

Міністерство освіти і науки України
Тернопільський національний технічний університет
імені І. Пулюя

Ігор БОЙКО, Михайло ПЕТРИК
**МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПРОЦЕСІВ ТА ОБ'ЄКТІВ
ПРЕДМЕТНИХ ОБЛАСТЕЙ. МОДЕЛЮВАННЯ У СЕРЕДОВИЩІ
WOLFRAM MATHEMATICA**
Навчально-методичний посібник

Тернопіль
Видавництво ТНТУ

УДК

ББК

Рецензент

Ясній Олег Петрович, доктор технічних наук, професор

Навчальний посібник розглянуто і затверджено на засіданні науково-методичної ради факультету комп'ютерно-інформаційних систем і програмної інженерії Тернопільського національного технічного університету імені Івана Пулюя.
Протокол № 1 від 02.09.2024 р.

Ігор Бойко, Михайло Петрик.

Математичне моделювання процесів та об'єктів предметних областей. Моделювання у середовищі Wolfram Mathematica): навчальний посібник. Тернопіль: : ТНТУ 2024 – 110 с.

Даний посібник написано згідно програми предметів “Математичне моделювання процесів та об'єктів предметних областей” та “Моделювання складних розподілених систем та об'єктів”, що читаються на факультеті комп'ютерно-інформаційних систем і програмної інженерії.

Для студентів спеціальності 121 – “Інженерія програмного забезпечення”, аспірантів та викладачів вищих навчальних закладів.

© Ігор Бойко, Михайло Петрик 2024

ЗМІСТ

Зміст.....	3
Вступ.....	5
РОЗДІЛ 1.	
РОБОТА В СИСТЕМІ MATHEMATICA ТА ІНТЕРФЕЙС.....	8
1.1. Історія появи системи Mathematica та її місце у застужанні до прикладних задач.....	8
1.2. Інсталяція і запуск системи Mathematica	9
1.3. Головне меню.....	10
1.4. Палітри математичних операторів і функцій.....	12
1.5. Перші навички роботи та поняття про ноутбук як робоче поле в системі Mathematica.....	13
1.6. Основні види файлів і пакети розширення.....	15
1.7. Команди позиції File меню.....	15
1.8. Робота з файлами документів.....	17
1.9. Команда завершення роботи з системою – Exit.....	17
1.10. Основні поняття про документи і їх стилях.....	18
1.11. Виділення в документах і використання миші.....	18
1.12. Підготовка текстових коментарів.....	20
1.13. Поняття про комірки документів.....	21
1.14. Команди опції Cell головного меню.....	22
1.15. Команди позиції Evaluation головного меню.....	23
1.16. Управління процесом обчислень.....	23
1.17. Вибір ядра системи.....	24
1.18. Динамічна інтерактивність при символічних обчисленнях.....	24
1.19. Управління графіками з допомогою миші.....	26
1.20. Динамічна інтерактивність при графічній візуалізації.....	28
РОЗДІЛ 2. MATHEMATICA ЯК СИСТЕМА ПРОГРАМУВАННЯ.....	32
2.1. Поняття про вхідну мову системи і мову реалізації.....	32
2.2. Можливості мови програмування системи Mathematica.....	32
2.3. Структура систем Mathematica.....	35

2.4. Ідеологія систем Mathematica.....	36
2.5. Концепція динамічної зміни змінних в Mathematica.....	37
2.6. Функції символьних обчислень. Поняття про аналітичні обчислення.....	39

ПРИКЛАДИ ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

ЛАБОРАТОРНА РОБОТА № 1.	
"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МОДЕЛЮВАННІ ПРОЦЕСІВ КОЛИВАНЬ ФІЗИЧНИХ ОБ'ЄКТІВ".....	42
ЛАБОРАТОРНА РОБОТА № 2	
"МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ДРУГОГО І ТРЕТЬОГО ПОРЯДКУ З ПЕРЕТВОРЕННЯМ ПРОДУКТУ".....	51
ЛАБОРАТОРНА РОБОТА № 3	
"МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ НЕЛІНІЙНОГО ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ДИФУЗІЇ /НАГРІВУ/ СУСПЕНЗІЙ ЗІ ЗМІННИМ КОЕФІЦІЄНТОМ ДИФУЗІЇ /ТЕПЛОПРОВІДНОСТІ/, ЩО ВРАХОВУЄ ВІДХИЛЕННЯ РЕАЛЬНОГО ПРОЦЕСУ ВІД КЛАСИЧНОГО".....	60
ЛАБОРАТОРНА РОБОТА № 4	
"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ У МОДЕЛЮВАННІ ЕНЕРГЕТИЧНОГО СПЕКТРУ ЗАКРИТИХ НАПІВПРОВІДНИКОВИХ НАНОСТРУКТУР".....	69
ЛАБОРАТОРНА РОБОТА № 5	
"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МОДЕЛЮВАННІ ПРОЦЕСІВ КОЛИВАНЬ ФІЗИЧНИХ ОБ'ЄКТІВ".....	76
ЛАБОРАТОРНА РОБОТА № 6	
"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ У МОДЕЛЮВАННІ КОЕФІЦІЄНТА ПРОЗОРОСТІ ТА ЕНЕРГЕТИЧНОГО СПЕКТРУ ДВОБАР'ЄРНИХ ВІДКРИТИХ НАПІВПРОВІДНИКОВИХ НАНОСТРУКТУР".....	85
ЛАБОРАТОРНА РОБОТА № 7	
"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ В ЧАСТИННИХ ПОХІДНИХ У МОДЕЛЮВАННІ ЯВИЩА ПЕРЕНЕСЕННЯ ТЕПЛА У НАГРІТИХ СТЕРЖНЯХ МЕТОДОМ ФУР'Є".....	99
ЛАБОРАТОРНА РОБОТА № 8	
"СИСТЕМИ НЕЛІНІЙНИХ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ЛОРЕНЦА У МОДЕЛЮВАННІ ГЕНЕРАТОРІВ ДИНАМІЧНОГО ХАОСУ".....	106

ВСТУП

Перші спроби створення комп'ютерних програм для виконання аналітичних обчислень були зроблені ще на перших лампових ЕОМ. Проте їх технічні можливості зумовлені мізерно малим обсягом пам'яті і низькою швидкістю обчислень не дозволили цьому напрямку розвиватися і дати реальні практичні, а не тільки абстрактні, результати.

Говорити про появу реальних систем комп'ютерної алгебри стало можливим лише після появи ЕОМ третього покоління класу «Мир-2» і «Мир-3», які були створені науковою школою академіка В. М. Глушкова. Дані ЕОМ, також мали досить скромні технічні характеристики, зате використовували мову програмування Аналітик, яка підтримує аналітичні обчислення [1].

Масовий розвиток комп'ютерна алгебра отримала після появи в 80-90-х роках минулого століття в 80-90-их роках минулого століття мікрокалькуляторів і персональних комп'ютерів, виконаних на напівпровідникових інтегральних схемах великої ступеня інтеграції (ВІС) [2]. В цей час вже з'явилися перші графічні калькулятори, що були орієнтовані на наукові розрахунки й моделювання з використанням систем символьних обчислень [7,8] і перші книги, що описували методи персональних масових обчислень і комп'ютерної алгебри [2-7, 9, 10]. З'явилося також і безліч розробок комерційних систем комп'ютерної математики (СКМ), добре описаних у великій кількості відповідної літератури [8-21,26-33].

Безпосереднім мотивом для підготовки пропонованої книги стала поява нових версій системи Mathematica – Mathematica 8.0, 9.0 і 10.0 та цілковита відсутність в нашій літературі книг по цих принципово нових реалізаціях цієї системи. На жаль, як старші так і сучасні версії системи Mathematica в українській навчальній літературі описані дуже мало. Варто виділити єдину книгу [] орієнтовану на використання розрахунків та моделювання фізичних процесів у системі Mathematica 5.0, що зі зрозумілих причин є вже морально застарілою. Також слід зауважити, що не зважаючи на постійно зростаючу потребу у навчанні майбутнього програміста дисциплін, пов'язаних з

використанням (СКМ) методичному та науковому забезпеченню такого навчання поки що приділяється мала увага.

Система Mathematica спочатку була лідером серед систем комп'ютерної алгебри. Версії Mathematica 8.0, 9.0, як і 5.0, 6.0 десять років тому, сьогодні є найбільш відомими і використовуваними серед користувачів системами цього класу, є спрямованими на збереження цього лідерства та розширення сфери застосування програмних продуктів даного типу. Слід зауважити, що хоча також існує проміжна версія Mathematica 7.0, між старими версіями 5.0, 6.0 і новими 8.0, 9.0 існують значні відмінності та вдосконалення, які необхідно було б висвітлити. Також істотним є відсутність книг з основ програмування в системах класу Mathematica.

Сучасні версії Mathematica 8.0, 9.0 - це вже не просто лідер. Це дійсно революційний програмний продукт з величезним числом новацій, які висувують систему в особливе становище, і що надає їй можливості, які, мають значні переваги порівняно з іншими системами. Це стосується новацій інтерфейсу документів системи, високій швидкості обчислень і величезного набору корисних функцій, а також, що дуже корисно, інтерфейсу користувача.

Особливістю цієї книги є те, що вона в першу чергу орієнтована на програмістів і описує систему Mathematica як потужний професійно-орієнтовану на математичні і науково-технічні розрахунки мову програмування надвисокого рівня. Величезне число вбудованих засобів обчислень дозволяє розв'язувати переважну більшість завдань на мові Mathematica, сильно наближеної до звичайної мови математичних обчислень та мови C++ - базової мови сучасного програміста. Тому користувач зможе програмувати в середовищі Mathematica, фактично, починаючи з перших сторінок книги. Проте книга буде корисною також і звичайному користувачеві, навіть не знайомому з програмуванням.

За кордоном система Mathematica широко застосовується не тільки в науково-технічних і математичних розрахунках, а також і при викладанні багатьох дисциплін в університетах, та коледжах [58-76]. Можливості

візуалізації обчислень у графічному та динамічному вигляді різко розширюють можливості останніх версій Mathematica в моделюванні, сфері точних і технічних наук, в сфері освіти.

Як уже відзначалося, з розширенням сфери діяльності сучасного програміста вміння використання сучасних систем комп'ютерної математики має велике значення. Цей пуста нішу в процесі навчання програміста і покликаний заповнити цей підручник.

Матеріал даної книги подається таким чином, що основна його частина може використовуватися користувачами будь-якої версії системи Mathematica, починаючи з перших її глав, а також вивчаючи лише спеціалізовані необхідні для навчання чи роботи глави. У кінці підручника приведений зміст та приклади виконання лабораторних робіт, пропонованих студентам спеціальності 121 – Інженерія програмного забезпечення під час вивчення предмету “Диференціальні рівняння у математичних моделях”, що демонструє призначення матеріалу, викладеного в книзі.

Автор вдячний своїм колегам з кафедри програмної інженерії ТНТУ ім. Івана Пулюя за цінну допомогу та зауваження при обговоренні рукопису книги.

РОЗДІЛ 1

РОБОТА В СИСТЕМІ MATHEMATICA ТА ІНТЕРФЕЙС КОРИСТУВАЧА

1. Історія появи системи Mathematica та її місце у застужванні до прикладних задач

Система Mathematica 1, яка з'явилася на початку 1988 р, стала першою серйозною системою комп'ютерної алгебри [22]. Наступна версія Mathematica 2 могла вже працювати під операційною системою MS-DOS і оболонкою Windows 3.0 [23-26]. Системи викликали жвавий інтерес з боку учнів, програмістів, викладачів вузів і університетів, аспірантів, інженерів і науковців у всьому світі.

Версії Mathematica 3/4 [26,27] забезпечили системі лідируюче місце серед систем комп'ютерної математики кінця XX століття - початку XXI століття. Величезне число розв'язаних математичних, фізичних і науково-технічних завдань [34-57] пов'язане з застосуванням саме цих версій системи. Число операторів і функцій в ядрі систем в цей час було доведено до більш ніж 1000, в пакетах розширення до 800.

Однак на початку XXI століття Mathematica стала відчувати сильну конкуренцію з боку інших систем комп'ютерної математики. В області комп'ютерної алгебри в лідери швидко пробилося система Maple [20, 21], що була створена спочатку для застосування до кола вузьких задач в університетських колах і швидко розвивається.

Серед систем для чисельних розрахунків та моделювання провідне місце зайняла потужна матрична система MATLAB з пакетом блочного математичного моделювання Simulink. Не зважаючи на значні можливості даної системи, в повному обсязі реалізувати усі її можливості можливо лише та сучасних мультипроцесорних ПК та обчислювальних кластерах.

Велику популярність здобула також і система Mathcad з її інтерфейсом, орієнтованим на розв'язання математичних задач і скрупульозним відбором функцій, входячих до її ядра [10,11]. Мінусом даної системи є її дуже вузька орієнтованість, що практично нівелює її можливості у розв'язанні складних задач фізики, моделювання фізико-технологічних процесів.

У зв'язку з вищесказаним цим лідируюче місце системи Mathematica в країнах пострадянського простору вже здавалось не таким беззаперечним. Нерідко починаючі користувачі відмовлялись від використання Mathematica просто тому, що їм був погано зрозумілою мова функціонального програмування цієї системи, а також їх лякала незвичайність фіксації введення клавішами Shift + Enter, незвичайне задання імен функцій, наприклад Tan[x], із зазначенням параметрів в квадратних дужках, і інші особливості мови програмування, успадковані від мови C. Крім того перші версії Mathematica мали мале число практичних прикладів застосування системи. Користувачам часто здавалося, що система слабо і довго модернізується, оскільки зміна номера версії у розробників системи (фірма Wolfram Research) займала кілька років, тоді як інші системи-конкуренти оновлювалися практично і щорічно. Це видно по номерам їх останніх версій, наприклад, Maple 14 і Mathcad 14.

Згадані недоліки системи Mathematica нерідко стають в подальшому її перевагами. Так, великий час між розробками версій Mathematica 6 та 7 і 8, 9 на практиці свідчить про їх ретельного доопрацювання. Особливо це стосується динамічної зміни змінних, інтерактивного динамічного інтерфейсу і інші дуже корисні та цікаві можливості, що мають активне застосування починаючи з сьомої версії системи.

Крім того, ще важливіше те, що нові реалізації Mathematica 8, 9- це дійсно універсальні математичні системи, що однаково швидко, ефективно і надійно виконують як аналітичні, так і чисельні обчислення. Дані системи завантажуються набагато швидше інших систем. Є всі підстави уважати, що Mathematica дійсно виконує обчислення швидше своїх конкурентів, зокрема, завдяки вперше реалізованій підтримці можливостей сучасних надшвидкісних мікропроцесорів - в тому числі багатоядерних та за рахунок багатопотокових обчислень.

2. Інсталяція і запуск системи Mathematica 5

Інсталяція системи Mathematica 9 нічим не відрізняється від інсталяції будь-якої програми в операційній системі Windows. Після інсталяції на робочому

стілі з'являється ярлик, активізація якого призводить до завантаження системи в пам'ять комп'ютера та появи вікон системи (рис. 2.1).



Рис. 2.1. Огляд початкового інтерфейсу системи Mathematica 9

Спочатку після запуску Mathematica з'являються панель меню і велике вікно робочого ноутбука. На рис. 2.1 показано також вікно з віртуальною книгою по роботі з системою Mathematica та навігатор вбудованих функцій ядра (Kernel). Усі вікна закриваються активізацією кнопки зі знаком «х».

3. Головне меню

Панель головного меню має всього два рядки:

- з назвами системи і завантаженого файлу;
- позиціями головного меню.

Праворуч і знизу вікна редагування містяться лінійки прокрутки з характерними повзунками, що можуть бути керованими мишею. У самому низу на початку лінійки прокрутки є так звана статусний рядок з інформацією про поточний режим роботи (Status bar).

Головне меню системи (рис. 2.1) містить такі вкладки:

- File - робота з файлами: створення нового файлу, вибір існуючого файлу з каталогу, закриття файлу, збереження поточного файлу, збереження файлу зі зміною імені, друк документа і вихід в Windows;

- Edit – виконання основних операцій редагування (відміна операції, копіювання виділених частин документа в буфер з їх подальшим видаленням і без нього, перенесення виділених частин, їх стирання);
- Insert – завдання елементів введення (графіків, матриць, гіперпосилань, додавання у робочий ноутбук елементів файлів, вибір кольору робочої комірки та нумерація комірок);
- Format - установка формату для документів;
- Cell - робота з функціональними комірками (об'єднання і роз'єднання комірок, установка статусу комірки, відкриття і закриття);
- Evaluation - управління ядром системи та його конфігурацією;
- Palettes – робота з палітрами математичних операторів та функцій, засоби введення математичних символів та їх опції;
- Window - операції з вікнами і їх розташуванням;
- Help - управління засобами довідкової системи.

Кожен елемент меню, при активації, відриває випадне підменю, що містить пов'язані з ним команди. Назви виконуваних команд виділяються чітким, а ті, які не виконуються в даний час - характерним сірим розпливчастим шрифтом.

Елементи інтерфейсу, зокрема вікно редагування, можна переміщати за допомогою миші та розтягувати в різні боки. Курсор миші зазвичай має вигляд стрілки, проте змінюється при становленні на окремі деталі елементів інтерфейсу. Наприклад, при установці на вертикальну межу вікна він набуває вигляду двосторонніх стрілок $\leftrightarrow$, розташованих по горизонталі. Вони вказують на можливість переміщення цієї лінії по горизонталі. Аналогічно можна розтягувати або стискати вікно переміщенням по вертикалі або діагоналі.

На початку титульних рядків головного меню і вікна редагування є кнопка з логотипом системи, що відкриває підменю з наступними командами:

- Відновити - відновити розміри елемента інтерфейсу;
- Перемістити - перемістити елемент інтерфейсу;
- Розмір - задати розміри елемента інтерфейсу;

- Згорнути - згорнути елемент в бирку в панелі завдань Windows;
- Розгорнути - розгорнути елемент інтерфейсу;
- Закрити - закрити елемент інтерфейсу.

Це підменю створюється засобами операційної системи Windows. Також, в кінці цих рядків є характерні кнопки, що повторюють три останні команди. Вони служать для управління вікнами відповідних елементів інтерфейсу.

4. Палітри математичних операторів і функцій

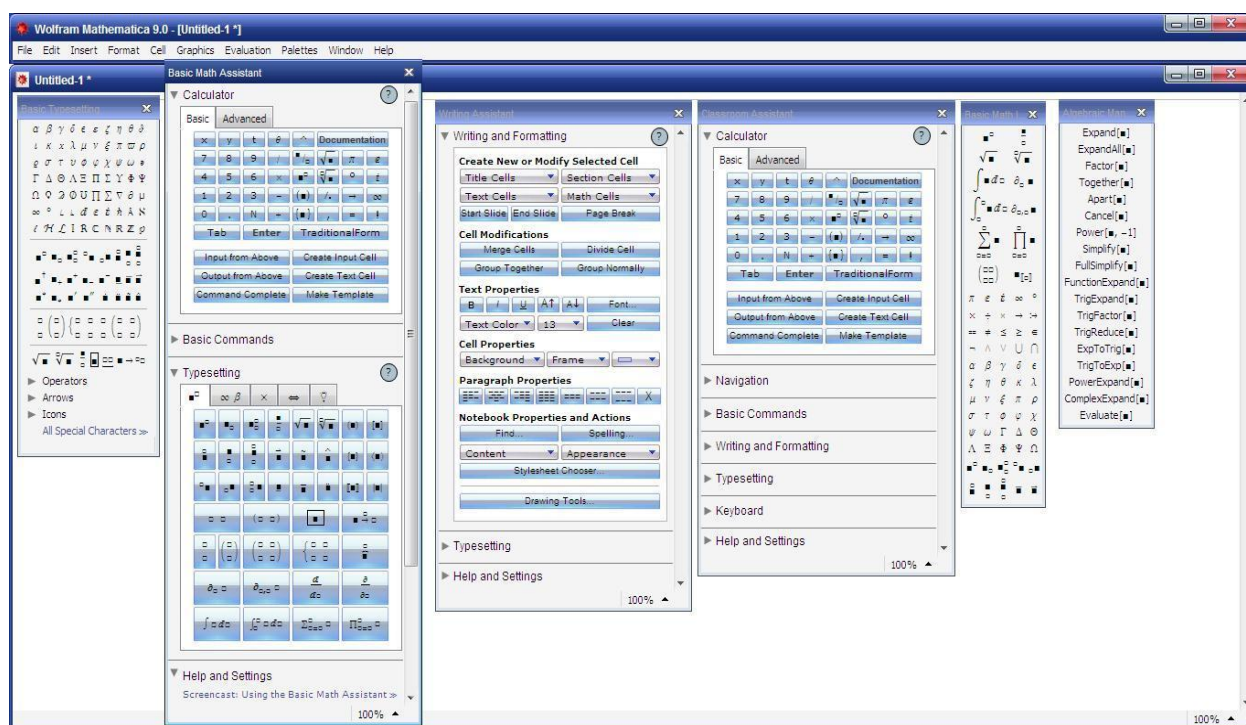


Рис. 4.1. Вікно системи Mathematica 9 з відритими палітрами та засобами введення алгебраїчних виразів і дій з ними

Для полегшення введення математичних виразів Mathematica містить виводжуванні користувачем та переміщувані по екрану в будь-яке його місце інструментальні панелі з безліччю піктограм введення математичних символів, функцій і команд управління системою. Вони виводяться за допомогою підменю Palettes головного меню системи (рис. 2.1).

Загальна кількість спеціальних математичних знаків (грецьких і латинських букв, операторів, функцій і команд), що вводяться за допомогою палітр, більше 700. Багато операцій та символів мають альтернативні варіанти їх введення з застосуванням комбінацій клавіш - їх можна знайти в

довідковій базі даних системи. Доцільно користуватися не більше ніж 2-3 панелями одночасно (наприклад загальною палітрою Basic Math Assistant). При необхідності видалення непотрібних панелей в правому верхньому куті кожної панелі розташовані маленькі кнопки зі знаком «х». Всі панелі максимально компактні і можуть бути переміщеними за допомогою миші в найбільш зручне місце екрана.

5. Перші навички роботи та поняття про ноутбук як робоче поле в системі Mathematica

Робота з документами зводиться до набору в комірках виразів (наприклад, математичних) і їх виконання. Для виконання виразів досить натиснути комбінацію клавіш **Shift** + **Enter** одночасно (оскільки клавіша **Enter** використовується тільки для падання перекладу рядка всередині поточного рядка введення). Наприклад, щоб обчислити суму $5 + 6$, необхідно спочатку ввести цей вислів в рядок введення. У Mathematica рядок введення формується по мірі введення об'єктів для виразів 5 і 6. Після натискання клавіш **Shift** і **Enter** отримаємо:

In[1]: = $5 + 6$

Out [1] = 11.

Слід зауважити, що при першому обчисленні Mathematica виконує його з відчутною затримкою. Це пов'язано із завантаженням ядра системи (Wolfram Mathematica Kernel). Подальші обчислення відбуваються практично миттєво.

Документ системи Mathematica своїми рядками (комірками) введення і виведення, текстовими коментарями, малюнками дуже нагадує сторінку блокнота вченого або інженера (рис. 5.1). Саме тому він і називається ноутбуком (від англійського notebook). Ноутбуки зберігаються з розширенням ***назва файлу*.nb**.

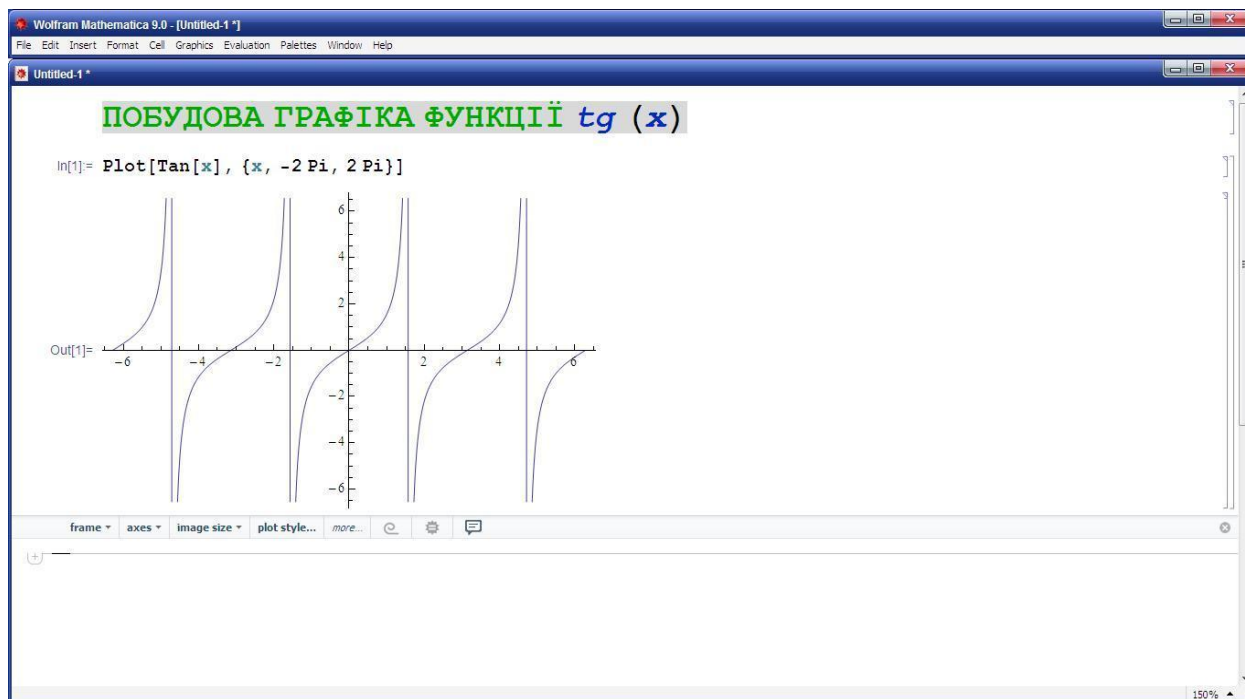


Рис. 5.1. Вікно системи Mathematica 9 відкритим ноутбуком та прикладом побудови графіка

Окремі комірки з математичними виразами і результатами їх обчислень позначаються в правій частині головного вікна редагування характерними тонкими квадратними дужками синього кольору. Це дозволяє відстежувати те, до чого відносяться математичні вирази - до вихідних даних або отриманих результатів. Крім того, комірки обчислень можуть мати різний статус, який позначається відповідними значками над квадратними дужками.

Для того, щоб документ мав наочний вид блокнота (ноутбука), необхідно зробити певні операції з форматування документа і надання йому потрібного вигляду. Перш за все, кожен крок обчислень слід супроводжувати пояснювальними написами. Вони можуть міститися в рядках введення, але потім їх необхідно відформатувати згідно текстового формату відповідного стилю. Для цього виділяється рядок введення з текстовим написом. Простір усередині дужки при затемнюється. Потім виконується команда **Format - Style - Text** (Alt + 7). Вона задає текстовий формат напису, який є не виконуваним.

За допомогою інших команд опції **Format** головного меню, можна задати напис різним шрифтом, різним кольором з виділенням фону. Як уже зазначалося, для введення математичного виразу згідно шаблону і для

представлення його в природній математичній формі використовується стандартний формат StandardForm комірок введення.

У робочих блокнотах бажано, щоб форма подання математичних виразів хоча б частково нагадувала загальноприйняту. Mathematica дозволяє задавати форми подання документів, прийняті в таких потужних мовах програмування, як Fortran, C і навіть TeX (мова для програмування засобів типографічного набору складних наукових текстів, що активно використовується в наукових журналах).

Кожен напис, математичний вираз або графік займають окрему комірку - Cell. Комірка може займати один або кілька рядків і завжди є виділеним своєю квадратною дужкою. Важливою властивістю комірок усіх систем Mathematica є можливість їх еволюції по всьому робочому документу. Таким способом здійснюється динамічний обмін даними в ході символічних перетворень.

РОБОТА З ФАЙЛАМИ (FILE)

6. Основні види файлів і пакети розширення

У нових версіях Mathematica 8/9 основним типом документів є блокноти - notebooks. Їм відповідають файли текстового формату з розширенням **.nb**. Ці файли можуть редагуватися будь-яким текстовим редактором, що підтримує формат ASCII. Файли містять докладний опис документа з вказівками типів шрифтів, деталей оформлення та розташування опрацьовуваних об'єктів.

Крім того, система має ряд пакетів розширення (в оригіналі доповнення -AddOn) системи, розташованих в каталозі ADDON. Дані пакети розширення містять значну кількість бібліотечних файлів з розширеннями .m, в кожному з яких визначено ряд нових функцій системи. З їх допомогою можна реалізувати нові алгоритми розв'язання математичних задач і постійно розширювати можливості системи.

7. Команди позиції File меню

Для роботи з файлами служить опція File меню. Вона дозволяє задавати такі команди:

- **New (Ctrl + N)** - відкривання вікна нового документа;
- **Open (Ctrl + O)** - відкривання вікна завантаження документа;
- **Close (Ctrl + F4)** - закриття поточного вікна;
- **Save (Ctrl + S)** - запис документа з поточним ім'ям;
- **Save As ... (Shift + Ctrl + S)** - запис документа зі зміною імені;
- **Save Selection As** – збереження виділеного фрагменту;
- **Install** – інсталяція доповнень до засобів та наборів інструментів системи;
- **CDF Preview** - попередній перегляд вмісту робочого ноутбука у форматі *.cdf*.
- **CDF Export** – експорт файлів у форматі *.cdf*.
- **Send To** – відправка виділеного фрагмента документа до відкритого файлу операційної системи;
- **Printing Settings** - установка параметрів друку;
- **Print (Ctrl + P)** – друк вмісту вікна поточного документа;
- **Print Preview** – попередній перегляд вмісту вікна поточного документа, що буде надруковано;
- **Exit (Alt + F4)** - завершення роботи з системою.

Вибір будь-якої команди в цій і в інших опціях головного меню можна здійснити кожним із чотирьох способів:

- Вибір позиції підменю за допомогою клавіш переміщення курсору і активізація цієї опції натисканням клавіші **Enter**.
- Вибір опції натисканням клавіші вибору (вона відзначена в позиціях підменю горизонтальною лінією знизу) і натисканням потім клавіші **Enter**.
- Використання комбінації клавіш прямого доступу до команди (не потребує активізації головного меню).
- Вибір опції підменю за допомогою мишки переміщенням її курсору і швидким подвійним натисканням лівої клавіші мишки в момент, коли курсор знаходиться на потрібній позиції. «

Слід зазначити, що хоча бібліотечні файли розширень можна завантажувати, як і файли з розширенням *.ta* у вікні редагування, та як правило, це робиться тільки при їх підготовці та редагуванні. Згадані файли

зазвичай завантажуються в поточний документ без відображення їх текстів за допомогою спеціальних команд.

8. Робота з файлами документів

Команда **New** використовується, коли потрібно почати роботу з новим документом. Ця команда повністю очищає екран із запитом про те, чи потрібно записати поточний документ, якщо він є і був редагованим. Документ отримує ім'я Untitled-N, де N - поточний номер документа. Важливо відзначити, що ця команда не скасовує змін, зроблених в попередньо редагованих документах і в раніше завантажені файли пакетів розширень. Лише повне перезавантаження операційної системи скасовує ці зміни.

