

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»**

## **МЕТОДИЧНІ ВКАЗІВКИ**

**до лабораторних занять та самостійної роботи  
для підготовки докторів філософії зі спеціальностей:  
122 — Комп'ютерні науки, 131 — Прикладна механіка**

## **ПРОБЛЕМИ КОНСТРУЮВАННЯ ОБ'ЄКТІВ У КОМП'ЮТЕРНИХ СИСТЕМАХ**



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

**МЕТОДИЧНІ ВКАЗІВКИ**

до лабораторних занять та самостійної роботи для підготовки  
докторів філософії зі спеціальностей: 122 — Комп'ютерні  
науки, 131 — Прикладна механіка

**ПРОБЛЕМИ КОНСТРУЮВАННЯ ОБ'ЄКТІВ  
У КОМП'ЮТЕРНИХ СИСТЕМАХ**

Затверджено  
редакційно-видавничою  
радою університету,  
протокол № 2 від 17.05.2019 р.

Харків 2019

Методичні вказівки до лабораторних занять та самостійної роботи для підготовки докторів філософії зі спеціальностей: 122 — Комп'ютерні науки, 131 — Прикладна механіка / Уклад. І. Ю. Адашевська. — Харків : «НТМТ», 2019. – 104 с.

Укладач: І. Ю. Адашевська

Рецензент О. Ю. Ніцин, д.т.н., проф.

Кафедра геометричного моделювання та комп'ютерної графіки

## Лабораторна робота 1

### Основи роботи з MathCAD

#### Математичні вирази~Типи даних~Оператори~Функції~Дискретні аргументи~Масиви~Текстові фрагменти~Графічні області~Створення анімаційного кліпу~Порядок виконання лабораторної роботи 1

Mathcad працює з *документами*. З точки зору користувача, документ — це чистий аркуш паперу, на якому можна розмішувати області трьох основних типів: математичні вирази, текстові фрагменти і графічні області.

#### **Математичні вирази**

До основних елементів математичних виразів Mathcad відносяться *типи даних, оператори, функції і керуючі структури*.

#### **Типи даних**

До типів даних відносяться числові константи, звичайні і системні змінні, масиви (вектори і матриці) і дані файлового типу.

*Константами* називають пойменовані об'єкти, що зберігають деякі значення, які не можуть бути змінені. Змінні є пойменованими об'єктами, що мають деякі значення, які можуть змінюватися по ходу виконання програми. Імена констант, змінних та інших об'єктів називають *ідентифікаторами*. Ідентифікатори в Mathcad є набором латинських або грецьких букв і цифр.

У Mathcad міститься невелика група особливих об'єктів, які не можна віднести ні до класу констант, ні до класу змінних,

значення яких визначені одразу після запуску програми. Їх вірніше вважати системними змінними, що мають визначені системою початкові значення.

Звичайні змінні відрізняються від системних тим, що вони повинні бути попередньо визначені користувачем, тобто їм необхідно хоча б один раз присвоїти значення. В якості оператора присвоєння використовується знак  $:=$ , тоді як знак  $=$  відведений для виводу значення чи константи змінної.

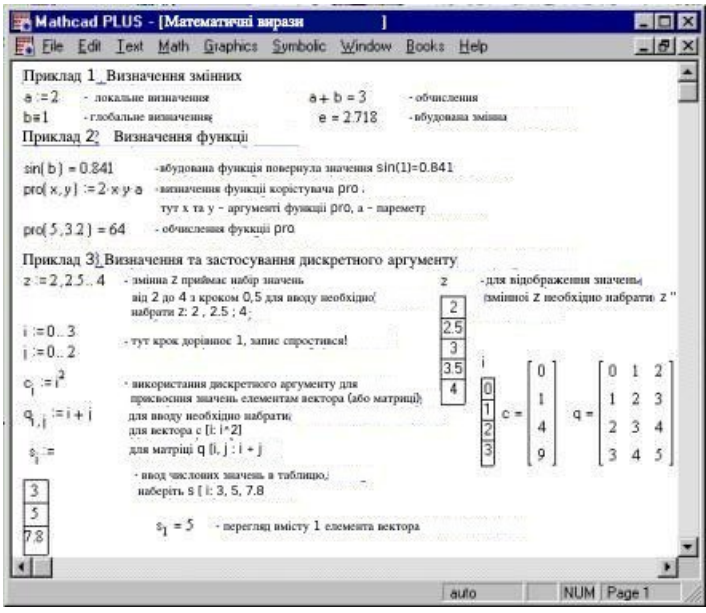


Рисунок 1. Математичні вирази

Якщо змінній присвоюється початкове значення за допомогою оператора  $:=$ , таке присвоєння називається локальним. До цього присвоєння змінна не визначена і її не можна використовувати. Однак за допомогою знака  $\equiv$  можна

забезпечити глобальне присвоєння (див. Приклад 1 рисунку 1). Існує також жирний знак рівності, який використовується, наприклад, як оператор наближеної рівності при вирішенні систем рівнянь.

### **Оператори**

Оператори — елементи Mathcad, за допомогою яких можна створювати математичні вирази. До них, наприклад відносяться символи арифметичних операцій, знаки обчислення сум, добутків, похідної та інтеграла і т.д. Після вказання операндів (параметрів операторів) оператори стають виконуваними по документу блоками, наприклад,  $2 + 5$ -оператор складання з двома операндами. У Додатку 2 даного посібника наведено список найбільш часто використовуваних операторів.

### **Функції**

У пакеті Mathcad є безліч вбудованих функцій, тобто функцій, завчасно введених розробниками (див. Додаток 3). Головною ознакою функції є повернення значення, тобто функція у відповідь на звернення до неї по імені з вказівкою її аргументів повинна повернути своє значення.

Важливою особливістю пакету є можливість завдання зовнішніх функцій, або функцій користувача. Слід особливо відзначити різницю між аргументами і параметрами функції. Змінні, зазначені в дужках після імені функції, є її аргументами і замінюються при обчисленні функції значеннями з дужок. Змінні в правій частині визначення функції, не зазначені дужках в лівій частині, є параметрами і повинні задаватися до визначення функції (див. Приклад 2, рис. 1).

### **Дискретні аргументи**

Дискретні аргументи — особливий клас змінних, який в пакеті Mathcad часто замінює керуючі структури, названі циклами (однак така заміна не є повноцінною). Ці змінні мають

ряд фіксованих значень, або цілочисельних, або у вигляді чисел з певним кроком, що міняються від початкового значення до кінцевого.

Дискретні аргументи значно розширюють можливості Mathcad, дозволяючи виконувати багаторазові обчислення чи цикли з повторними обчисленнями, формувати вектори і матриці (Приклад 3, рис. 1).

### **Масиви**

Масив – сукупність (що має унікальне ім'я) кінцевого числа числових чи символічних елементів, впорядкованих деяким чином, що мають певні адреси. У пакеті Mathcad використовуються масиви двох найбільш поширених типів: одномірні (вектори) і двовимірні (матриці).

Порядковий номер елемента, який є його адресою, називається індексом. Індеси можуть мати тільки цілочисельні значення. Вони можуть починатися з нуля або одиниці, у відповідності зі значенням системної змінної ORIGIN (див. Додаток 1).

