтерфейс за допомогою якого користувач може міняти вхідні параметри математичної моделі, геометричні та фізико-хімічні параметри матеріалів за допомогою слайдерів. Для прикладу вибору параметрів зразку на основі цеоліту ZSM-5 таке меню користувача подано далі:

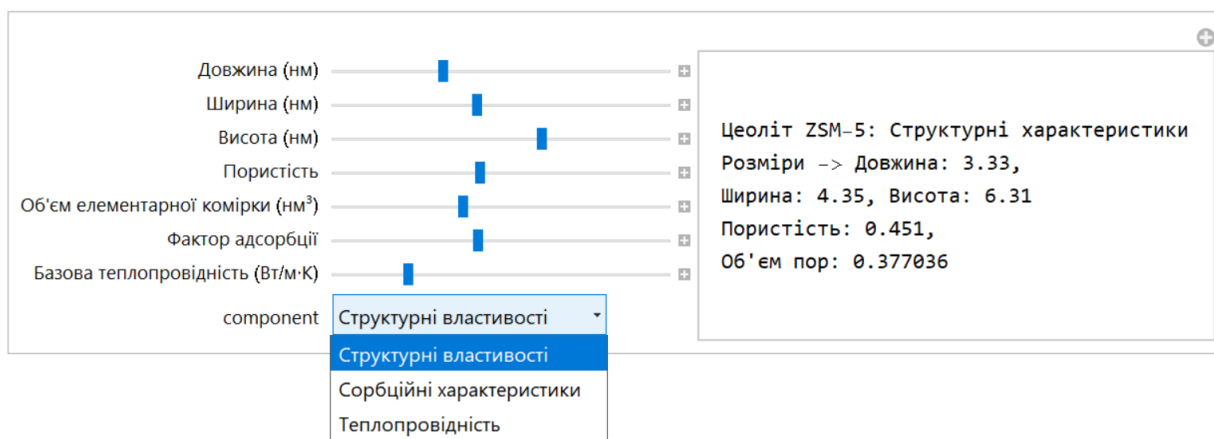


Рис. 3.4. Інтерфейс користувача при роботі з вхідними параметрами досліджуваного зразка

Як видно із Рис. 3.4. користувач програмної системи може ефективно змінювати геометричні параметри досліджуваного зразка у математичній моделі, при цьому вхідні фізико-хімічні параметра можна варіювати лише в реалістичних межах для заданого матеріалу (в даному випадку цеоліту ZSM-5). Після вибору вхідних параметрів користувач може вибрати у випадному меню одну із трьох компонент програмної системи, що відповідають за моделювання структурних властивостей зразка, його сорбційні характеристики та теплопровідність відповідно.

Таким чином, реалізований інтерфейс користувача забезпечує повний цикл взаємодії з математичною моделлю: від задання вхідних параметрів до отримання графічних і числових результатів. Завдяки інтерактивним елементам керування, користувач може оперативно змінювати умови моделювання, аналізувати вплив різних характеристик зразка на результати та отримувати візуалізацію процесів дифузії у зручному форматі. Обрана концепція побудови інтерфейсу дозволяє поєднати обчислювальну потужність системи з високою доступністю її функцій для широкого кола користувачів — від студентів до науковців.

3.5 Можливості інтеграції з іншими системами

Розроблена програмна система проєктувалася з урахуванням відкритості архітектури та гнучкості у використанні, що забезпечує широкі можливості інтеграції з іншими програмними засобами для обчислень, візуалізації, аналізу та формування звітності. Така інтеграційна здатність дозволяє застосовувати систему не лише як самодостатній інструмент для моделювання процесів дифузії в багатошарових зразках, а й як частину більших наукових чи інженерних обчислювальних комплексів, у тому числі в рамках багаторівневих досліджень та цифрових лабораторій.

Однією з базових інтеграційних функцій є можливість експорту результатів чисельного моделювання у поширені формати — зокрема, CSV, TXT, XML, JSON, які підтримуються у MATLAB, Python, Excel, OpenFOAM та інших інструментах. Це забезпечує подальший аналіз отриманих даних у різних аналітичних середовищах, зокрема при виконанні спектрального аналізу, побудові багатовимірних залежностей, проведенні оптимізаційних досліджень або статистичних оцінок. Графічні результати виводяться у форматах PNG, SVG, PDF, що дозволяє зручно використовувати їх у наукових публікаціях, презентаціях, звітах або під час порівняльного аналізу моделей. Крім того, у систему може бути інтегровано автоматичне формування звітів у форматах LaTeX або PDF із виведенням вихідних параметрів моделі, чисельних результатів, графіків і пояснень, що суттєво прискорює науково-аналітичну роботу.

Окрему увагу приділено можливості взаємодії з іншими платформами моделювання. Зокрема, структура програми дозволяє сформувати вхідні конфігураційні файли для середовищ COMSOL Multiphysics, ANSYS, FEniCS або MATLAB PDE Toolbox. Таким чином, обчислювальні параметри, геометричні характеристики та граничні умови, задані в межах розробленої системи, можуть бути використані як основа для подальших розрахунків у середовищах, орієнтованих на багатофізичне моделювання або моделювання з урахуванням інших типів взаємодій (наприклад, теплоперенесення, конвекції, електростатики тощо). Крім того, формат виводу параметрів у вигляді JSON або XML відкриває

можливості автоматизації передачі даних у суміжні додатки або реалізації REST-орієнтованих API-з'єднань у межах хмарної або клієнт-серверної архітектури.

Варто зазначити й потенціал застосування розробленої системи як генератора навчальних вибірок для методів машинного навчання, зокрема у задачах прогнозування, класифікації або регресії. Структуровані масиви даних, що зберігаються у табличній формі, можуть бути безпосередньо використані для тренування моделей штучного інтелекту, зокрема глибоких нейронних мереж. Особливо перспективною є інтеграція з так званими фізично-обґрунтованими нейронними мережами (PINNs — Physics-Informed Neural Networks), що дозволяє поєднувати точність аналітичних та чисельних методів із гнучкістю штучного інтелекту, знижуючи потребу в обчислювальних ресурсах і підвищуючи узагальнюваність результатів. Подібний підхід уже активно розвивається в сучасній науці, і завдяки відкритій архітектурі розробленої системи може бути реалізований без суттєвих модифікацій основного ядра.

Крім того, система демонструє високу сумісність з іншими інженерними інструментами та обчислювальними платформами, що використовуються для автоматизованого збору, обробки або візуалізації даних. Вона може використовуватись як частина програмного ланцюга в межах цифрового двійника реального об'єкта — наприклад, при створенні моделі пористого матеріалу для використання у сенсорах, фільтрах, акумуляторах або хімічних реакторах. За рахунок модульної структури та універсальності форм представлення даних, програмна система може бути також адаптована до роботи в хмарному середовищі (наприклад, Wolfram Cloud) або як частина серверного додатку із чисельним бекендом.

У підсумку, широкі інтеграційні можливості розробленої програмної системи роблять її не лише самодостатнім інструментом моделювання, а й потужним компонентом комплексних науково-інженерних платформ. Вони сприяють розширенню її функціональності, забезпечують гнучкість застосування, відкривають перспективи міждисциплінарного використання та дозволяють

ефективно поєднувати моделювання, аналіз, візуалізацію та оптимізацію в єдиному цифровому середовищі.