Завантаження готових файлів для раніше створених документів - одна з найбільш поширеніших операцій. Вона реалізується командою **Open**, яка слугує для завантаження раніше створеного документа з його пошуком в файлової системі комп'ютера. Ця команда виводить стандартне діалогове вікно, типове для операційної системи Windows і призначене для зручного пошуку файлів. Робота з цим вікном для користувача Mathematica добре знайома і додаткового опису не потребує.

Після роботи з документом виникає необхідність зберегти його. Для цього служать команди **Save** і **Save As**. Команда **Save** виконує збереження документа без зміни його імені. Тому вона виконується швидко і без будь-яких додаткових дій. Збереження виконується у форматі notebook.

Команда **Save As** дозволяє змінити ім'я файлу і помістити його в будь-яке місце на довільному диску. Ця команда викликає появу стандартного діалогового вікна. У цьому вікні крім налаштувань диска і потрібного підкаталогу треба задати ім'я записуваного файлу або підтвердити пропонуване ім'я. Запис йде в форматі notebook.

9. Команда завершення роботи з системою - Exit

Команда **Exit** використовується для закінчення роботи з системою. Якщо все документи, з якими користувач працював, були збережені на диску, то при виконанні цієї команди відбудеться закриття одного за другим вікон

цих документів. Якщо якийсь із документів не був збереженим після внесення змін, то команда **Exit** виконає запит про необхідність такого збереження. Виконання команди **Exit** завершується виходом користувача в оболонку Windows.

Унизу опції File головного меню видно також перелік файлів, з якими останнім часом працював користувач. Вибір будь-якого з файлів веде до його завантаження в поточному вікні редагування. Це робить для активного користувача роботу з системою більш зручною, оскільки позбавляє його від пошуку найбільш потрібних файлів на дисках у папках.

РЕДАГУВАННЯ ДОКУМЕНТА (EDIT)

10. Основні поняття про документи і їх стилі

Редагуванням документа вважається будь-яка зміна тексту коментарів, вихідних даних і математичних формул з метою надання документу більш підходящого виду та отримання нових результатів. До редагування відноситься також і зміна формату графіків.

У загальному випадку документи характеризуються стилем оформлення. Від вибору стилю документа багато в чому залежить його наочність і естетичність сприйняття. Тому в Mathematica передбачені значні можливості по зміні стилю документів і їх частин. Вони зосереджені в підменю команди **Format** головного меню. Проте велика кількість засобів установки стилю породжує проблему сумісності стилів, оскільки нерідко стилі бувають несумісними. Для вирішення цієї проблеми використовуються спеціальні засоби перетворення стилів. Для комірок вони розміщені в підменю опції Cell головного меню.

11. Виділення в документах і використання миші

При редагуванні документа курсор миші необхідно переміщати з одної комірки в іншу і звертатися до команд головного меню для виконання тих чи інших операцій, наприклад копіювання вмісту комірки в буфер чи зміни шрифту. Слід враховувати, що вигляд курсору при цьому змінюється і дозволяє встановлювати його місцезнаходження.

Важливим моментом в роботі з документами є виділення їх елементів - комірок введення і виведення, їх вмісту і т.д. Для виділення комірки достатньо встановити курсор миші на її дужку і натиснути ліву клавішу - дужка виділиться синім кольором. Якщо при цьому натиснути праву кнопку миші, з'явиться контекстно-залежне меню правої клавіші з підменю позиції **Copy As**, що дозволяє скопіювати вміст виділеної комірки в буфер обміну.

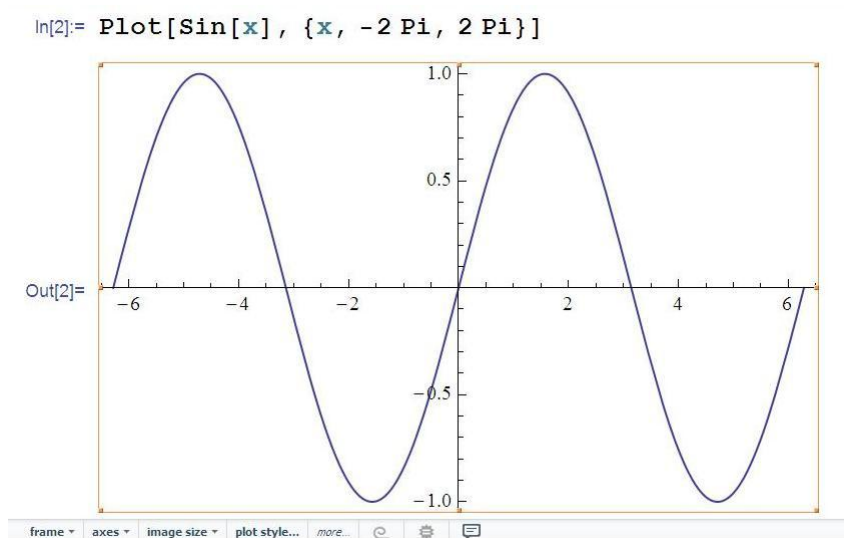


Рис. 11.1. Вікно системи Mathematica 9 відкритим ноутбуком та виділеним графіком

Можливим також є виділення виразів всередині комірок, а також виділення графіків (рис. 1.4). Для виділення графіка достатньо увести курсор в область малюнка і клацнути лівою клавішею миші. Малюнок буде обведеним рамкою з характерними прямокутниками. Тоді, натиснувши і утримуючи ліву кнопку миші, можна розтягувати графік в різні сторони і міняти його розмір. В цьому випадку також можна вивести контекстно-залежне меню правої кнопки миші.

Контекстно-залежні меню правої кнопки миші дуже зручні при професійній роботі з системою Mathematica. Вони дають повний перелік команд, які можна використовувати для виділеного об'єкта, не використовуючи головне меню.

Коли курсор знаходиться в межах комірки введення або виведення, натискання лівої клавіші миші викликає виділення деякої частини цієї комірки. Виділення при цьому можна розширювати повторним натисканням

лівої кнопки миші. Можна розширювати область виділення і просто переміщенням маркера введення при постійно натиснутій лівій клавіші миші.

Якщо курсор знаходиться в комірці введення, то він використовується для точного задання місця, в якому починається редагування. Для фіксації курсору на цьому місці потрібно натиснути ліву кнопку миші - курсор при цьому перетворюється в жирну вертикальну риску. За межами робочих комірок курсор мишки має звичайний вигляд похилої жирної стрілки.

12. Підготовка текстових коментарів

Важливою частиною професійно створено і опрацьованого ноутбука є текстові коментарі (див. Рис. 12.1). Без них документ через деякий час стає незрозумілим навіть його розробнику. Тому правилом хорошого тону є застосування досить докладних текстових коментарів.

Тестові коментарі вводяться прямо, в поточний рядок введення. Однак не слід завершувати введення натисканням комбінації клавіш **Shift + Enter**, так як це призведе до висновку коментаря в рядок виводу з можливими повідомленнями про помилку. Вони обумовлені тим, що в текстових коментарях зазвичай не дотримуються синтаксису вхідного мови системи Mathematica, що і спричиняє появу помилок.

Щоб зазначена ситуація не повторювалася, достатньо встановити курсор миші під рядок введення з коментарем, а потім клацнути лівою клавішею миші; в новому рядку введення можна розмішувати новий коментар чи математичні вирази для обчислень.

Часто в ході редагування доводиться змінювати текстові коментарі, наприклад великі написи в документах. Для цього достатньо виділити ту клітинку, в якій знаходиться напис. Для цього необхідно підвести курсор мишки в квадратній дужці в правому кінці комірки - курсор при цьому перетворюється в стрілку з вертикальною лінією. Вказавши стрілкою необхідну комірку, треба натиснути ліву клавішу мишки. Дужка виділеної осередки заповнюється чорним кольором.

Ряд розширених можливостей редагування представляють команди підменю, що відноситься до опції **Edit** головного меню. В основному ці

операції пов'язані з обміном інформацією між виділеної (зазначеної) осередком або групою осередків і спеціальним буфером.

РОБОТА З КОМІРКАМИ (CELL)

13. Поняття про комірки документів

Отже, комірки (Cells) є основними об'єктами документів. Комірки відрізняються їх статусом, тобто сукупністю властивостей, що визначають поведінку комірки в різних ситуаціях і її типом. Важливим поняттям, що відноситься до комірок і відображає особливості роботи систем символічної математики, є поняття зміни (еволюції) і модифікації вмісту комірок.

Наприклад, комірки виведення містять текстові написи - коментарі не еволюціонують, тобто не змінюються в ході виконання документа. Комірки введення, навпаки, еволюціонують (і їх зміст змінюється) і породжують комірки виведення з різним вмістом; наприклад комірка, вираз якої містить функцію $f[x]$, буде змінюватися відповідно до зміни $f[x]$. Комірки можуть бути заблокованими від модифікації і розблокованими. Статус комірок постійно перевіряється за допомогою операції оцінювання в ході виконання документа.

Статус комірок можна розпізнати і без виконання документа по ряду характерних ознак. Один з них - вид курсору миші при його розміщенні в області комірок - був описаний вище. Інша ознака - малопомітний розпізнавальний знак у верхній частині квадратної дужки, що обрамляє комірку. Відсутність знаку означає те, що це звичайна комірка введення. Знак «-» (коротка горизонтальна риска) відзначає комірку виведення зі статусом Innpactive. Заблокована (закрита) комірка (Locked) позначається знаком «х», а ініціалізована комірка (Initialization) - знаком «т». Крім того, що еволюціонуючи комірки відзначаються маленьким трикутником. Про тип комірки можна також судити по їх стилю, зокрема за шрифтом використовуваних в ній символів.

Для перемикавання на потрібний тип комірки досить помістити курсор миші на потрібний напис і натиснути ліву клавішу мишки. Відповідний напис про тип комірки з'явиться у вікні селектора типів осередків.

14. Команди опції Cell головного меню

Комірки (cells) - це найбільш характерна деталь ноутбуків системи Mathematica. У позиції Cell головного меню передбачені команди для роботи з комірками.

При активній позиції Cell головного меню з'являється підменю, що має ряд команд.

- **Convert To** - перетворення формату комірок;
- **Cell Properties** - установка властивостей комірок;
- **Cell Tags** - установка тегів для комірок;
- **Grouping** - групування комірок;
- **Notebook History** – історія редагування вмісту комірок документа;
- **Delete All Output** – видалення вмісту комірок уведеного під час поточної роботи з документом;
- **Show Expression** – показ вмісту комірки у розгорнутому вигляді.

Команда ShowExpression служить для управління показом виразів в стандартному та розгорнутому вигляді.

Наприклад, при відключеній команді уведемо у комірку простий вираз

Sin[x]/Log[x].

Sin[x]/Log[x].

Тут вигляд комірок є стандартним. Далі, виділивши ці комірки та виконавши команду **Show Expression (Shift+Ctrl+E)** отримаємо представлення цих виразів в розгорнутому форматі:

```
Cell[BoxData[
  RowBox[{
    RowBox[{"Sin", "[", "x", "]"}, "/",
    RowBox[{"Log", "[", "x", "]"}, "Input",
    CellChangeTimes->{{3.680553853375*^9, 3.6805538589375*^9}}]
```

```
Cell[BoxData[
  FractionBox[
```

```
RowBox[{"Sin", "[", "x", "]" }],
RowBox[{"Log", "[", "x", "]" }]]], "Output",
CellChangeTimes->{3.680553928109375*^9}]
```

Такий формат є внутрішнім в тому сенсі, що він характерний для внутрішнього подання висновку на екран дисплея, прийнятого в мові програмування системи Mathematica. Це типова програма для виведення вибраних виразів. Чим складнішим є вираз, тим довше і менш зрозумілим для невідготовленого читача виглядає розгорнуте уявлення у внутрішньому форматі.

УПРАВЛІННЯ РОБОТОЮ ЯДРА СИСТЕМИ (KERNEL)

15. Команди позиції *Evaluation* головного меню

Позиція головного меню **Evaluation** служить для управління діями системи, виконуваними ядром системи над комірками робочого документу. Розглянемо команди цього підменю більш докладно.

16. Управління процесом обчислень.

Загальні команди з управління процесом обчислень зосереджені в меню Evaluation:

- **Evaluate Cells (Shift + Enter)** - виконання вибраної комірки;
- **Evaluate Place (Shift + Ctrl + Enter)** - виконання виділеного виразу згідно місця його розташування;
- **Evaluate Subsection (F7)** виконання субсекції документа;
- **Evaluate Notebook** - виконання всього документа;
- **Evaluate Initialization Cells** - виконання ініційованих осередків;
- **Find Currently Evaluating Cell** – пошук поточної виконуваної комірки.

Дана група команд управляє еволюцією (виконанням) комірок. Перед виконанням кожна комірка оцінюється за своїми ознаками. Команда **Evaluation Cells** оцінює всі виділені осередки, викликає їх еволюцію і поміщає результат еволюції кожного осередку відразу після неї. Це одна з найбільш поширених команд команд. Слід пам'ятати, що, здавалося б, природне натискання клавіші **Enter** викликає лише переривання рядка, а не еволюцію виділених комірок введення. При управлінні з клавіатури еволюція обраних комірок відбувається при одночасному натисканні клавіш **Shift** і

Enter. Перед цією командою має бути завантажено ядро системи. Якщо цього не було зроблено, то може пройти певний час для завантаження ядра, перш ніж дана команда буде виконана.

17. Вибір ядра системи

Сучасні версії системи Mathematica мають можливість роботи не тільки з установленим локальним ядром, а й з іншими ядрами, орієнтованими на безпосередньо на специфічні класи обчислень. Це призвело до появи ряду нових Команд:

- **Start Kernel** - задає старт обраного ядра;
- **Quit Kernel** - задає вихід з обраного ядра;
- **Default Kernel** - задає ядро, яке використовується за умовчанням;
- **Notebook's Kernel** - задає ядро робочого документа;
- **Kernel Configuration Options** - виводить вікно установки властивостей ядер;
- **Parallel Kernel Configuration** – конфігурація паралельних ядер (при наявності багатоядерного процесора на ПК);
- **Parallel Kernel Status** – статус паралельних ядер при виконанні комірок документа (при наявності багатоядерного процесора на ПК).

ІНТЕРАКТИВНІСТЬ ПІД ЧАС РОБОТИ З СИСТЕМОЮ

18. Динамічна інтерактивність при символічних обчисленнях

Динамічна інтерактивність - нова якість ноутбуків системи Mathematica, починаючи з шостої її версії. Вона полягає в застосуванні досить простих засобів, що дозволяють перетворювати ноутбуки (документи) системи в діалогові (інтерактивні) вікна з елементами, що дозволяють динамічно управляти параметрами вихідних даних для обчислень (в тому числі символічних) і побудови графіків.

Одним з таких засобів є керуюча функція Manipulate. Це типовий програмний модуль, який буде детально описаний в подальшому. Поки ж

розглянемо приклад його застосування для обчислення інтегралу з підінтегральною функцією $\cos(x^n)$ для різних цілих n від 0 до 10, що показаний на Рис. 18.1. Функція Manipulate в осередку виведення виводить інтерактивне вікно зі слайдером, бігунок якого можна переміщати мишею і вікном виведення результатів аналітичного інтегрування. При цьому n змінюється і міняється символічне значення інтегралу. На Рис. 18.1 показаний випадок, коли слайдером задано значення $n=1$. Воно виведено в кінці слайдера.

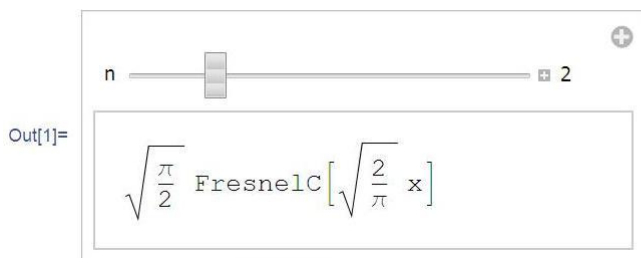
```
In[1]:= Manipulate[Integrate[Cos[x^n!], x], {n, 0, 10, 1, Appearance -> "Labeled"}]
```



Рис. 18.1. Динамічне обчислення інтеграла з підінтегральною функцією $\cos(x^n)$ при $n=1$

Для демонстрації динамічної інтерактивності на рис. 18.2 і показано приклади для $n=2$ і $n=3$. Ці значення n отримані пересуванням повзунка слайдера. Добре видно, що при зміні n змінюється формула, яка визначає символічне значення інтеграла.

```
In[1]:= Manipulate[Integrate[Cos[x^n!], x], {n, 0, 10, 1, Appearance -> "Labeled"}]
```



```
In[1]:= Manipulate[Integrate[Cos[x^n!], x], {n, 0, 10, 1, Appearance -> "Labeled"}]
```

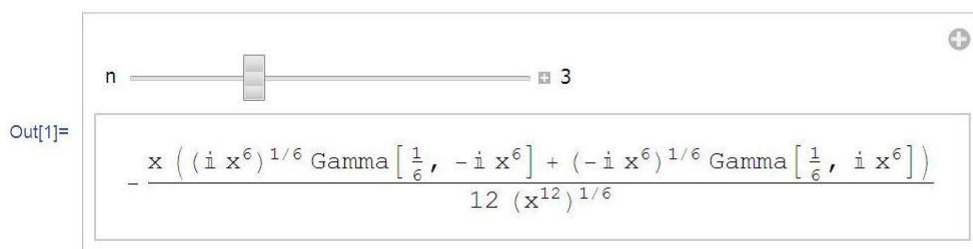


Рис. 18.2. Динамічне обчислення інтеграла з підінтегральною функцією $\cos(x^n)$ при $n=2$ і $n=3$

Можливості автоматичного обчислення заданого інтеграла разом з розгорнутою панеллю анімаційного програвача обчислення наведено на Рис 18.3.

```
In[1]:= Manipulate[Integrate[Cos[x^n!], x], {n, 0, 10, 1, Appearance -> "Labeled"}]
```

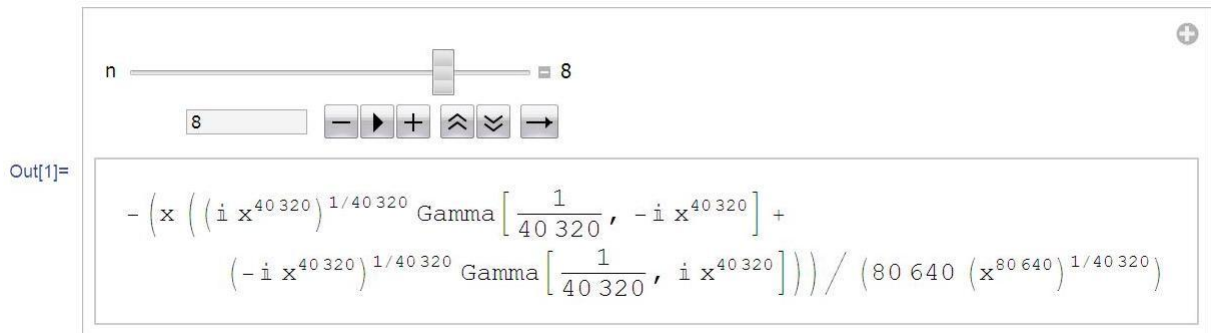


Рис. 18.3. Динамічне обчислення інтеграла з підінтегральною функцією $\cos(x^{n!})$ з розгорнутою панеллю анімаційного програвача

19. Управління графіками з допомогою миші

Графіка системи Mathematica довгий час мала явний недолік - відсутність можливості повороту тривимірних графіків фігур мишею. Ця можливість вже давно була реалізована в інших системах комп'ютерної математики Mathcad, Maple і MATLAB. Правда, в Mathematica 4/5 можна було завантажити ноутбук з процедурою, що дає таку можливість. Однак тільки починаючи з Mathematica 6 і подальших версіях системи можливість управління графіками мишею присутня вже як властивість тривимірних графіків, побудова яких задана в ядрі системи.

На Рис. 19.1 показано класичний приклад на побудову поверхні, описуваної функцією двох змінних - $x^2 + y^2 - 1$. Поверхня (параболоїд обертання) будується за допомогою функції **Plot3D** і показана при побудові за замовчуванням - опуклістю вниз.

```
In[1]:= Plot3D[x^2 + y^2 - 1, {x, -1, 1}, {y, -1, 1}]
```

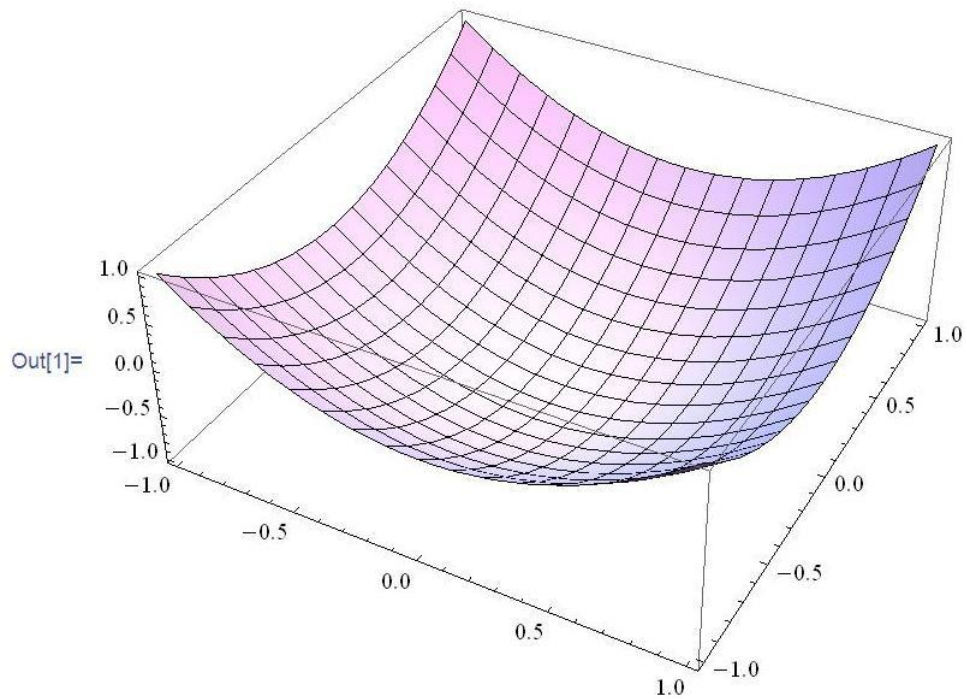


Рис. 19.1. Побудова параболоїда обертання.

Якщо ввести курсор миші всередину малюнка, він зміниться на зображення двох вигнутих стрілок, що вказує на обертання. Натиснувши ліву кнопку миші і утримуючи її натиснутою, можна почати повертати фігуру і спостерігати її обертання в просторі. На Рис. 19.2 показаний результат таких маніпуляцій, в результаті якого параболоїд виявилася побудованим опуклістю вгору.

Можливість обертання тривимірних графічних об'єктів мишею дозволяє розглядати їх з різних сторін і виявляти спочатку невидимі їх особливості. Якщо натиснути клавішу **Shift**, то при натиснутій лівій клавіші миші можна переміщати графік по полю графічного вікна, а, утримуючи клавішу **Ctrl** можна плавно наближати або віддаляти графік (операція **Zoom**).

```
In[2]:= Plot3D[x^2 + y^2 - 1, {x, -1, 1}, {y, -1, 1}]
```

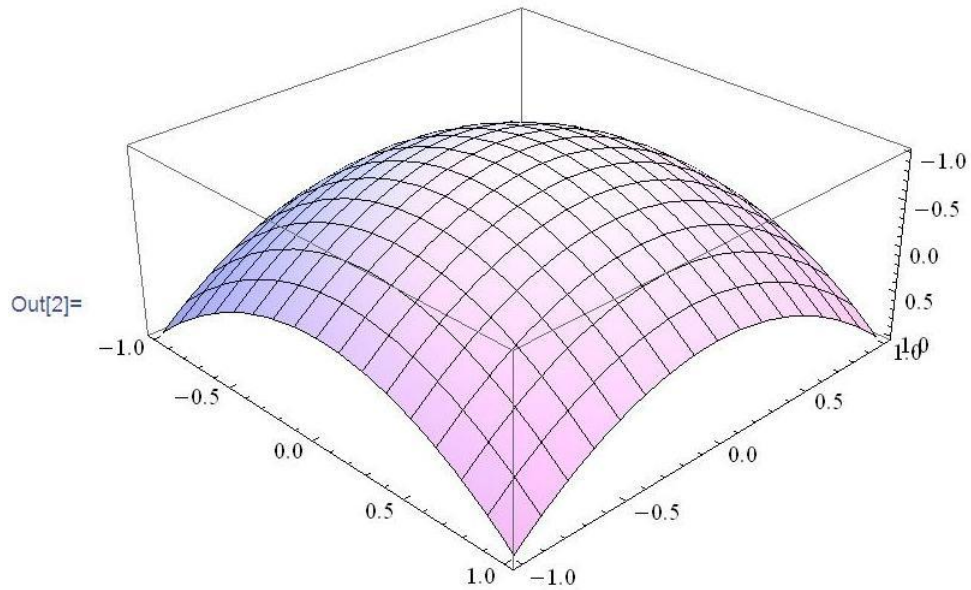


Рис. 19.2. Побудова оберненого параболоїда обертання

20. Динамічна інтерактивність при графічній візуалізації

```
In[1]:= Manipulate[Plot[Sin[n x] / (n x), {x, -2 Pi, 2 Pi},
  PlotRange -> {-0.3, 1.3}], {n, 1, 10, Appearance -> "Labeled"}]
```

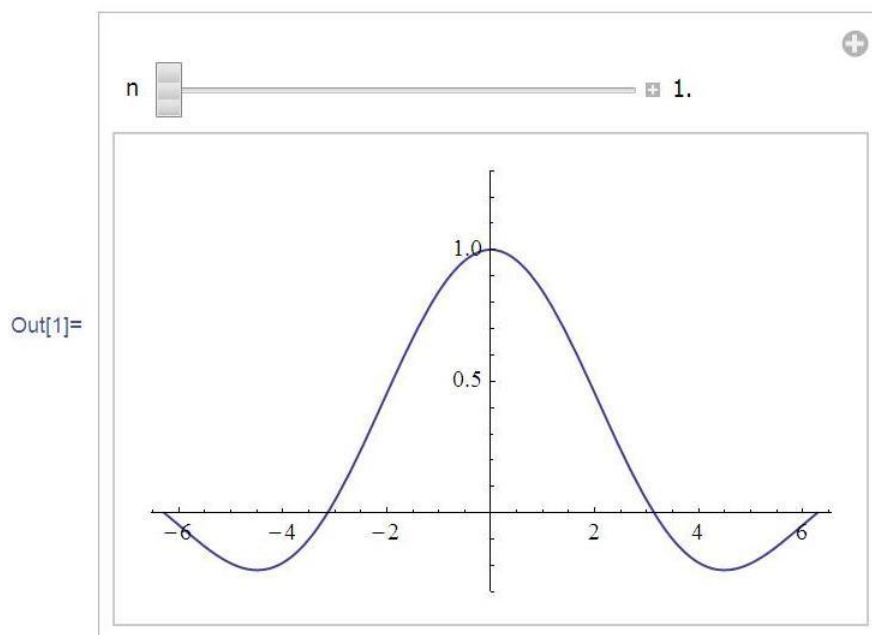


Рис. 20.1. Побудова графіка функції $\sin nx / nx$ із слайдером для зміни параметра n для випадку $n = 1$

Можливість обертати тривимірні фігури та графіки мишею - це лише один з компонентів графічної динамічної візуалізації. Розробники Mathematica пішли далі і ввели можливості спостереження за поведінкою

графіків різного типу (в тому числі тривимірних) при зміні будь-якого з їх параметрів. Ці можливості також забезпечує функція **Manipulate**.

На Рис. 20.1 показано побудову графіка за висловом $\sin nx/nx$ за допомогою функцій **Plot** і **Manipulate**. При цьому задано зміну параметра n від 1 до 10. Графік на Рис. 20.1 побудований для $n = 1$.

Тепер, переміщаючи повзунок слайдера, можна спостерігати за змінами графіка, що істотно підвищує його наочність. На Рис. 20.2 показаний той же ноутбук, але при положенні повзунка слайдера, відповідного $n = 3,65$. При цьому характер зміни графіка очевидний.

```
In[1]:= Manipulate[Plot[Sin[n x] / (n x), {x, -2 Pi, 2 Pi},
  PlotRange -> {-0.3, 1.3}], {n, 1, 10, Appearance -> "Labeled"}]
```

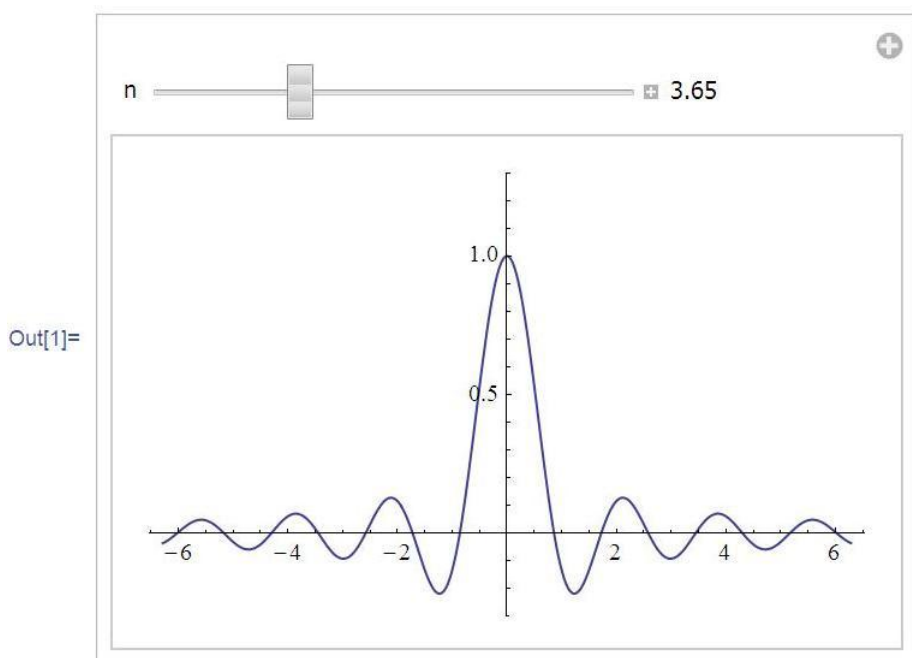


Рис. 20.2. Побудова графіка функції $\sin nx/nx$ із слайдером для зміни параметра n для випадку $n = 3,65$

Число слайдерів і змінюваних параметрів неважко збільшити, просто надавши відповідні параметри декільком змінним в списку параметрів функції **Manipulate**. На Рис. 20.3 показаний приклад побудови гіперболічної поверхні, у якій змінюються два параметри - коефіцієнти n_1 і n_2 при змінних x і y . Поверхня побудована для значень $n_1 = n_2 = 1$ і повернена мишею для отримання більш наочного зображення.

```
In[1]:= Manipulate[Plot3D[n1 x^2 - n2 y^2,
  {x, -2, 2}, {y, -2, 2}], {n1, 1, 10, Appearance -> "Labeled"}, {n2, 1, 10, Appearance -> "Labeled"}]
```

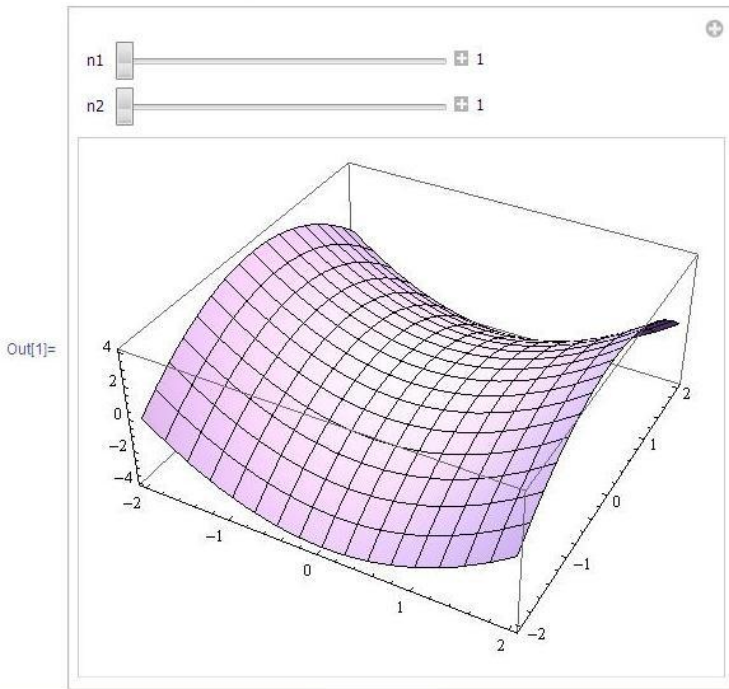


Рис. 20.3. Побудова гіперболічної поверхні при двох змінюваних параметрах (при $n_1 = n_2 = 1$)

На рис. 20.4 показана та ж поверхню після переміщення повзунків слайдерів з установкою значень $n_1 = -0,62$; $n_2 = 1,56$. Добре видно, як змінюється вигляд поверхні. Можна спостерігати плавну зміну цієї поверхні при переміщенні повзунків слайдерів. При зусилля на їх виведення є мінімальними – вони задаються параметрами функції **Manipulate**. В інших системах для реалізації таких інтерактивних можливостей довелося б скористатися процедурним або візуально-орієнтованим програмуванням, попередньо освоївши його.