Вектори і матриці можна задавати різними способами:

- за допомогою команди **Math** → **Matrics**,
- з використанням дискретного аргументу (Приклад 3, рис. 1).

### **Текстові фрагменти**

Текстові фрагменти являють собою шматки тексту, які користувач хотів би бачити в своєму документі. Існують два види текстових фрагментів — текстова область (**region**) і текстовий діапазон (**band**). Текстові області призначені для невеликих шматків тексту — підписів, коментарів і т.п. Текстові діапазони застосовуються в тому випадку, якщо необхідно працювати з абзацами або сторінками.

## Графічні області

Графічні області поділяються на три основних типи — двовимірні графіки, тривимірні графіки й імпортовані графічні образи. Двовимірні і тривимірні графіки будуються самим Mathcad на підставі оброблених даних.

### Створення анімаційного кліпу

Mathcad має вбудовану змінну **FRAME**, чиє єдине призначення — управління анімаціями:

- Створіть об'єкт, чий вигляд залежить від **FRAME**.
- Виберіть **Windows** → **Animation** → **Create** для виклику діалогового вікна.
- Укладіть в пунктирний прямокутник, що виділяє, частину робочого документа, яку потрібно анімувати.
- Встановіть нижні і верхні межі **FRAME**.
- В поле **At (Temp)** введіть значення швидкості відтворення (кадр / сек).
- Виберіть **Animate**. Зараз анімація тільки створюється.
- Збережіть анімацію як **AVI** файл (Save as).
- Розпочніть відтворення збереженої анімації **Windows** → **Animation** → **Playback**.

### Повідомлення про помилки

При виконанні обчислень можливі помилки. Повідомлення про помилку в Mathcad виводиться в червоному прямокутнику, від якого відходить лінія, яка вказує на місце помилки. У Додатку 4 наведено список повідомлень про помилки.

### Порядок виконання лабораторної роботи 1

**Завдання 1.** Обчислити:

$$\sqrt{100} = , | -10 | = , 10! = .$$

Це і всі інші завдання забезпечити коментарями, використовуючи команди **Text** → **Create Text Region** або **Text** → **Create Text Paragraph**.

**Завдання 2.** Визначити змінні:  $a = 3.4$ ,  $b = 6.22$ ,  $c = 0.149$  (причому змінну  $c$  — глобально) і вирази:

$$Z := \frac{2ab + \sqrt[3]{c}}{\sqrt{(a^2 + b^{a+c})} \cdot c}, \quad N := e^{\sin c} \cos \frac{a}{b}$$

- обчислити вирази.
- За допомогою команди **Math** → **Numerical Format** → **Displayed Precision** змінити точність відображення результатів обчислення *глобально*.

**Завдання 3.** Вивести на екран значення *системної константи  $\pi$*  і встановити максимальний формат її відображення *локально*.

**Завдання 4.** Виконати наступні операції з комплексними числами:

$$Z := -3 + 2i, |Z| =, \operatorname{Re}(Z) =, \operatorname{Im}(Z) =, \arg(Z) =,$$

$$\sqrt{Z} =, \sqrt{-5} =, 2Z =,$$

$$Z1 := 1 + 2i, Z2 := 3 + 4i, Z1 + Z2 =, Z1 - Z2 =, Z1 Z2 =, Z1 / Z2 =.$$

**Завдання 5.** Виконати наступні операції:

$$i := 1 \dots 10, \sum_i i \quad \prod_i (i+1) =, =,$$

$$\int_{b^8}^{0.4} x^2 \cdot \lg(x+2) dx \quad \int_{b^8}^{1.2} \frac{\operatorname{ctg} 2x}{(\sin 2x)^2} dx =, =,$$

$$x := 2, \frac{d}{dx} x^5 =, \frac{d}{dx} \sin(x) =.$$

$i$	$d_i$	$S_i$	$R_i$
0	0.5	3.3	2
1	1	5.9	3.9
2	1.5	7	4.5
3	2	6.3	3.7
4	2.5	4.2	1.2

**Завдання 6.** Визначити вектори  $d$ ,  $S$  і  $R$  через дискретний аргумент  $i$ . Відобразити графічно таблично задані функції  $R_i(d_i)$  і  $S_i(d_i)$ , використовуючи команду

**Graphics** → **CreateX-YPlot**. Щоб оформити графік, необхідно виконати наступні команди:

- Клацнути мишею на графіку, щоб виділити його, при цьому MathCAD замінить меню **Graphics** на меню **X-Y Plot**.
- Обрати **X-Y Plot** → **Format** (з'явиться діалогове вікно "**Formatting Currently Selected X-Y Plot**") і відформатувати графік так, щоб в кожній вузловій точці графіка функції  $S_i(d_i)$  стояв знак виду (**Traces**→**Symbol**→**box**), а графік функції  $R_i(d_i)$  відобразити у вигляді гістограми (**Trace**→**tupe**→**bar**).
- Нанести лінії сітки на графік (**X-Y Axes**→**Grid Lines**) і відобразити легенду (**Traces**→**Hide Legend**).

**Завдання 7.** Побудувати декартові (**X-YPlot**) і полярні (**PolarPlot**) графіки наступних функцій:

$$X(\alpha) := \cos(\alpha) \cdot \sin(\alpha),$$

$$Y(\alpha) := 1.5 \cos(\alpha)^2 - 1,$$

$$P(\alpha) := \cos(\alpha).$$

Для цього необхідно визначити  $\alpha$  як дискретний аргумент на інтервалі від 0 до  $2\pi$  з кроком  $\pi/30$ .

Визначити за графіком **X-Y Plot** координати будь-якої з точок перетину графіків  $Y(\alpha)$  і  $P(\alpha)$ , для цього необхідно:

- Виділити графік і вибрати **X-Y Plot**→**Zoom** (з'явиться діалогове вікно "**X-Y Zoom**") для збільшення частини графіка в області точки перетину.
- На кресленні виділити пунктирним прямокутником околицю точки перетину графіків  $Y(\alpha)$  і  $P(\alpha)$ , яку потрібно збільшити.
- Натиснути кнопку **Zoom**, щоб перемалювати графік.
- Щоб зробити це зображення постійним, вибрати **Accept**.
- Вибрати **X-Y Plot**→**Trace** (з'явиться діалогове вікно "**X-Y Trace**").
- Всередині креслення натиснути кнопку миші і перемістити курсор миші на точку, чії координати потрібно побачити.

- Вибрати **Copy X** (чи **Copy Y**), на вільному полі документа набрати  $X_{per}:=$  (або  $Y_{per}:=$ ) і вибрати пункт меню **Edit**  $\Rightarrow$  **Paste**.

Обчислити значення функцій  $X(\alpha)$  і  $Y(\alpha)$  при  $\alpha := \pi/2$ .

**Завдання 8.** Використовуючи команду **Math** $\rightarrow$ **Matrics** створити матрицю  $Q$  розміром 6 на 6, заповнити її довільно і відобразити графічно за допомогою команди **Graphics**  $\rightarrow$  **Create Surface Plot**.