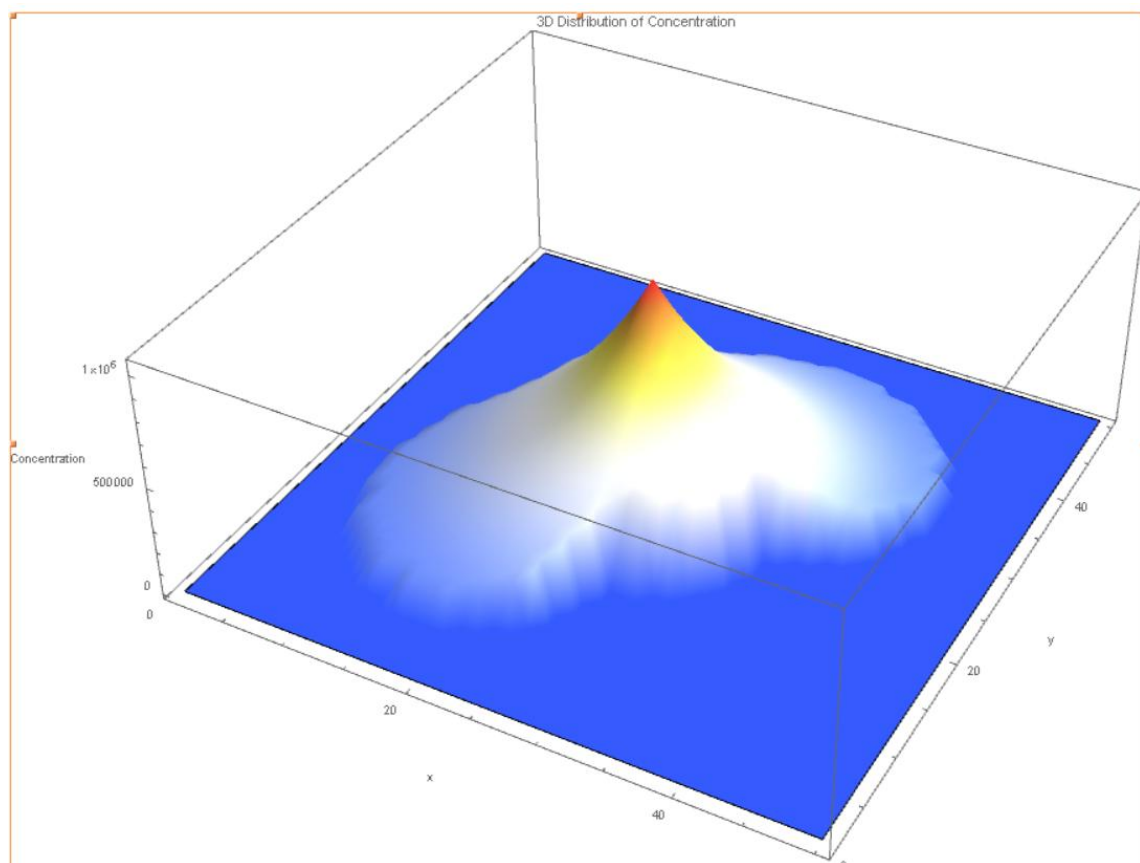
3.6 Тестування роботи з програмною системою. Аналіз дифузії у різнотипних зразках, візуалізація результатів.

Після завершення етапів побудови математичної моделі, її чисельної апроксимації та реалізації архітектури програмної системи постає завдання перевірки її коректної роботи. Тестування є критичним етапом у розробці програмного забезпечення, особливо якщо йдеться про системи для наукових обчислень, де навіть незначна помилка може призвести до спотворення результатів та хибних висновків. Метою даного етапу є верифікація точності реалізованої чисельної схеми, перевірка стабільності алгоритмів прямої та зворотної прогонки, а також аналіз поведінки програмної системи при варіюванні параметрів вхідних даних.

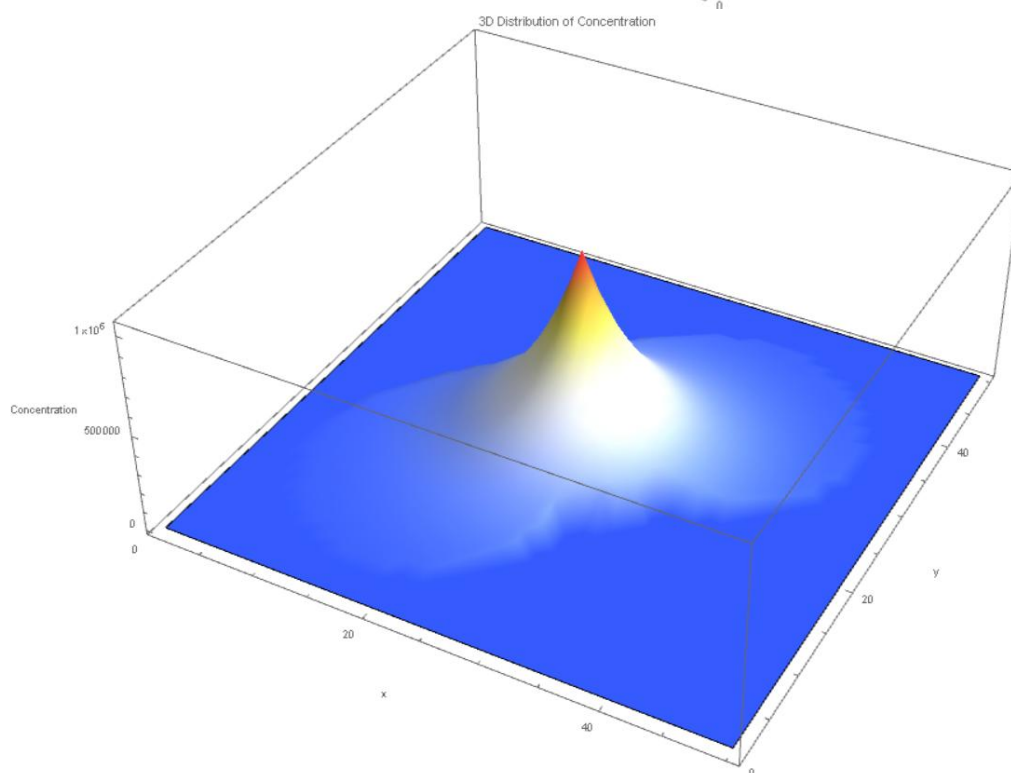
У рамках тестування було обрано кілька типових прикладів як з аналітичними, так і числовими розв'язками, зокрема для моношарових та багатошарових структур. Це дозволило не лише порівняти результати моделювання з теоретично очікуваними, але й виявити потенційні обмеження системи в контексті точності, продуктивності та стабільності. Візуалізація результатів у вигляді просторових та часових розподілів концентрацій надала можливість додаткового аналізу коректності функціонування програми на інтуїтивному рівні, що особливо важливо для подальшого її використання дослідниками, інженерами або в освітніх цілях.

Таким чином, тестування виконує не лише технічну, але й методологічну функцію: воно підтверджує працездатність побудованої моделі, а також демонструє потенціал системи як інструменту для аналізу складних фізико-хімічних процесів у багатошарових структурах.

На першому етапі верифікації розвинутих математичних моделей виконувалось порівняння результатів, що їх дає аналітична модель (2.28)-(2.29) з результатами, отриманими на основі реалізації скінченно-різницевої схеми. Для розрахунків вибиралась мікроструктура зразка сферичної форми із лінійними розмірами 50 мкм, на основі цеоліту ZSM-12 й пористістю 0.3, та розмірами пор 50 нм. В якості дифундованого агента вибирався метан при температурі 300K. Результати застосування компонентів програмної системи на основі аналітичної і дискретизованої математичної моделей подано на Рис. 3.5.