Однак і на цьому можливості Mathematica 9 не закінчуються. Можна реалізувати анімацію побудованих графічних об'єктів. На рис. 20.4 показано меню операцій, яке з'являється при активізації мишею кружка зі знаком «+» у верхньому правому кутку ноутбука. Внизу цього меню є опція **Autorun**, вибір якої викликає появу зверху вікна панелі анімаційного програвача **Autorun**.

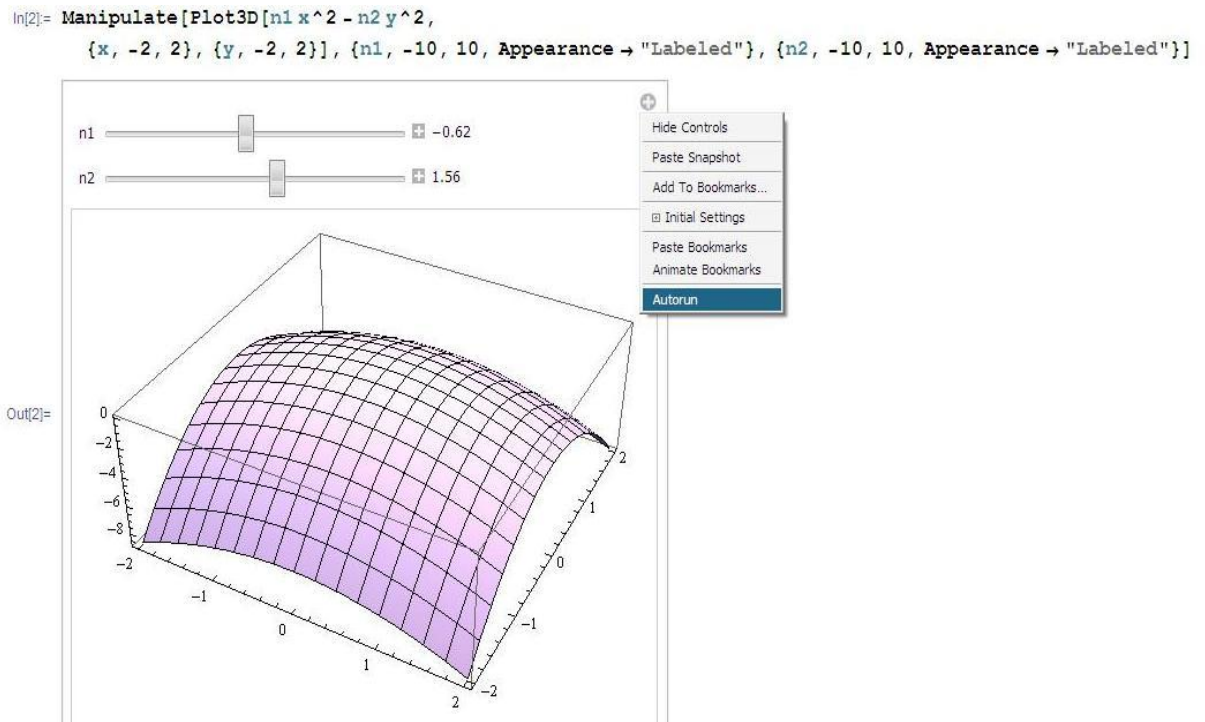


Рис. 20.4. Побудова гіперболічної поверхні при двох змінюваних параметрах (при $n_1 = -0,62$; $n_2 = 1,56$)

За замовчуванням програвач **Autorun** забезпечує перегляд анімації поверхні при плавному автоматичному переміщенні спочатку одного повзунка, а потім другого. Можна втрутитися в роботу програвача, наприклад, зупинивши його, змінивши швидкість і напрямок переміщення повзунків, і, нарешті, закрити його. Відповідні кнопки з очевидним мнемонічним зображенням є на панелі програвача.

РОЗДІЛ 2

MATHEMATICA ЯК СИСТЕМА ПРОГРАМУВАННЯ

1. *Поняття про вхідну мову системи і мову реалізації*

Система Mathematica здатна без загальноприйнятого програмування розв'язувати велику кількість математичних і науково-технічних завдань. Однак всі засоби системи (включаючи алфавіт її вхідної мови, його букви, цифри, оператори і численні спеціальні знаки) по суті є частиною проблемному орієнтованої мови програмування надвисокого рівня [34,45,46]. За своїми можливостями у виконанні математичних і науково-технічних обчислень ця мова набагато перевершує звичайні універсальні мови програмування, такі як Fortran, Basic, Pascal або C.

Важливо підкреслити, що тут мова йде про мову програмування системи Mathematica, а не про мову реалізації самої системи. Мовою реалізації є універсальна мова програмування C++, яка показала свою високу ефективність як мова системного програмування.

2. *Можливості мови програмування системи Mathematica*

Перевага системи Mathematica, як засобу програмування розв'язків математичних задач, зумовлена незвично великим (в порівнянні зі звичайними мовами програмування) набором функцій, серед яких чимало таких, які реалізують складні і практично корисні математичні перетворення та чисельні та аналітичні обчислювальні методи. Число функцій в Mathematica б досягає 4000.

Мова програмування системи Mathematica є типовим інтерпретатором і не є призначеною для створення виконуваних файлів. Проте, для окремих виразів ця мова може виконувати компіляцію з допомогою функції **Compile**, що корисно при необхідності збільшення швидкості рахунків в старих реалізаціях системи. У нових версіях алгоритми поліпшені настільки, що особливої необхідності в застосуванні функції **Compile** майже немає.

Мова системи Mathematica увібрала в себе кращі засоби ряду поколінь мов програмування, таких як Basic, Fortran, Pascal і C. Завдяки цьому вона дозволяє легко реалізувати всі відомі типові концепції програмування:

функціональне, структурне, об'єктно-орієнтоване, математичне, логічне, рекурсивне. У нього включені і засоби візуально-орієнтованого програмування на основі застосування шаблонів математичних символів, таких як знаки диференціювання, інтегралу, підсумовування, добутку, розкладання в ряд і т.д.

Внутрішнє представлення всіх обчислень базується на застосуванні повних форм виразів, представлених функціями. І взагалі, функціям в системі Mathematica належить вирішальна роль в організації будь-яких обчислень. Тому, Mathematica фактично спочатку реалізує як головний **функціональний метод програмування** - один з найефективніших і надійних. Наявна велика кількість логічних операторів і функцій дозволяє повноцінно реалізувати логічний метод програмування. Безліч операцій перетворення виразів і функцій дозволяють здійснювати програмування на основі **правил перетворення**.

Слід також відзначити, що мова системи дозволяє розбивати програми на окремі модулі (блоки) і зберігати ці модулі в тексті документа або на жорсткому диску. Можливе створення повністю самостійних блоків – процедур з присвоєними іменами і функцій з локальними змінними. Все це, поряд з типовими структурами управління, дозволяє реалізувати **структурне і модульне** програмування.

Настільки ж природно мову системи реалізує об'єктно-орієнтоване програмування, що стало новою віхою теперішнього часу. Воно, перш за все, базується на узагальненому понятті об'єкта і можливості завдання безлічі пов'язаних один із одним об'єктів. В системі Mathematica кожна комірка документа є окремим об'єктом і породжується іншими, попередніми об'єктами. При цьому вмістом об'єктів можуть бути математичні вирази, вхідні і вихідні дані, графіки та малюнки, звуки і т.д.

З поняттям об'єкта тісно пов'язані три основних властивості: **інкапсуляція, успадкування і поліморфізм**. Всі вони органічно притаманні об'єктам системи Mathematica і не потребують для своєї реалізації будь-яких спеціальних засобів.

Інкапсуляція означає об'єднання в одному об'єкті як даних, так і методів їх опрацювання.

Успадкування означає, що кожен об'єкт, отриманий від інших об'єктів, успадковує їх властивості.

Поліморфізм - властивість, що дозволяє передати ряду об'єктів повідомлення, яке буде оброблятися кожним об'єктом відповідно до його індивідуальних особливостей.

Наведений нижче приклад об'єктно-орієнтованого програмування дає три визначення, асоційовані з об'єктом h :

$h/:h[x_]+h[y_]:=hplus[x,y]$

$h/:p[h[x_],x]:=hp[x]$

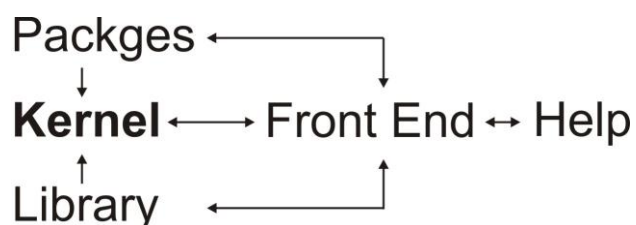
$h/:f[h[x_]]:=fh[f,x]$

В принципі, мова програмування системи Mathematica спеціально створена для реалізації будь-якого з перерахованих підходів до програмування, а також до ряду інших, наприклад рекурентного програмування, при якому наступний крок обчислень базується на інформації, отриманій на попередніх етапах обчислення. Можливим є і рекурсивне програмування, коли функція в загальному випадку неодноразово звертається до себе самої. Наочним прикладом цього може служити обчислення факторіалу рекурсивним методом: $n! = n \cdot (n-1)!$.

Засоби мови Mathematica дозволяють реалізувати і елементи візуально-орієнтованого програмування. Його зміст полягає в автоматичній генерації програмних модулів шляхом вказівки візуально зрозумілого та простого об'єкта - найчастіше кнопки. Mathematica дозволяє створювати палітри і панелі з різними кнопками, що дозволяють управляти програмою, або вводити нові програмні об'єкти.

3. Структура систем Mathematica

Щоб зрозуміти, як виконуються обчислення в системі Mathematica, потрібно хоча б у загальних рисах знати структуру систем Mathematica. Її можна зручно представити в наступному вигляді:



Для орієнтації системи на конкретну машинну платформу служить інтерфейсний процесор **Front End**. Саме він визначає, який вигляд має інтерфейс користувач системи. Інтерфейсні процесори систем Mathematica для різних платформ можуть мати свої нюанси, але особливих відмінностей порівняно з інтерфейсний процесором системи Windows немає.

Саме через інтерфейсний процесор здійснюється зв'язок користувача з системою. Користувач вводить обчислювані вирази за допомогою клавіатури і слідує за введенням за допомогою дисплея. Інтерфейсний процесор запитує у ядра потрібні оператори, функції і правила їх перетворення, обчислює з їх допомогою вираз, і результат обчислень виводить на екран дисплея.

Центральне місце в системах класу Mathematica займає машинно-незалежне ядро математичних операцій - **Kernel**. Воно містить набір операторів і функцій, правил обчислень і перетворень математичних виразів. Ядро зроблено досить компактним для того, щоб будь-яка функція з нього могла бути викликаною досить швидко. Для розширення набору функцій служить бібліотека **Library** і набір пакетів розширення **Packages**. Пакети розширень формуються власною мовою програмування системи Mathematica і є головним засобом розширення можливостей системи і їх адаптації до вирішення безпосередніх класів задач користувача. Крім того, системи мають

вбудовану довідкову систему - **Help**. Вона містить ряд електронних книг з великою кількістю прикладів.

4. Ідеологія систем Mathematica

Ідеологія систем Mathematica базується на двох положеннях:

- розв'язання більшості математичних задач в системі може відбуватися в діалоговому режимі без традиційного програмування;
- вхідна мова спілкування системи є одним з найпотужніших мов функціонального програмування, орієнтованих на розв'язання різноманітних завдань (в тому числі математичних).

Виникаюча з першого погляду суперечливість цих положень чисто номінальною. Насправді Mathematica - типова система програмування з проблемно-орієнтованою мовою програмування надвисокого рівня. Її можна віднести до класу інтерпретаторів. Як відомо, мови такого типу послідовно аналізують (інтерпретують) кожен вираз і тут же виконують його. Таким чином, робота з системою відбувається в діалоговому (інтерактивному) режимі: користувач задає системі завдання, а вона тут же виконує його. Mathematica містить достатній набір керуючих структур процедурного програмування для створення умовних виразів, розгалуження в програмах, циклів і т.д., Для повної автоматизації обчислень і легкого створення користувачем своїх процедур і функцій.

У нових реалізаціях Mathematica до вказаних двох принципами доданий цілий ряд нових принципів:

- за рахунок усунення обмежень по швидкості реалізації чисельних методів системи Mathematica стали універсальними системами комп'ютерної математики;
- системи стали істотно розширюваними за рахунок застосування вбудованих (Add-On) і зовнішніх пакетів розширення, а також пакетів розширення, створюваних користувачем;
- системи реалізують найвищу якість підготовки ноутбуків - документів, що містять одночасно текстові записи, аналітичні вирази, таблиці, графіки, малюнки та інші компоненти;

- системи дозволяють створювати повністю завершені високоякісні електронні уроки, статті та книги з високим рівнем візуалізації всіх видів обчислень;
- системи стали інтелектуальними системами надання знань в галузі фундаментальної та прикладної математики.

Все це істотно розширює можливості системи Mathematica і робить їх застосування більш привабливим порівняно з іншими системами.

5. Концепція динамічної зміни змінних в Mathematica

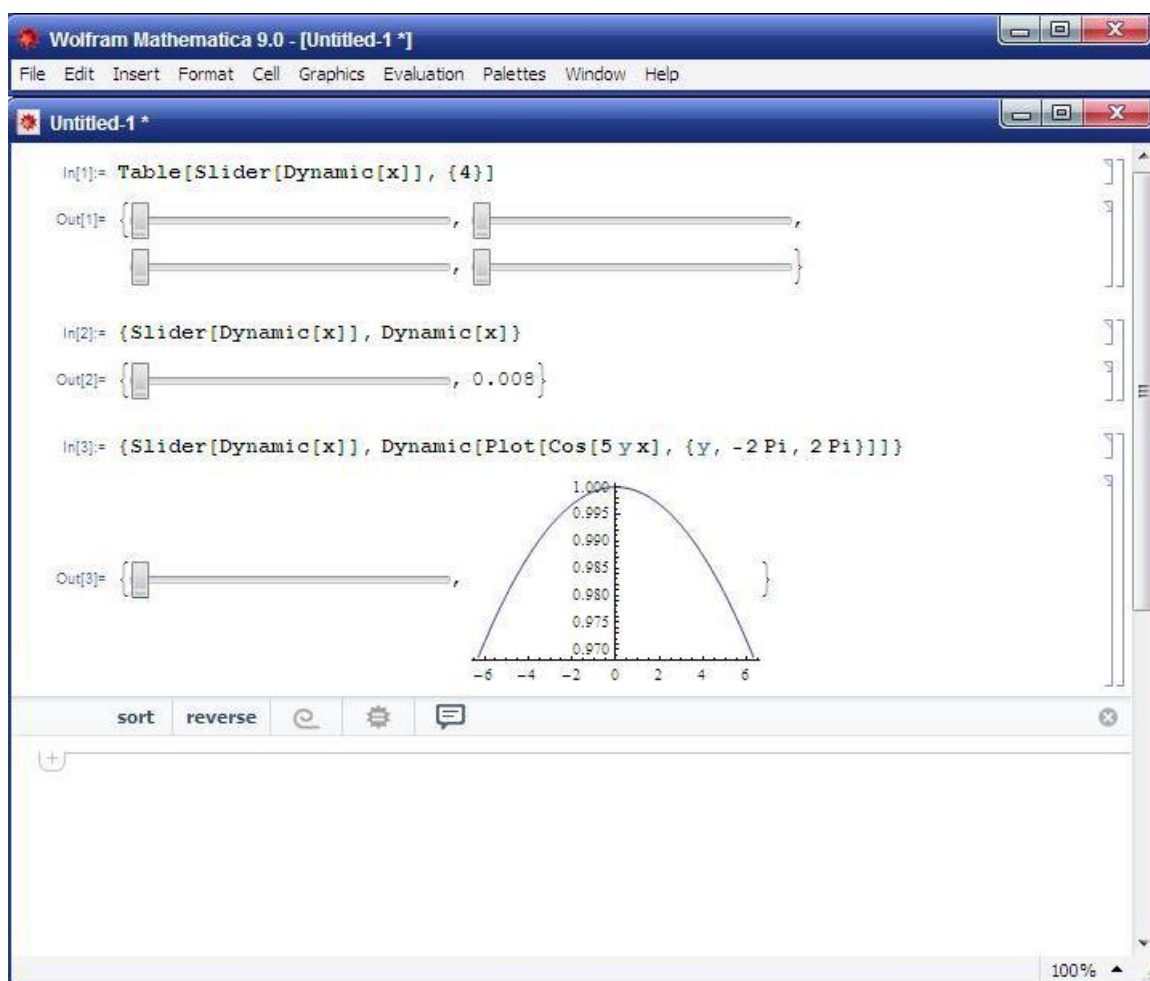


Рис. 5.1.Задання динамічної змінної x

Мова програмування Mathematica має такі добре відомі об'єкти як **змінні**. Слід відзначити, що змінні мають імена-ідентифікатори і їм можна присвоювати ті чи інші чисельні, стрічкові або символічні значення. При цьому змінні діляться на глобальні (їх значення діють у всьому ноутбукі) і локальні (їх область дії обмежена заданими програмними процедурами або функціями зі списками параметрів - формальними змінними).

Поки обмежимося обговоренням глобальних змінних. Таким змінним, можна присвоїти довільне значення в будь-якому місці ноутбука, і це значення збережеться в наступних частинах ноутбука аж до чергового переприсвоєння. Проте часто виникає необхідність змінити значення змінної так, щоб нове значення зберігалось як після моменту привласнення, так і до нього. Особливо цінною була б така можливість при використанні засобів динамічної зміни значень глобальних змінних. Саме ці можливості і введені в систему Mathematica. Вони реалізовані функцією-ознакою **Dynamic**.

Застосування функції **Dynamic** ілюструє ноутбук (рис. 5.1). На ньому представлені: побудова таблиці з чотирма слайдерами, одиничний слайдер і слайдер з графіком синусоїдальної функції. Скрізь змінна x вказана як параметр функції **Dynamic**. Це означає, що зміна положення повзунка будь-якого слайдера (або просто зміна значень змінної) автоматично веде і зміни її значення вище і нижче точки присвоювання змінної заданого значення. Іншими словами, це означає, що положення повзунків всіх слайдерів задається зміною повзунка будь-якого слайдера. На це вказує ідентичність положення повзунків всіх слайдерів.

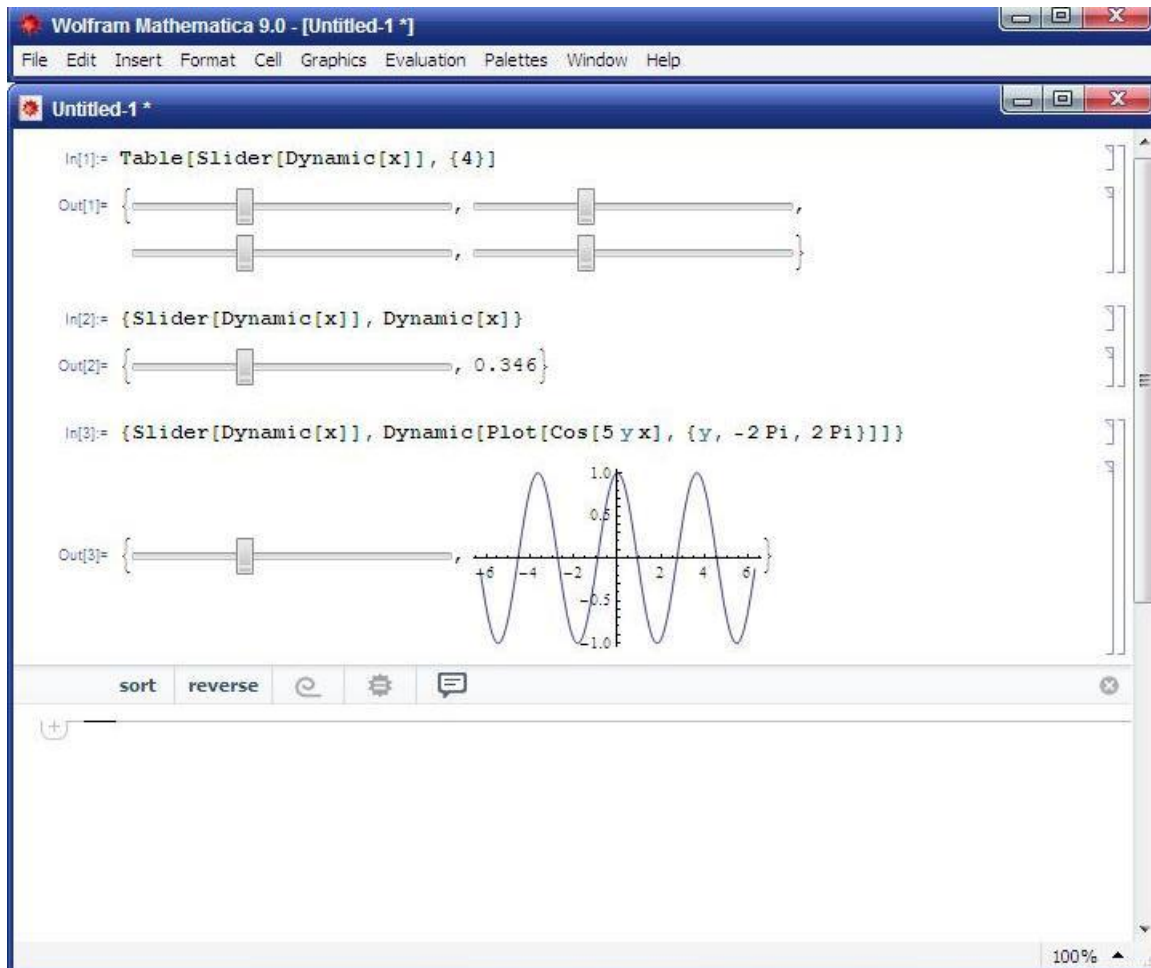


Рис. 5.2. Динамічна зміна значення змінної x

На Рис. 5.2 показаний другий випадок - переміщення повзунка нижнього слайдера, що викликає зміну графіка синусоїдальної функції під ним. Легко помітити, що задане положення слайдера (і значення змінної x) автоматично переноситься на вищі комірки ноутбука.

6. Функції символьних обчислень. Поняття про аналітичні обчислення

Символьні операції - це та ознака, яка кардинально відрізняє систему Mathematica і інші подібні їй системи комп'ютерної математики) від систем для виконання чисельних розрахунків, наприклад таких, як примітивні табличні процесори Excel. Системи з символьними операціями часто називаються також системами комп'ютерної алгебри, що, однак, не цілком відображає куди більші можливості систем класу Mathematica

Розглянемо деякі прості приклади застосування символьної математики, які реалізуються в системі Mathematica. При символьних операціях, які також називають аналітичними, завдання на обчислення

подаються у вигляді символьних (формульних) виразів, і результати обчислень також отримуються в символьному вигляді. Чисельні результати при цьому є окремими випадками результатів символьних обчислень.

Наприклад, спроба обчислити в загальному вигляді вираз $\sin^2 x + \cos^2 x = 1$, що є тотожність з допомогою чисельних математичних систем або програм на звичайних мовах програмування до успіху не приведе. Замість очікуваного результату з'явиться повідомлення виду: «Змінна x невизначена». Тому комп'ютер буде чекати введення конкретного значення для x . Так буде незалежно від того, чи запрограмоване обчислення на простенькому Basic або мовою професіоналів-програмістів C++. І лише системи символьної математики при обчисленнях дадуть довгоочікуване і абсолютно точне значення 1 (Рис. 6.1).

При розгляді навіть найпростішого прикладу, показаного на Рис. 6.1, можна зробити кілька важливих висновків. Перш за все, видно, що спроба виведення незалежної змінної x дає саму змінну. Функції $\sin x$ і $\cos x$ в системі Mathematica позначаються як **Sin[x]** і **Cos[x]**. Це відповідає двом важливим особливостям системи: імена функцій зазвичай починаються з великої літери, а параметри функції задаються в квадратних, а не круглих дужках. Останні використовуються для конструювання виразів і завдання пріоритету операцій.

Вираз $\sin^2 x + \cos^2 x$ після введення просто повторюється, а для безпосереднього обчислення використовується функція **Simplify** (спростити), аргументом якої є знак %, що означає підстановку попереднього виразу в аргумент функції виконуваної в поточній обчислювальній комірці.

Слід звернути увагу на те, що система виділяє області введення визначником In[N], а області виведення - визначником Out[N], де N - номер рядка, який автоматично перевіряється.

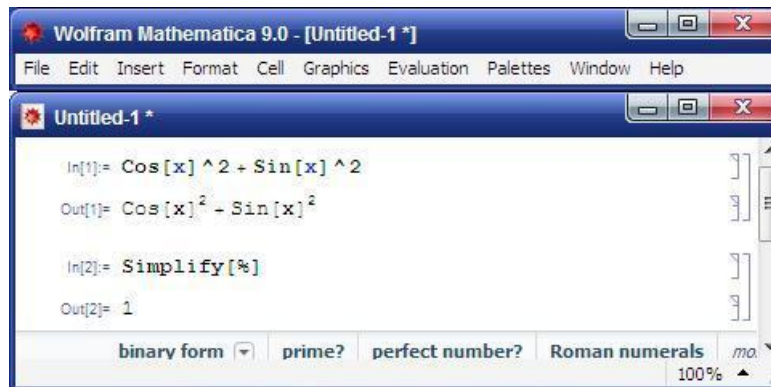


Рис. 6.1. Обчислення значення виразу $\sin^2 x + \cos^2 x$ в системі Mathematica

Комірки нумеруються по мірі їх використання. При цьому можна з кінця документу повернутися до його початку або середини і, змінивши вміст раніше використовуваних комірок, знову виконати обчислення. При цьому комірки міняють свої номери введення та виведення. При завантаженні файлу комірки перенумеровуються в строго послідовному порядку. Таким чином, номери комірок не є строго фіксованими і є суто технічним засобом.

ПРИКЛАДИ ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

ЛАБОРАТОРНА РОБОТА № 1

"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МОДЕЛЮВАННІ ПРОЦЕСІВ КОЛИВАНЬ ФІЗИЧНИХ ОБ'ЄКТІВ"

1. Фізична постановка задачі

Розглядається технологічний процес дифузійного розчинення (солі, солоду, тощо) у водному розчині.

2. Математична модель процесу: диференціальні рівняння процесу

Процес дифузійного розчинення продукту описується за допомогою найпростішої математичної моделі дифузійного типу у вигляді:

$$\frac{\partial C}{\partial t} = -k_r C \quad (1)$$

з початковою умовою:

$$C(t)|_{t=0} = C_0 \quad (2)$$

Тут $C(t)$ – розподіл приведених концентрацій розчинюваної речовини у твердій фазі від часу t ; k_r – константа швидкості дифузійного розчинення продукту, що визначається із даних фізичного експерименту; C_0 – початкове значення приведеної концентрації розчинюваного продукту у твердій фазі.

3. Вхідні дані

$t = 1, 10..1000$ - діапазон зміни часу t для визначення неперервного розподілу концентрацій розчинюваного продукту;

$n = 100$ - кількість точок обчислення;

$t_0 = 10$ - початкове значення часу t для визначення неперервного розподілу концентрацій розчинюваного продукту;

$\Delta t = 10$ - крок зміни часу t для визначення дискретного розподілу концентрацій розчинюваного продукту;

$i = 1..1000$;
 $\tau_i = t_0 + i \cdot \Delta t$ - діапазон інтервалів зміни часу для розрахунку дискретного
 концентрацій розчинюваного продукту;
 $C_0 := 1$ - початкове значення приведеної концентрації розчинюваного
 продукту.

4. Параметри і залежності для симуляції фізичного експерименту

$k_e := 0,005$ - параметр, що симулює експоненціальний ріст(спад)
 концентрації; :

$\alpha := 0,0045$ - параметр, що симулює ступінь імпульсивності випадкових
 вимірювань концентрацій;

RandomReal - генератор дійсних випадкових величин.