**Завдання 9.** Побудувати графік поверхні (**Surface Plot**) і карту ліній рівня (**Contour Plot**) для функції двох змінних:

$$X(t, \alpha) := t \cdot \cos \alpha \cdot \sin \alpha$$

- Визначити функцію  $X(t, \alpha)$
- Задати на осях змінних  $t$  і  $\alpha$  по 41 точці ( $i:= 0..40, j:= 0..40$ ): для змінної  $t_i$  зі значеннями, що змінюються від -5 до 5 з кроком 0.25 ( $t_i := -5 + 0.25 i$ ), а для змінної  $\alpha_j$  — від 0 до  $2\pi$  з кроком  $\pi/20$  ( $\alpha_j := \pi/20 j$ ).
- Визначити матрицю  $M_{i,j} := X(t_i, \alpha_j)$  і відобразити її графічно.

За допомогою команди **Graphics** $\rightarrow$ **3D Plot Format** викликати діалогове вікно "**3D Plot Format**" і змінити:

- характеристики перегляду (**View** $\rightarrow$ **Rotation, Tilt**),
- кольори і лінії поверхні (**Color & Lines** $\rightarrow$ **Shading**),
- параметри осей (**Axes**),
- видазаголовка графіка (**Title**).

**Завдання 10.** Використовуючи змінну **FRAME** і команду **Animation** $\rightarrow$ **Create**, створити анімаційні кліпи за допомогою даних наведених у таблиці:

### Варіанти завдання 10

№ варіанта	Змінні	Функції	FRAME	Тип графіка
<b>1</b>	$x := 0,0.1..30$	$f(x) := x +$ FRAME	від 0 до 20	полярний <b>(Polar Plot)</b>
<b>2</b>	$i := 0..FRAME + 1$	$g_i := 0.5 i$ $\cos(i)$ $h_i := i \sin(i)$ $k_i := 2 i$	від 0 до 50	тривимірний точковий графік <b>(3D Scatter Plot)</b> кордону на осях Min Max $x-50\ 50$ $y-50\ 50$ $z0\ 50$
<b>3</b>	$i := 0..20$ $j := 0..20$	$f(x, y) := \sin(x^2 + y^2 +$ FRAME) $x_i := -1.5 + 0.15i$ $y_j := -1.5 + 0.15j$	від 0 до 50	графік поверхні <b>(Surface Plot)</b>
<b>4</b>	$r :=$ FRAME, $R := 6$ $m := 0..20$ $n := 0..20$	$x_{m,n} := (R + r \cdot \cos(v_n)) \cos(w_m)$ $y_{m,n} := (R + r \cos(v_n)) \sin(w_m)$ $z_{m,n} := r \sin(v_n)$	від 0 до 11	графік параметричної поверхні <b>(Surface Plot)</b> (межі на всіх осях встановити від -11 до 11)

5	$r :=$ FRAME, $R := 6$ $m := 0..20$ $n := 0..20$	$x_{m,n} := (R$ + $r \cdot \cos(v_n))$ $\cos(w_m)$ $y_{m,n} := (R$ + $r \cdot \cos(v_n))$ $\cdot \sin(w_m)$ $z_{m,n} :=$ $r \cdot \sin(v_n)$	від 0 до 11	графік параметричної поверхні <b>(Surface Plot)</b> (межі на всіх осях встановити від -11 до 11)
---	--	---	----------------	---

## Лабораторна робота 2

### Вирішення рівнянь

Ітераційні методи~Рішення рівнянь засобами  
Mathcad~Символьний розв'язок рівнянь і систем  
рівнянь~Порядок виконання лабораторної роботи 2

Як відомо, багато рівнянь і системи рівнянь не мають аналітичних рішень. У першу чергу, це відноситься до більшості трансцендентних рівнянь. Доведено також, що не можна побудувати формулу, по якій можна було б вирішити довільне алгебраїчне рівняння ступеня вище четвертої. Однак такі рівняння можуть вирішуватися ітераційним методом із заданою точністю.

### Ітераційні методи

Задача знаходження кореня рівняння  $f(x) = 0$  ітераційними методами полягає в наступному:

- *відділення коренів* — відшукання наближеного значення кореня (наприклад, графічним методом);
- *уточнення коренів* — доведення їх значень до заданого ступеня точності  $\varepsilon$ .

При використанні *методу Н'ютона* необхідно задатися початковим наближенням  $x_0$ , розташованим досить близько до точного значення кореня. Ітераційний процес будується за формулою:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, f'(x_i) \neq 0, i = 0, 1, \dots \quad (1)$$

Метод *простих ітерацій* рішення рівняння  $f(x) = 0$  полягає в заміні вихідного рівняння еквівалентним йому рівнянням  $x = \varphi(x)$  і побудові ітераційної послідовності за формулою:

$$x_{i+1} = \varphi(x_i), i = 0, 1 \dots \quad (2)$$

Достатньою умовою збіжності розглянутих ітераційних процесів є виконання нерівності

$$|x_i - x_{i-1}| \leq \varepsilon \quad (3)$$

на кожному кроці ітерації.

*until(a, z)* повертає  $z$ , поки вираз  $a$  не стає негативним;  $a$  повинно містити дискретний аргумент.

Малюнок 2 ілюструє використання функції *until* для реалізації *методу Н'ютона*.

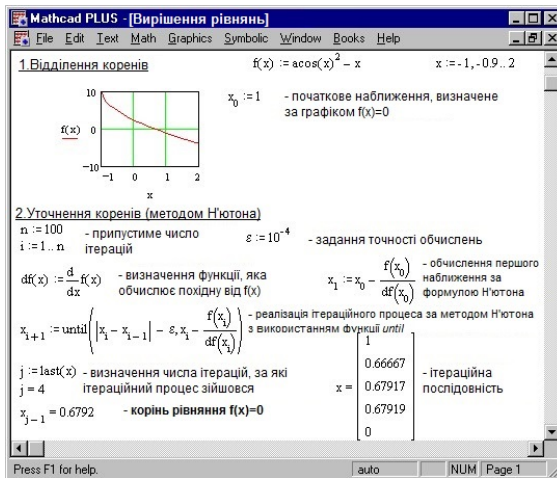


Рисунок 1

## Рішення рівнянь засобами Mathcad

Для найпростіших рівнянь виду  $f(x) = 0$  рішення знаходять за допомогою функції *root*.

$\text{root}(f(z), z)$	Повертає значення $z$ , при якому вираз чи функція $f(z)$ звертається до 0. Обидва аргументи цієї функції повинні бути скалярами. Функція повертає скаляр.
------------------------	--

Перший аргумент — або функція, визначена де-небудь в робочому документі, або вираз. Другий аргумент — ім'я змінної, яка використовується у виразі. Цій змінній перед використанням функції *root* необхідно присвоїти числове значення.

Для знаходження коренів виразу, що має вигляд

$$v_n x^n + \dots + v_2 x^2 + v_1 x + v_0,$$

краще використовувати функцію *polyroots*, ніж *root*. На відміну від функції *root*, функція *polyroots* не вимагає початкового наближення і повертає відразу всі корені, як речові, так і комплексні.

$\text{polyroots}(v)$	Повертає корені полінома ступеня $n$ . Коефіцієнти полінома знаходяться у векторі $v$ довжини $n + 1$ . Повертає вектор довжини $n$ , що складається з коренів полінома.
-----------------------	--

Системи лінійних рівнянь зручно вирішувати за допомогою функції *lsolve*.