(a)



б)

Рис. 3.5. Концентраційні розподіли в моношаровому цеоліті ZSM-12 у аналітичній (а) та сітковій моделі (б)

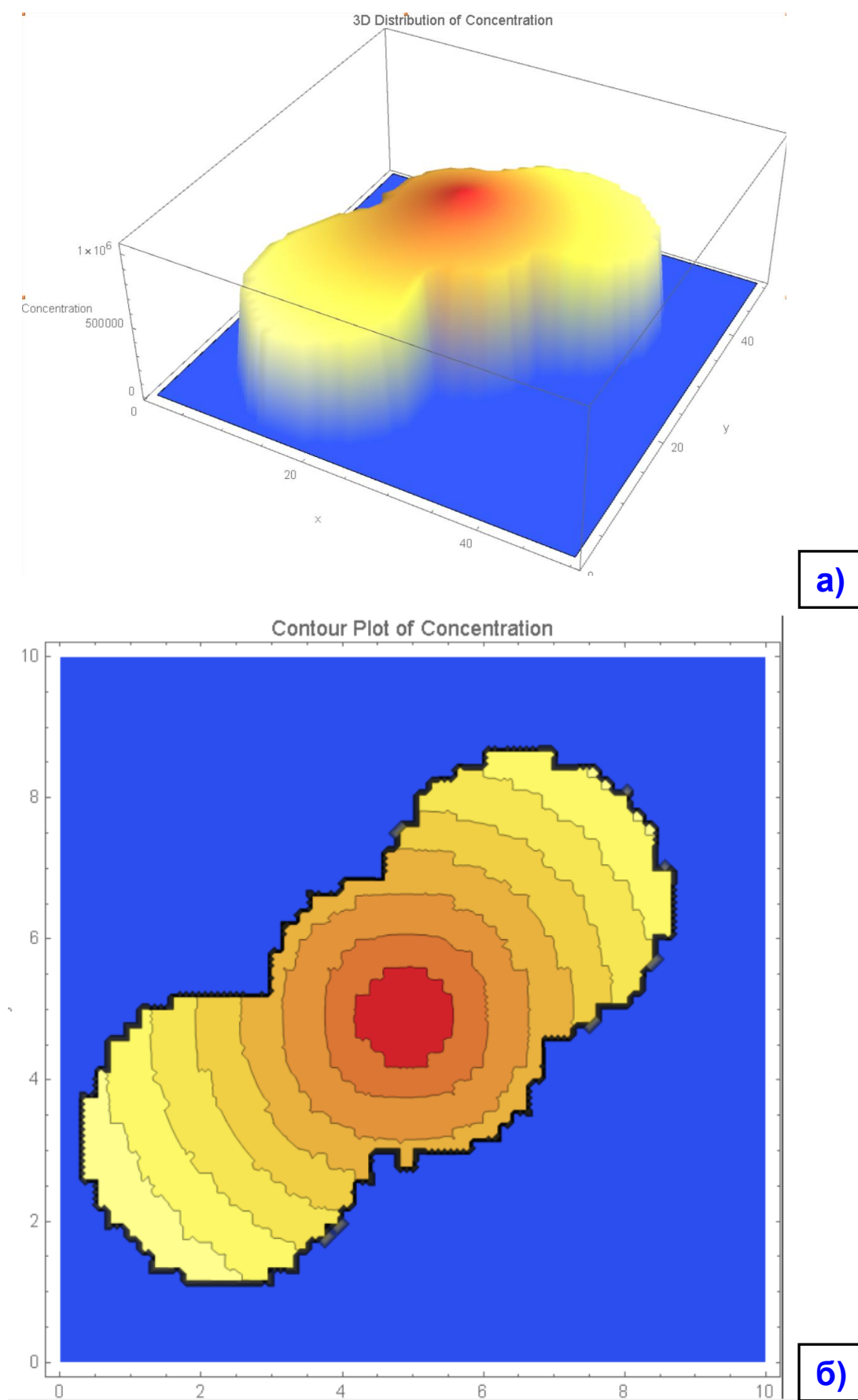


Рис. 3.6. Концентраційні розподіли цеоліті ZSM-5 з 8 шарами просторова залежність (а) та переріз (б)

Результати реалізації обох математичних моделей дають схожий та близький результат, що вказує на надійність цих методів у застосуванні до моношарових зразків.

Далі за допомогою розробленого програмного забезпечення для скінченно різницевого схем виконувалось математичне моделювання просторових розподілів та їх перерізів для різного типу зразків, що створені із різних матеріалів, але їх кількість шарів відповідає експериментально реалізованим конфігураціям. На Рис. 3.5 подано результати таких розрахунків для зразка із 8 шарами цеоліту ZSM-5 із пористістю від 0.1 до 0.45. Товщина кожного шару 10 нм.

Матеріали на основі цеоліту ZSM-11 зазвичай мають велику кількість шарів, розраховані для зразка із 20 шарами результати подано на Рис. 3.6. З розрахункових залежностей чітко видно, що цеоліт ZSM-11 має набагато більшу адсорбційну здатність ніж цеоліт ZSM-5, зокрема за рахунок більш рівномірного заповнення пор у кожному із шарів зразка.

На останок ми застосували програмний комплекс для зразка, створеного із поширеного і часто використовуваного матеріалу – силікагелю. Розраховані залежності подано на Рис. 3.8. Як видно зразок із силікагелю сильно програє у адсорбційних властивостях всім зразкам, що створені на основі цеолітів. Максимальна концентрація поглинутого агента отримується в центрі зразка, причому значна частина його не бере участі в процесі поглинання.

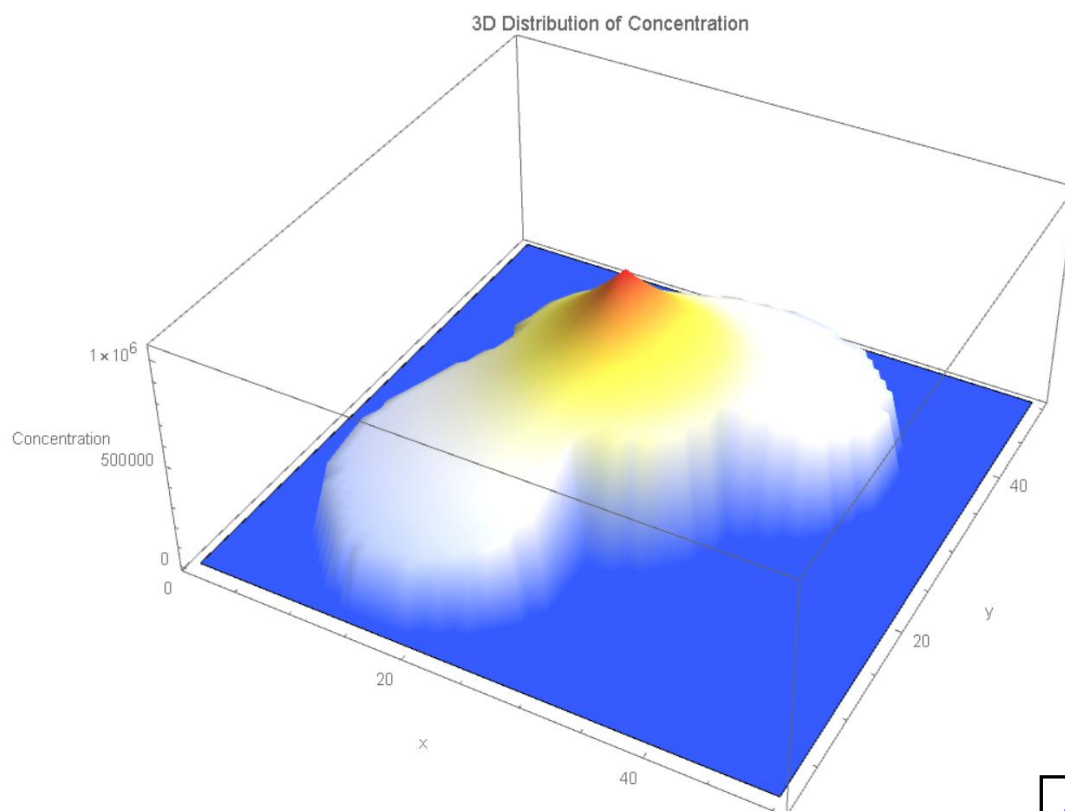
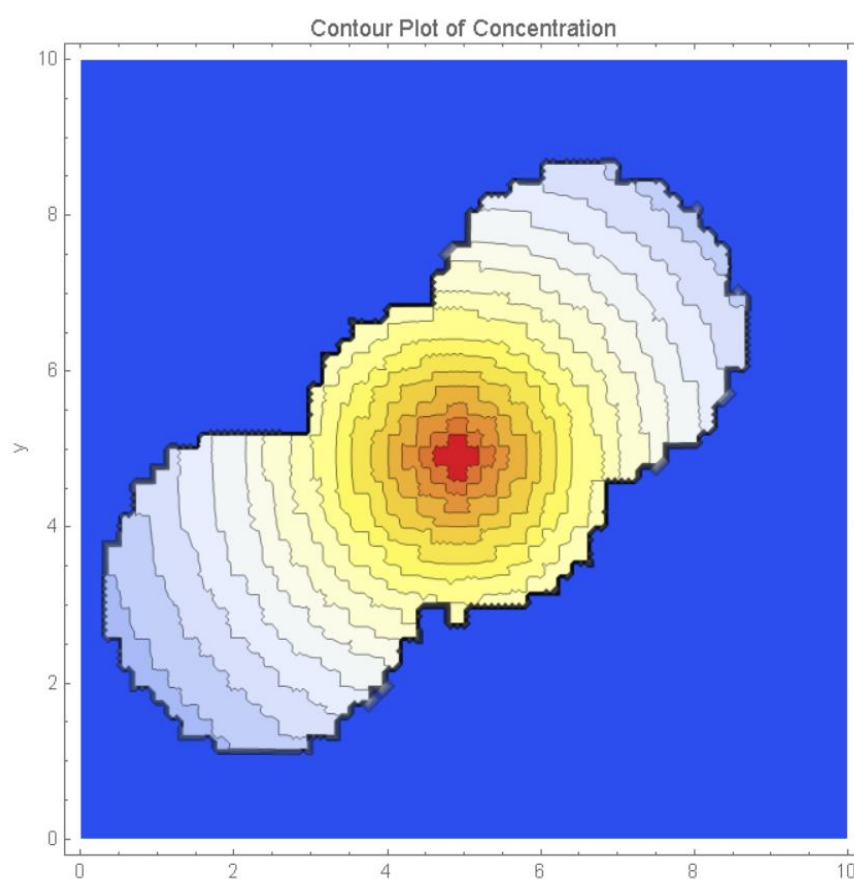
**a)**