Залежність концентрації розчинюваної речовини у експерименті задається
 співвідношенням:

$$C_{\text{exp}_i} = C_0 e^{-k_e (1 + \alpha \cdot \text{Random}(t_i)) t_i}.$$

Задана залежність має вигляд:

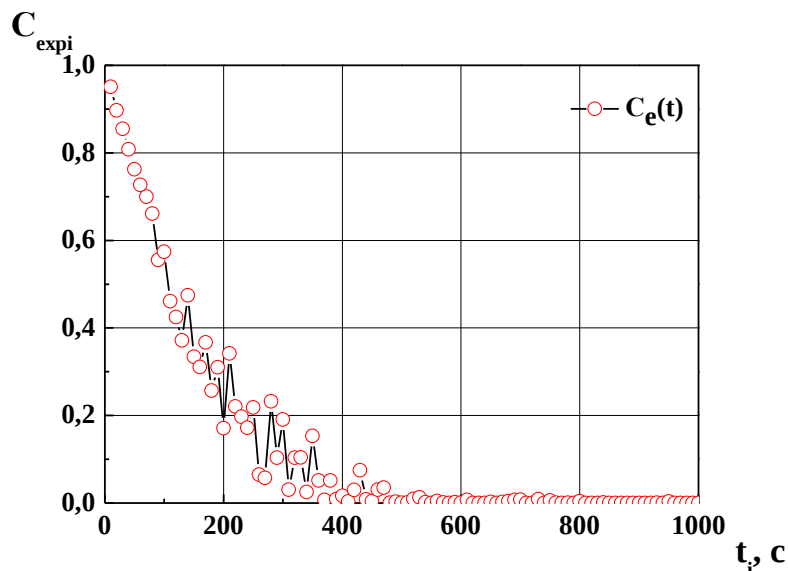


Рис. 1 - Графік експериментальних значень

5. Алгоритмізація побудови розв'язку математичної моделі

Розв'язком математичної моделі, що описується диференціальним рівнянням першого порядку (1) та початковими умовами (2) є залежність:

$$C(t) = C_0 e^{-k_r t} \quad (4)$$

де значення константи швидкості дифузійного розчинення, що визначає внутрішню кінетику протікання процесу, визначається із експериментальних даних (рис.1) шляхом розв'язання зворотної задачі:

$$C_e(t) = C_0 e^{-k_e (1 + \alpha \cdot \text{Random}(t)) t} \quad (5)$$

6. Перевірка математичної моделі на адекватність

З метою перевірки математичної моделі на адекватність (відповідність) експериментальним даним (рис.1), використовуємо математичний розв'язок зворотної задачі:

$$k_i = -\frac{1}{t_i} \ln \left(\frac{C_{\text{exp}_i}}{C_0} \right).$$

Тут k_i - точкові (дискретні) значення константи швидкості дифузійного розчинення продукту, що одержані на основі експериментальних розподілів концентрацій C_{exp_i} (рис.1). Далі, за одержаним значенням k_i , шляхом їх усереднення визначаємо значення константи швидкості дифузійного розчинення k_r , що відповідає вказаним експериментальним даним (рис.1):

$$k_r = \frac{1}{n} \sum_{i=1}^k k_i.$$

та будуємо їх відповідні графічні розподіли:

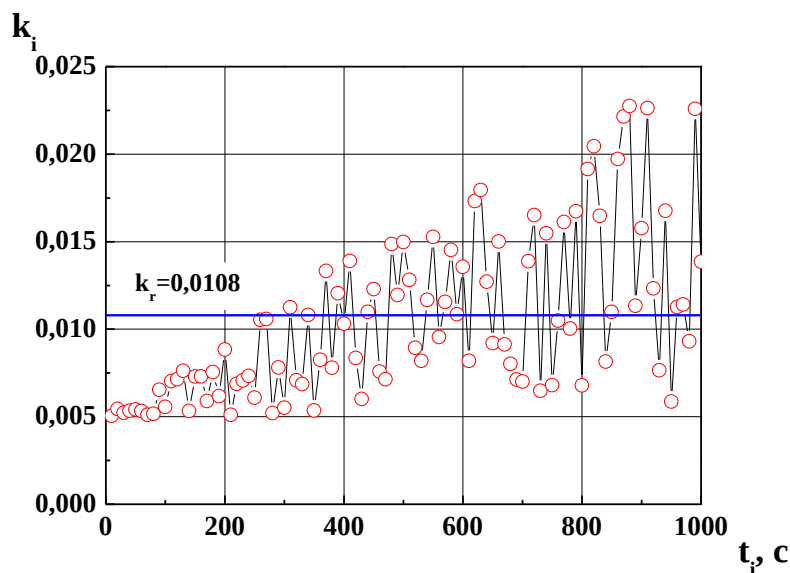


Рис.2 - Графічний розподіл константи швидкості дифузійного розчинення $k_r = 0,0108$.

7. Перевірка математичної моделі на адекватність

З використанням результату перевірки математичної моделі на адекватність, визначивши константу швидкості дифузійного розчинення продукту k_r як розв'язок зворотної задачі (1)-(2), одержуємо модельний розподіл концентрації розчиненої речовини (в твердій фазі) (рис.3):

$$C_{\text{mod}}(t) = C(t) = C_0 e^{-k_r t}.$$

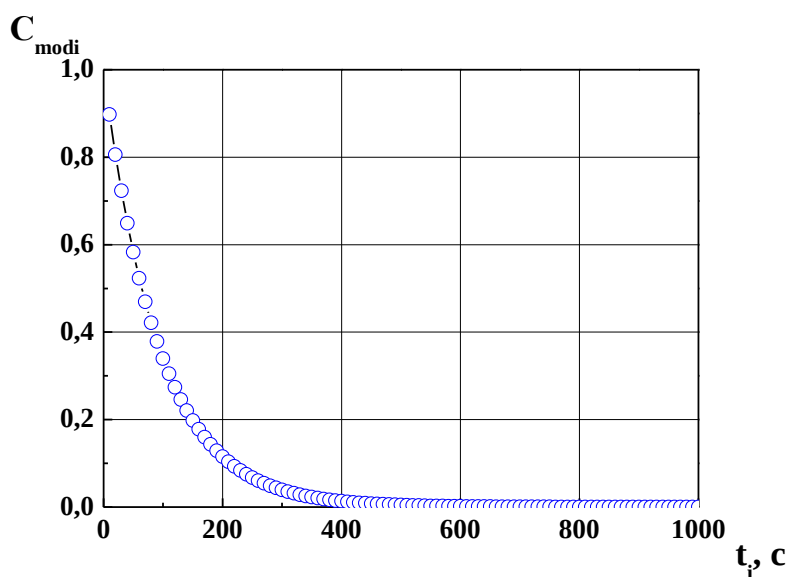


Рис. 3 - Модельний розподіл концентрації розчиненої речовини

8. Порівняльний аналіз з даними експерименту

Наклавши на графік модельного розподілу концентрацій (рис.3) експериментальні точки (рис.1), отримуємо порівняльну якісну картину щодо відповідності результатів математичного моделювання (C_{mod}) результатам фізичного експерименту (C_{exp}) (рис.4):

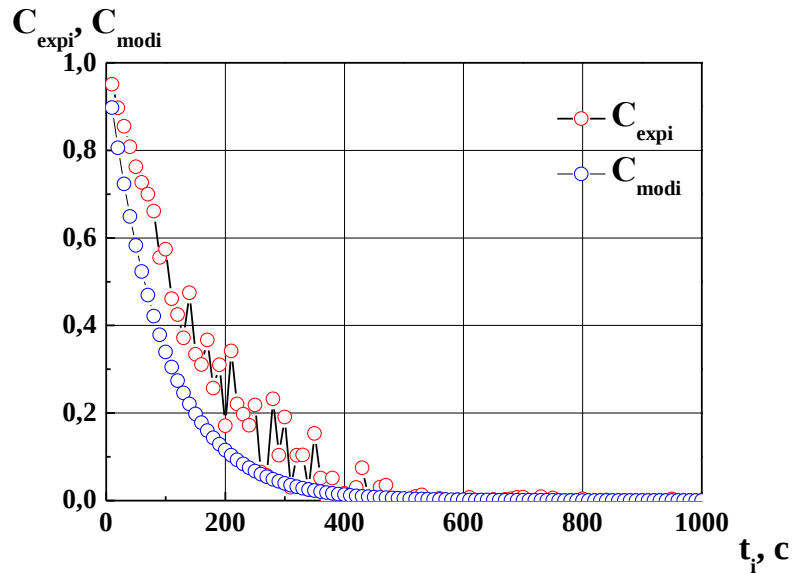


Рис. 4 - Порівняння результатів математичного моделювання і фізичного експерименту

Як видно з рис.4, результати математичного моделювання добре корелюють з результатами фізичного експерименту. Величини абсолютної похибки ΔC_i для кожної з розрахованих точок залежностей визначаються за формулою:

$$\Delta C_i = |C_{\text{mod}_i} - C_{\text{exp}_i}|.$$

Результати розрахунку похибки, що характеризує оцінку ефективності математичної моделі подано на Рис. 5.

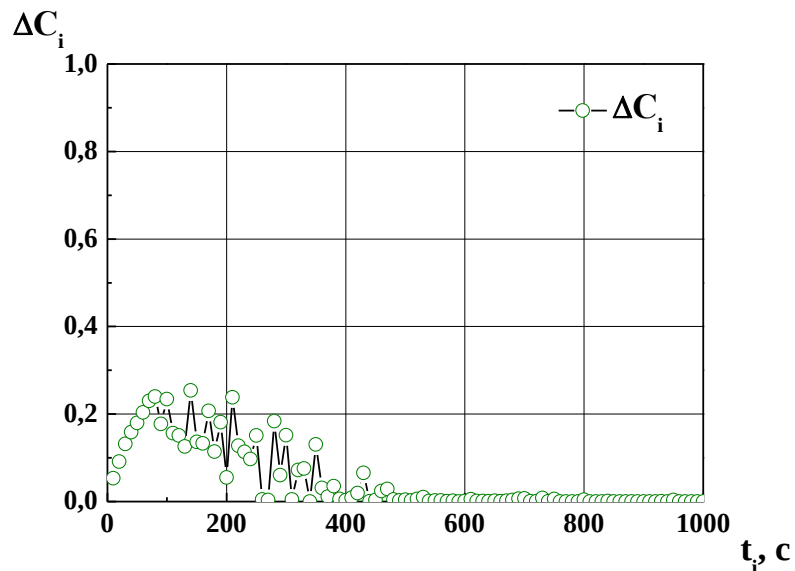


Рис. 5 - Залежність похибки абсолютної похибки ΔC_i при оцінці ефективності математичної моделі

Як видно з Рис. 5, на початковому етапі процесу (стадія експоненціального спуску — діапазон часу 0—50 с), спостерігається достатньо добра кореляція розрахункових (модельних) і експериментальних параметрів. Для проміжного діапазону (стадія пологого зниження концентрації, діапазон 200 - 300 с) спостерігається майже повне накладання модельних і експериментальних залежностей. Для стадії стабілізації (діапазон часу більше 400 с) спостерігаємо практично повне спів падіння модельних і експериментальних значень розрахованих залежностей.

Висновки. Результатом проведеного дослідження з математичного моделювання процесу розчинення з використанням даних фізичного експерименту є розробка ефективного програмного інструментарію для оцінки ефективності використовуваної математичної моделі. Розраховано точне значення константи швидкості дифузійного розчинення, що відповідає проведеним експериментам і забезпечує високий ступінь відповідності і узгодженості модельних і експериментальних розподілів концентрацій розчиненого продукту впродовж всієї тривалості протікання технологічного процесу. Запропонований якісний і кількісний аналіз кінетики процесу дозволяє суттєво знизити затрати часу на проведення експериментальних

досліджень, підвищити їх якість, отримати оптимальні енергозберігаючі технологічні параметри, що забезпечують досліджуваного інтенсифікацію процесу в цілому.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання процесів вільних та вимушених коливань механічної системи

Equations

```
DSolve[{C'[t] == -kr C[t], C[0] == C0}, C[t], t]
{{C[t] -> C0 e^{-kr t}}}
```

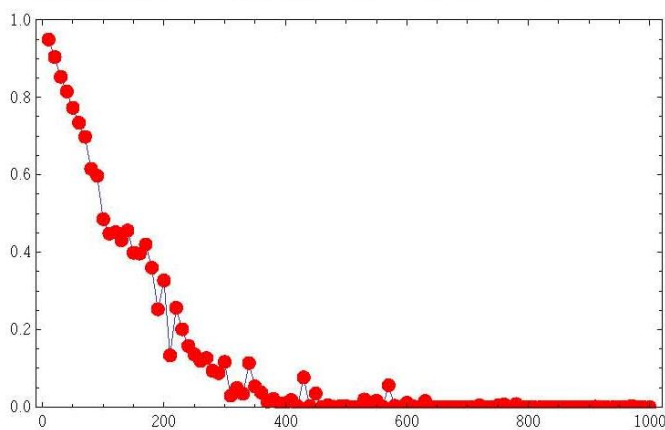
Problem data

```
C0 := 1;
ke := 0.005;
α := 0.0045;
Cexp[t_] := C0 Exp[-ke * (1 + α * RandomReal[t]) t];
```

Experimental dependence

```
SetDirectory["f:\\"];
a = 0;
i = 0;
For[t = 10, t ≤ 1000,
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];
  WriteString[qwe, t, "\\t"];
  Close[qwe];
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[N[Cexp[t]]], "\\n"];
  Close[qwe];
  a = a + N[ $\frac{-1}{t} * \text{Log}[Cexp[t]]$ ];
  i = i + 1;
  kr = a / i;
  Print["Concentration= ", CForm[N[Cexp[t]]]];
  Print["i= ", CForm[i]];
  Print["a= ", CForm[a]];
  Print["krec= ", CForm[krec]];
  t = t + 10];

ListLinePlot[Table[{t, Cexp[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 1},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Red],
  PlotLegends -> "Expressions", Frame -> True]
```



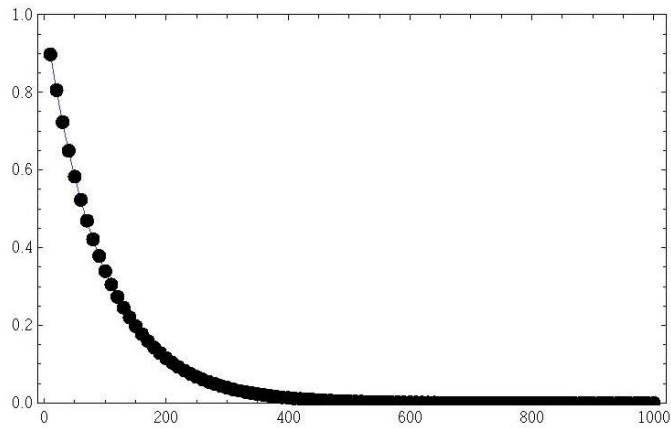
Modelling dependence

```

C0 := 1;
kr := 0.0108;
Cmod[t_] := C0 Exp[-kr * t];

ListLinePlot[Table[{t, Cmod[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 1},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Black],
  PlotLegends -> "Expressions", Frame -> True]

```



```

SetDirectory["f:\\"];
For[t = 10, t ≤ 1000,
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, t, "\\t"];
  Close[qwe];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[N[Cmod[t]]], "\\n"];
  Close[qwe];
  Print["Concentration= ", CForm[N[Cmod[t]]]];
  t = t + 10];

```

ЛАБОРАТОРНА РОБОТА № 2

"МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ДРУГОГО І ТРЕТЬОГО ПОРЯДКУ З ПЕРЕТВОРЕННЯМ ПРОДУКТУ"

1. Фізична постановка задачі

Розглядається технологічний процес другого (або третього) порядку з утворенням продукту шляхом хімічного перетворення (реакції).

2. Математична модель процесу: диференціальні рівняння процесу

Процес такого технологічного процесу може описуватися за допомогою такої математичної моделі з диференціальним рівнянням типу другого порядку у вигляді:

$$\frac{\partial C}{\partial t} = -k_r (a - C)(b - C) \quad (1)$$

з початковою умовою:

$$C(t)|_{t=0} = C_0 \quad (2)$$

Тут $C(t)$ – розподіл приведених концентрацій утворюваного продукту від часу t ; k_r – константа швидкості технологічного процесу, що визначається із даних фізичного експерименту; a , b – початкові значення безрозмірних концентрацій вихідних компонентів сировини, що беруть участь у технологічному процесі.

3. Вхідні дані

$t = 1, 10..1200$ - діапазон зміни часу t для визначення неперервного розподілу концентрацій продукту;

$n = 120$ - кількість точок обчислення;

$t_0 = 10$ - початкове значення часу t для визначення неперервного розподілу концентрацій продукту;

$\Delta t = 10$ - крок зміни часу t для визначення дискретного розподілу концентрацій продукту;

$i = 1..1000$;

$\tau_i = t_0 + i \cdot \Delta t$ - діапазон інтервалів зміни часу для розрахунку дискретного

концентрацій розчинюваного продукту;

$a = 0,003631$

$b = 0,00005$ - початкове значення приведених концентрацій вхідних

компонентів.

4. Параметри і залежності для симуляції фізичного експерименту

$k_e := 0,003631$ - параметр, що симулює експоненціальний ріст(спад) концентрації; :

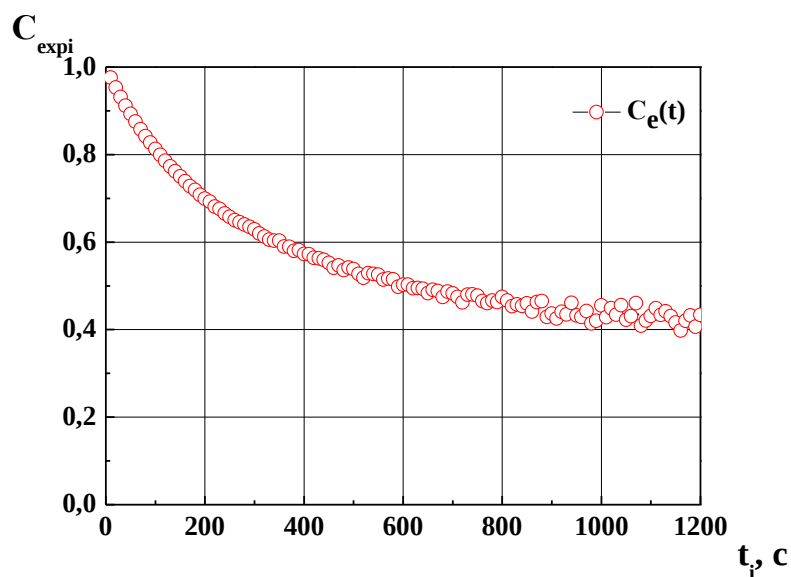
$\alpha := 0,0005$ - параметр, що симулює ступінь імпульсивності випадкових вимірювань концентрацій;

RandomReal - генератор дійсних випадкових величин.

Залежність концентрації розчинюваної речовини у експерименті задається співвідношенням:

$$C_{\text{exp}_i} = \frac{a(b-1)e^{a(1+\alpha\text{Random}(t_i))k_e t_i} - b(a-1)e^{b(1+\alpha\text{Random}(t_i))k_e t_i}}{(b-1)e^{a(1+\alpha\text{Random}(t_i))k_e t_i} - (a-1)e^{b(1+\alpha\text{Random}(t_i))k_e t_i}} \cdot$$

Задана залежність має вигляд:



5. Алгоритмізація побудови розв'язку математичної моделі

Розв'язком математичної моделі, що описується диференціальним рівнянням першого порядку (1) та початковими умовами (2) є залежність:

$$C_e(t) = \frac{a(b-1)e^{ak_e t} - b(a-1)e^{bk_e t}}{(b-1)e^{ak_e t} - (a-1)e^{bk_e t}} \quad (4)$$

де значення константи швидкості внутрішньої кінетики протікання процесу визначається із експериментальних даних (рис.1) шляхом розв'язання зворотної задачі.:

$$C_e(t) = \frac{a(b-1)e^{a(1+\alpha\text{Random}(t))k_e t} - b(a-1)e^{b(1+\alpha\text{Random}(t))k_e t}}{(b-1)e^{a(1+\alpha\text{Random}(t))k_e t} - (a-1)e^{b(1+\alpha\text{Random}(t))k_e t}} \quad (5)$$

6. Перевірка математичної моделі на адекватність

З метою перевірки математичної моделі на адекватність (відповідність) експериментальним даним (рис.1) використовуємо математичний розв'язок зворотної задачі:

$$k_i = -\frac{1}{t_i \left(1 - \frac{b}{a}\right)} \ln \left(\frac{a - aC_{\text{exp}_i}}{b - aC_{\text{exp}_i}} \frac{b}{a} \right), \quad k_r = \frac{1}{n} \sum_{i=1}^{120} k_i, \quad k_r = 0,0006.$$

Тут k_i – точкові (дискретні) значення константи швидкості протікання технологічного процесу, одержані на основі експериментальних розподілів зміни концентрацій вхідних компонентів (рис.1). Далі, за одержаним значенням k_i , шляхом їх усереднення визначаємо значення константи швидкості k_r , що відповідає вказаним експериментальним даним (рис.1) та будуємо їх графічні розподіли (рис.2):

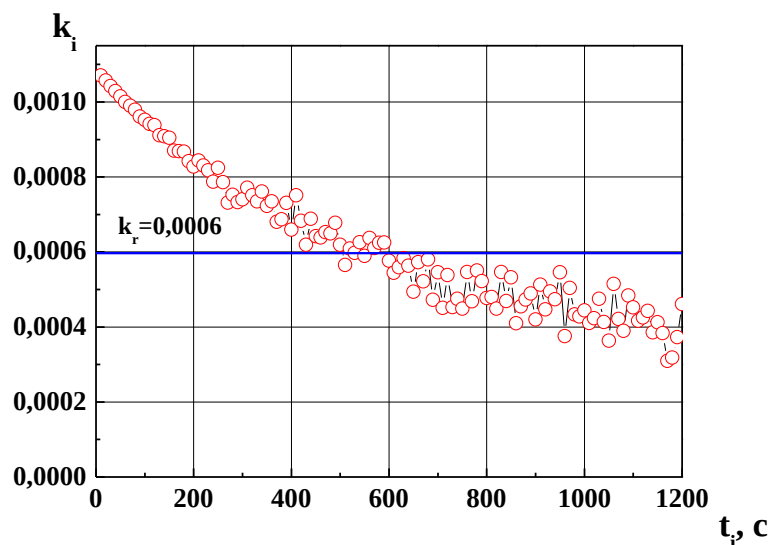


Рис.2 - Розподіл розрахункових залежностей констант швидкості протікання технологічного процесу і їх середнього значення від часу $k_r = 0,0006$.

7. Перевірка математичної моделі на адекватність

Використовуючи результат перевірки математичної моделі на адекватність, визначивши константу швидкості протікання технологічного процесу k_r як розв'язок зворотної задачі (1)-(2), одержуємо модельний розподіл концентрації продукту (рис.3):

$$C_{\text{mod}}(t) = C(t) = \frac{a(b-1)e^{ak_e t} - b(a-1)e^{bk_e t}}{(b-1)e^{ak_e t} - (a-1)e^{bk_e t}}.$$

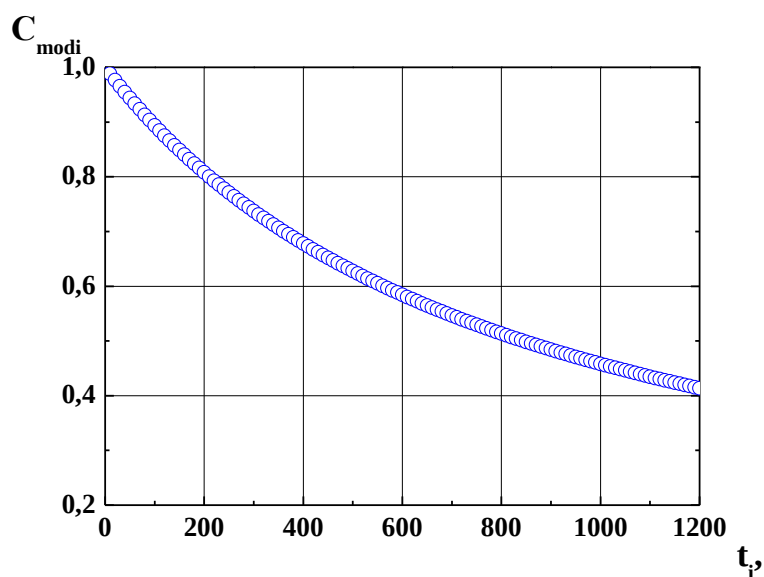


Рис. 3 - Модельний розподіл концентрації продукту

8. Порівняльний аналіз з даними експерименту

Наклавши на графік модельного розподілу концентрацій (рис.3) експериментальні точки (рис.1), отримуємо порівняльну якісну картину щодо відповідності результатів математичного моделювання (C_{mod}) результатам фізичного експерименту (C_{exp_i}), (рис.4):

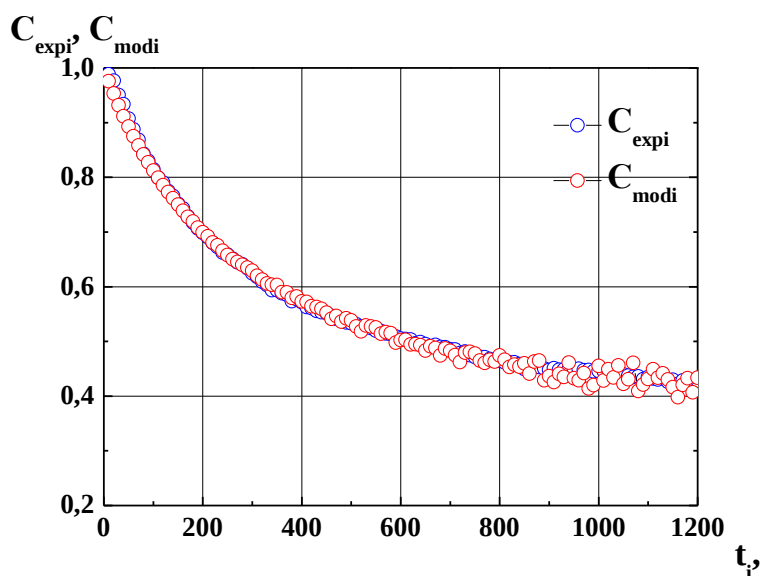


Рис. 4 - Порівняльний аналіз модельних розподілів зміни концентрацій вихідних компонентів з даними фізичного експерименту

Як видно з рис.4, результати математичного моделювання добре корелюються з результатами фізичного експерименту. Величина відносного відхилення (похибки) ΔC_i визначається за формулою (5). На рис.5 показана графічна залежність похибки для оцінки ефективності математичної моделі.

$$\Delta C_i = |C_{\text{mod}_i} - C_{\text{exp}_i}|.$$

Результати розрахунку похибки, що характеризує оцінку ефективності математичної моделі подано на Рис. 5.

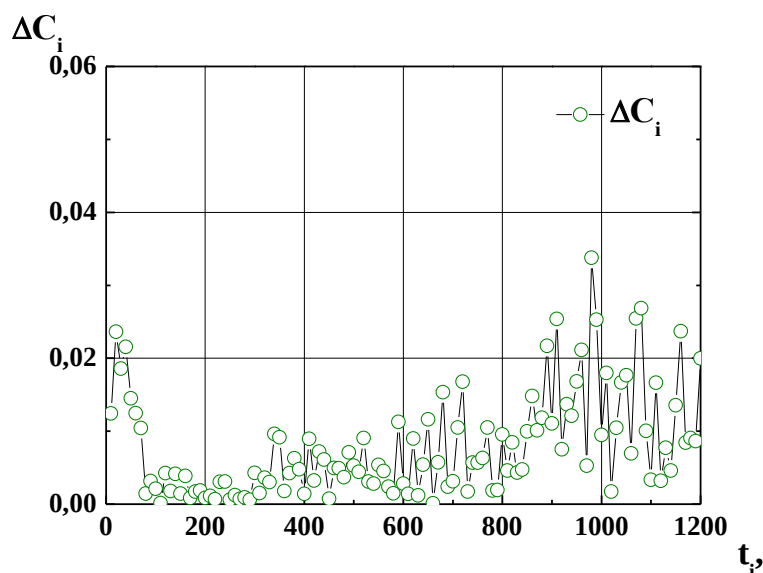


Рис5 - Залежність похибки абсолютної похибки ΔC_i при оцінці ефективності математичної моделі

Як видно з рис.4,5, на початковій стадії процесу (стадія експоненціального спуску – діапазон 0-350 с), ми спостерігаємо відносно добру кореляцію розрахункових (модельних) і експериментальних параметрів. Максимальне значення відносної похибки для цього випадку не перевищує 2%. Для проміжного діапазону (стадія пологого зниження концентрації, діапазон 350-900 с) спостерігаємо майже повне співпадіння модельних і експериментальних значень. Найбільше значення похибки для цього випадку не перевищує 4%. Для стадії стабілізації (діапазон 900 с і вище) спостерігаємо практично повне співпадіння модельних і експериментальних значень, значення відносної похибки в окремих точках цього діапазону досягає 4%, що знаходиться в допустимих межах розрахунків інженерного аналізу.

Висновки. Результатом проведеного дослідження з математичного моделювання технологічного процесу 2-го (3-го) порядку з хімічним перетворенням продукту з використанням даних фізичного експерименту є розробка ефективного інструментарію для оцінки ефективності математичної моделі. Розраховано точне значення константи швидкості такого виду масопереносу, що відповідає проведеним експериментам і забезпечує

високий ступінь відповідності і узгодженості модельних і експериментальних розподілів концентрацій вихідних компонентів перетворення та утворюваного продукту впродовж всієї тривалості протікання технологічного процесу. Такий якісний і кількісний аналіз кінетики процесу дозволить суттєво знизити витрати на проведення експериментальних досліджень, підвищити їх якість, отримати оптимальні енергозберігаючі технологічні параметри, що забезпечує інтенсифікацію процесу в цілому.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання процесів вільних та вимушених коливань механічної системи

Equations

```
DSolve[{C'[t] == -kr C[t], C[0] == C0}, C[t], t]
{{C[t] -> C0 e^{-kr t}}}
```

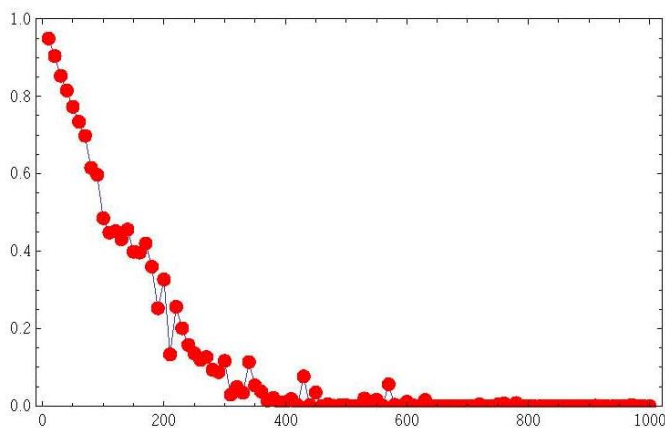
Problem data

```
C0 := 1;
ke := 0.005;
α := 0.0045;
Cexp[t_] := C0 Exp[-ke * (1 + α * RandomReal[t]) t];
```

Experimental dependence

```
SetDirectory["f:\\"];
a = 0;
i = 0;
For[t = 10, t ≤ 1000,
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];
  WriteString[qwe, t, "\\t"];
  Close[qwe];
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[N[Cexp[t]]], "\\n"];
  Close[qwe];
  a = a + N[ $\frac{-1}{t} * \text{Log}[Cexp[t]]$ ];
  i = i + 1;
  kr = a / i;
  Print["Concentration= ", CForm[N[Cexp[t]]]];
  Print["i= ", CForm[i]];
  Print["a= ", CForm[a]];
  Print["krec= ", CForm[krec]];
  t = t + 10];

ListLinePlot[Table[{t, Cexp[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 1},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Red],
  PlotLegends -> "Expressions", Frame -> True]
```



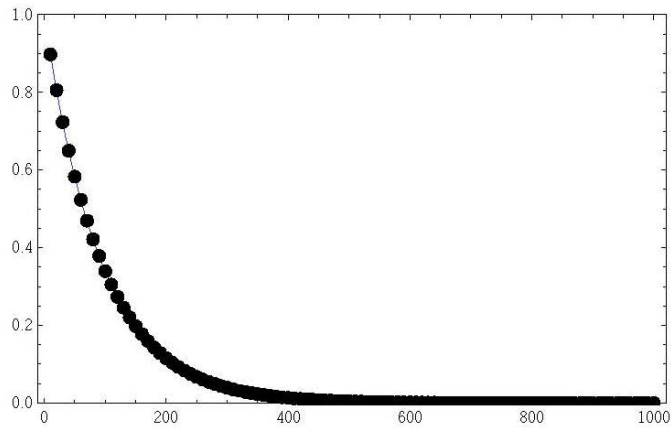
Modelling dependence

```

C0 := 1;
kr := 0.0108;
Cmod[t_] := C0 Exp[-kr * t];

ListLinePlot[Table[{t, Cmod[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 1},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Black],
  PlotLegends -> "Expressions", Frame -> True]

```



```

SetDirectory["f:\\"];
For[t = 10, t ≤ 1000,
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, t, "\\t"];
  Close[qwe];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[N[Cmod[t]]], "\\n"];
  Close[qwe];
  Print["Concentration= ", CForm[N[Cmod[t]]]];
  t = t + 10];

```

ЛАБОРАТОРНА РОБОТА № 3

"МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ НЕЛІНІЙНОГО ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ДИФУЗІЇ /НАГРІВУ/ СУСПЕНЗІЙ ЗІ ЗМІННИМ КОЕФІЦІЄНТОМ ДИФУЗІЇ /ТЕПЛОПРОВІДНОСТІ/, ЩО ВРАХОВУЄ ВІДХИЛЕННЯ РЕАЛЬНОГО ПРОЦЕСУ ВІД КЛАСИЧНОГО"

1. Фізична постановка задачі

Розглядається технологічний процес дифузії харчових суспензій, що супроводжується нагрівом суспензій. Це призводить до деякого відхилення кінетики процесу від класичної схеми. Врахування цього важливого фізичного чинника враховується шляхом використання змінного коефіцієнту дифузії, як функції від концентрації суспензії (розчину).