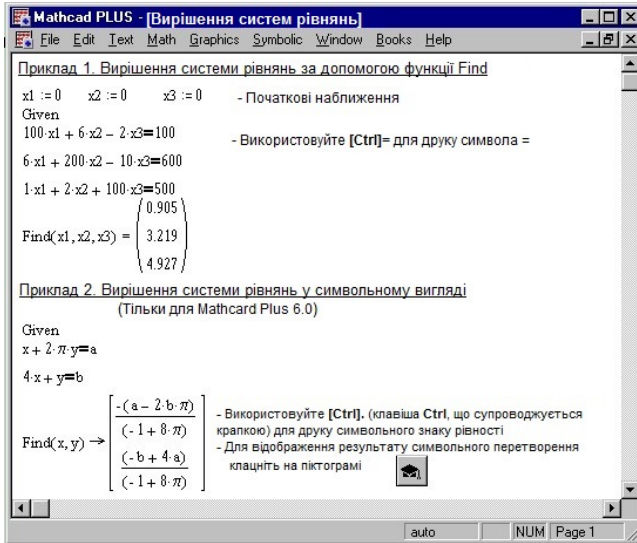


Рисунок 2. Вирішення систем рівнянь

$\text{Isolve}(\mathbf{M}, \mathbf{v})$	Повертається вектор рішення $\mathbf{z}$ такий, що $\mathbf{M} \cdot \mathbf{z} = \mathbf{v}$ .
---	---

При вирішенні систем рівнянь використовується спеціальний обчислювальний блок, що відкривається службовим словом *Given* і закінчується вираженням з функціями *Find* або *Minerr*.

$\text{Find}(z_1, z_2, \dots)$	Повертає точний розв'язок системи рівнянь. Число аргументів має дорівнювати числу невідомих.
$\text{Minerr}(z_1, z_2, \dots)$	Повертає наближений розв'язок системи рівнянь. Число аргументів має дорівнювати числу невідомих.

Приклад 1 на Рисунку 2 ілюструє рішення системи рівнянь за допомогою обчислювального блоку *Given ... Find*.

## Символьний розв'язок рівнянь і систем рівнянь

Якщо задано деякий вираз  $f(x)$  і відзначена змінна  $x$ , то команда **Symbolic**⇒**Solve for Variable**(Вирішити щодо змінної) повертає символьні значення зазначеної змінної  $x$ , при якій  $f(x) = 0$ .

Якщо ви працюєте з пакетом Mathcad PLUS 5.0, не забудьте попередньо використовувати команду **Symbolic**→**Load Symbolic Processor** для завантаження символьного процесора.

Якщо ви працюєте з пакетом Mathcad PLUS 6.0, то зможете вирішувати символьно не тільки рівняння, але і системи рівнянь. Щоб вирішити систему рівнянь у символьному вигляді, не потрібно задавати початкові наближення. Приклад 2 Малюнку 2 показує рішення системи рівнянь у символьному вигляді.

### Порядок виконання лабораторної роботи 2

**Завдання 1.** Побудувати графік функції  $f(x)$  і приблизно визначити один з коренів рівняння.

Вирішити рівняння  $f(x) = 0$  з точністю  $\epsilon = 10^{-4}$ :

- за допомогою вбудованої функції Mathcad *root*;
- методом Ньютона (дотичних), використовуючи функцію *until*;
- методом ітерацій, використовуючи функцію *until*.

Визначити число ітерацій в кожному методі, за допомогою функції *last*.

## Варіанти завдання 1

№ варіанта	$f(x)$	№ варіанта	$f(x)$	№	$f(x)$
1	$3 \sin(\sqrt{x}) + 0.35x - 3.8$ $x \in [2, 3]$	6	$0.25x^3 + x - 2$ $x \in [0, 2]$	11	$\sqrt{2x^2 + 1.2 - \cos x} - 1$ $x \in [0, 1]$
2	$x - \frac{1}{3 + \sin(3.6x)}$ $x \in [0, 1]$	7	$\frac{1 - x^2}{1 + x^2}$ $x$ $x \in [2, 3]$	12	$\cos\left(\frac{2}{x}\right) - 2 \sin\left(\frac{1}{x}\right) + \frac{1}{x}$ $x \in [1, 2]$
3	$\arccos x - \sqrt{1 - 0.3x^3}$ $x \in [0, 1]$	8	$3x - 4 \ln x - 5$ $x \in [2, 4]$	13	$0.1x^2 - x \ln x$ $x \in [1, 2]$
4	$\sqrt{1 - 0.4x^2} - \arcsin x$ $x \in [0, 1]$	9	$e^x - e^{-x} - 2$ $x \in [0, 1]$	14	$1 - x + \sin x - \ln(1 + x)$ $x \in [0, 2]$
5	$3x - 14 + e^x - e^{-x}$ $x \in [1, 3]$	10	$\sqrt{1 - x} - \operatorname{tg} x$ $x \in [0, 1]$	15	$e^{x-1} - x^3 - x$ $x \in [0, 1]$

**Завдання 2.** Для полінома  $g(x)$  виконати наступні дії:

- за допомогою команди **Symbolic**→**Polynomial Coefficients** створити вектор  $V$ , що містить коефіцієнти полінома;
- вирішити рівняння  $g(x) = 0$  за допомогою функції *polyroots*;
- вирішити рівняння символно, використовуючи команду **Symbolic**→**Solve for Variable**;
- розкласти на множники, використовуючи **Symbolic** → **Factor Expression**.

## Варіанти завдання 2

№ варіанта	$g(x)$	№ варіанта	$g(x)$
1	$x^4 - 2x^3 + x^2 - 12x + 20$	9	$x^4 + x^3 - 17x^2 - 45x - 100$
2	$x^4 + 6x^3 + x^2 - 4x - 60$	10	$x^4 - 5x^3 + x^2 - 15x + 50$
3	$x^4 - 14x^2 - 40x - 75$	11	$x^4 - 4x^3 - 2x^2 - 20x + 25$
4	$x^4 - x^3 + x^2 - 11x + 10$	12	$x^4 + 5x^3 + 7x^2 + 7x - 20$
5	$x^4 - x^3 - 29x^2 - 71x - 140$	13	$x^4 - 7x^3 + 7x^2 - 5x + 100$
6	$x^4 + 7x^3 + 9x^2 + 13x - 30$	14	$x^4 + 10x^3 + 36x^2 + 70x + 75$
7	$x^4 + 3x^3 - 23x^2 - 55x - 150$	15	$x^4 + 9x^3 + 31x^2 + 59x + 60$
8	$x^4 - 6x^3 + 4x^2 + 10x + 75$		

**Завдання 3.** Вирішити систему лінійних рівнянь:

- використовуючи функції *Find*;
- матричним способом, використовуючи функцію *lsolve*.