Рис. 3.7. Концентраційні розподіли цеоліті ZSM-11 з 20 шарами: просторова залежність (а) та переріз (б)

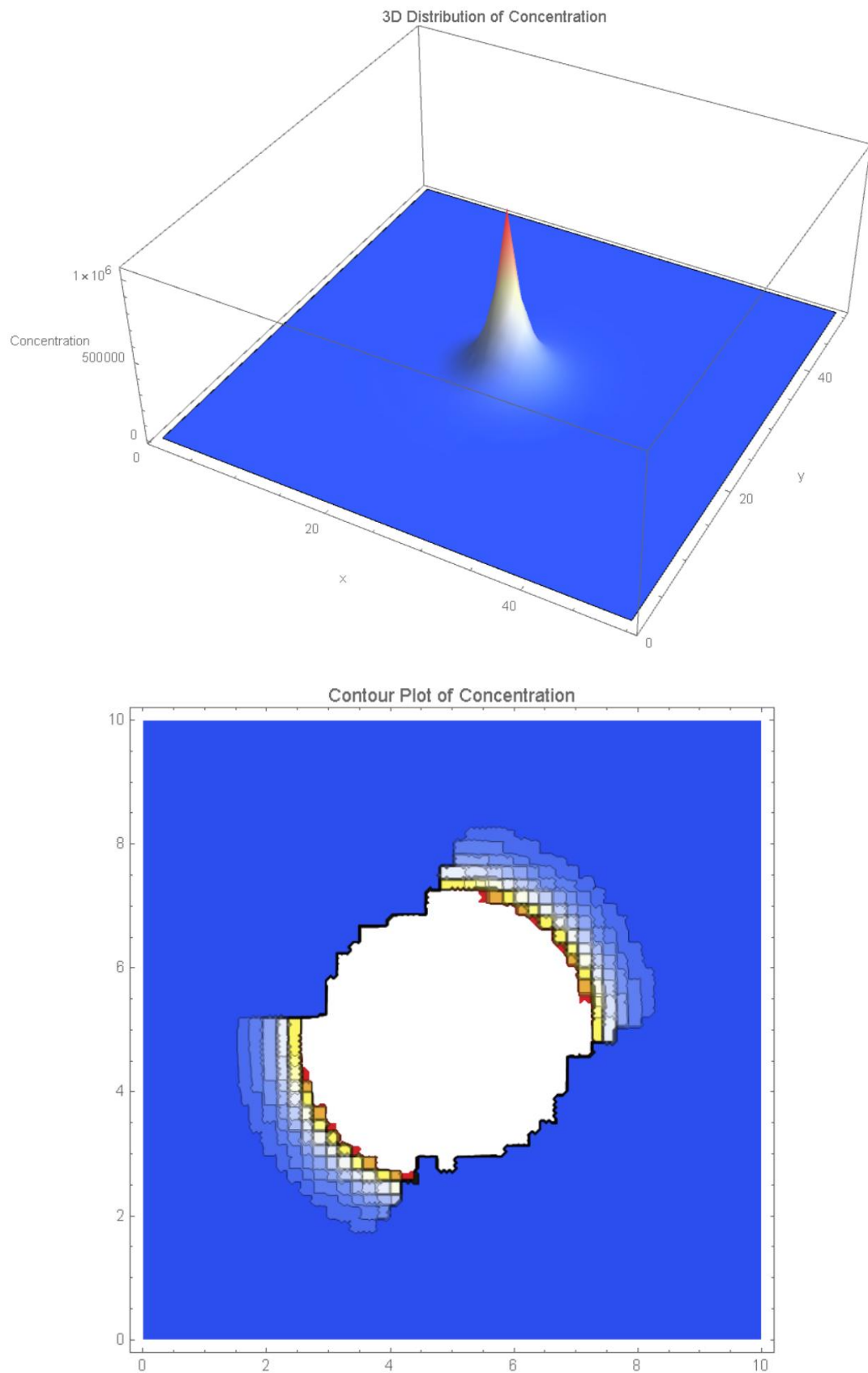


Рис. 3.8. Концентраційні розподіли силікагелі з 10 шарами: просторова залежність (а) та переріз (б)

Проведене тестування підтвердило правильність реалізації математичної моделі дифузії та ефективність чисельних алгоритмів, покладених в основу програмної системи. Отримані результати добре узгоджуються з аналітичними розв'язками для простих випадків та демонструють очікувану поведінку системи при зміні вхідних параметрів. Зокрема, модель виявила здатність адекватно описувати просторово-часову еволюцію концентраційного профілю в складних багат шарових структурах.

Візуалізація результатів моделювання дозволила наочно переконатися в достовірності отриманих залежностей та забезпечила зручність їх інтерпретації. Програмна система показала стабільну роботу в широкому діапазоні значень, зберігаючи високу швидкість обчислень навіть при великій кількості шарів.

Таким чином, програмна реалізація успішно пройшла перевірку, демонструючи готовність до подальшого застосування в дослідницьких та освітніх цілях, а також до розширення функціоналу відповідно до вимог користувача.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Ергономічні проблеми безпеки життєдіяльності

Ергономіка — це міждисциплінарна наука, яка вивчає особливості взаємодії людини з технічними системами, середовищем та інформацією. Основна мета ергономіки — створення умов праці, які враховують фізіологічні, психологічні, анатомічні й когнітивні особливості людини, щоб забезпечити безпеку, ефективність та комфорт у трудовій діяльності.