2. Математична модель процесу: диференціальні рівняння процесу

Процес такого масопереносу може бути описаний за допомогою такої математичної моделі дифузійного типу у вигляді:

$$\frac{\partial C}{\partial t} = -k_r(C)C = -(D_0 + \beta C)C \quad (1)$$

з початковою умовою:

$$C(t)|_{t=0} = C_0 \quad (2)$$

Тут $C(t)$ – розподіл приведених концентрацій дифундованої речовини від часу t ; D_0, β – компоненти коефіцієнту дифузії (константа швидкості процесу, що визначається із даних фізичного експерименту); C_0 – початкове значення приведеної концентрації дифундованого продукту.

3. Вхідні дані

$t = 1, 10..1000$ - діапазон зміни часу t для визначення неперервного розподілу концентрацій дифундованої речовини;

$n = 100$ - кількість точок обчислення;

$t_0 = 10$ - початкове значення часу t для визначення неперервного розподілу концентрацій дифундованої речовини;

$\Delta t = 10$ - крок зміни часу t для визначення дискретного розподілу концентрацій розчинюваного продукту;

$i = 1..1000$;

$\tau_i = t_0 + i \cdot \Delta t$ - діапазон інтервалів зміни часу для розрахунку дискретного

концентрацій дифундованої речовини;

$C_0 := 1$ - початкове значення приведеної концентрації дифундованої речовини;

$\beta = 0,5$ - значення компоненти дифузії.

4. Параметри і залежності для симуляції фізичного експерименту

$D_0 := 0,005$ - параметр, що симулює експоненціальний ріст(спад) концентрації;

$\alpha := 0,002$ - параметр, що симулює ступінь імпульсивності випадкових вимірювань концентрацій;

RandomReal - генератор дійсних випадкових величин.

Залежність концентрації дифундованої речовини у експерименті задається співвідношенням:

$$C_{\text{exp}_i} = \frac{D_0}{-\beta + e^{D_0 (1 + \alpha \cdot \text{Random}(t_i)) t_i} (D_0 + \beta)}.$$

Задана залежність має вигляд:

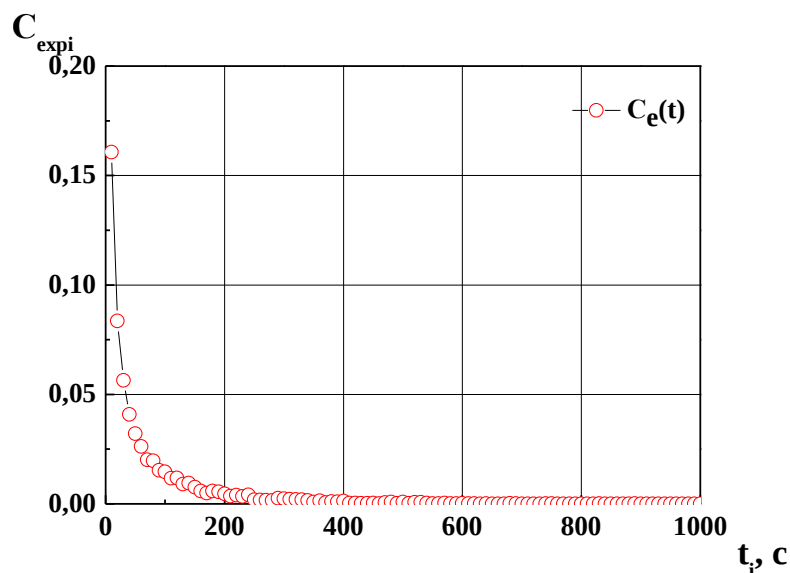


Рис. 1 - Експериментальні дані розподілу концентрацій дифундованої речовини

5. Алгоритмізація побудови розв'язку математичної моделі

Розв'язком математичної моделі, що описується диференціальним рівнянням першого порядку (1) та початковими умовами (2) є залежність:

$$C(t) = \frac{D_0}{-\beta + e^{D_0 t} (D_0 + \beta)} \quad (4)$$

де значення константи швидкості дифузійного розчинення, що визначає внутрішню кінетику протікання процесу, визначається із експериментальних даних (рис.1) шляхом розв'язання зворотної задачі.

6. Перевірка математичної моделі на адекватність

З метою перевірки математичної моделі на адекватність (відповідність) експериментальним даним (рис.1), використовуємо математичний розв'язок зворотної задачі:

$$D_i = \frac{1}{t_i} \ln \left(\frac{D_0 + \beta C_{\text{exp}_i}}{C_{\text{exp}_i} (D_0 + \beta)} \right).$$

Тут D_i – точкові (дискретні) значення константи швидкості процесу, одержані на основі експериментальних розподілів концентрацій C_{exp_i} (рис.1).

Далі, за одержаним значенням D_i , шляхом їх усереднення визначаємо значення константи швидкості процесу D_0 , що відповідає вказаним експериментальним даним (рис.1):

$$D_0 = \frac{1}{n} \sum_{i=1}^{100} D_i, D_0 = 0,00735.$$

та будуємо їх відповідні графічні розподіли:

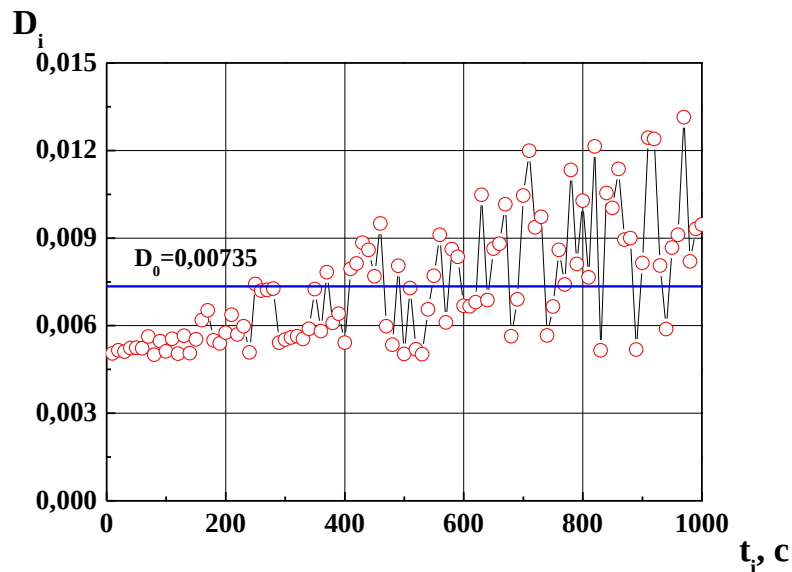


Рис.2 - Графічний розподіл константи швидкості дифузійного розчинення

$$D_0 = 0,00735.$$

7. Перевірка математичної моделі на адекватність

Використовуючи результат перевірки математичної моделі на адекватність, визначивши константу швидкості D , як розв'язок зворотної задачі (1)-(2) одержуємо модельний розподіл концентрації дифундованої речовини (рис.3):

$$C_{\text{mod}_i}(t) = C_i(t) = \frac{D_0}{-\beta + e^{D_0 t_i} (D_0 + \beta)}.$$

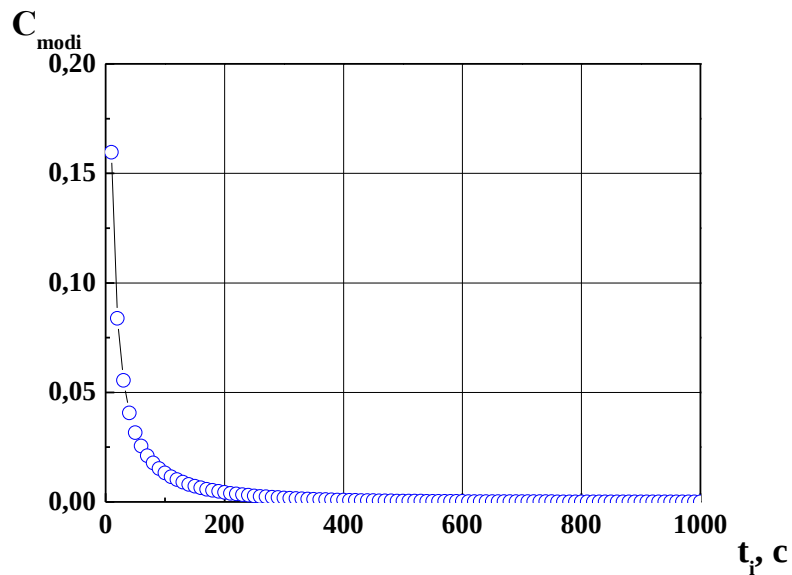


Рис. 3 - Модельний розподіл концентрації продукту

8. Порівняльний аналіз з даними експерименту

Наклавши на графік модельного розподілу концентрацій (рис.3) експериментальні точки (рис.1), отримуємо порівняльну якісну картину щодо відповідності результатів математичного моделювання (C_{mod}) результатам фізичного експерименту (C_{exp}) (рис.4):

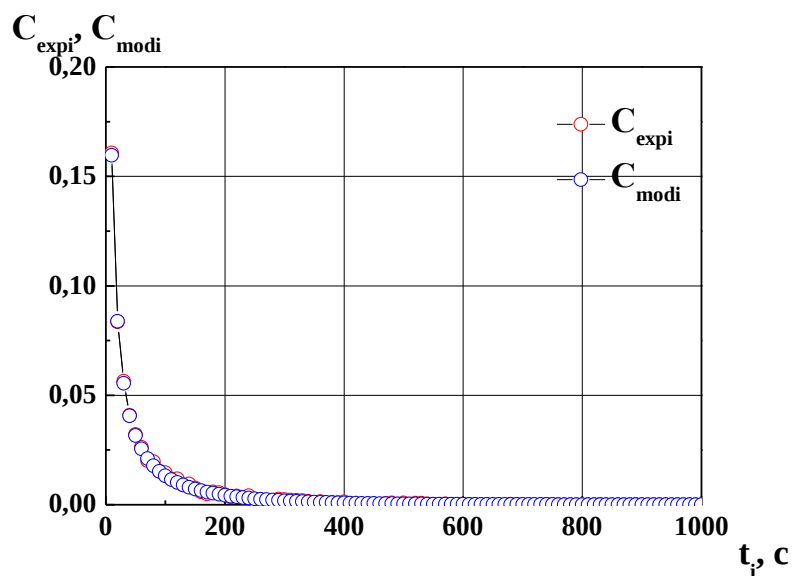


Рис. 4 - Порівняльний аналіз модельного розподілу концентрації дифундованої речовини з даними фізичного експерименту

Як видно з рис.4, результати математичного моделювання добре корелюють з результатами фізичного експерименту. Величини абсолютної похибки ΔC_i для кожної з розрахованих точок залежностей визначаються за формулою:

$$\Delta C_i = |C_{\text{mod}_i} - C_{\text{exp}_i}|.$$

Результати розрахунку похибки, що характеризує оцінку ефективності математичної моделі подано на Рис. 5.

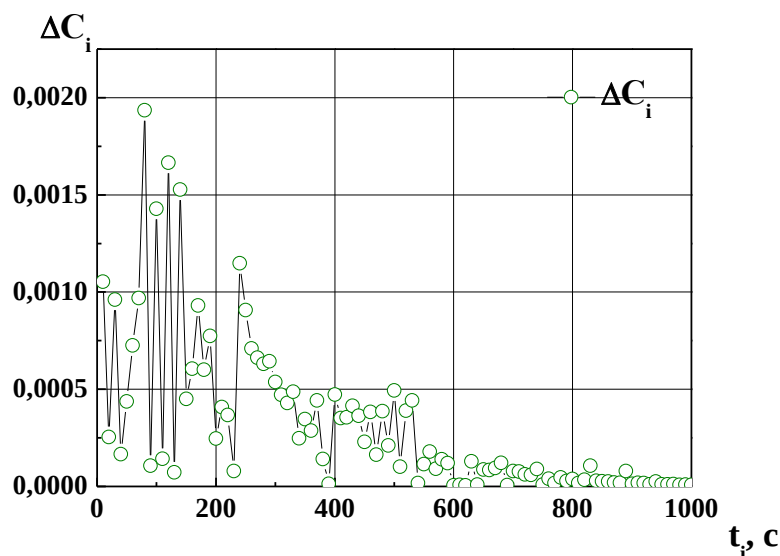


Рис5 - Залежність похибки абсолютної похибки ΔC_i при оцінці ефективності математичної моделі

Як видно з рис.5, на початковій стадії процесу (стадія експоненціального спуску – діапазон 0-100 с), ми спостерігаємо відносно добру кореляцію розрахункових (модельних) і експериментальних параметрів. Максимальне значення відносної похибки для цього випадку становить 0.1%. Для проміжного діапазону (стадія пологого зниження концентрації, діапазон 100-600 с) спостерігаємо майже повне співпадіння модельних і експериментальних значень. Найбільше значення похибки для цього випадку не перевищує 0,13%. Для стадії стабілізації (діапазон 600-1000 с) спостерігаємо практично повне співпадіння модельних і експериментальних значень, значення відносної похибки в окремих точках

цього діапазону досягає 0.03-0.01%, що знаходиться в допустимих межах розрахунків інженерного аналізу.

Висновки. Результатом проведеного дослідження з математичного моделювання процесу дифузії з використанням даних фізичного експерименту є розробка ефективного інструментарію для оцінки ефективності математичної моделі. Розраховано точне значення компонентів коефіцієнту дифузії, що відповідає проведеним експериментам і забезпечує високий ступінь відповідності і узгодженості модельних і експериментальних розподілів концентрацій дифундованого продукту впродовж всієї тривалості протікання технологічного процесу. Такий якісний і кількісний аналіз кінетики процесу дозволить суттєво знизити витрати на проведення експериментальних досліджень, підвищити їх якість, отримати оптимальні енергозберігаючі технологічні параметри, що забезпечує інтенсифікацію процесу в цілому.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання нелінійного технологічного процесу дифузії /нагріву/ суспензій зі змінним коефіцієнтом дифузії /теплопровідності/, що враховує відхилення реального процесу від класичного

Equations

```
In[4]= FullSimplify[DSolve[{C'[t] == -(D0 + β * C[t]) * C[t], C[0] == 1}, C[t], t]]
```

Solve::ifun : Inverse functions are being used by Solve, so
some solutions may not be found; use Reduce for complete solution information. >>

```
Out[4]= {{C[t] -> \frac{D0}{-β + e^{D0 t} (D0 + β)}}}
```

```
In[1]= FullSimplify[Solve[\frac{D0}{-β + e^{D1 t} (D0 + β)} = C1, D1]]
```

```
Out[1]= {{D1 -> ConditionalExpression[\frac{2 i \pi C[1] + \text{Log}[\frac{D0+C1 \beta}{C1 (D0+\beta)}}]{t}, C[1] \in \text{Integers}}]}
```

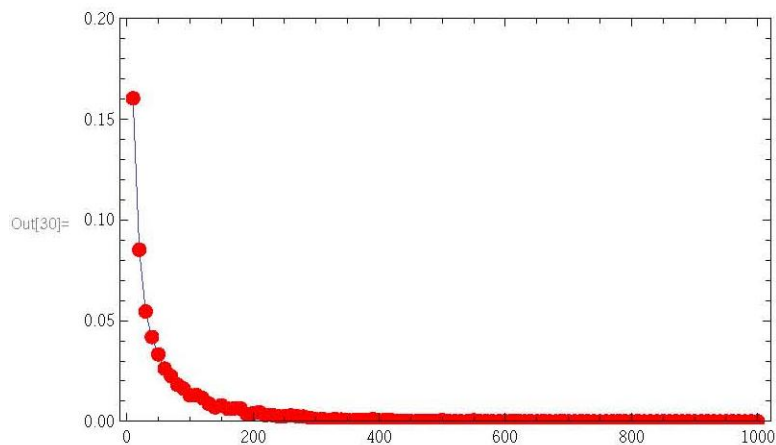
Problem data

```
In[24]= C0 := 1;  
D0 := 0.005;  
α := 0.002;  
β := 0.5;  
Cexp[t_] := \frac{D0}{-β + e^{D0 (1+α \text{RandomReal}[t]) t} (D0 + β)};
```

Experimental dependence

```
In[15]= SetDirectory["f:\\"];  
a = 0;  
i = 0;  
For[t = 10, t ≤ 1000,  
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];  
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];  
  WriteString[qwe, t, "\\t"];  
  Close[qwe];  
  qwe = OpenAppend["Ce.dat", DOSTextFormat -> True];  
  WriteString[qwe, CForm[N[Cexp[t]]], "\\n"];  
  Close[qwe];  
  a = a + N[\frac{\text{Log}[\frac{D0+Cexp[t] \beta}{Cexp[t] (D0+\beta)}}{t}];  
  i = i + 1;  
  kr = a / i;  
  Print["Concentration= ", CForm[N[Cexp[t]]]];  
  Print["i= ", CForm[i]];  
  Print["a= ", CForm[a]];  
  Print["krec= ", CForm[kr]];  
  t = t + 10];
```

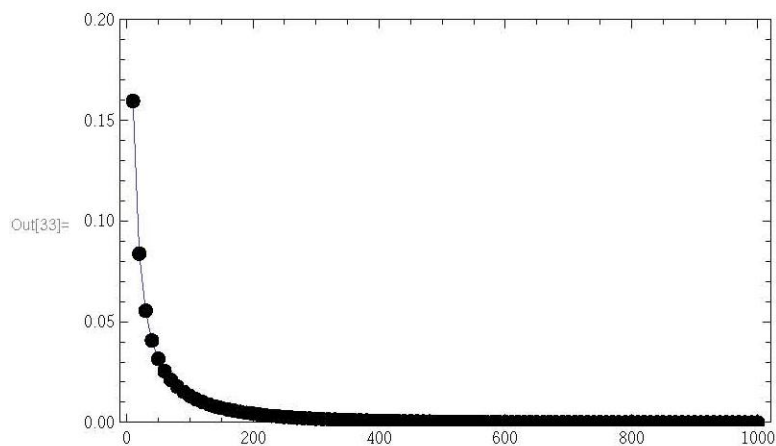
```
In[30]= ListLinePlot[Table[{t, Cexp[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 0.2},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Red],
  PlotLegends -> "Expressions", Frame -> True]
```



Modelling dependence

```
In[29]= C0 := 1;
D0 := 0.00735;
β := 0.5;
Cmod[t_] := 
$$\frac{D0}{-\beta + e^{D0 t} (D0 + \beta)}$$
;
```

```
In[33]= ListLinePlot[Table[{t, Cmod[t]}, {t, 10, 1000, 10}], PlotRange -> {0, 0.2},
  MaxPlotPoints -> 100, Mesh -> Full, MeshStyle -> Directive[PointSize[Large], Black],
  PlotLegends -> "Expressions", Frame -> True]
```



```
In[34]= SetDirectory["f:\\"];
For[t = 10, t ≤ 1000,
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, t, "\\t"];
  Close[qwe];
  qwe = OpenAppend["Cmod.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[N[Cmod[t]]], "\\n"];
  Close[qwe];
  Print["Concentration= ", CForm[N[Cmod[t]]]];
  t = t + 10];
```

ЛАБОРАТОРНА РОБОТА № 4

"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ У МОДЕЛЮВАННІ ЕНЕРГЕТИЧНОГО СПЕКТРУ ЗАКРИТИХ НАПІВПРОВІДНИКОВИХ НАНОСТРУКТУР"

1. Фізична постановка задачі

Розглядається фізичний процес поглинання електронів напівпровідниковою наноструктурою з відомими геометричними та фізичними параметрами (рис. 1).

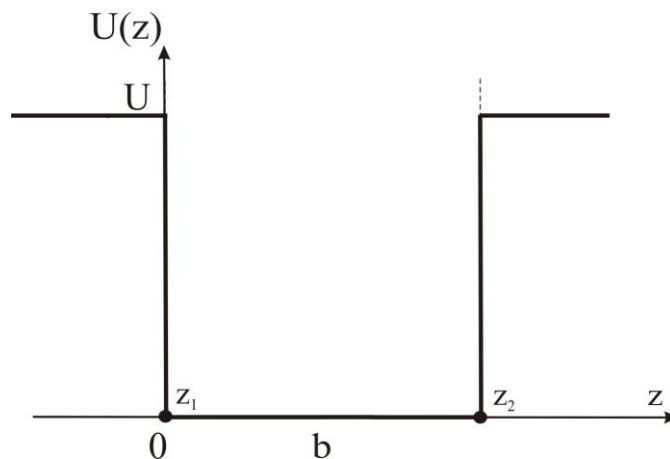


Рис. 1. Потенціальний профіль закритої наноструктури

2. Математична модель процесу: диференціальні рівняння

Шредінгера

Процес поглинання електронів закритою напівпровідниковою наноструктурою описується диференціальними рівняннями другого порядку – рівняннями Шредінгера, записаними для кожної з областей наноструктури :

$$\begin{cases} \frac{d^2\psi_1}{dz^2} - \chi^2\psi_1 = 0, z < 0, \\ \frac{d^2\psi_2}{dz^2} + k^2\psi_2 = 0, 0 \leq z \leq b, \\ \frac{d^2\psi_3}{dz^2} - \chi^2\psi_3 = 0, z > b. \end{cases} \quad (1)$$

де

$$\chi = \sqrt{\frac{2m_e(U - E)}{\hbar^2}}; k = \sqrt{\frac{2m_e E}{\hbar^2}}.$$

Для розв'язків рівнянь (1) виконуються граничні умови:

$$\begin{cases} \psi_1(z)|_{z=0} = \psi_2(z)|_{z=0}, \\ \left. \frac{d\psi_1(z)}{dz} \right|_{z=0} = \left. \frac{d\psi_2(z)}{dz} \right|_{z=0}, \\ \psi_2(z)|_{z=b} = \psi_3(z)|_{z=b}, \\ \left. \frac{d\psi_2(z)}{dz} \right|_{z=b} = \left. \frac{d\psi_3(z)}{dz} \right|_{z=b}. \end{cases} \quad (2)$$

Розв'язки рівнянь (1) мають вигляд:

$$\begin{cases} \psi_1 = A_1 e^{\chi z}, z < 0, \\ \psi_2 = A_2 \cos kz + B_2 \sin kz, 0 \leq z \leq b, \\ \psi_3 = B_3 e^{-\chi z}, z > b. \end{cases} \quad (3)$$

Тоді з системи (2) знаходимо:

$$\begin{cases} A_1 = A_2, \\ \chi A_1 = k B_2, \\ A_2 \cos kb + B_2 \sin kb = B_3 e^{-\chi b}, \\ k(-A_2 \sin kb + B_2 \cos kb) = -\chi B_3 e^{-\chi b}, \end{cases} \quad \begin{cases} A_1 - A_2 = 0, \\ \chi A_1 - k B_2 = 0, \\ A_2 \cos kb + B_2 \sin kb - B_3 e^{-\chi b} = 0, \\ -k A_2 \sin kb + k B_2 \cos kb + \chi B_3 e^{-\chi b} = 0. \end{cases} \quad (4)$$

Спектр електрона в потенціальній ямі визначається з умови однозначності існування розв'язків системи (4), тобто з умови, що визначник системи рівний нулю, тобто:

$$\Delta = \begin{vmatrix} 1 & -1 & 0 & 0 \\ \chi & -k & 0 & 0 \\ 0 & \cos kb & \sin kb & -e^{-\chi b} \\ 0 & -k \sin kb & k \cos kb & \chi e^{-\chi b} \end{vmatrix} = 0 \quad (5)$$

або (використовуючи функцію Det [] для обчислення визначника):

$$\Delta = e^{-\chi b} (2k\chi \cos kb + \sin kb)(\chi^2 - k^2). \quad (6)$$

Енергетичний спектр електрона визначається розв'язками трансцендентного рівняння (6):

$$e^{-\chi b} (2k\chi \cos kb + \sin kb)(\chi^2 - k^2) = 0,$$

зокрема зразу знаходимо два асимптотичні розв'язки $E_0 = 0$, що не має фізичного змісту, та $E_\infty = U$.

3. Вхідні дані

$b = 10$ нм - товщина потенціальної ями;

$m_e = 0,04585 \cdot 10^{-31}$ кг - ефективна маса електрона;

$\hbar = 1,055 \cdot 10^{-34}$ Дж·с - стала Планка.

4. Алгоритмізація побудови розв'язку математичної моделі

Знайдемо графічно корені трансцендентного рівняння (6). Для цього побудуємо графік залежності $\Delta = \Delta(E)$, де $0 \leq E \leq U$:

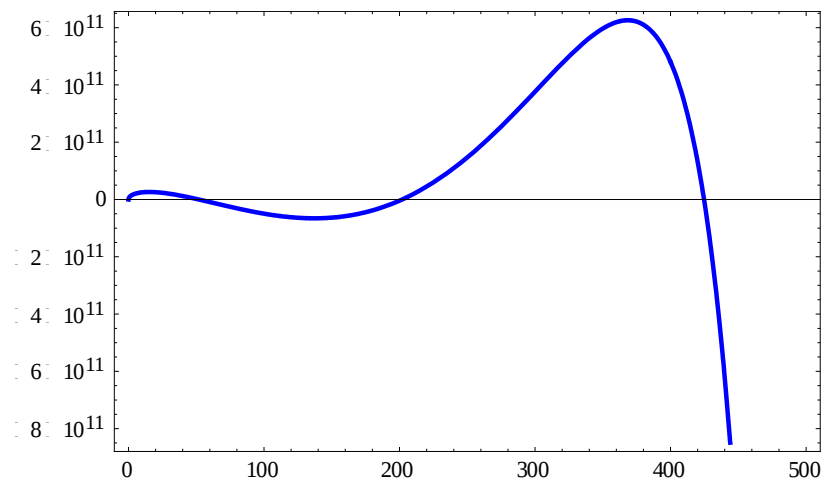


Рис. 1. Графік залежності $\Delta = \Delta(E)$

Знаходимо корені рівняння, користуючись функцією FindRoot:

$$E_1 = 51,777 \text{ меВ}, E_2 = 201,937 \text{ меВ}, E_3 = 424,688 \text{ меВ}.$$

5. *Моделювання залежності енергетичного спектру електрона від геометричних параметрів квантової ями*

З метою отримання залежності $E = E(b)$ будемо змінювати величину b у межах:

$$2 \text{ нм} \leq b \leq 10 \text{ нм}$$

з кроком:

$$\delta = 1 \text{ нм}$$

кожного разу знаходячи корені трансцендентного рівняння $\Delta = \Delta(E)$, для кожної з точок:

$$b_i = n\delta, i = 1, \dots, 10.$$

використовуючи метод ділення відрізка навпіл і вбудовану функцію FindRoot.

Результатом розрахунків є графічна залежність:

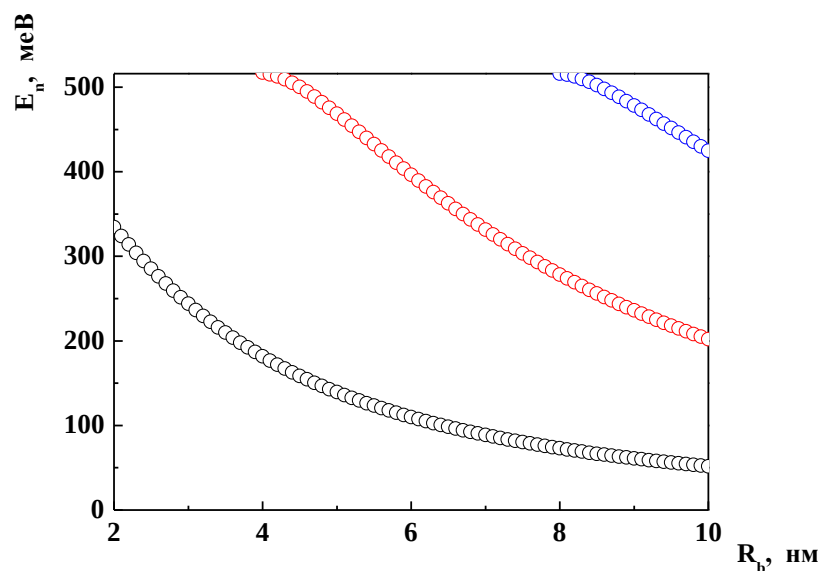


Рис. 1 - Залежність перших трьох енергетичних рівнів електрона від ширини потенціальної ями

Як видно з Рис. 1, залежності перших трьох енергетичних рівнів електрона, локалізованого у потенціальній квантовій ямі зі скінченною висотою зовнішніх потенціальних бар'єрів є однотипними. Їх поведінка у межах використовуваної моделі корелює з теоретичною залежністю, згідно з якою $E_n \sim \frac{n^2}{b^2}$, де b - ширина потенціальної ями, n - номер енергетичного рівня. При цьому всі енергетичні рівні розміщені в інтервалі: $0 < E_n \leq U$.