## Варіанти завдання 3

№ варіанта	Система лінійних рівнянь	№ варіанта	Система лінійних рівнянь
1	$\begin{cases} 2x_1 + x_2 + 2x_3 + 3x_4 = 8 \\ 3x_1 + 3x_3 = 6 \\ 2x_1 - x_2 + 3x_4 = 4 \\ x_1 + 2x_2 - x_3 + 2x_4 = 4 \end{cases}$	9	$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = -4 \\ x_1 - 3x_2 - 6x_4 = -7 \\ 2x_2 - x_3 + 2x_4 = 2 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -2 \end{cases}$

2	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 22 \\ 2x_1 + 3x_2 + x_3 + 2x_4 = 17 \\ x_1 + x_2 + x_3 - x_4 = 8 \\ x_1 - 2x_3 - 3x_4 = -7 \end{cases}$	10	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 26 \\ 2x_1 + 3x_2 + 4x_3 + x_4 = 34 \\ 3x_1 + 4x_2 + x_3 + 2x_4 = 26 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$
3	$\begin{cases} 9x_1 + 10x_2 - 7x_3 - x_4 = 23 \\ 7x_1 - x_3 - 5x_4 = 37 \\ 5x_1 - 2x_3 + x_4 = 22 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$	11	$\begin{cases} 2x_1 - 8x_2 - 3x_3 - 2x_4 = -18 \\ x_1 - 2x_2 + 3x_3 - 2x_4 = 28 \\ x_2 + x_3 + x_4 = 10 \\ 11x_2 + x_3 + 2x_4 = 21 \end{cases}$
4	$\begin{cases} 6x_1 - x_2 + 10x_3 - x_4 = 158 \\ 2x_1 + x_2 + 10x_3 + 7x_4 = 128 \\ 3x_1 - 2x_2 - 2x_3 - x_4 = 7 \\ x_1 - 12x_2 + 2x_3 - x_4 = 17 \end{cases}$	12	$\begin{cases} 2x_1 - x_2 + 4x_3 + x_4 = 66 \\ 2x_2 - 6x_3 + x_4 = -63 \\ 8x_1 - 3x_2 + 6x_3 - 5x_4 = 146 \\ 2x_1 - 7x_2 + 6x_3 - x_4 = 80 \end{cases}$
5	$\begin{cases} x_1 - 2x_2 + 6x_3 + x_4 = 88 \\ 5x_1 + 2x_3 - 3x_4 = 88 \\ 7x_1 - 3x_2 + 7x_3 + 2x_4 = 181 \\ 3x_1 - 7x_2 + 5x_3 + 2x_4 = 99 \end{cases}$	13	$\begin{cases} 2x_1 - 3x_3 - 2x_4 = -16 \\ 2x_1 - x_2 + 13x_3 + 4x_4 = 213 \\ 3x_1 + x_2 + 2x_3 + x_4 = 72 \\ x_1 - 12x_3 - 5x_4 = -159 \end{cases}$
6	$\begin{cases} x_1 - 2x_2 - 8x_4 = -7 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -8 \\ x_1 + x_2 - 5x_3 + x_4 = -10 \\ 2x_1 - x_2 + 2x_4 = 7 \end{cases}$	14	$\begin{cases} 7x_1 + 7x_2 - 7x_3 - 2x_4 = 5 \\ 3x_1 + 4x_2 + 5x_3 + 8x_4 = 60 \\ 2x_1 + 2x_2 + 2x_3 + x_4 = 27 \\ 2x_1 - 2x_3 - x_4 = -1 \end{cases}$
7	$\begin{cases} 2x_1 + 2x_2 + 6x_3 + x_4 = 15 \\ -x_2 + 2x_3 + x_4 = 18 \\ 4x_1 - 3x_2 + x_3 - 5x_4 = 37 \\ 3x_1 - 5x_2 + x_3 - x_4 = 30 \end{cases}$	15	$\begin{cases} 6x_1 - 9x_2 + 5x_3 + x_4 = 124 \\ 7x_2 - 5x_3 - x_4 = -54 \\ 5x_1 - 5x_2 + 2x_3 + 4x_4 = 83 \\ 3x_1 - 9x_2 + x_3 + 6x_4 = 45 \end{cases}$
8	$\begin{cases} 4x_1 - 5x_2 + 7x_3 + 5x_4 = 165 \\ 2x_1 + x_2 - 3x_3 - x_4 = -15 \\ 9x_1 + 4x_3 - x_4 = 194 \\ x_1 - x_2 - 2x_3 - 3x_4 = -19 \end{cases}$		

**Завдання 4.** Перетворити нелінійні рівняння системи до виду  $f_1(x) = y$  і  $f_2(y) = x$ .

Побудувати їх графіки і визначити початкове наближення рішення. Вирішити систему нелінійних рівнянь, використовуючи функцію *Minerr*.

### Варіанти завдання 4

№ варіанта	Система нелінійних рівнянь	№ варіанта	Система нелінійних рівнянь
1	$\begin{cases} \sin x + 2y = 2, \\ \cos(y-1) + x = 0.7. \end{cases}$	9	$\begin{cases} \sin y + x = -0.4, \\ 2y - \cos(x+1) = 0. \end{cases}$
2	$\begin{cases} \sin(x+0.5) - y = 1, \\ \cos(y-2) + x = 0. \end{cases}$	10	$\begin{cases} \sin(x+2) - y = 1.5, \\ \cos(y-2) + x = 0.5. \end{cases}$
3	$\begin{cases} \cos x + y = 1.5, \\ 2x - \sin(y-0.5) = 1. \end{cases}$	11	$\begin{cases} \cos(x+0.5) - y = 2, \\ \sin y - 2x = 1. \end{cases}$
4	$\begin{cases} \cos(x+0.5) + y = 0.8, \\ \sin y - 2x = 1.6. \end{cases}$	12	$\begin{cases} \cos(x-2) + y = 0, \\ \sin(y+0.5) - x = 1. \end{cases}$
5	$\begin{cases} \sin(x-1) = 1.3 - y, \\ x - \sin(y+1) = 0.8. \end{cases}$	13	$\begin{cases} \cos(x+0.5) + y = 1, \\ \sin(y+0.5) - x = 1. \end{cases}$
6	$\begin{cases} \cos(x+0.5) + y = 1, \\ \sin y - 2x = 2. \end{cases}$	14	$\begin{cases} \sin(x) - 2y = 1, \\ \cos(y+0.5) - x = 2. \end{cases}$
7	$\begin{cases} -\sin(x+1) + y = 0.8, \\ \sin(y-1) + x = 1.3. \end{cases}$	15	$\begin{cases} 2y - \sin(x-0.5) = 1, \\ \cos(y) + x = 1.5. \end{cases}$
8	$\begin{cases} \sin(x) - 2y = 1, \\ \sin(y-1) + x = 1.3. \end{cases}$		

**Завдання 5.** Символьно вирішити системи рівнянь:

$$\begin{cases} 3x + 4\pi y = a, \\ 2x + y = b. \end{cases} \quad \begin{cases} 2y - \pi z = a, \\ \pi z - z = b, \\ 3y + x = c. \end{cases}$$

## Лабораторна робота 3

### Інтерполяція і передбачення

#### Інтерполяція~Передбачення~Порядок виконання роботи

Апроксимація функцій полягає в наближеній заміні заданої функції  $f(x)$  деякою функцією  $\varphi(x)$  так, щоб відхилення функції  $\varphi(x)$  від  $f(x)$  в заданій області було найменшим. Функція  $\varphi(x)$  при цьому називається *апроксимуючою*. Типовим завданням апроксимації функцій є завдання *інтерполяції*. Необхідність *інтерполяції* функцій в основному пов'язана з двома причинами:

1. Функція  $f(x)$  має складний аналітичний опис, що викликає певні труднощі при його використанні (наприклад,  $f(x)$  є спецфункцією: гамма-функцією, еліптичної функцією та ін.).
2. Аналітичний опис функції  $f(x)$  невідомо, тобто  $f(x)$  задана таблично. При цьому необхідно мати аналітичний опис, що наближено представляє  $f(x)$  (наприклад, для обчислення: значень  $f(x)$  в довільних точках, визначення інтегралів і похідних від  $f(x)$  і т.п.)