Термін "ергономіка" походить від грецьких слів *ergon* — праця і *nomos* — закон. Його вперше запропонував польський вчений Войцех Ястшембовський у 1857 році. У сучасному контексті ергономіка виконує важливу роль у сфері безпеки життєдіяльності, оскільки правильне проектування робочих систем безпосередньо впливає на зниження виробничих ризиків, профілактику професійних захворювань та загальний добробут працівників [19].

Основні завдання ергономіки:

1. Проектування людино-машинних систем, що враховують людські можливості та обмеження;
2. Оптимізація інтерфейсів та органів керування для забезпечення інтуїтивного й безпечного використання;
3. Мінімізація фізичного та психоемоційного навантаження на працівника;
4. Адаптація робочого середовища до індивідуальних антропометричних, фізіологічних та когнітивних характеристик людини;
5. Підвищення продуктивності праці через зменшення стомлюваності та створення комфортних умов.

Основні нормативи:

- ДСТУ EN 614-1:2018 — «Безпека машин. Ергономічні принципи проектування. Частина 1: Загальні принципи». Визначає загальні ергономічні

принципи, які потрібно враховувати при проектуванні безпечних машин і робочих місць [21].

- ДСТУ ISO 6385:2018 — «Основні принципи ергономіки при проектуванні систем». Встановлює загальні вимоги до врахування ергономічних факторів у робочому середовищі [22].

- ДСТУ ISO 7250-1:2018 — «Базові антропометричні вимірювання людини для технічного проектування». Забезпечує стандарти для проектування, які враховують антропометричні параметри населення.

- ДБН В.2.5-28:2018 — «Природне і штучне освітлення». Встановлює нормативи освітлення робочих зон [23].

- СанПіН 3.3.2.007-98 — норми роботи з відеотерміналами [24].

- СанР 173–96 — гранично допустимі рівні шуму на робочих місцях [25].

- Закон України «Про охорону праці»;

- Кодекс законів про працю України (КЗпП).

Основні ергономічні принципи проектування:

- Адаптація до людини, а не навпаки — умови праці повинні бути пристосовані до працівника.

- Антропометрична відповідність — розміри меблів та обладнання мають відповідати фізичним параметрам людини (наприклад, висота стола та стільця).

- Зони досяжності — елементи керування та інструменти мають розміщуватися в межах зручного доступу (рухи від ліктя та плеча).

- Інформаційна ергономіка — інформація (на екранах, у вигляді сигналів тощо) повинна бути легкою для сприйняття, логічною, читабельною та недвозначною.

- Зорове навантаження — екран повинен мати оптимальний кут нахилу та яскравість, рівень освітлення повинен бути не менше 300 лк.

- Психологічний комфорт — робоче середовище має бути спокійним, передбачуваним, без зайвих дратівливих факторів.

- Зниження навантаження — чергування видів діяльності (фізична, розумова), обов'язкові перерви, зміна положення тіла.
- Можливість індивідуального налаштування — висота стільця, нахил спинки, підлокітники, температура в приміщенні.
- Безпечна взаємодія з технікою — відсутність ризику травм при виконанні дій, аварійне вимикання, чіткі інструкції.
- Сприяння відновленню — ергономіка передбачає наявність зон відпочинку, вентиляції, зелених зон, можливості гімнастики.
- Наслідки недотримання ергономічних принципів:
- Професійні захворювання (остеохондроз, варикоз, тунельний синдром, короткозорість);
- Підвищення рівня помилок та аварійних ситуацій;
- Психоемоційне виснаження, стрес, депресивні стани;
- Зниження продуктивності та концентрації;
- Травматизм, особливо в умовах інтенсивної або монотонної праці.

Ергономіка — це елемент системи безпеки, яка охоплює організацію праці, проектування робочого місця, взаємодію людини з технікою та інформацією. Її мета — зниження професійних ризиків, підвищення продуктивності праці та формування здорового робочого середовища. Дотримання принципів ергономіки є обов'язковим як на етапі проектування, так і в процесі експлуатації робочого простору.

4.2 Значення автоматизації виробничих процесів в питаннях охорони праці.

Автоматизація виробничих процесів у сфері охорони праці передбачає впровадження технічних засобів, які забезпечують контроль, аналіз та реагування на відхилення у режимах роботи виробничого обладнання, умовах праці та

поведінці персоналу, з метою запобігання аварійним ситуаціям, нещасним випадкам і професійним захворюванням.

Згідно зі статтею 13 Закону України «Про охорону праці», роботодавець зобов'язаний впроваджувати сучасні засоби виробництва та технології з урахуванням вимог безпеки, гігієни праці та охорони здоров'я. Водночас, автоматизація повинна бути інтегрована не лише у виробничі процеси, а й у системи управління охороною праці, формуючи єдиний комплекс заходів технічного і організаційного характеру [26].

Проведені дослідження (А.П. Бочковський, Н.Ю. Сапожнікова, 2017) довели, що більшість виробничих травм в Україні мають організаційні (67%) та психофізіологічні (21%) причини, зокрема:

- невиконання інструкцій з охорони праці;
- перебування працівника у небезпечній зоні;
- незадовільний технічний стан обладнання.

Щоб мінімізувати вплив "людського фактора", запропоновано впроваджувати системи автоматизованого контролю і підвищення безпеки виробництв. До їх складу входять:

- датчики руху, деформації, температури, вологості тощо;
- відеоспостереження та фіксація дій працівників;
- електронні термінали для входу до системи і контролю доступу до обладнання;
- автоматичне блокування небезпечних ділянок та устаткування;
- світло-звукові сигнали тривоги;
- електронна база даних, де фіксуються порушення, інструктажі, медичні огляди тощо [27].

Впровадження таких автоматизованих систем дає змогу:

- зменшити кількість нещасних випадків за рахунок попередження порушень у реальному часі;
- підвищити рівень контролю за технічним станом будівель і устаткування;

- вести персоніфікований облік знань, інструктажів і дотримання правил охорони праці;
- формувати індивідуальні навчальні модулі для підвищення компетентності персоналу;
- своєчасно сигналізувати про небезпечні ситуації й відхилення від норм.

Крім того, автоматизація дозволяє реалізувати системний підхід до безпеки, передбачений у ДСТУ EN ISO 12100:2016 («Безпечність машин. Загальні принципи проєктування [29]. Оцінка ризику та зменшення ризику»), а також відповідає вимогам ДСТУ EN ISO 45001:2019 — міжнародного стандарту щодо систем управління охороною праці та безпекою на робочих місцях [30].

Такі системи інтегруються у виробництво шляхом застосування програмованих логічних контролерів (ПЛК), систем SCADA, сенсорних мереж та аналітики великих даних для виявлення тенденцій до порушень чи несправностей задовго до того, як вони стануть причиною аварії.

Отже, автоматизація охоплює створення комплексної адаптивної системи забезпечення безпечних умов праці, що поєднує моніторинг, аналіз, управління та навчання. Це відповідає міжнародному підходу до управління ризиками в промисловій безпеці, орієнтованому не лише на усунення наслідків, а й на попередження причин виникнення небезпек. У сучасному виробництві автоматизація виробничих процесів є необхідністю для збереження життя, здоров'я та працездатності працівників.