Висновки. Результатом проведеного дослідження з математичного моделювання енергетичного спектру електрона локалізованого у закритій наносистемі з потенціальними стінками скінченної висоти є отримана графічна залежність $E_n = E_n(b)$, що добре корелює з теоретичними результатами. Розглядувана модель може бути використана для розрахунку спектру електрона в багатошарових напівпровідникових наноструктурах.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання енергетичного спектру закритих напівпровідникових наноструктур

```

Dipsersion equation

In[1]= FullSimplify[
  Det[
    {
      {1, -1, 0, 0},
      {χ[En], 0, -k[En], 0},
      {0, Cos[k[En] Rb], Sin[k[En] Rb], -e-χ[En] Rb},
      {0, -k[En] Sin[k[En] Rb], k[En] Cos[k[En] Rb], χ[En] e-χ[En] Rb}
    }
  ]
]

Out[1]= e-Rb χ[En] (2 Cos[Rb k[En]] k[En] χ[En] + Sin[Rb k[En]] (-k[En]2 + χ[En]2))

Problem data

In[1]= V := 516; (*potential barrier*)
m0 = 0.04585; (*electron effective mass*)

ab := 0.529 * 10-8
Ry := 13 605 (* meV *)
hp := 6.58 * 10-13 (* meV c *)
el := 4.8 * 10-10;

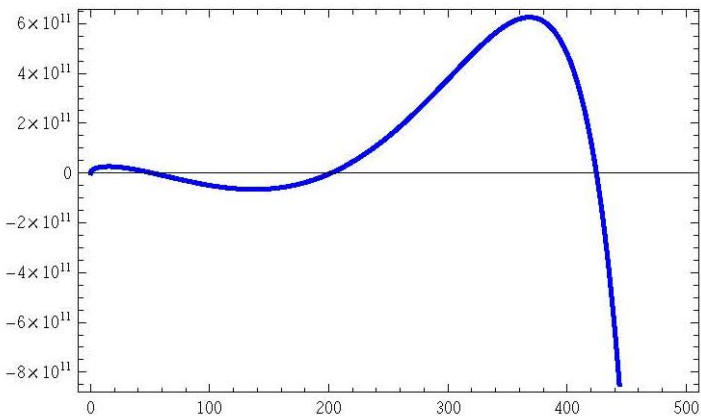
Rb = 10.0 * 10-7; (*L*)

k[En_] := Sqrt[ $\frac{m_0 \text{En}}{Ry}$ ] / ab;

χ[En_] := Sqrt[ $\frac{m_0 (V - \text{En})}{Ry}$ ] / ab;

dispeq[En_] :=
  e-Rb χ[En] (2 Cos[Rb k[En]] k[En] χ[En] + Sin[Rb k[En]] (-k[En]2 + χ[En]2));

In[33]= Plot[dispeq[En], {En, 0, 500}, PlotStyle -> {Thick, Blue}, Frame -> True]

Out[33]=


FindRoot[dispeq[En] == 0, {En, 50}]

{En -> 51.7726}

```

```

FindRoot[dispeq[En] == 0, {En, 200}]

FindRoot::lstol:
  The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was
  unable to find a sufficient decrease in the merit function. You may need more
  than MachinePrecision digits of working precision to meet these tolerances. >>
{En -> 201.937}

FindRoot[dispeq[En] == 0, {En, 420}]

FindRoot::lstol:
  The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was
  unable to find a sufficient decrease in the merit function. You may need more
  than MachinePrecision digits of working precision to meet these tolerances. >>
{En -> 424.688}

= SetDirectory["f:\\"];
For[Rb = 1.5 * 10-7, Rb ≤ 10.0 * 10-7,
  Print[Style["Rb= ", 16, Blue, Bold], Style[Rb, 16, Blue, Bold]];
  qwe = OpenAppend["energy.dat", DOSTextFormat -> True];
  WriteString[qwe, CForm[Rb], "\t"];
  Close[qwe];

  jmax = 4;
  st := 0.1;
  b = 0;
  i = 0;
  x = Table[0, {i, 1, jmax}];
  For[En = 10, En ≤ 516,
    If[i == 0, f1 = Re[dispeq[En]]; f2 = Re[dispeq[En + st]]; i = i + 1];
    If[i > 0, f1 = f2; f2 = Re[dispeq[En + st]]; i = i + 1];
    If[(f1 * f2 < 0), (*Print["En= ", En, "    f1= ", f1, "    f2= ", f2];*)
      a = FindRoot[Re[dispeq[u]] == 0, {u, En}];
      x[[++b]] = a[[1, 2]];
      qwe = OpenAppend["energy.dat", DOSTextFormat -> True];
      WriteString[qwe, x[[b]], "\t"];
      Close[qwe];
      Print["E= ", x[[b]]];
    ]; (*If*)
    If[(b - jmax) == 0, En = 1501];
    En = En + st;
  ]; (*for*)

  qwe = OpenAppend["energy.dat", DOSTextFormat -> True];
  WriteString[qwe, "\n"];
  Close[qwe];
  Rb = Rb + 0.1 * 10-7];

```

ЛАБОРАТОРНА РОБОТА № 5

"ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ В МОДЕЛЮВАННІ ПРОЦЕСІВ КОЛИВАНЬ ФІЗИЧНИХ ОБ'ЄКТІВ"

1. *Фізична постановка задачі*

Розглядається фізичний процес коливання динамічної системи, яка знаходиться під впливом зовнішнього збуджуючого навантаження, величина якого змінюється за періодичним законом. Прикладні аспекти застосування диференціальної математичної моделі, що описує такі важливі процеси, можна продемонструвати на дослідженні вимушених коливань моста складної конструкції під дією періодично збуджуючих потоків автомобілів (рис.1). Внаслідок недостатньої керованості такого руху конструкція моста може зазнати пошкоджень аж до руйнувань в найбільш навантажених точках конструкції з рахунок виникнення явища резонансу.

В припущенні, що такою найбільш навантаженою точкою є центр мас системи і вся маса системи m разом з потоком автомобілів зосереджена в цій точці, коливання такої системи можна аналогізувати з коливанням пружної лінії із сконцентрованою масою .

2. *Математична модель процесу: диференціальні рівняння процесу*

Процес вимушених коливань такої одномасової системи буде описуватися диференціальним рівнянням 2-го порядку з правою частиною, що визначає вплив періодично змінної збуджуючої сили (її проекції на напрям поперечного коливного руху) у вигляді:

$$\frac{d^2}{dt^2}u(t) + \phi^2 u = P_{vibr} \cdot \cos \omega t \quad (1)$$

з початковими умовами

$$u(t)|_{t=0} = U_0, \quad \frac{d}{dt}u(t)|_{t=0} = V_0 \quad (2)$$

Тут $u(t)$ - величина прогину серединної поверхні моста при коливному русі, м; $\phi = \frac{1}{\sqrt{\delta_{11}m}}$ (с^{-1}) - власна частота системи, що її внутрішньою характеристикою (є функцією від одиничного прогину та масою для найпростіших коливних систем), ω (с^{-1}) - частота вимушених коливань (періодичного збурюючого потоку автомобілів, с^{-1} ; P_{vibr} - амплітуда збурюючого потоку; U_0 (м)- величина початкового відхилення серединної поверхні мосту від положення рівноваги, V_0 (м/с)- початкова швидкість коливання поверхні мосту.

3. Вхідні дані

$t = 1, 10..1000$ - діапазон зміни часу t для визначення неперервного розподілу параметрів коливного руху;

$n = 100$ - кількість точок обчислення;

$t_0 = 10$ - початкове значення часу t для визначення неперервного розподілу параметрів коливного руху;

$\Delta t = 10$ - крок зміни часу t для визначення дискретного розподілу зміщення;

$i = 1..1000$;

$\tau_i = t_0 + i \cdot \Delta t$ - діапазон інтервалів зміни часу для розрахунку дискретного

розподілу функції зміщення;

$U_0 = 1$ (м) - амплітудна величина прогину;

$V_0 = 10$ (м/с) - амплітудне значення швидкості;

$\phi = 10$ (с^{-1}) - частота власних коливань;

$\omega = 1,1$ (с^{-1}) - частота вимушених коливань усередненої змушуючої сили;

$P_{\text{vibr}} = 1,1$ (м) - значення амплітуди вимушених коливань.

4. Алгоритмізація побудови розв'язку математичної моделі

Розв'язком математичної моделі, що описується диференціальним рівнянням другого порядку та початковими умовами (2) є залежність:

$$u(t) = U_0 \cos \phi t + \frac{V_0}{\phi} \sin \omega t + \frac{P_{vibr}}{(\phi^2 - \omega^2)} (\cos \omega t - \cos \phi t) \quad (3)$$

5. Чисельне та графічне моделювання процесів коливання механічної системи та дослідження явища резонансу

5.1. Моделювання процесу власних коливань механічної системи.

Використовуючи отриманий аналітичних розв'язок задачі Коші для неоднорідної моделі вимушених коливань поверхні моста (1)-(2), що визначається залежністю (3), побудуємо графіки залежностей компонент однорідного розв'язку. При цьому проілюструємо внесок кожної з компонент в сумарну залежність, що описує однорідну модель, без урахування впливу доданка, що визначає власні коливання системи.

$$u_{model}(t) = U_0 \cos \phi t + \frac{V_0}{\phi} \sin \omega t \quad (4)$$

$$u_{model_i}(t) = U_{0_i} \cos \phi t_i + \frac{V_0}{\phi} \sin \phi t_i; \tau_i = t_0 + i \cdot \Delta t; i = 1..1000. \quad (5)$$

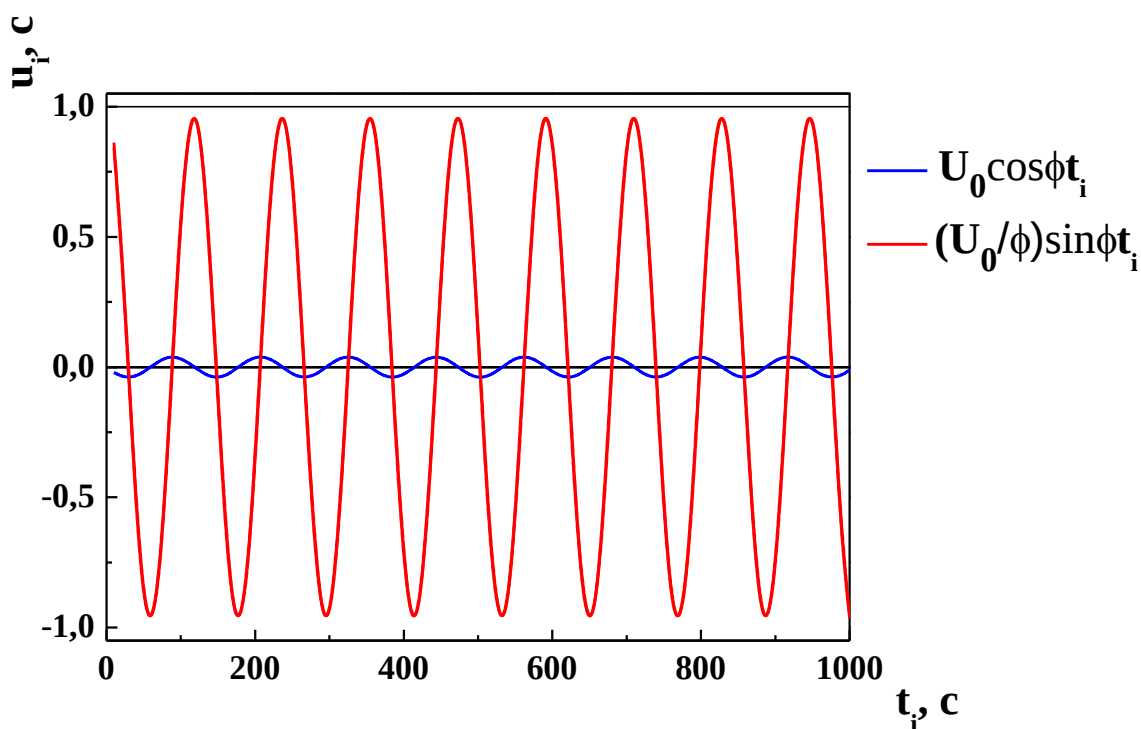


Рис. 1. Залежності від часу t компонент зміщення $u_{model}(t)$ вільних коливань у однорідній моделі.

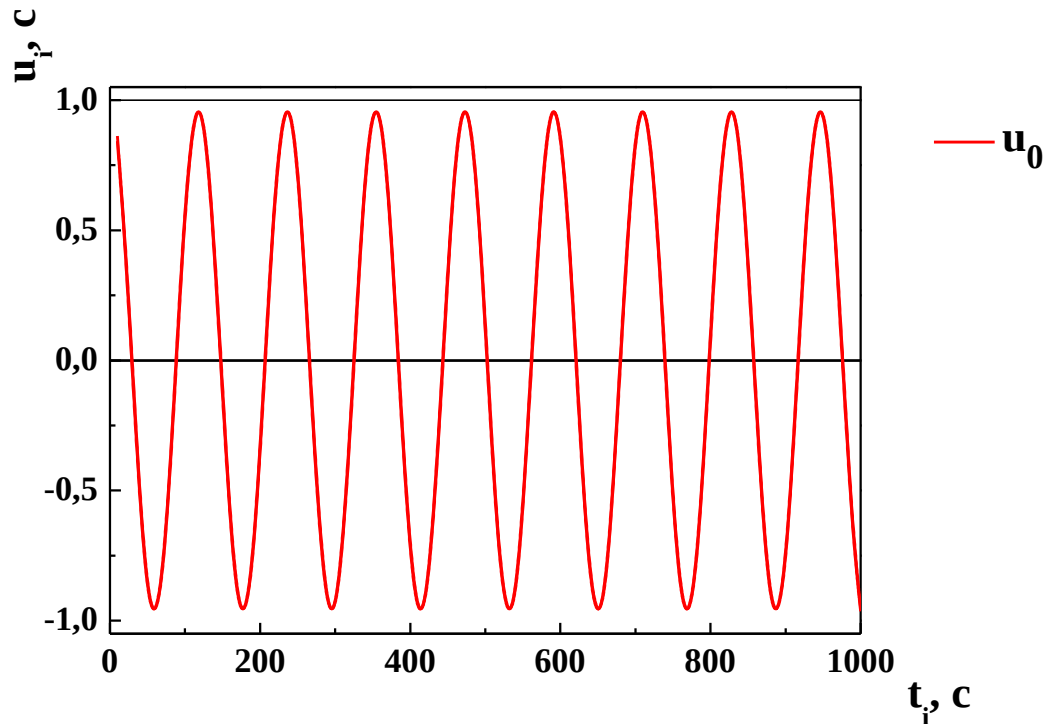


Рис. 2. Залежність від часу t зміщення $u_{\text{model}}(t)$ вільних коливань у однорідній моделі.

5.2. Моделювання процесу вимушених коливань системи та явища резонансу (биття).

Використовуючи повний аналітичний розв'язок задачі Коші для неоднорідної моделі коливального процесу (3), виконаємо побудову графіків залежностей зміщення від часу, щоб проілюструвати вплив періодичної збуджуючої сили на механічну систему, що спричиняє вільні затухаючі коливання системи.

$$u_{\text{vym}_i}(t) = u_{\text{model}_i}(t) + \frac{P_{\text{vibr}}}{(\phi^2 - \omega^2)} (\cos \omega t_i - \cos \phi t_i); \quad \tau_i = t_0 + i \cdot \Delta t; \quad i = 1..1000.$$

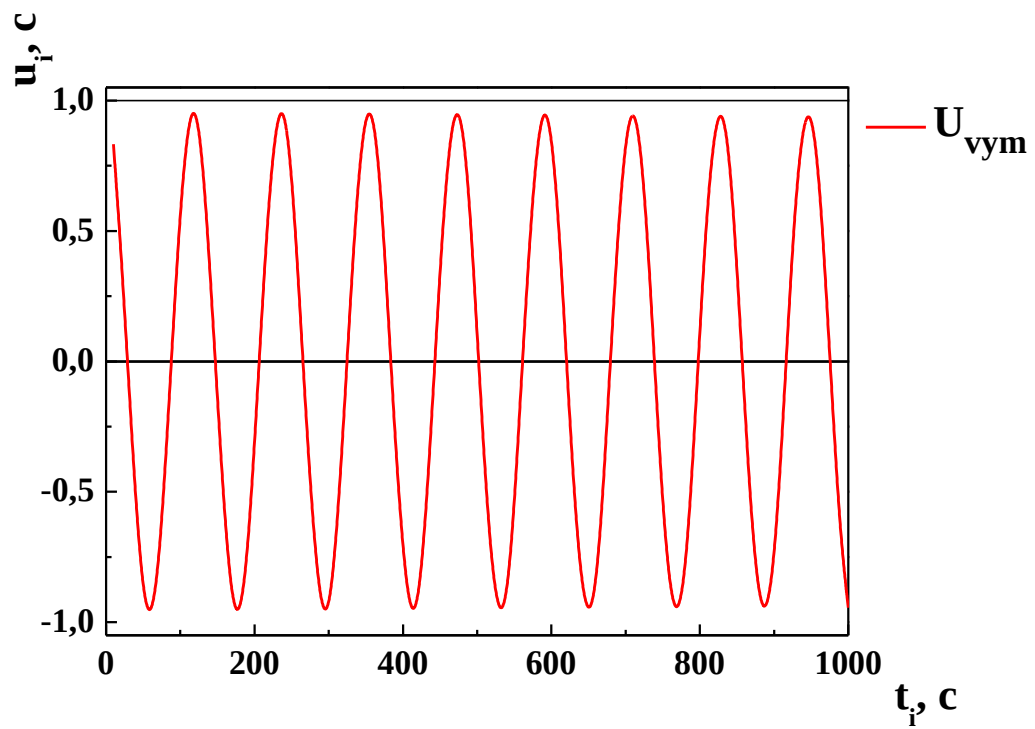


Рис. 3. Залежність від часу t зміщення $u_{\text{vym}}(t)$ у моделі системи з вільними коливаннями при $\phi = 10$ ($\tilde{n}^{-1}$) і $\omega = 1,1$ ($\tilde{n}^{-1}$).

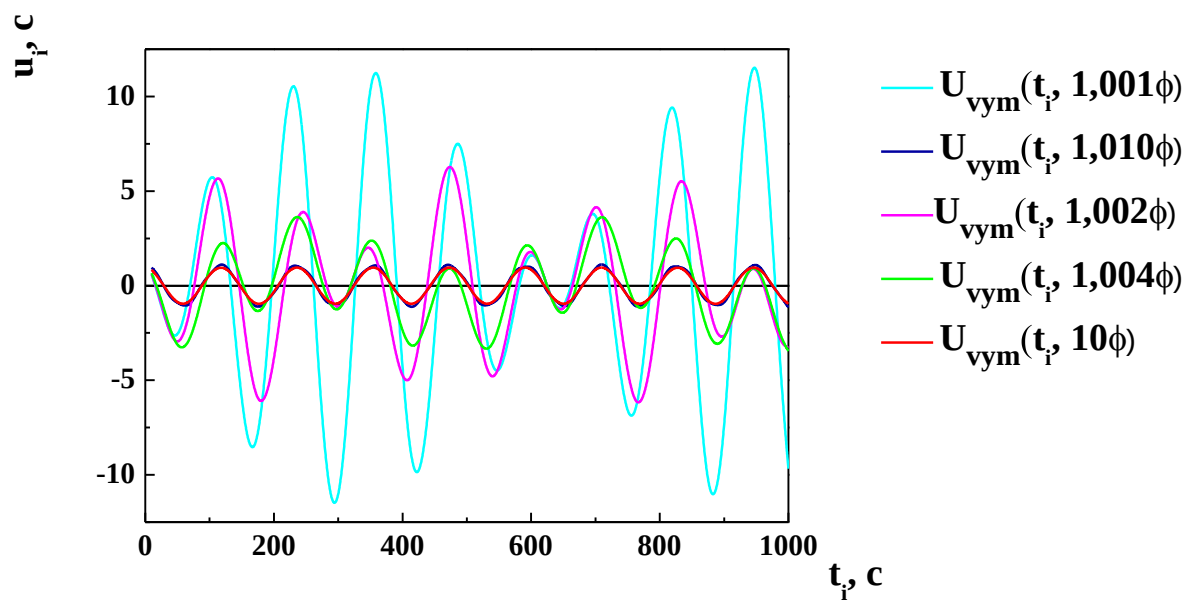


Рис. 4. Залежність від часу t зміщення $u_{\text{vym}}(t)$ у моделі системи з вимушеними коливаннями при для різних значень частот змушуючої сили ω .

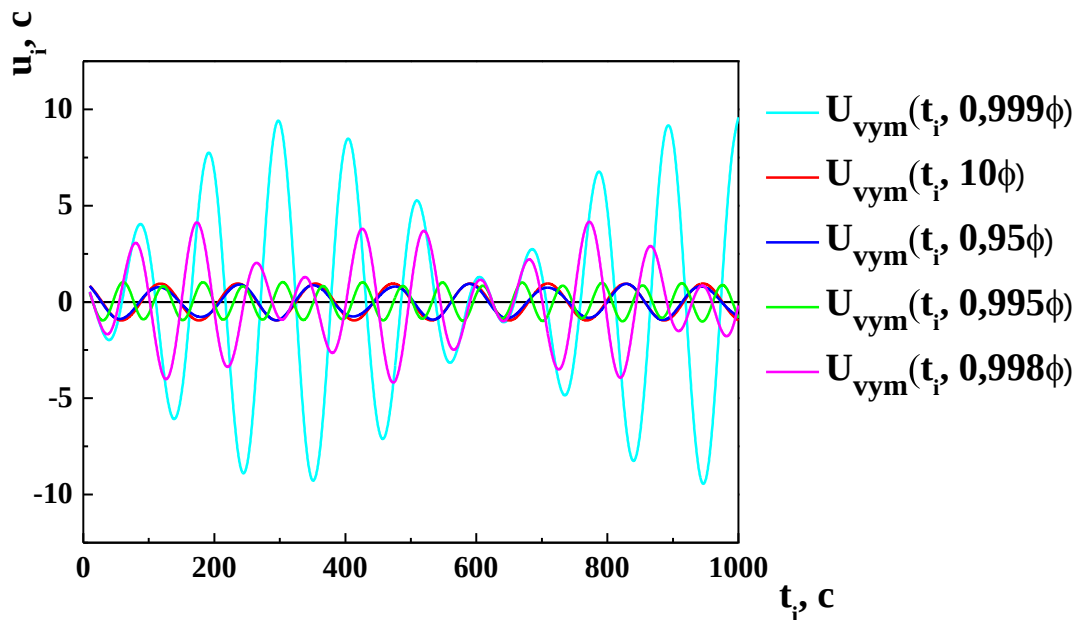


Рис. 5. Залежності від часу t зміщення $u_{\text{vym}}(t)$ у моделі системи з вимушеними коливаннями при для різних значень частот змушуючої сили ω .

Аналіз параметрів, що визначають явище резонансу.

Як видно з побудованих графіків залежностей величини зміщення для різних частот змушуючої сили, а також для вільних коливань, наближення частоти вимушених коливань справа до частоти власних коливань спричиняє поступовий ріст амплітуди коливань. В межах наближення частоти коливань змушуючої сили до частоти вільних коливань у межах до 0,1 % спостерігається більш ніж 10-ти кратне збільшення амплітуди коливань, що за реальних умов спричиняє руйнування конструкції. Аналогічна ситуація виникає при наближенні частоти вимушених коливань справа до частоти власних коливань спричиняє зліва. Слід зауважити, що при порядку 0,1 % наближенні частот власних коливань і змушуючої сили, мають місце осциляції амплітуди коливань.

Висновки. Виконавши лабораторну роботу, було здійснено математичне моделювання процесів вільного коливання механічної системи під впливом зовнішньої осцилюючої сили. Окремо розглянуто процеси вільного коливання та коливання під дією зовнішнього навантаження. Встановлено, що при наближенні частоти збуджуючої сили до частоти вільних коливань як зліва та справа виникають явища резонансу, що проявляються у збільшенні

та осциляціях амплітуди коливань та на практиці викликають руйнування механічної конструкції.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання процесів вільних та вимушених коливань механічної системи

```
FullSimplify[DSolve[u''[t] +  $\phi^2$  u[t] == Pv * Cos[ $\omega$  t], u[t], t]]
{{u[t] -> C[1] Cos[t  $\phi$ ] +  $\frac{Pv \text{Cos}[t \omega]}{\phi^2 - \omega^2}$  + C[2] Sin[t  $\phi$ ]}}
```

$$\left\{ \left\{ u[t] \rightarrow \left(- \left(Pv + U0 \left(-\phi^2 + \omega^2 \right) \right) \text{Cos}[t \phi] + Pv \text{Cos}[t \omega] \right) / \left((\phi - \omega) (\phi + \omega) \right) + \frac{V0 \text{Sin}[t \phi]}{\phi} \right\} \right\}$$

```
FullSimplify[DSolve[{u''[t] +  $\phi^2$  u[t] == 0, u[0] == U0, u'[0] == V0}, u[t], t]]
{{u[t] -> U0 Cos[t  $\phi$ ] +  $\frac{V0 \text{Sin}[t \phi]}{\phi}$ }}
```

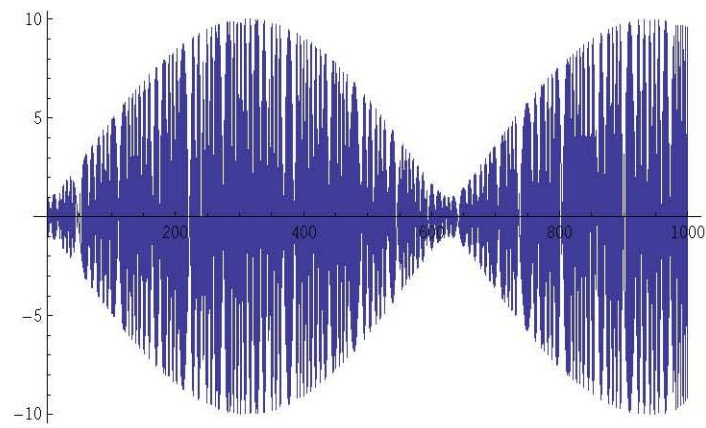
```
Collect[
  (- (Pv + U0 (- $\phi^2$  +  $\omega^2$ )) Cos[t  $\phi$ ] + Pv Cos[t  $\omega$ ]) / (( $\phi$  -  $\omega$ ) ( $\phi$  +  $\omega$ )) +  $\frac{V0 \text{Sin}[t \phi]}{\phi}$ , Pv]
-  $\frac{U0 (-\phi^2 + \omega^2) \text{Cos}[t \phi]}{(\phi - \omega) (\phi + \omega)}$  + Pv  $\left( -\frac{\text{Cos}[t \phi]}{(\phi - \omega) (\phi + \omega)} + \frac{\text{Cos}[t \omega]}{(\phi - \omega) (\phi + \omega)} \right)$  +  $\frac{V0 \text{Sin}[t \phi]}{\phi}$ 
FullSimplify[-  $\frac{U0 (-\phi^2 + \omega^2) \text{Cos}[t \phi]}{(\phi - \omega) (\phi + \omega)}$ ]
U0 Cos[t  $\phi$ ]
FullSimplify[Pv  $\left( -\frac{\text{Cos}[t \phi]}{(\phi - \omega) (\phi + \omega)} + \frac{\text{Cos}[t \omega]}{(\phi - \omega) (\phi + \omega)} \right)$ ]
 $\frac{Pv (-\text{Cos}[t \phi] + \text{Cos}[t \omega])}{\phi^2 - \omega^2}$ 
D[C[1] Cos[t  $\phi$ ] +  $\frac{Pv \text{Cos}[t \omega]}{\phi^2 - \omega^2}$  + C[2] Sin[t  $\phi$ ], t]
 $\phi$  C[2] Cos[t  $\phi$ ] -  $\phi$  C[1] Sin[t  $\phi$ ] -  $\frac{Pv \omega \text{Sin}[t \omega]}{\phi^2 - \omega^2}$ 

U0 := 1;
V0 := 0.4;
 $\phi$  := 10;
 $\omega$  := 0.999  $\phi$ ;
Pv := 1.1

u0[t_] := U0 Cos[t  $\phi$ ] +  $\frac{V0 \text{Sin}[t \phi]}{\phi}$ ;
u0sym[t_] := U0 Cos[t  $\phi$ ];
u0ant[t_] :=  $\frac{V0 \text{Sin}[t \phi]}{\phi}$ ;

u[t_] := U0 Cos[t  $\phi$ ] +  $\frac{V0 \text{Sin}[t \phi]}{\phi}$  +  $\frac{Pv (\text{Cos}[t \omega] - \text{Cos}[t \phi])}{\phi^2 - \omega^2}$ ;
```

```
Plot[u[t], {t, 0, 1000}]
```



```
SetDirectory["f:\\"];  
For[t = 10, t ≤ 1000,  
  Print[Style["t= ", 14, Blue, Bold], Style[t, 14, Blue, Bold]];  
  qwe = OpenAppend["u.dat", DOSTextFormat -> True];  
  WriteString[qwe, t, "\\t"];  
  Close[qwe];  
  qwe = OpenAppend["u.dat", DOSTextFormat -> True];  
  WriteString[qwe, CForm[N[u[t]]], "\\n"];  
  Close[qwe];  
  Print["vibration= ", CForm[N[u[t]]]];  
  t = t + 10];
```

ЛАБОРАТОРНА РОБОТА № 6

" ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ У МОДЕЛЮВАННІ КОЕФІЦІЄНТА ПРОЗОРОСТІ ТА ЕНЕРГЕТИЧНОГО СПЕКТРУ ДВОБАР'ЄРНИХ ВІДКРИТИХ НАПІВПРОВІДНИКОВИХ НАНОСТРУКТУР "

1. Фізична постановка задачі

Розглядається фізичний процес тунелювання електронів крізь напівпровідникову наноструктуру з відомими геометричними та фізичними параметрами (рис. 1).