#### Інтерполяція

Найпростіша задача *інтерполяції* полягає в наступному. Для заданих  $n + 1$  точок  $x_i = x_0, x_1, \dots, x_n$ , які називаються *вузлами інтерполяції*, і значень в цих точках деякої функції  $f(x_i) = y_0, y_1, \dots, y_n$  побудувати поліном  $\varphi(x)$  (*інтерполяційний поліном*) ступеня  $n$  виду

$$\varphi(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0, \quad (1)$$

що приймає у вузлах інтерполяції  $x_i$  ті ж значення  $y_i$ , що і функція  $f(x_i)$ :

$$\varphi(x_0) = y_0, \varphi(x_1) = y_1, \dots, \varphi(x_n) = y_n, i = 0, 1, \dots, n. \quad (2)$$

### Глобальна інтерполяція

Найпростішим видом глобальної інтерполяції є параболічна інтерполяція, коли, використовуючи описані вище умови (2), для відшукування невідомих  $n + 1$  коефіцієнтів  $a_0, a_1, \dots, a_n$  виразу (1) отримують систему з  $n + 1$  рівнянь:

Інтерполяційна формула Лагранжа:

$$L_n(x) = \sum_{i=0}^n y_i \frac{(x - x_0)(x - x_1) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n)}{(x_i - x_0)(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)}. \quad (4)$$

Для побудови інтерполяційної формули Лагранжа в Mathcad зручно використовувати функцію *if*

if (cond, tval, fval)	Повертає значення <i>tval</i> , якщо <i>cond</i> відмінний від 0 (істина). Повертає значення <i>fval</i> , якщо <i>cond</i> дорівнює 0 (брехня).
-----------------------	--

Часто інтерполювання ведеться для функцій, заданих таблично з рівновіддаленими значеннями аргументу ( $h_i = x_{i+1} - x_i = \text{const}$ ). Введемо попередньо поняття кінцевих різниць:

$$\begin{aligned} \Delta y_i &= y_{i+1} - y_i, (i = 0, 1, \dots, n-1) \\ \Delta^2 y_i &= \Delta y_{i+1} - \Delta y_i, (i = 0, 1, \dots, n-2) \\ \Delta^k y_i &= \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i, (i = 0, 1, \dots, n-k) \end{aligned}$$

З урахуванням введених позначень перша інтерполяційна формула Ньютона має вигляд:

$$t = \frac{x - x_0}{h}, \quad (5)$$

$$P_{n1}(x) = P_{n1}(x_0 + th) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!} \Delta^2 y_0 + \dots + \frac{t(t-1)\dots(t-n+1)}{n!} \Delta^n y_0.$$

Друга інтерполяційна формула має вигляд:

$$t = \frac{x - x_n}{h}, \quad (6)$$

$$P_{n2}(x) = P_{n2}(x_n + th) = y_n + t\Delta y_{n-1} + \frac{t(t+1)}{2!} \Delta^2 y_{n-2} + \dots + \frac{t(t+1)\dots(t+n-1)}{n!} \Delta^n y_0.$$

Однак, інтерполяція при великому числі вузлів призводить до необхідності працювати з многочленами високого ступеня (наприклад, 50-й або навіть 100-й), що є неприйнятним як з точки зору обчислень, так і через схильність таких многочленів до осциляції (коливань) між вузлами сітки. Тому на практиці часто використовують інтерполяцію шматковими многочленами (або *локальну інтерполяцію*).

### Локальна інтерполяція

При *локальній* інтерполяції між різними вузлами обираються різні многочлени невисокого ступеня. У середовищі Mathcad є для цього інструментарій: засоби *лінійної інтерполяції* (функція *linterp*) і *інтерполяції сплайном* (функція *interp*) — лінійним (*lspline*), параболічним (*pspline*) і кубічним (*cspline*). Малюнок 4 показує деякі приклади локальної інтерполяції.

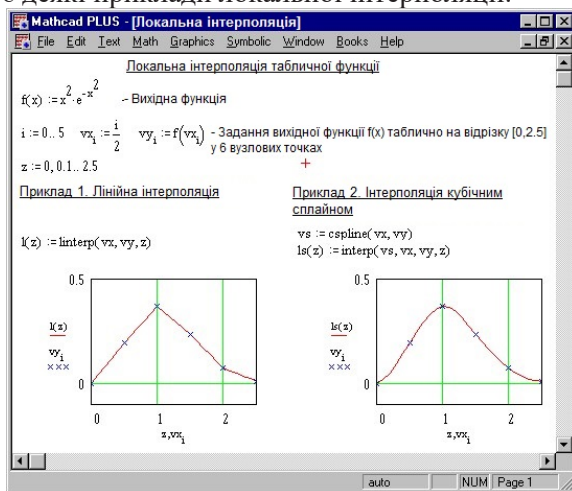


Рисунок 4. Локальна інтерполяція

$\text{linterp}(vx, vy, x)$	Використовує вектори даних $vx$ і $vy$ , щоб повернути лінійно інтерпольовані значення $y$ , відповідне третьому аргументу $x$ .
$\text{lspline}(vx, vy)$ $\text{pspline}(vx, vy)$ $\text{cspline}(vx, vy)$	Всі ці функції повертають вектор коефіцієнтів других похідних, який ми будемо називати $vs$ . Вектор $vs$ , використовується в функції $\text{interp}$ :
$\text{interp}(vs, vx, vy, x)$	Повертає інтерпольовані значення $y$ , відповідному аргументу $x$ .

### Передбачення

Якщо необхідно оцінити значення функції в точках, які не належать до відрізка  $[x_0, x_n]$ , використовуйте функцію  $\text{predict}$  (рис. 5).

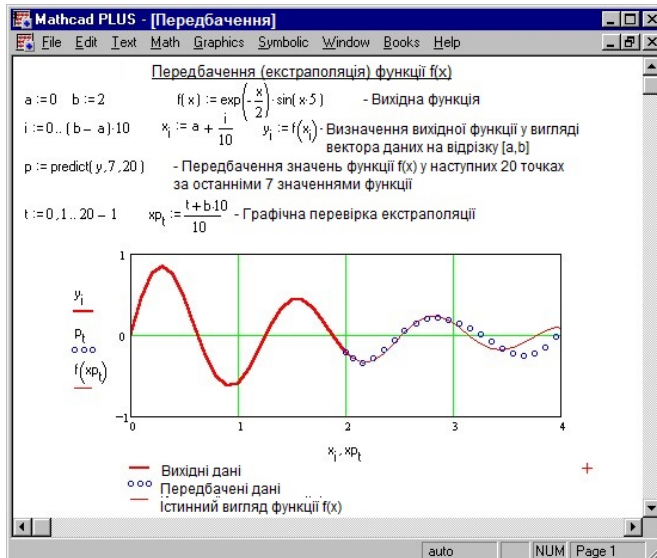


Рисунок 5. Екстраполяція функцій

predict ( $v, m, n$ )	Повертає $n$ передбачених значень, заснованих на $m$ послідовних значеннях вектора даних $v$ .
-----------------------	--

### Порядок виконання лабораторної роботи 3

**Завдання 1.** Обчислити значення заданої функції  $y_i = f(x_i)$  в вузлах інтерполяції  $x_i = a + hi$ , де  $h = (b - a) / 10$ ,  $i = 0, 1, \dots, 10$ , на відріжку  $[a, b]$ .