ВИСНОВКИ

Основні результати, які отримано в дипломній роботі такі:

- Побудовано математичні моделі, що описують дифузійні процеси у матеріалах із нано- та мікропорами (цеолітах). Проведено їх порівняння за придатністю до моделювання реальних технологічних умов, що дозволяє вибрати найбільш адекватні моделі для практичного застосування;
- Виконано дискретизацію моделей методом скінченних різниць для рівнянь дифузійного переносу та граничних умов, які описують баланс маси в досліджуваних зразках;
- Отримані дискретизовані моделі представлені у вигляді сіткових задач. Розроблено алгоритми їх розв'язання методами прямої та зворотної прогонки, реалізовані на мові C++ у середовищі Wolfram Mathematica, що забезпечує ефективність та точність обчислень;
- Розроблено архітектуру програмної системи, яка реалізує розглянуті сіткові задачі та включає інтерфейс користувача. Інтерфейс дозволяє зручно змінювати параметри математичних моделей, геометричні розміри зразків та фізико-хімічні властивості, що підвищує гнучкість та практичну користь системи;
- Проведено верифікацію математичних моделей та програмного забезпечення на основі експериментальних даних та тестування на продуктивність. Отримані результати підтвердили адекватність моделювання і реалістичність симуляцій.

Розроблена програмна система є ефективним інструментом для дослідників у галузі тепло- і масопереносу, а також для фахівців, що займаються розв'язанням прикладних технологічних задач. Вона дозволяє швидко отримувати необхідні параметри моделей, що значно спрощує і прискорює процес аналізу і проектування матеріалів із нано- та мікропорами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. L. Zheng, Q. Zhao, MJ. Adams. Molecular dynamics study of shale oil adsorption and diffusion behavior in reservoir nanopores: Impact of hydrocarbon composition and surface type // *Journal of Molecular Liquids*. 2024. №15. P. 126110.
2. J. Wang, K. Tian, Y. Li, W. Wang, H. Jin. Diffusion coefficients of polycyclic aromatic hydrocarbons in supercritical carbon dioxide: A molecular dynamics simulation study // *Journal of Molecular Liquids*. 2024. №409. P. 125457.
3. S. Shafieiyoun, A. Ling, B. Ramsay. Effects of size and composition of bitumen drops on intra-oil diffusion and dissolution of hydrocarbon solvents in froth treatment tailings ponds // *Chemosphere*. 2024. №362. P. 142540.
4. Z. Liu , Q. Wang , Y. Feng. The self-adjuvant heterocyclic lipid nanoparticles encapsulated with vaccine and STAT3 siRNA boost cancer immunotherapy through DLN-targeted and STING pathway // *Chemical Engineering Journal*. 2024. №475. P. 146474.
5. H. Abdelsalam, Q. Zhang. Spintronic properties of 2D heterostructures from laterally connected graphene and hBN quantum dots // *Chemical Physics Letters*. 2024. №825. P. 140591.
6. G-H. Kim, H-Y. Choi. Topological quantum oscillation of magnetization in the Josephson φ_0 junction // *Journal of Magnetism and Magnetic Materials*. 2024. №491. P. 165535.
7. W-P. Wu, Z-Z. Yao. Molecular dynamics simulation of stress distribution and microstructure evolution ahead of a growing crack in single crystal nickel // *Theoretical and Applied Fracture Mechanics*. 2012. №62. P. 67-75.
8. H. Yu, Q. Meng, Y. Ning. Microstructure control and DRX characteristics of Ni–Co–W superalloys affected by changing deformation direction on [001] columnar grain // *Journal of Materials Research and Technology* 2024. №33. P. 785-795.
9. М. Р. Петрик, І. В. Бойко, О. М. Хіміч, М. М. Петрик. Високопродуктивні суперкомп'ютерні технології моделювання та ідентифікації

складних нанопористих кіберсистем зі зворотними зв'язками для n-компонентної конкуритивної адсорбції // Кібернетика та системний аналіз. 2021. т. 57. №2. с. 170-183.

10. М. Р. Петрик, І. В. Бойко, О. М. Хіміч, М. М. Петрик. Високопродуктивні суперкомп'ютерні технології моделювання та ідентифікації складних нанопористих кіберсистем зі зворотними зв'язками для багатоконпонентної конкуритивної адсорбції // Кібернетика та системний аналіз. 2020. в. 5. с. 174-186.

11. М. Р. Петрик, О. М. Хіміч, М. М. Петрик. Моделювання адсорбції та десорбції вуглеводнів нанопористих каталізаторів систем нейтралізації вихлопних газів з використанням нелінійної ізотерми ленгмюра з урахуванням енергії активації // Міжнародний науково-технічний журнал Проблеми керування та інформатики. 2018. №5. с.59-72.

12. D. Liao , Q. Zhong, X. Hou. High-quality thickness-tunable InAs nanowire crosses grown by molecular-beam epitaxy // Vacuum. 2024. №230. P. 113657.

13. W-J. Lee, J. Seo, J.Cheol Shin. Interfacial characteristics dependence on interruption times in InGaAs/InAlAs superlattice grown by molecular beam epitaxy // Journal of Alloys and Compounds. 2024. №1006. P. 76297.

14. M. Yang, H. Ye, Y. Wang. Comparison of AlN/GaN heterojunctions grown by molecular beam epitaxy with Al and Ga assistance // Journal of Alloys and Compounds. 2024. №1008. P. 176559.

15. B. Di Pierro , S. Hank. CPU and GPU parallel efficiency of ARM based single board computing cluster for CFD applications // Computers & Fluids. 2024. №1272. P. 106187.

16. J.O. Takhirov, M.I. Boborakhimova. On the mathematical model of the concentration of pollutants and their impact on the population of the river // Results in Applied Mathematics. 2024. №21. P. 100414.

17. A. Alla, M. Hinze. HJB-POD feedback control of advection-diffusion equation with a model predictive control snapshot sampling // IFAC-PapersOnLine.2015. №23. P. 527-532.

18. В.С. Дейнека, І.В. Сергієнко. Оптимальне управління неоднорідними системами : Монографія. Київ: Наукова думка. 2003. 505 с.
19. Стищенко Т.Є., Пронюк Г.В., Сердюк Н.М., Хондак І.І. «Безпека життєдіяльності»: навч. посібник / Т.Є Стищенко, Г.В. Пронюк, Н.М. Сердюк, І.І. Хондак. – Харків: ХНУРЕ, 2018. – 336 с.
20. Електронне навчання ТНТУ. Курс “Безпеки життєдіяльності та охорони праці” [Електронний ресурс]. – 2025. – Режим доступу: <https://dl.tntu.edu.ua/index.php>
21. ДСТУ EN 614-1:2018. Безпека машин. Ергономічні принципи проектування. Частина 1: Загальні принципи від 18.07.2018 № 226 [Електронний ресурс]. – 2018. – Режим доступу: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=77654
22. ДСТУ ISO 6385:2018. Основні принципи ергономіки при проектуванні систем від 19.12.2018 № 513 [Електронний ресурс]. – 2018. – Режим доступу: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=80890
23. ДБН В.2.5-28:2018. Природне і штучне освітлення від 03.10.2018 № 264 [Електронний ресурс]. – 2018. – Режим доступу: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=79885
24. СанПіН 3.3.2.007-98. Гігієнічні вимоги до відеотерміналів персональних електронно-обчислювальних машин від 10.12.98 № 7 [Електронний ресурс]. – 2025. – Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0007282-98#Text>
25. СанР 173–96. Гранично допустимі рівні шуму на робочих місцях від 22.02.2019 № 463 [Електронний ресурс]. – 2025. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0281-19#Text>
26. Закон України «Про охорону праці» від 14.10.1992 № 2694-ХІІ [Електронний ресурс] / Верховна Рада України. – 2025. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/2694-12#Text>
27. Бочковський А. П. Формалізація системи автоматизованого контролю і підвищення безпеки виробництв / А. П. Бочковський, Н. Ю. Сапожнікова // Вісник

Львівського державного університету безпеки життєдіяльності. - 2017. - № 15. - С. 114-123. - Режим доступу: http://nbuv.gov.ua/UJRN/Vldubzh_2017_15_17.