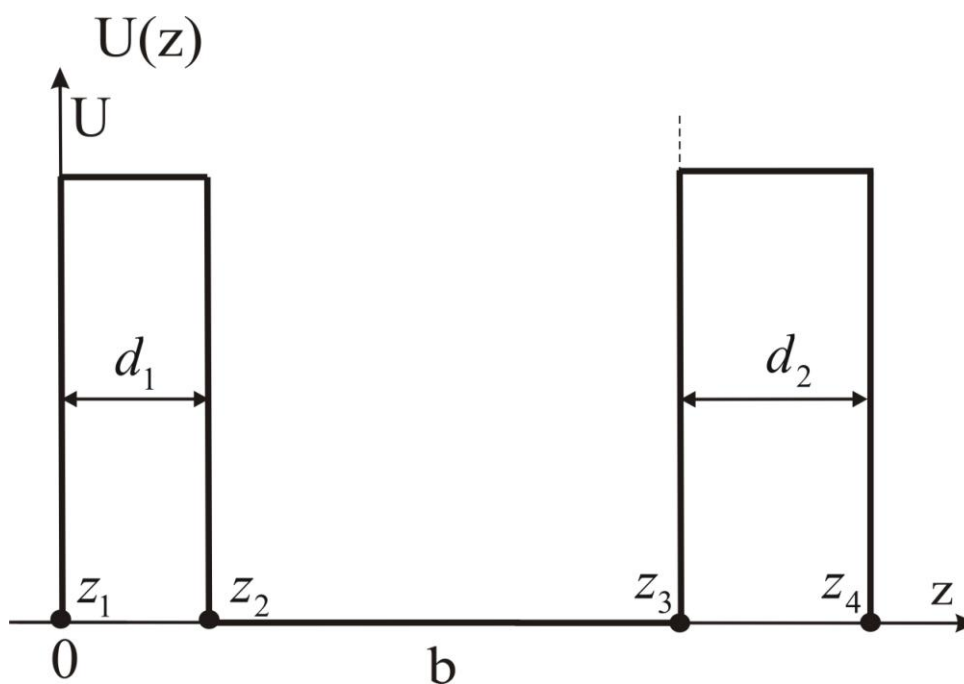


Рис. 1. Потенціальний профіль відкритої двобар'єрної наноструктури

2. Математична модель процесу: диференціальні рівняння Шредінгера

Процес тунелювання електронів крізь відкриту напівпровідникову наноструктуру описується диференціальними рівняннями другого порядку – рівняннями Шредінгера, записаними для кожної з областей наноструктури :

$$\left\{ \begin{array}{l} \frac{d^2 \psi_1}{dz^2} + k^2 \psi_1 = 0, \quad z < 0, \\ \frac{d^2 \psi_2}{dz^2} - \chi^2 \psi_2 = 0, \quad 0 \leq z < d_1 \\ \frac{d^2 \psi_3}{dz^2} + k^2 \psi_3 = 0, \quad d_1 \leq z \leq d_1 + b, \\ \frac{d^2 \psi_4}{dz^2} - \chi^2 \psi_4 = 0, \quad d_1 + b \leq z \leq d_1 + b + d_2, \\ \frac{d^2 \psi_5}{dz^2} + k^2 \psi_5 = 0, \quad z > d_1 + b + d_2. \end{array} \right. \quad (1)$$

де

$$\chi = \sqrt{\frac{2m_e(U - E)}{\hbar^2}}; \quad k = \sqrt{\frac{2m_e E}{\hbar^2}}.$$

Для розв'язків рівнянь (1) виконуються граничні умови:

$$\left\{ \begin{array}{l} \psi_1(z)|_{z=0} = \psi_2(z)|_{z=0}, \\ \frac{d\psi_1(z)}{dz}|_{z=0} = \frac{d\psi_2(z)}{dz}|_{z=0}, \\ \psi_2(z)|_{z=z_1} = \psi_3(z)|_{z=z_1}, \\ \frac{d\psi_2(z)}{dz}|_{z=z_1} = \frac{d\psi_3(z)}{dz}|_{z=z_1}, \\ \psi_3(z)|_{z=z_2} = \psi_4(z)|_{z=z_2}, \\ \frac{d\psi_3(z)}{dz}|_{z=z_2} = \frac{d\psi_4(z)}{dz}|_{z=z_2}, \\ \psi_4(z)|_{z=z_3} = \psi_5(z)|_{z=z_3}, \\ \frac{d\psi_4(z)}{dz}|_{z=z_3} = \frac{d\psi_5(z)}{dz}|_{z=z_3}. \end{array} \right. \quad (2)$$

У (2): $z_1 = d_1$; $z_2 = d_1 + b$; $z_3 = d_1 + b + d_2$.

Розв'язки рівнянь (1) мають вигляд:

$$\left\{ \begin{array}{l} \psi_1 = e^{ikz} + B_1 e^{-ikz}, \quad z < 0, \\ \psi_2 = A_2 e^{\chi z} + B_2 e^{-\chi z}, \quad 0 \leq z < d_1 \\ \psi_3 = A_3 e^{ikz} + B_3 e^{-ikz}, \quad d_1 \leq z \leq d_1 + b, \\ \psi_4 = A_4 e^{\chi z} + B_4 e^{-\chi z}, \quad d_1 + b \leq z \leq d_1 + b + d_2, \\ \psi_5 = A_5 e^{ikz}, \quad z > d_1 + b + d_2. \end{array} \right. \quad (3)$$

Тоді з системи (2) знаходимо:

$$\begin{cases} 1 + B_1 = A_2 + B_2, \\ ik(1 - B_1) = \chi(A_2 - B_2), \\ A_2 e^{\chi z_1} + B_2 e^{-\chi z_1} = A_3 e^{ikz_1} + B_3 e^{-ikz_1}, \\ \chi(A_2 e^{\chi z_1} - B_2 e^{-\chi z_1}) = ik(A_3 e^{ikz_1} - B_3 e^{-ikz_1}), \\ A_3 e^{ikz_2} + B_3 e^{-ikz_2} = A_4 e^{\chi z_2} + B_4 e^{-\chi z_2}, \\ ik(A_3 e^{ikz_2} - B_3 e^{-ikz_2}) = \chi(A_4 e^{\chi z_2} - B_4 e^{-\chi z_2}), \\ A_4 e^{\chi z_3} + B_4 e^{-\chi z_3} = A_5 e^{ikz_3}, \\ \chi(A_4 e^{\chi z_3} - B_4 e^{-\chi z_3}) = ikA_5 e^{ikz_3}. \end{cases} \quad (4)$$

Звідси:

$$\begin{cases} B_1 - A_2 - B_2 = -1, \\ ikB_1 + \chi A_2 - \chi B_2 = ik, \\ A_2 e^{\chi z_1} + B_2 e^{-\chi z_1} - A_3 e^{ikz_1} - B_3 e^{-ikz_1} = 0, \\ \chi A_2 e^{\chi z_1} - \chi B_2 e^{-\chi z_1} - ikA_3 e^{ikz_1} + ikB_3 e^{-ikz_1} = 0, \\ A_3 e^{ikz_2} + B_3 e^{-ikz_2} - A_4 e^{\chi z_2} - B_4 e^{-\chi z_2} = 0, \\ ikA_3 e^{ikz_2} - ikB_3 e^{-ikz_2} - \chi A_4 e^{\chi z_2} + \chi B_4 e^{-\chi z_2} = 0, \\ A_4 e^{\chi z_3} + B_4 e^{-\chi z_3} - A_5 e^{ikz_3} = 0, \\ \chi A_4 e^{\chi z_3} - \chi B_4 e^{-\chi z_3} - ikA_5 e^{ikz_3} = 0. \end{cases} \quad (5)$$

Знаходимо згідно методу Крамера:

$$B_1 = \frac{\Delta_1}{\Delta}; A_2 = \frac{\Delta_2}{\Delta}; B_2 = \frac{\Delta_3}{\Delta}; A_3 = \frac{\Delta_4}{\Delta}; B_3 = \frac{\Delta_5}{\Delta}; A_4 = \frac{\Delta_6}{\Delta}; B_4 = \frac{\Delta_7}{\Delta}; A_5 = \frac{\Delta_8}{\Delta}, \text{ де:}$$

$$\Delta = \begin{vmatrix} 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ ik & \chi & -\chi & 0 & 0 & 0 & 0 & 0 \\ 0 & e^{\chi z_1} & e^{-\chi z_1} & -e^{ikz_1} & e^{-ikz_1} & 0 & 0 & 0 \\ 0 & \chi e^{\chi z_1} & -\chi e^{-\chi z_1} & -ike^{ikz_1} & ike^{-ikz_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{ikz_2} & e^{-ikz_2} & -e^{\chi z_2} & -e^{-\chi z_2} & 0 \\ 0 & 0 & 0 & ike^{ikz_2} & -ike^{-ikz_2} & -\chi e^{\chi z_2} & \chi e^{-\chi z_2} & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{\chi z_3} & e^{-\chi z_3} & -e^{ikz_3} \\ 0 & 0 & 0 & 0 & 0 & \chi e^{\chi z_3} & -\chi e^{-\chi z_3} & -ike^{ikz_3} \end{vmatrix}. \quad (6)$$

Усі визначники $\Delta_1, \Delta_2, \Delta_3, \Delta_4$ отримуються заміною відповідного стовбця у (6) стовбцем вільних членів з системи (5).

Коефіцієнт прозорості наноструктури визначається як:

$$D = D(E) = |A_5|^2, \quad (7)$$

Енергетичний спектр електрона визначається максимумами коефіцієнта прозорості на шкалі енергії E (7):

$$E_n = \max D(E). \quad (8)$$

Густина ймовірності знаходження електрона в межах наноструктури визначається квадратом його хвильової функції для відповідного значення його резонансної енергії $E = E_n$:

$$W = W(E_n) = |\psi_n(E_n)|^2 = \begin{cases} |\psi_1|^2, & z < 0, \\ |\psi_2|^2, & 0 \leq z < d_1 \\ |\psi_3|^2, & d_1 \leq z \leq d_1 + b, \\ |\psi_4|^2, & d_1 + b \leq z \leq d_1 + b + d_2, \\ |\psi_5|^2 = A_5 e^{ikz}, & z > d_1 + b + d_2. \end{cases} \quad (9)$$

3. Вхідні дані

$b = 10$ нм - товщина потенціальної ями;

$d_1 = d_2 = 2$ нм - товщини потенціальних бар'єрів;

$m_e = 0,04585 \cdot 10^{-31}$ кг - ефективна маса електрона;

$\hbar = 1,055 \cdot 10^{-34}$ Дж·с - стала Планка.

4. Алгоритмізація побудови розв'язку математичної моделі

Побудуємо графік залежності $D = D(E)$ (7) для значень енергії електрона $0 \leq E \leq U$:

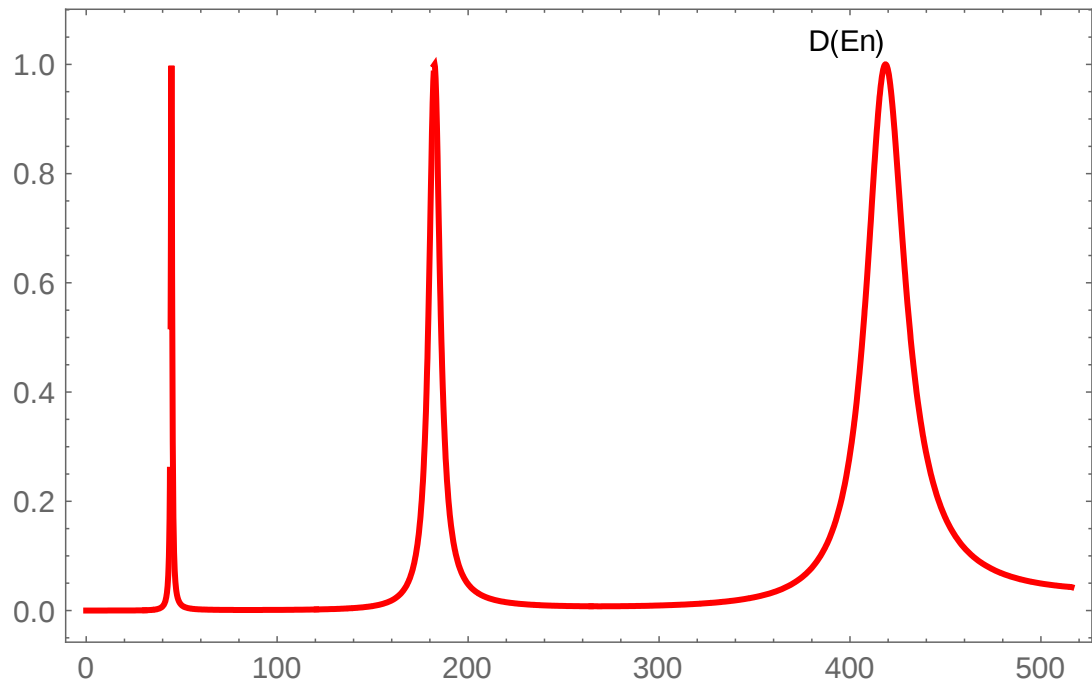


Рис. 1. Графік залежності $D = D(E)$

Знаходимо резонансні енергії – максимуми функції $D = D(E)$, користуючись функцією FindMaximum:

$$E_1 = 44,6269 \text{ meV}, E_2 = 182,392 \text{ meV}, E_3 = 418,62 \text{ meV}. \quad (9)$$

5. *Моделювання залежності енергетичного спектру електрона від геометричних параметрів квантової ями*

З метою отримання залежності $E = E(b)$ будемо змінювати величину b у межах:

$$2 \text{ нм} \leq b \leq 15 \text{ нм}$$

з кроком:

$$\delta = 0,2 \text{ нм}$$

кожного разу знаходячи максимуми залежності $D = D(E)$, для кожної з точок:

$$b_i = n\delta, i = 1, \dots, 10.$$

використовуючи вбудовану функцію FindMaximum.

Результатом розрахунків є графічна залежність:

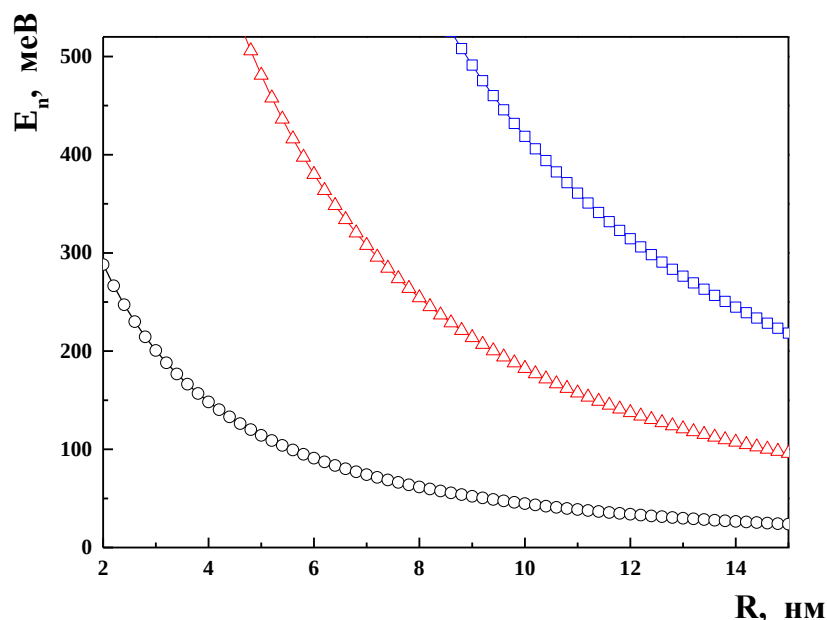


Рис. 1 - Залежність перших трьох енергетичних рівнів електрона від ширини потенціальної ями

Як видно з Рис. 1, залежності перших трьох енергетичних рівнів електрона, локалізованого у потенціальній квантовій ямі зі скінченною висотою зовнішніх потенціальних бар'єрів є однотипними. Їх поведінка у межах використовуваної моделі корелює з теоретичною залежністю, згідно з якою $E_n \sim \frac{n^2}{b^2}$, де b - ширина потенціальної ями, n - номер енергетичного рівня. При цьому всі енергетичні рівні розміщені в інтервалі: $0 < E_n \leq U$.

Для кожного із значень енергії (9) побудуємо залежності $W = W(E_n) = |\psi_n(E_n)|^2$, $n = 1, 2, 3$.

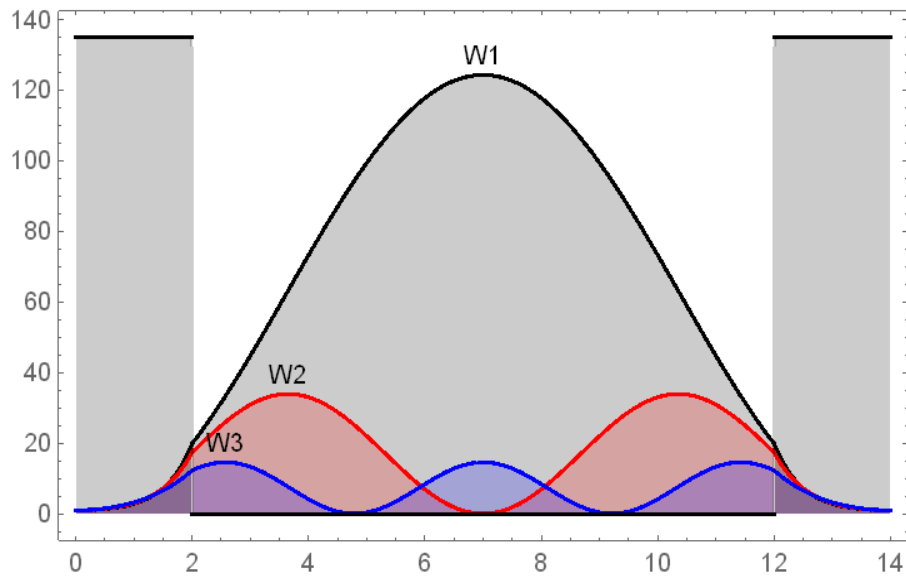


Рис. 2 – Залежність $W = W(E_n)$ для перших трьох енергетичних рівнів електрона

Висновки. Результатом проведеного дослідження з математичного моделювання енергетичного спектру електрона тунельованого крізь відкриту наносистему з потенціальними стінками скінченної висоти є отримані графічні залежності $E_n = E_n(b)$ і $W = W(E_n)$, що добре корелюють з теоретичними результатами. Розглядувана модель може бути використана для розрахунку спектру електрона в багатошарових напівпровідникових наноструктурах.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання коефіцієнта прозорості та енергетичного спектру електронів у відкритих напівпровідникових наноструктур

```
V := 516; (*у внутрішньому бар'єрі*)
m0 = 0.04585; (*в ямі*)
m1 = 0.08856; (*у бар'єрах*)
x1 = 0.53; (*яма*)
```

```
ab := 0.0529
Ry := 13605 (* meV *)
```

```
R = 10; (*ями*)
l1 = 2.0; (*бар'єри*)
l2 = 2.0;
```

```
Δ1 := l1 (*бар'єри*)
Δ2 := l2
z1 := Δ1 + R
z2 := z1 + Δ2
```

```
k[En_] := Sqrt[m0 * En / Ry] / ab
```

└─квадратный корень

```
χ[En_] := Sqrt[ $\frac{m1 (V - En)}{Ry}$ ] / ab
```

└─квадратный корень

```
Ψ0[En_, z_] := (ei k[En] z + B0[En] e-i k[En] z);
Ψ1[En_, z_] := (A1[En] eχ[En] z + B1[En] e-χ[En] z);
Ψ2[En_, z_] := (A2[En] ei k[En] (z-Δ1) + B2[En] e-i k[En] (z-Δ1));
Ψ3[En_, z_] := (A3[En] eχ[En] (z-z1) + B3[En] e-χ[En] (z-z1));
Ψ4[En_, z_] := A4[En] ei k[En] (z-z2);
```

```
determ[En_] := Det[
```

└─детерминант

$$\begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ \frac{i k[En]}{m0} & \frac{\chi[En]}{m1} & -\frac{\chi[En]}{m1} & 0 & 0 & 0 \\ 0 & e^{\Delta1 \chi[En]} & e^{-\Delta1 \chi[En]} & -1 & -1 & 0 \\ 0 & \frac{e^{\Delta1 \chi[En]} \chi[En]}{m1} & -\frac{e^{-\Delta1 \chi[En]} \chi[En]}{m1} & -\frac{i k[En]}{m0} & \frac{i k[En]}{m0} & 0 \\ 0 & 0 & 0 & e^{i k[En] (z1-\Delta1)} & e^{-i k[En] (z1-\Delta1)} & -1 \\ 0 & 0 & 0 & \frac{i e^{i k[En] (z1-\Delta1)} k[En]}{m0} & -\frac{i e^{-i k[En] (z1-\Delta1)} k[En]}{m0} & \frac{\chi[En]}{m1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En]} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En]} (-1)}{m1} \end{pmatrix}$$

$$B0[En_]:=Det\left[\begin{array}{cccccc} -1 & -1 & -1 & 0 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & 0 & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & -1 & -1 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 \\ 0 & 0 & 0 & e^{i k[En] (z1-\Delta 1)} & e^{-i k[En] (z1-\Delta 1)} & -1 \\ 0 & 0 & 0 & \frac{i e^{i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En] (-z1+z2)} \chi[En]}{n1} \end{array}\right]$$

детерминант

$$A1[En_]:=Det\left[\begin{array}{cccccc} 1 & -1 & -1 & 0 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{i k[En]}{n0} & -\frac{\chi[En]}{n1} & 0 & 0 & 0 \\ 0 & 0 & e^{-\Delta 1 \chi[En]} & -1 & -1 & 0 \\ 0 & 0 & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 \\ 0 & 0 & 0 & e^{i k[En] (z1-\Delta 1)} & e^{-i k[En] (z1-\Delta 1)} & -1 \\ 0 & 0 & 0 & \frac{i e^{i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En] (-z1+z2)} \chi[En]}{n1} \end{array}\right]$$

детерминант

$$B1[En_]:=Det\left[\begin{array}{cccccc} 1 & -1 & -1 & 0 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & \frac{i k[En]}{n0} & 0 & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & 0 & -1 & -1 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & 0 & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 \\ 0 & 0 & 0 & e^{i k[En] (z1-\Delta 1)} & e^{-i k[En] (z1-\Delta 1)} & -1 \\ 0 & 0 & 0 & \frac{i e^{i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En] (-z1+z2)} \chi[En]}{n1} \end{array}\right]$$

детерминант

$$A2[En_]:=Det\left[\begin{array}{cccccc} 1 & -1 & -1 & -1 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & \frac{i k[En]}{n0} & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & 0 & -1 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & 0 & \frac{i k[En]}{n0} & 0 \\ 0 & 0 & 0 & 0 & e^{-i k[En] (z1-\Delta 1)} & -1 \\ 0 & 0 & 0 & 0 & -\frac{i e^{-i k[En] (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En] (-z1+z2)} \chi[En]}{n1} \end{array}\right]$$

детерминант

$$B2[En_]:=Det\left[\begin{array}{ccccccc}1 & -1 & -1 & 0 & -1 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & 0 & \frac{i k[En]}{n0} & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & -1 & 0 & 0 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{i k[En]}{n0} (z1-\Delta 1)} & 0 & -1 & -1 \\ 0 & 0 & 0 & \frac{i e^{\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & 0 & -\frac{\chi[En]}{n1} & \frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} & e^{-\chi[En]} \\ 0 & 0 & 0 & 0 & 0 & \frac{e^{\chi[En] (-z1+z2)} \chi[En]}{n1} & -\frac{e^{-\chi[En]}}{n1}\end{array}\right]$$

детерминант

$$A3[En_]:=Det\left[\begin{array}{ccccccc}1 & -1 & -1 & 0 & 0 & -1 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & 0 & 0 & \frac{i k[En]}{n0} & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & -1 & -1 & 0 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{i k[En]}{n0} (z1-\Delta 1)} & e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} & 0 & -1 \\ 0 & 0 & 0 & \frac{i e^{\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & 0 & \frac{\chi[En]}{n1} \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{-\chi[En]} \\ 0 & 0 & 0 & 0 & 0 & 0 & -e^{\chi[En]}\end{array}\right]$$

детерминант

$$B3[En_]:=Det\left[\begin{array}{ccccccc}1 & -1 & -1 & 0 & 0 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & 0 & 0 & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & -1 & -1 & 0 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{i k[En]}{n0} (z1-\Delta 1)} & e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} & -1 & 0 \\ 0 & 0 & 0 & \frac{i e^{\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} & \frac{e^{\chi[En] (-z1+z2)}}{n1}\end{array}\right]$$

детерминант

$$A4[En_]:=Det\left[\begin{array}{ccccccc}1 & -1 & -1 & 0 & 0 & 0 & 0 \\ \frac{i k[En]}{n0} & \frac{\chi[En]}{n1} & -\frac{\chi[En]}{n1} & 0 & 0 & 0 & 0 \\ 0 & e^{\Delta 1 \chi[En]} & e^{-\Delta 1 \chi[En]} & -1 & -1 & 0 & 0 \\ 0 & \frac{e^{\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{e^{-\Delta 1 \chi[En]} \chi[En]}{n1} & -\frac{i k[En]}{n0} & \frac{i k[En]}{n0} & 0 & 0 \\ 0 & 0 & 0 & e^{\frac{i k[En]}{n0} (z1-\Delta 1)} & e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} & -1 & 0 \\ 0 & 0 & 0 & \frac{i e^{\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & -\frac{i e^{-\frac{i k[En]}{n0} (z1-\Delta 1)} k[En]}{n0} & -\frac{\chi[En]}{n1} & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{\chi[En] (-z1+z2)} & \frac{e^{\chi[En] (-z1+z2)}}{n1}\end{array}\right]$$

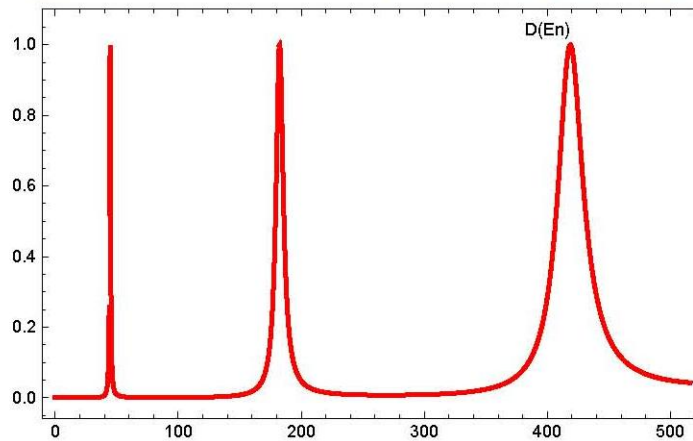
детерминант

$$T[En_]:=Abs[A4[En_]]^2;$$

$$W[En_, z_]:=$$

```
Which[(z < 0), Abs[Ψ0[En, z]]^2, (z > 0) && (z ≤ Δ1), Abs[Ψ1[En, z]]^2, (z > Δ1) && (z ≤ z1),
условный оператор с множественными ветвями
Abs[Ψ2[En, z]]^2, (z > z1) && (z ≤ z2), Abs[Ψ3[En, z]]^2, (z > z2), Abs[Ψ4[En, z]]^2];
```

```
Plot[T[En], {En, 0, 516}, PlotRange → All, PlotStyle → {Red, Thick},
график функции отображаем... всё стиль графика кр... жирный
Frame → True, PlotLabels → Placed[{"D(En)", Above}]
рамка ист... пометки на г... располо... диффер... сверху
```



```
FindMaximum[T[En], {En, 40, 60}]
```

```
найти максимум
```

```
{1., {En → 44.6269}}
```

```
FindMaximum[T[En], {En, 180, 210}]
```

```
найти максимум
```

```
{1., {En → 182.392}}
```

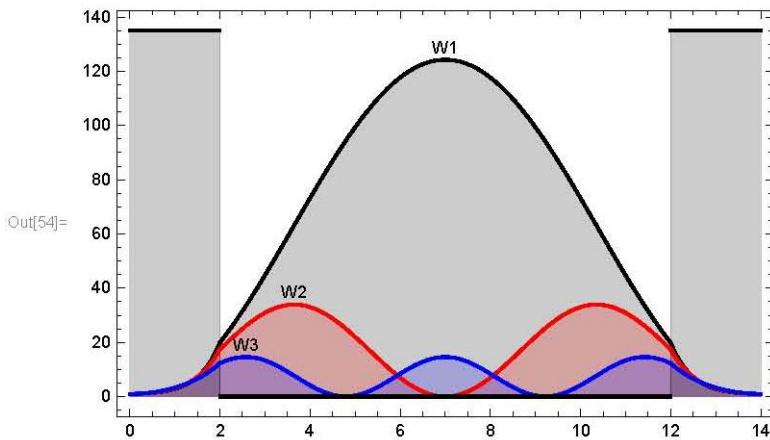
```
In[34]= FindMaximum[T[En], {En, 380, 500}]
```

```
найти максимум
```

```
Out[34]= {1., {En → 418.62}}
```

```
In[53]= f := Which[(z < 0), 0, (z > 0) && (z ≤ 11), 135, (z > 11) && z ≤ (11 + R),
условный оператор с множественными ветвями
0, z > (11 + R) && z ≤ (11 + R + 12), 135, (z > 11 + R + 12), 0];
```

```
In[54]= Plot[{W[44.626854479774984, z], W[182.39195776466073, z], W[418.61976777425434, z], f},
{z, 0, 14}, PlotRange -> All, PlotStyle -> {Black, Red, Blue}, Frame -> True,
PlotLabels -> Placed[{"W1", "W2", "W3"}, Above], Filling -> Bottom]
```



```
SetDirectory["G:\\"];
For[R = 1.4, R ≤ 15,
Print[Style["R ", 16, Blue, Bold], Style[R, Blue, Bold]];
qwe = OpenAppend["Ed_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
WriteString[qwe, R "\t"];
Close[qwe];
qwe = OpenAppend["Gd_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
WriteString[qwe, R "\t"];
Close[qwe];

(*Розрахунок енергій*)
jmax = 5;
imax := jmax;
st := 0.1;
b1 = 0;
b2 = 0;
b = 0;
i = 0;
x = Table[0, {i, 1, jmax}];
Print[Style[енергії, 16, Red, Bold]];
For[En = 0, En ≤ 1500,
```

```

Label[povtor];
отметка
If[(Max[T1, T2] - Min[T1, T2]) / Max[T1, T2] ≥ 0.01, st = st * 0.9;
усл... максимум минимум максимум
    GoTo[povtor], st = st * 1.1];
T1 = T[En];
T2 = T[En + st];
If[i == 0, f1 = T1; f2 = T2; b2 = (f2 - f1); i = i + 1];
условный оператор
If[i > 0, b1 = b2; f1 = f2; f2 = T2; b2 = (f2 - f1); i = i + 1];
условный оператор
If[(b1 b2 < 0) && (b2 < 0),
условный оператор

    a = FindMaximum[T[u], {u, N[En - 2 * st], En}];
найти максимум численное приближение
    x[[++b]] = a[[2, 1, 2]];
    (*Print["E= ", x[[b]], "      maxT= ", T[x[[b]]]];*)
печата... основание натурального логарифма
    qwe = OpenAppend["Ed_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
открыть для добавления в конец истина
    WriteString[qwe, CForm[x[[b]]], "\t"];
записать строку С форма
    Close[qwe];
синхронизирующее число
]; (*If*)
условный оператор
If[(b - jmax) == 0, En = 1501];
условный оператор
En = En + st;
]; (*for*)
jmax = b;

(*Print[Style["ШИРИНЫ МАКСИМУМІВ", 14, Blue, Bold]];*)
печа... стиль синий жирный шрифт

Riv[En_, i_] := T[En] - T[x[[i]]] / 2;
G = Table[0, {i, 1, imax}, {j, 1, 2}];
таблица значений

For[i = 1, i ≤ imax - 1,
цикл для

    s = 0;
    x[[0]] = 0;
    y1 = Abs[(x[[i]] - x[[i - 1]])] / 50;
абсолютное значение
    y2 = Abs[(x[[i]] - x[[i + 1]])] / 50;
абсолютное значение
    step = -Min[y1, y2];
минимум
    Em2 = x[[i]];
    Label[zanovo];
    .....