### Варіанти завдань

№ варіанта	$f(x)$	$[a, b]$	№ варіанта	$f(x)$	$[a, b]$
1	$\sin x^2$	$[0, 2]$	9	$x \cdot \cos(x + \ln(1 + x))$	$[1, 5]$
2	$\cos x^2$	$[0, 2]$	10	$10 \cdot \ln 2x / (1 + x)$	$[1, 5]$
3	$e^{\sin x}$	$[0, 5]$	11	$\sin x^2 \cdot e^{-(x/2)^2}$	$[0, 3]$
4	$1 / (0.5 + x^2)$	$[0, 2]$	12	$\cos(x + \cos^3 x)$	$[0, 2]$
5	$e - (x + \sin x)$	$[2, 5]$	13	$\cos(x + e \cos x)$	$[3, 6]$
6	$1 / (1 + e^{-x})$	$[0, 4]$	14	$\cos(2x + x^2)$	$[0, 1]$
7	$\sin(x + e \sin x)$	$[0, 3]$	15	$e^{\cos x^2} \cos x^2$	$[0, 2]$
8	$e - (x + 1/x)$	$[1, 3]$			

**Завдання 2.** За обчисленою таблицею  $(x_i, y_i)$  провести параболічну інтерполяцію.

Для знаходження коефіцієнтів шуканого полінома (1) необхідно скласти систему лінійних алгебраїчних рівнянь (3).

Систему рівнянь вирішити матрично з використанням функції *lsolve*.

Побудувати графік інтерполяційного многочлена і відзначити на ньому вузлові точки  $(x_i, y_i)$ .

**Завдання 3.** Для обчисленої табличної функції скласти формулу інтерполяційного многочлена *Лагранжа*, використовуючи оператори підсумовування і множення по дискретному аргументу, а також функцію *if*.

Побудувати графік інтерполяційного многочлена і відзначити на ньому вузлові точки  $(x_i, y_i)$ .

**Завдання 4.** Провести інтерполювання заданої функції за допомогою 1<sup>ої</sup> і 2<sup>ої</sup> інтерполяційних формул *Ньютона*.

Побудувати графіки інтерполяційних многочленів і відзначити на ньому вузлові точки  $(x_i, y_i)$ .

**Завдання 5.** Провести *лінійну інтерполяцію* заданої функції за допомогою вбудованої інтерполяційної функції *linterp*.

Побудувати графік функції *linterp* і відзначити на ньому вузлові точки  $(x_i, y_i)$ .

**Завдання 6.** Провести *сплайн-інтерполяцію* за допомогою функцій *lspline*, *pspline*, *cspline* і *interp*.

Побудувати графік функції *interp* і відзначити на ньому вузлові точки  $(x_i, y_i)$ .

**Завдання 7.** Обчислити значення заданої функції  $y_i = f(x_i)$  в точках  $x_i = a + i/10$ , де,  $i = 0, 1, \dots, 10 (b - a)$ , на відрізку  $[a, b]$ .

З використанням функції *predict* виконати передбачення (*екстраполяцію*) отриманого вектора даних  $y_i$  в наступних 10 точках за останніми 7 значеннями функції.

Відобразити графічно наявні дані, передбачені дані і справжній вид функції  $f(x)$ .

## Лабораторна робота 4

### Математична обробка результатів експериментальних даних

#### Метод найменших квадратів~Регресійний аналіз~Згладжування~Порядок виконання лабораторної роботи 4

Вивчаючи в лабораторній роботі № 3 теорію інтерполяції, ви познайомилися з інтерполяційними формулами, які в точності відтворюють значення даної функції в вузлах інтерполяції. Однак в ряді випадків виконання цієї умови затруднене або навіть недоцільне:

1. Якщо задані величини  $x$  і  $y$  є експериментальними даними, то можуть містити в собі істотні помилки, тому що отримані в результаті вимірів або спостережень. Тому побудова апроксимуючого многочлена, що відтворює в точності задане значення функції, означала б ретельне копіювання допущених при вимірах помилок.

2. Якщо є точні значення функції в деяких точках, але число таких точок  $n$  досить велике, то інтерполяційний многочлен буде дуже високого ступеня (якщо тільки різниці НЕ будуть ставати постійними).

Тому виникає задача побудови многочлена деякого цілком визначеного ступеня, але меншого ніж  $n - 1$ , який хоча і не дає точних значень функції у вузлах інтерполяції, але досить близько до них підходить.

#### Метод найменших квадратів

*Метод найменших квадратів* (МНК) полягає в наступному: для даних значень  $x = x_0, x_1, \dots, x_n$  і  $y = y_0, y_1, \dots, y_n$  підібрати многочлен заданого ступеня  $m < n$  виду

$$\varphi(x) = a_m x^m + a_{m-1} x^{m-1} + \dots + a_1 x + a_0, \quad (1)$$

що приймає в заданих точках  $x_i$  значення якомога більш близькі до табличних значень  $y_i$ . Коефіцієнти  $a_i$  многочлена (1) знаходять із рішення системи

$$\begin{cases} b_{00}a_0 + b_{01}a_1 + \dots + b_{0m}a_m = c_0, \\ b_{10}a_0 + b_{11}a_1 + \dots + b_{1m}a_m = c_1, \end{cases} \quad (2)$$

$$\begin{cases} b_{j0}a_0 + b_{j1}a_1 + \dots + b_{jm}a_m = c_j. \end{cases} \quad (3)$$

де  $b_{k,l} = \sum_{i=0}^n x_i^{k+l}, c_k = \sum_{i=0}^n x_i^k y_i, k, l = 0, 1, \dots, m$ .

## Регресійний аналіз

Нехай є два ряди чисел  $x = x_0, x_1, \dots, x_n$  і  $y = y_0, y_1, \dots, y_n$ , при цьому передбачається, що ряд  $y$  будь-яким способом залежить від ряду  $x$ . Завдання *регресійного аналізу* полягає у відновленні математичної залежності (*регресії*)  $y(x)$  за результатами вимірювань  $(x_i, y_i), i = 0, 1, \dots, n$ .

Mathcad включає ряд функцій для обчислення регресії. Функції відрізняються перш за все типом кривої, яку вони використовують, щоб апроксимувати дані.