28. Кодекс законів про працю України від 10.12.1971 № 322-VIII [Електронний ресурс] / Верховна Рада України. – 2025. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/322-08#Text>

29. ДСТУ EN ISO 12100:2016. Безпечність машин. Загальні принципи проєктування. Оцінка ризику та зменшення ризику від 13.12.2016 № 426 [Електронний ресурс]. – 2016. – Режим доступу: https://online.budstandart.com/ua/catalog/doc-page?id_doc=54659

30. ДСТУ EN ISO 45001:2019. Системи управління охороною здоров'я та безпекою праці. Вимоги та настанови щодо застосування від 26 грудня 2019 року № 506 [Електронний ресурс]. – 2019. – Режим доступу: https://online.budstandart.com/ua/catalog/doc-page?id_doc=87509

ДОДАТКИ

Додаток А - Лістинг програми

Введення геометричних та фізичних параметрів:

```

alpha = 5.0; Lx = 10; Ly = 10; Nx = 50; Ny = 50;
Tmax = 2;
Nt = 200;
dx = Lx/(Nx - 1);
dy = Ly/(Ny - 1);
dt = Tmax/Nt;
(*Коефіцієнти для схеми Кранка-Ніколсон*)
rx = alpha dt/(2 dx^2);
ry = alpha dt/(2 dy^2);
u = ConstantArray[0, {Nx, Ny}];
Do[If[EuclideanDistance[{i, j}, {Nx/2, Ny/2}] <= 10,
    u[[i, j]] = Exp[-0.1 EuclideanDistance[{i, j}, {Nx/2, Ny/2}]]], {i,
    Nx}, {j, Ny}];
applyBoundaryConditions[mat_] := Module[{m = mat},
    m[[1, All]] = 0; m[[-1, All]] = 0;
    m[[All, 1]] = 0;
    m[[All, -1]] = 0; m];
(*Основний цикл по часу*)
Do[(*Збереження старих значень концентрації*)uOld = u;
    rhs = Table[
        rx (uOld[[i + 1, j]] - 2 uOld[[i, j]] + uOld[[i - 1, j]]) +
        ry (uOld[[i, j + 1]] - 2 uOld[[i, j]] + uOld[[i, j - 1]]) +
        uOld[[i, j]], {i, 2, Nx - 1}, {j, 2, Ny - 1}];
    uNew = Table[(1 + 2 rx + 2 ry) u[[i, j]] -
        rx (u[[i + 1, j]] + u[[i - 1, j]]) -
        ry (u[[i, j + 1]] + u[[i, j - 1]]) + rhs[[i - 1, j - 1]], {i, 2,
        Nx - 1}, {j, 2, Ny - 1}];
    Do[u[[i, j]] = uNew[[i - 1, j - 1]], {i, 2, Nx - 1}, {j, 2, Ny - 1}];
    u = applyBoundaryConditions[u];
    If[Mod[n, 20] == 0,
        Print[ListPlot3D[u, PlotRange -> All,
            ColorFunction -> "TemperatureMap",
            AxesLabel -> {"x", "y", "Concentration"},
            PlotLabel -> "3D Distribution of Concentration", Boxed -> True,
            Mesh -> False]];
    Print[ContourPlot[
        u[[Round[x/dx] + 1, Round[y/dy] + 1]], {x, 0, Lx}, {y, 0, Ly},
        ColorFunction -> "TemperatureMap", Contours -> 20,
        FrameLabel -> {"x", "y"},
        PlotLabel -> "Contour Plot of Concentration"]];, {n, 1, Nt}]

```

Реалізація чисельної схеми Кранка-Ніколсона для 2D рівняння дифузії:

```

(*Параметри задачі*)alpha = 5.0;(*Збільшений коефіцієнт дифузії*)Lx = \
10; Ly = 10;(*Розміри області*)Nx = 50; Ny = 50;(*Більша кількість \
точок сітки*)Tmax = 2;(*Максимальний час моделювання*)Nt = \
200;(*Більше кроків за часом*)(*Кроки сітки*)dx = Lx/(Nx - 1);
dy = Ly/(Ny - 1);

```

```

dt = Tmax/Nt;

(*Коефіцієнти для схеми Кранка-Ніколсон*)
rx = alpha dt/(2 dx^2);
ry = alpha dt/(2 dy^2);

(*Ініціалізація матриці концентрації u*)
u = ConstantArray[0, {Nx, Ny}];

(*Початкові умови:декілька імпульсів у центрі області*)
Do[If[EuclideanDistance[{i, j}, {Nx/3, Ny/3}] <= 10 ||
     EuclideanDistance[{i, j}, {2 Nx/3, 2 Ny/3}] <= 10 ||
     EuclideanDistance[{i, j}, {Nx/2, Ny/2}] <= 10,
    u[[i, j]] = Exp[-0.1 EuclideanDistance[{i, j}, {Nx/2, Ny/2}]]], {i,
    Nx}, {j, Ny}];

(*Граничні умови (нульова концентрація на межах області*)
applyBoundaryConditions[mat_] :=
Module[{m = mat}, m[[1, All]] = 0; (*Лівий край*)
m[[-1, All]] = 0; (*Правий край*) m[[All, 1]] = 0; (*Верхній край*)
m[[All, -1]] = 0; (*Нижній край*) m];

(*Основний цикл по часу*)
Do[(*Збереження старих значень концентрації*) uOld = u;
  (*Проміжний крок:обчислення правої частини*)
  rhs = Table[
    rx (uOld[[i + 1, j]] - 2 uOld[[i, j]] + uOld[[i - 1, j]]) +
    ry (uOld[[i, j + 1]] - 2 uOld[[i, j]] + uOld[[i, j - 1]]) +
    uOld[[i, j]], {i, 2, Nx - 1}, {j, 2, Ny - 1}];
  (*Оновлення значень концентрації*)
  uNew = Table[(1 + 2 rx + 2 ry) u[[i, j]] -
    rx (u[[i + 1, j]] + u[[i - 1, j]]) -
    ry (u[[i, j + 1]] + u[[i, j - 1]]) + rhs[[i - 1, j - 1]], {i, 2,
    Nx - 1}, {j, 2, Ny - 1}];
  (*Вставка оновлених значень в основну матрицю*)
  Do[u[[i, j]] = uNew[[i - 1, j - 1]], {i, 2, Nx - 1}, {j, 2, Ny - 1}];
  (*Застосування граничних умов*) u = applyBoundaryConditions[u];
  (*Візуалізація концентрації у 3D-графіку та контурному перерізі*)
  If[Mod[n, 20] == 0,
    Print[ListPlot3D[u, PlotRange -> All,
      ColorFunction -> "TemperatureMap",
      AxesLabel -> {"x", "y", "Concentration"},
      PlotLabel -> "3D Distribution of Concentration", Boxed -> True,
      Mesh -> False]];
    Print[ContourPlot[
      u[[Round[x/dx] + 1, Round[y/dy] + 1]], {x, 0, Lx}, {y, 0, Ly},
      ColorFunction -> "TemperatureMap", Contours -> 20,
      FrameLabel -> {"x", "y"},
      PlotLabel -> "Contour Plot of Concentration"]];, {n, 1, Nt}]

```

Розв'язання рівняння дифузії з використанням двонапрямної прогонки:

```

(*Визначення параметрів*) Manipulate[L = 1; (*Довжина області*)
T = 0.1; (*Часова межа*) nx = nxVal; (*Кількість вузлів по x*)
nt = ntVal; (*Кількість часових кроків*) dx = L/nx;
dt = T/nt;
alpha = 0.01; (*Коефіцієнт дифузії*) r = alpha dt/dx^2;