```

```

For [l = 1, l < 250, l++,
    цикл ДЛЯ
    Em1 = Em2; Em2 = Em2 + step; h1 = Riv[Em1, i]; h2 = Riv[Em2, i];
    If[h2 ≤ 0, t = FindRoot[Riv[Em, i] == 0, {Em, Em1, Em2}];
    условный опер... найти корень
    l = 251;
    s = s + 1;
    G[[i, s]] = Abs[t[[1, 2]] - x[[i]]];
    абсолютное значение
]; (*If*)
    условный оператор
]; (*For*)
    цикл ДЛЯ
If[s == 1, step = -step; Goto[zanovo]];
условный оператор перейти
Gam = 2 * Min[G[[i, 1]], G[[i, 2]]];
минимум
Print["En= ", x[[i]], "      G= ", Gam];
печатать
qwe = OpenAppend["Gd_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
открыть для добавления в конец истина
WriteString[qwe, Gam, "\t"];
записать строку
Close[qwe];
синхронизирующее число
i = i + 1
]; (*For*)
    цикл ДЛЯ

qwe = OpenAppend["Ed_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
открыть для добавления в конец истина
WriteString[qwe, "\n"];
записать строку
Close[qwe];
синхронизирующее число
qwe = OpenAppend["Gd_3.4_x_1.6_21.4-x_3.4nm.dat", DOSTextFormat -> True];
открыть для добавления в конец истина
WriteString[qwe, "\n"];
записать строку
Close[qwe];
синхронизирующее число

R = R + 0.2];

```

ЛАБОРАТОРНА РОБОТА № 7

" ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ ДРУГОГО ПОРЯДКУ В ЧАСТИННИХ ПОХІДНИХ У МОДЕЛЮВАННІ ЯВИЩА ПЕРЕНЕСЕННЯ ТЕПЛА У НАГРІТИХ СТЕРЖНЯХ МЕТОДОМ ФУР'Є "

1. Фізична постановка задачі

Розглядається фізичний процес охолодження стержня довжиною L який був попередньо нагрітий до температури u_0 . В момент часу $t=0$ стержень теплоізолюють, а його кінці утримують при температурі 0°C . Необхідно встановити залежність за якою відбувається охолодження стержня, та температуру всередині нього для заданого моменту часу t_0 (рис. 1).

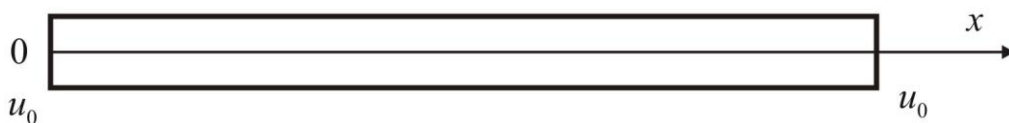


Рис. 1. Геометрична схема теплоізолизованого стержня, торці якого утримуються при постійній температурі

2. Математична модель процесу: рівняння теплопереносу та граничні умови на кінцях стержня

Крайова задача, тобто задача в якій задані граничні умови для функції розподілу температури в с стержні $u(x,t)$ має вигляд:

$$\begin{cases} \frac{\partial u}{\partial t} = k \frac{\partial^2 u}{\partial x^2}, \\ u(0,t) = u(L,t) = 0, \\ u(x,0) = u_0. \end{cases} \quad (1)$$

де k - коефіцієнт теплопровідності матеріалу стержня.

Розв'язок крайової задачі (1) для нагрітого стержня з нульовими значеннями температури в його кінцях шукається у вигляді ряду:

$$u(x,t) = \sum_{n=1}^{\infty} a_n \exp(-n^2 \pi^2 k t / L^2) \sin \frac{n \pi x}{L}, \quad (2)$$

де a_n - коефіцієнти розкладу за в ряд Фур'є за синусами функції $f(x) = u(x, 0)$, що задає початковий розподіл температури в стержні.

Значення цих коефіцієнтів визначаються як:

$$a_n = \frac{2}{L} \int_0^L f(x) \sin \frac{n \pi x}{L} dx. \quad (3)$$

В нашому випадку:

$$f(x) = \begin{cases} u_0, & 0 < x < L, \\ 0, & -L < x < 0. \end{cases} \quad (4)$$

Тоді підставляючи (4) в (3) і виконуючи інтегрування:

$$\begin{aligned} a_n &= \frac{2}{L} \int_0^L f(x) \sin \frac{n \pi x}{L} dx = \frac{2u_0}{L} \int_0^L \sin \frac{n \pi x}{L} dx = -\frac{2u_0}{L} \cdot \frac{L}{n\pi} \cos \frac{n \pi x}{L} \Big|_0^L = \\ &= -\frac{2u_0}{n\pi} \left(\cos \frac{n \pi L}{L} - \cos \frac{n \pi \cdot 0}{L} \right) = -\frac{2u_0}{n\pi} (\cos n\pi - \cos 0) = -\frac{2u_0}{n\pi} ((-1)^n - 1) = \\ &= \frac{2u_0}{n\pi} (1 - (-1)^n) = \begin{cases} \frac{4u_0}{n\pi}, & n = 2k + 1, k \in \mathbb{Z} \\ 0, & n = 2k, k \in \mathbb{Z}. \end{cases} \end{aligned} \quad (4)$$

Цей же результат можна було отримати з використанням системи Wolfram Mathematica:

$$\begin{aligned} \text{In}[1]:= & \text{Assuming}[n \in \text{Integers}, \left\{ \frac{2}{L} \int_0^L u_0 \sin\left[\frac{n \pi x}{L}\right] dx \right\}] \\ & \text{[предполагая [множество целых чисел] [синус]} \\ \text{Out}[1]= & \left\{ \frac{2 \left(1 + (-1)^{1+n}\right) u_0}{n \pi} \right\} \end{aligned}$$

Таким чином, розв'язок крайової задачі:

$$u(x,t) = \frac{4u_0}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \exp(-n^2 \pi^2 k t / L^2) \sin \frac{n \pi x}{L}. \quad (5)$$

3. Вхідні дані

$L = 50$ м - довжина стержня;

$k = 0,15 \frac{\text{Дж}}{\text{м}^2 \cdot \text{К}}$ - коефіцієнт теплопровідності стержня (відповідає матеріалу

залізного стержня);

$u_0 = 100$ °С - температура стержня;

$t = \{0, 2000 \text{ с}\}$ - діапазон часу від початку охолодження стержня.

4. Алгоритмізація побудови розв'язку математичної моделі

Побудуємо графік залежності $u(x, t)$ для температури посередині стержня

$x = \frac{L}{2}$ для різних значень n :

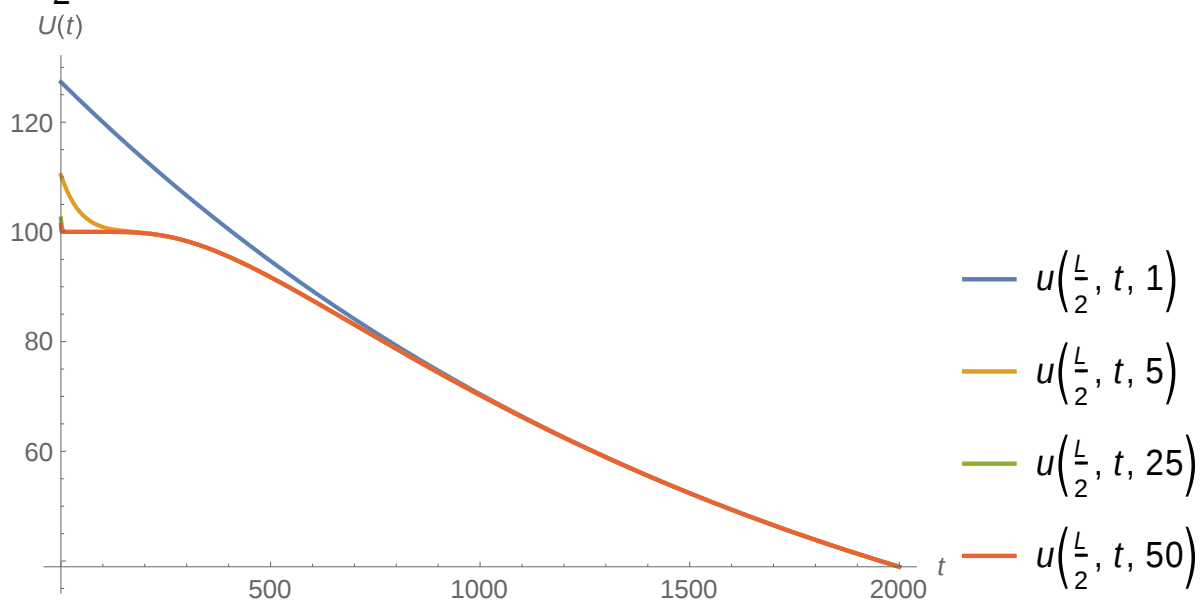


Рис. 1. Графік залежності $u = u\left(\frac{L}{2}, t\right)$

Як видно з рисунка при $n = 25$ і при $n = 50$ графіки залежностей $u = u\left(\frac{L}{2}, t\right)$

уже майже співпадають, що свідчить про збіжність ряду (5).

Побудуємо залежності $u = u(x)$ для різних моментів часу:

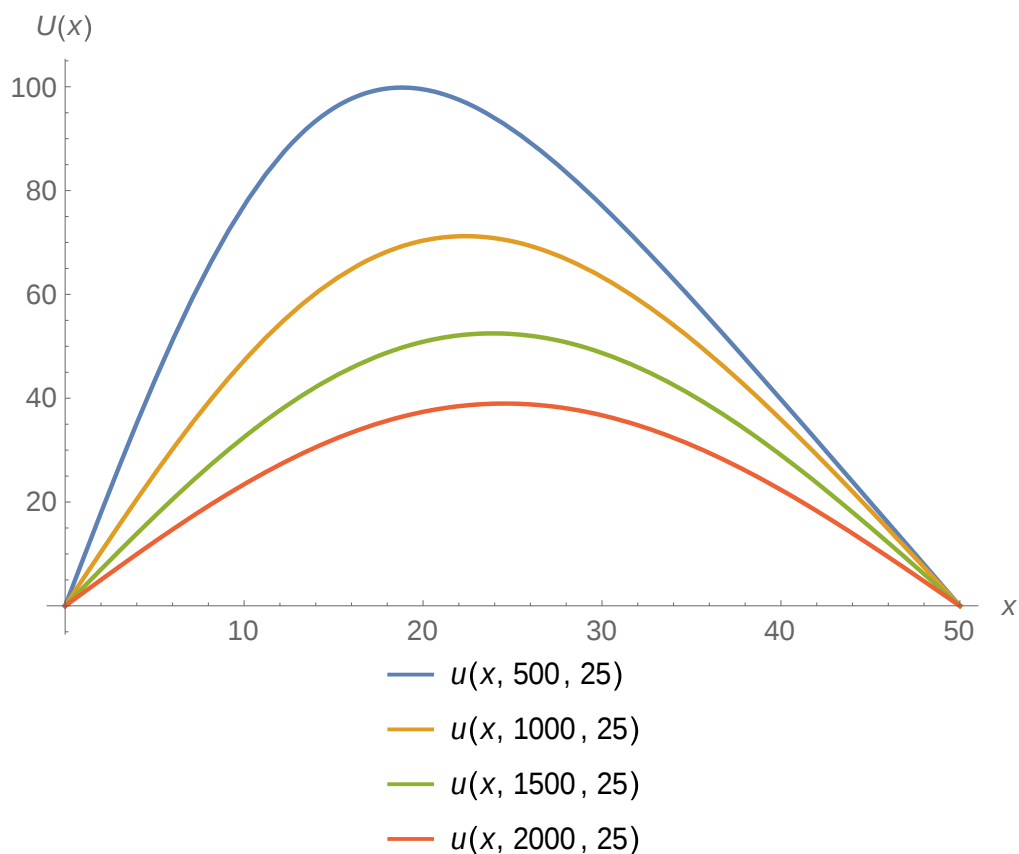


Рис. 2. Графік залежності $u = u(x)$

Як видно з графіків залежності $u = u(x)$ при охолодженні стержня максимум його температури зміщується до його середини $x = \frac{L}{2}$.

Узагальнюючи отримані вище результати побудуємо просторовий розподіл температури $u(x, t)$:

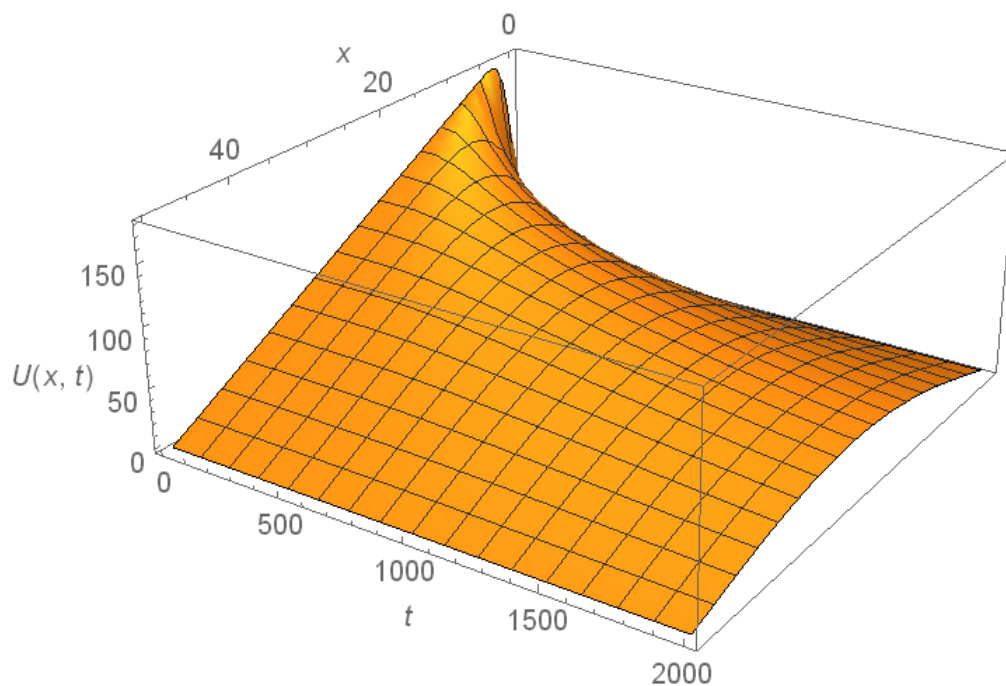


Рис. 3. Просторовий розподіл температури в стержні виготовленого з заліза

Доцільно виконати розрахунок просторового розподілу температури для стержня з іншого матеріалу, який характеризується меншим коефіцієнтом теплопровідності, наприклад для бетону для якого, $k = 0,005 \frac{\text{Дж}}{\text{м}^2 \cdot \text{К}}$.

Маємо:

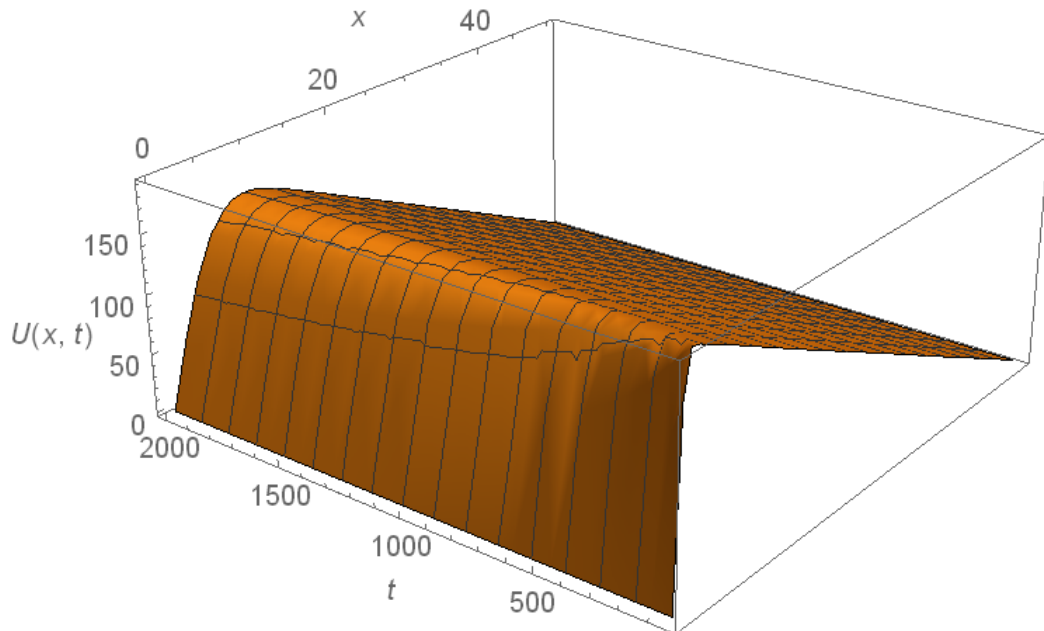


Рис. 4. Просторовий розподіл температури в стержні виготовленого з бетону

Як видно з порівняння рисунків 3 і 4, для стержня, виготовленого з бетону остигання відбувається повільність, ніж для стержня, виготовлено з заліза, оскільки коефіцієнт його теплопровідності менший.

Більш наочно це можна показати обчисливши температуру всередині стержня $x = \frac{L}{2}$ для досить великого значення часу t .

При $t = 10000 \text{ c}$ маємо:

$$u_{\text{бетон}} = 97,516^{\circ}\text{C};$$

$$u_{\text{залізо}} = 0,341^{\circ}\text{C}.$$

Таким чином можна зробити висновок, що бетон є хорошим теплоізолятором.

Висновки. Результатом проведеного дослідження з математичного моделювання теплопереносу в нагрітих стержнях встановлено залежності

температури в стержні від часу та розподіл температури всередині стержня. Показано, що основним чинником для швидкості остигання стержня є теплопровідність матеріалу, з якого він виготовлений.

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для моделювання явища перенесення тепла у нагрітих стержнях методом Фур'є

"Розрахунок коефіцієнтів розкладу у ряд Фур'є"

`Assuming[n ∈ Integers, { $\frac{2}{L} \int_0^L u_0 \sin\left[\frac{n \pi x}{L}\right] dx$ }]`
[предполагая [множество целых чисел] ... [синус]

$$\left\{ \frac{2 \left(1 + (-1)^{1+n} \right) u_0}{n \pi} \right\}$$

"Вхідні дані задачі"

`u0 := 100;`

`L := 50;`

`k = 0.15;`

`u[x_, t_, n_] :=  $\frac{4 u_0}{\pi} \sum_{i=1}^n \frac{1}{i} \exp\left[-\frac{i^2 \pi^2 k t}{L^2}\right] \sin\left[\frac{i \pi x}{L}\right]$`
[показательная ... [синус]

"Розрахунок залежностей температури всередині стержня від часу"

`Plot[{u[L/2, t, 1], u[L/2, t, 5], u[L/2, t, 25], u[L/2, t, 50]}, {t, 0, 2000},`
[график функции]

`PlotRange → All, AxesLabel → {t, U[t]}, PlotLegends → "Expressions"]`
[отображаем ... [всё [обозначения на осях [легенды графика]

"Розрахунок розподілу температури всередині стержня "

`Plot[{u[x, 500, 25], u[x, 1000, 25], u[x, 1500, 25], u[x, 2000, 25]},`
[график функции]
`{x, 0, L}, PlotRange → All, AxesLabel → {x, U[x]}, PlotLegends → "Expressions"]`
[отображаем ... [всё [обозначения на осях [легенды графика]

"Просторовий розподіл температури для залізного стержня"

`Plot3D[u[x, t, 50], {x, 0, 50}, {t, 10, 2000}, PlotRange → All, AxesLabel → {x, t, U[x, t]}]`
[график функции 2-х переменных [отображаем ... [всё [обозначения на осях]

"Просторовий розподіл температури для бетонного стержня"

`k = 0.005;`

`Plot3D[u[x, t, 50], {x, 0, 50}, {t, 50, 2000}, PlotRange → All, AxesLabel → {x, t, U[x, t]}]`
[график функции 2-х переменных [отображаем ... [всё [обозначения на осях]

"Температура всередині бетонного стержня"

`k = 0.005;`

`N[u[L/2, 10000, 50]]`
[численное приближение]

97.5161

"Температура всередині залізного стержня"

`k = 0.15;`

`N[u[L/2, 10000, 50]]`
[численное приближение]

0.341288

ЛАБОРАТОРНА РОБОТА № 8

" СИСТЕМИ НЕЛІНІЙНИХ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ЛОРЕНЦА У МОДЕЛЮВАННІ ГЕНЕРАТОРІВ ДИНАМІЧНОГО ХАОСУ "

Розглядається динамічна система, стан якої описується системою нелінійних диференціальних рівнянь Лоренца, що генератором динамічного хаосу.

1. *Математична модель генератора динамічного хаосу*

Математична модель генератора динамічного хаосу задається системою нелінійних диференціальних рівнянь Лоренца:

$$\begin{cases} \frac{dx}{dt} = -ax + ay, \\ \frac{dy}{dt} = -xz + bx - y, \\ \frac{dz}{dt} = xy - cz, \end{cases} \quad (1)$$

де $x = x(t)$; $y = y(t)$; $z = z(t)$.

Дана система є класичною, тривимірною та забезпечує отримання двопетлевого атрактора Лоренца для заданих значень параметричних коефіцієнтів a, b, c та при зміні аргумента t .

Для побудови математичної моделі задачі, для системи (1) застосуємо метод скінченних різниць.

Надамо функціям $x = x(t)$; $y = y(t)$; $z = z(t)$ приростів $\Delta x, \Delta y, \Delta z$ по вихідних параметрах та для приросту аргумента Δt .

Тоді система (1) набуває вигляду:

$$\begin{cases} \frac{\Delta x}{\Delta t} = -ax_{i-1} + ay_{i-1}, \\ \frac{\Delta y}{\Delta t} = -xz_{i-1} + bx_{i-1} - y_{i-1}, \\ \frac{\Delta z}{\Delta t} = x_{i-1}y_{i-1} - cz_{i-1}, \end{cases} \quad (2)$$

звідки:

$$\begin{cases} \Delta x = (-ax_{i-1} + ay_{i-1})\Delta t, \\ \Delta y = (-xz_{i-1} + bx_{i-1} - y_{i-1})\Delta t, \\ \Delta z = (x_{i-1}y_{i-1} - cz_{i-1})\Delta t. \end{cases} \quad (3)$$

У (3): i - номер поточної точки, $i=1..N$, Δt - скінченний приріст аргумента t .

2. Вхідні дані

$a=5$; $b=1$; $c=1$ - значення параметричних коефіцієнтів;

$\Delta t=0,05$ - крок розрахунку по аргументу t ;

$N=1000$ - кількість розрахункових точок;

$t=\{0..1000\}$ - діапазон зміни аргумента t .

3. Алгоритмізація побудови розв'язку математичної моделі

Побудуємо динамічні атрактори задані системою рівнянь Лоренца у представленні методу скінченних різниць для різних значень параметричних аргументів.

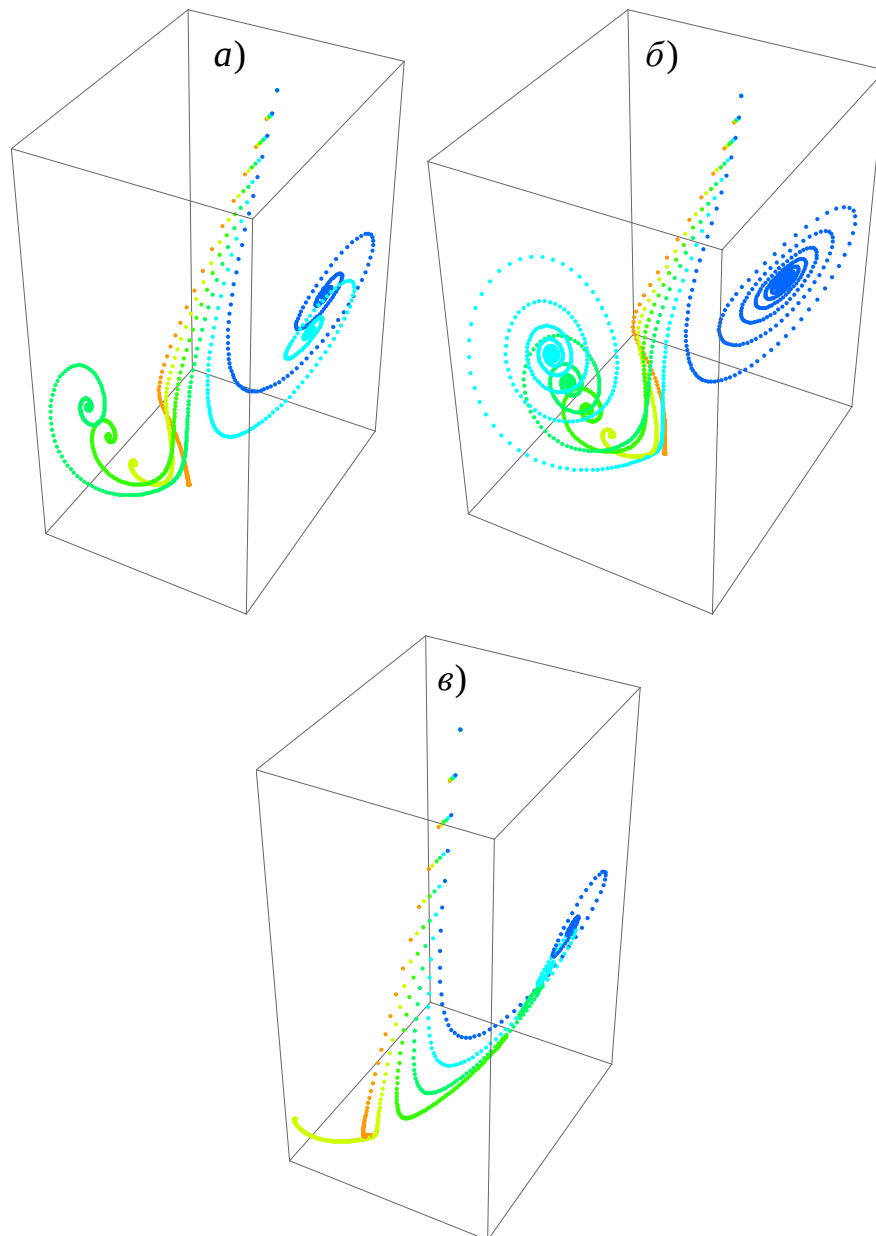


Рис.1. Атрактори Лоренца, генеровані з системи (3) для різних значень параметричних аргументів. а) $a=1; b=1; c=1$; б) $a=2; b=1; c=1$; в) $a=1; b=1; c=2$.

Побудуємо динамічні атрактори задані системою рівнянь Лоренца у представленні методу скінченних різниць для різного значення кількості витків у генерованому динамічному атракторі при значенні значень параметричних аргументів $a=1; b=1; c=1$.

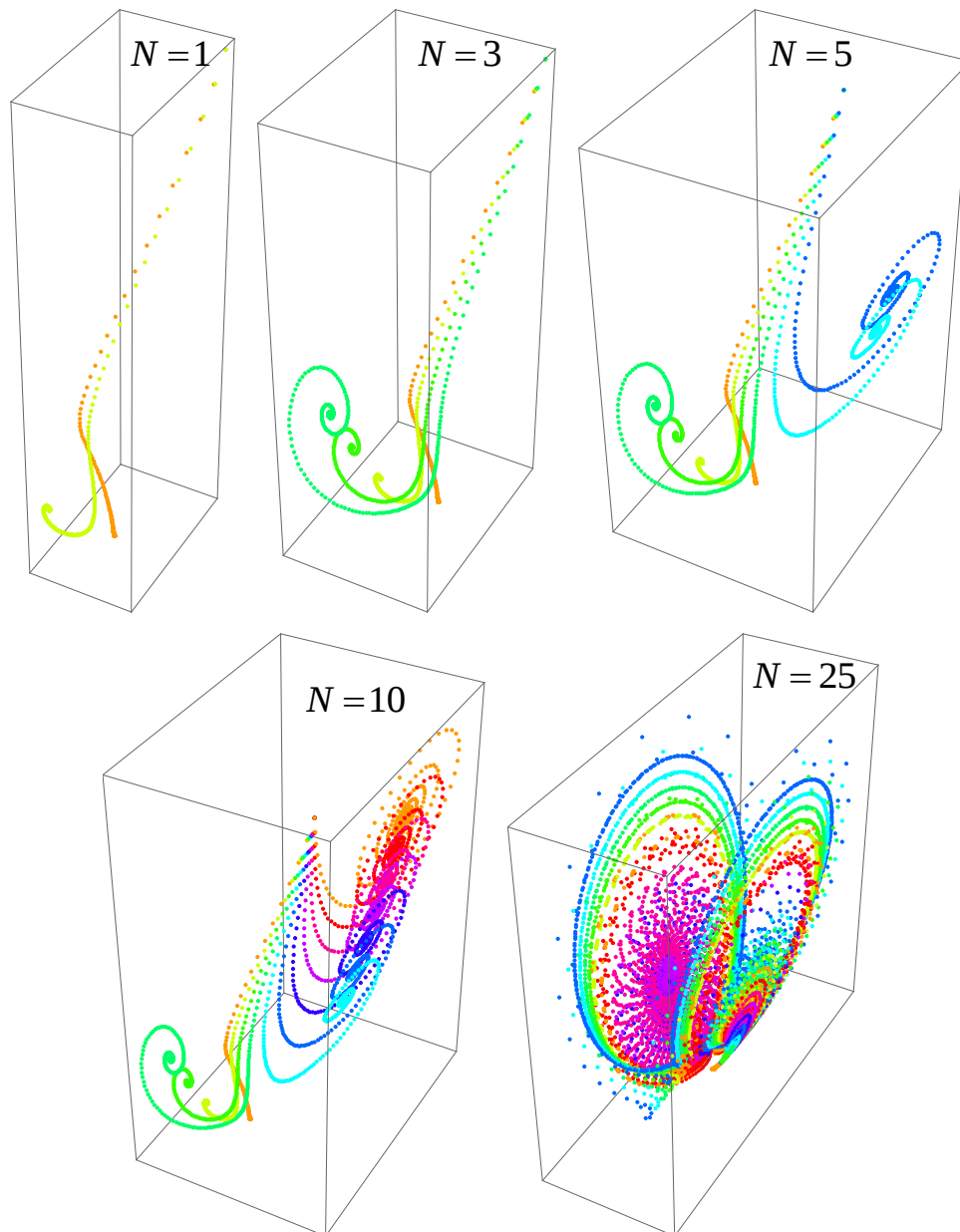


Рис.2. Атрактори Лоренца, генеровані з системи (3) для різних значень кількості витків у генерованому динамічному атракторі при значенні значень параметричних аргументів $a=1$; $b=1$; $c=1$.

Висновки. Результатом проведеного дослідження з математичного моделювання генераторів динамічного хаосу на основі системи нелінійних рівнянь Лоренца є просторові атрактори, отримані для різних значень параметричних коефіцієнтів та різного діапазону аргумента t .

Додатки.

Лістинг програми у середовищі Wolfram Mathematica для генераторів динамічного хаосу

(*Розрахунок просторових значень координат динамічного атрактора згідно методу скінченних різниць*)

```
In[74]:= data = Table[With[{N = 1000, dt = 0.05, a = 1, b = 1 + j, c = 1},  
  Табл...  испо...  численное приближение  
  NestList[Module[{x, y, z, x1, y1, z1}, {x, y, z} = #;  
    список и...  программный модуль  
    x1 = x + a (-x + y) dt;  
    y1 = y + (b x - y - z x) dt;  
    z1 = z + (-c z + x y) dt;  
    {x1, y1, z1}] &, {1, 1, 1}, N]], {j, 0, 25}];  
                                         численное приближение
```

(*Параметрична побудова просторового динамічного атрактора*)

```
Graphics3D@MapIndexed[{Hue[0.1 First[#2]], Point[#1]} &, data]  
3-мерная г...  преобразоват...  тон  первый  точка
```