```

```

(*Початкові та граничні умови*)u = Table[0, {i, 0, nx}, {j, 0, nt}];
Do[u[[i + 1, 1]] = Sin[Pi i dx], {i, 0, nx}];
Do[u[[1, j + 1]] = 0; u[[nx + 1, j + 1]] = 0, {j, 1, nt}];
(*Пряма прогонка*)A = Table[0, {i, 1, nx - 1}, {j, 1, nx - 1}];
B = Table[0, {i, 1, nx - 1}];
Do[Do[A[[i, i]] = 1 + 2 r; If[i > 1, A[[i, i - 1]] = -r];
  If[i < nx - 1, A[[i, i + 1]] = -r], {i, 1, nx - 1}];
Do[B[[i]] = u[[i + 1, j]], {i, 1, nx - 1}];
sol = LinearSolve[A, B];
Do[u[[i + 1, j + 1]] = sol[[i]], {i, 1, nx - 1}, {j, 1, nt}];
(*Зворотна прогонка*)Do[sol = LinearSolve[A, B];
  Do[u[[i + 1, j + 1]] = sol[[i]], {i, 1, nx - 1}, {j, nt, 1, -1}];
(*Візуалізація результату*)
Grid[{{ListPlot3D[u, Mesh -> All, ColorFunction -> "TemperatureMap",
  AxesLabel -> {"x", "t", "u"}],
  ListLinePlot[Transpose[u][[pereriz]], Mesh -> All,
  PlotStyle -> Thick, AxesLabel -> {"x", "u"}],
  ContourPlot[
  u[[Round[x nx] + 1, Round[t nt] + 1]], {x, 0, 1}, {t, 0, T},
  Contours -> 20, ColorFunction -> "TemperatureMap",
  AxesLabel -> {"x", "t"}]}],
{{nxVal, 10, "nx"}, 5, 50, 1}, {{ntVal, 100, "nt"}, 50, 500,
10}, {{pereriz, 10, "t-slice"}, 1, nt, 1}}

```

Інтерактивний інтерфейс для візуалізації параметрів моделі та результатів симуляції:

```

Manipulate[Module[{result, poreVolume, surfaceArea,
adsorptionCapacity,
  thermalConductivity}, poreVolume = porosity*unitCellVolume;
surfaceArea = 2 length width + 2 width height + 2 length height;
adsorptionCapacity = surfaceArea*adsorptionFactor;
thermalConductivity = baseThermalConductivity*(1 - porosity);
result = Switch[component, "Структурні властивості",
  "Цеоліт ZSM-5: Структурні характеристики\n" <>
  "Розміри -> Довжина: " <> ToString[length] <> ",
  Ширина: " <> ToString[width] <> ", Висота: " <>
  ToString[height] <> "\n" <>
  "Пористість: " <> ToString[porosity] <> ",
  Об'єм пор: " <> ToString[poreVolume], "Сорбційні характеристики",
  "Цеоліт ZSM-5: Сорбційні параметри\n" <> "Площа поверхні:
  " <> ToString[surfaceArea] <> "\n" <> "Фактор адсорбції: " <>
  ToString[adsorptionFactor] <> "\n" <>
  "Ємність адсорбції: " <> ToString[adsorptionCapacity],
  "Теплопровідність",
  "Цеоліт ZSM-5: Теплофізичні параметри\n" <>
  "Базова теплопровідність: "
  <> ToString[baseThermalConductivity] <> "\n" <>
  "Теплопровідність з урахуванням пористості: " <>
  ToString[thermalConductivity]];
result], (*Геометричні параметри*){{length, 1, "Довжина (нм)"}, 0.1,
10},
, {{width, 1, "Ширина (нм)"}, 0.1, 10}, {{height, 1, "Висота (нм)"},
0.1, 10},
{{porosity, 0.3, "Пористість"}, 0.1, 0.9},

```

```

{{unitCellVolume, 0.5, "Об'єм елементарної комірки (нм\.3)"}, 0.1,
 2.0},
{{adsorptionFactor, 10, "Фактор адсорбції"}, 1, 50},
{{baseThermalConductivity, 0.2,
  "Базова теплопровідність (Вт/м\.[CenterDot]К)"}, 0.1,
 1.0}, {component, {"Структурні властивості",
  "Сорбційні характеристики", "Теплопровідність"},
ControlType -> PopupMenu}]

```

Додаток Б - Тези доповіді на конференції

Міністерство освіти і науки України
 Тернопільський національний технічний університет
 імені Івана Пулюя
 Маріборський університет (Словенія)
 Технічний університет в Кошице (Словаччина)
 Каунаський технологічний університет (Литва)
 Львівський національний університет
 імені Івана Франка,
 Гірничо-металургійна академія ім. Станіслава Сташиця (Польща)
 Луцький національний технічний університет,
 Чернівецький національний університет
 імені Юрія Федьковича,
 Вроцлавський економічний університет (Польща)
 Університет технологій та економіки
 імені Хелени Ходковської (Польща)
 Донбаська державна машинобудівна академія



*Студентське наукове
товариство*



VIII МІЖНАРОДНА

студентська науково - технічна конференція

"ПРИРОДНИЧІ ТА ГУМАНІТАРНІ НАУКИ.

АКТУАЛЬНІ ПИТАННЯ"

24-25 квітня 2025 р.

(збірник тез конференції)

Тернопіль 2025

УДК 004.942

Гикава В.–ст. гр. СП-41

Тернопільський національний технічний університет імені Івана Пулюя

ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ПРОЦЕСІВ ДИФУЗІЇ В РІЗНОМАНІТНИХ ЗРАЗКАХ ЦЕОЛІТУ

Науковий керівник: к.ф.-м.н., доцент Бойко І. В.

Hykava V.

Ternopil Ivan Puluj National Technical University

SOFTWARE COMPLEX FOR MATHEMATICAL MODELING OF DIFFUSION PROCESSES IN VARIOUS ZEOLITE SAMPLES

Supervisor: Ph.D. in Physics and Mathematics, associate professor Boyko I.V.

Ключові слова: обернена задача, математична модель, ідентифікація параметрів дифузії
Keywords: inverse problem, mathematical model, identification of diffusion parameters

Моделювання процесів дифузії у багатошарових матеріалах, таких як цеоліти та силікагель, має важливе значення для вирішення завдань у сферах інженерії, хімії та енергетики. Ці матеріали вирізняються своєю мікропористою структурою, яка впливає на ефективність тепломасоперенесення. Детальне дослідження дифузійних процесів є ключем до вдосконалення технологій сорбції, каталізу, мембранного розділення та енергетичного зберігання. Водночас, класичні методи моделювання часто стикаються з обмеженнями при аналізі таких складних систем. Це зумовлює необхідність створення універсальних математичних моделей і програмних систем, які дозволять дослідникам точно прогнозувати властивості матеріалів.

Для вирішення цих завдань були розроблені математичні моделі, які враховують особливості багатошарових структур. В основі дослідження лежать рівняння дифузії, що описують тепломасоперенесення у матеріалах із різною пористістю. Використано числові методи, такі як методи прямої та зворотної прогонки, для розв'язання задач із високою точністю та ефективністю. Важливим результатом роботи стало створення програмного комплексу, який дозволяє інтерактивно моделювати геометричні параметри матеріалів, налаштовувати фізико-хімічні характеристики і виконувати графічну візуалізацію 2D та 3D розподілу концентрацій або тепла в зразках. Система зручна для користувачів та адаптована до широкого спектру інженерних і дослідницьких завдань.

У рамках роботи проведено тестування розробленого програмного комплексу на різних типах матеріалів, таких як цеоліти різних класів (ZSM-5, ZSM-11, ZSM-12) та силікагель. Виконано аналіз концентраційних розподілів, просторових залежностей і перерізів для багатошарових систем з пористістю різного ступеня. Зокрема, досліджено ефективність процесів дифузії в цеолітах, які виявилися значно вищими порівняно із силікагелем. Ці експерименти дозволили оцінити реалістичність отриманих результатів, підтвердити високу продуктивність розробленої системи та її потенціал для використання в галузях науки і промисловості.

Додаток В - Диск з роботою