### Лінійна регресія

Вбудовані функції *intercept* (to intercept - відкласти відрізок на лінії) і *slope* (нахил) вирішують найпростішу і поширену задачу *лінійної регресії* експериментальних даних:

$$f(x) = \text{slope}(vx, vy) x + \text{intercept}(vx, vy)$$

slope (vx, vy)	Повертає скаляр: нахил лінії для даних з vx і vy.
intercept (vx, vy)	Повертає скаляр: зсув по осі ординат лінії регресії для даних з vx і vy.

## Поліноміальна регресія



Використовуйте функцію *regress*, коли потрібно отримати єдиний поліном довільного ступеня, щоб наблизити всі дані. Не рекомендується робити ступінь апроксимуючого полінома вище 4–6, оскільки похибки реалізації регресії сильно зростають.

$regress(vx, vy, n)$	Повертає вектор $vs$ , необхідний <i>interp</i> (див. Лабораторну роботу № 3), щоб знайти поліном порядку $n$ , який найкращим чином наближає дані з $vx$ і $vy$ .
----------------------	--

Приклад 1. Рисунок 6 ілюструє використання функції *regress*. Так як *regress* наближає всі крапки даних, використовуючи один поліном, це не дає хороший результат, коли дані не пов'язані єдиною поліноміальною залежністю.



Функція *loess* полегшує ці проблеми, виконуючи локальне наближення. Замість одного полінома *loess* створює різні поліноми другого порядку в залежності від розташування на кривій (див. Приклад 2 рис. 6).

$Loess(vx, vy, span)$	Повертає вектор $vs$ , необхідний <i>interp</i> , щоб знайти набір поліномів другого порядку, які найкращим чином наближають певні околиці вибіркової точки, визначених у векторах $vx$ і $vy$ . Аргумент $span > 0$ визначає, наскільки великі околиці <i>loess</i> буде використовувати при виконанні локального наближення.
-----------------------	--

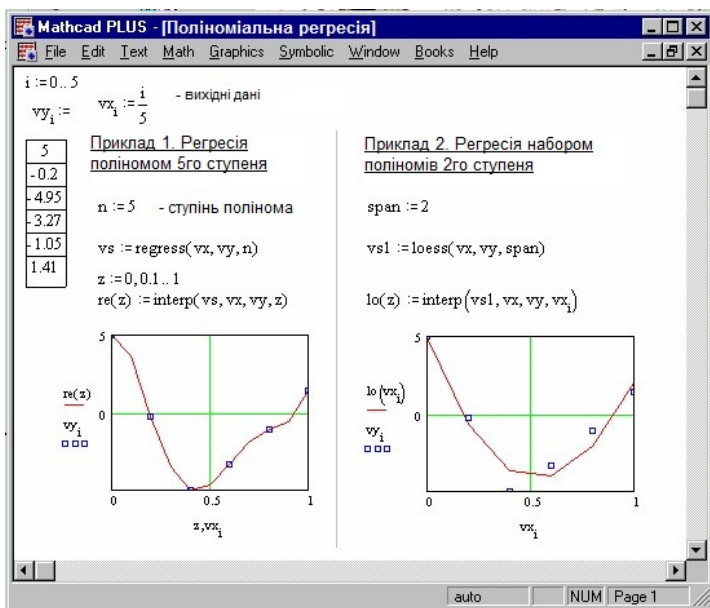


Рисунок 6. Поліноміальна регресія

### Узагальнена регресія

Лінійна або поліноміальна регресія не у всіх випадках підходять для опису залежності даних. Буває, що потрібно шукати цю залежність у вигляді лінійних комбінацій довільних функцій, жодна з яких не є поліномом. Якщо передбачається, що дані могли б бути змодельовані у вигляді лінійної комбінації довільних функцій

$$f(x) = a_0 f_0(x) + a_1 f_1(x) + \dots + a_n f_n(x),$$

слідвикористовуватилinfit, щобобчислити $a_i$ . Це так звана лінійна регресія загального вигляду (Приклад 1, рис. 7).

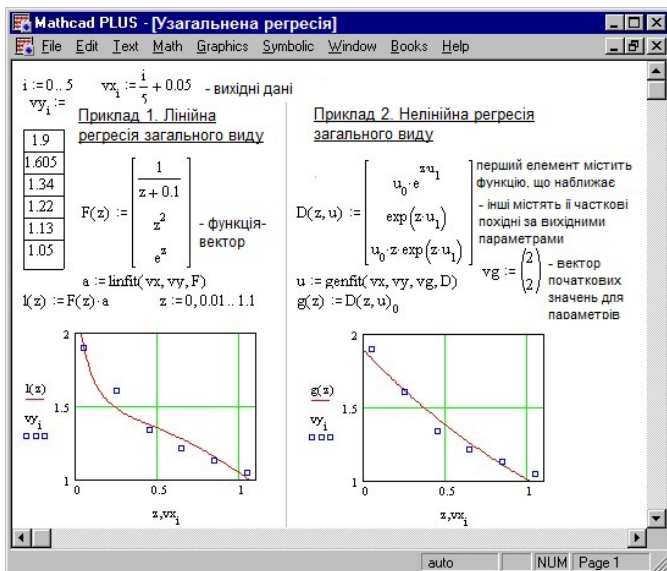


Рисунок 7. Узагальнена регресія

$\text{linfit}(vx, vy, F)$	Повертає вектор коефіцієнтів лінійної регресії загального вигляду, щоб створити лінійну комбінацію функцій з $F$ , що дає найкращу апроксимацію даних з векторів $vx$ і $vy$ . $F$ — функція-вектор, що складається з функцій, які потрібно об'єднати у вигляді лінійної регресії.
----------------------------	--

Якщо дані повинні бути змодельовані у вигляді  $f(x) = f(x, u_0, u_1, \dots, u_n)$ , потрібно використовувати функцію  $\text{genfit}$ , щоб знайти невідомі параметри  $u_i$ . Це *нелінійна регресія загального вигляду* (Приклад 2 рис. 7).

$\text{genfit}(vx, vy, vg, F)$	Повертає вектор $n$ параметрів $u_0, u_1, \dots, u_{n-1}$ , які забезпечують найкраще наближення даних з $vx$ і $vy$ функцією
--------------------------------	---

	<p><math>f</math>, що залежить від <math>x</math> і параметрів <math>u_0, u_1, \dots, u_{n-1}</math>. <math>F</math> — функція-вектор, що складається з <math>f</math> та її приватних похідних (ось тут-то і знадобляться засоби символічної математики (Детальніше див. Лабораторну роботу 5)) щодо параметрів. <math>vg</math> - <math>n</math>-мірний вектор початкових значень для <math>n</math> параметрів.</p>
--	--

### Згладжування



*Згладжування* передбачає використання набору значень  $y$  (і можливо  $x$ ) і повернення нового набору значень  $y$ , який є більш гладким, ніж вихідний набір. На відміну від регресії і інтерполяції, згладжування призводить до нового набору значень  $y$ , а не до функції, яка може оцінювати значення між заданими точками даних.

$ksmooth(vx, vy, b)$	Повертає $n$ -мірний вектор, створений згладжуванням за допомогою гауссова ядра даних з $n$ -мірного вектора $vy$ . Параметр $b$ управляє вікном згладжування і повинен бути в кілька разів більше величини інтервалу між точками $x$ .
$medsmooth(vy, m)$	Повертає $n$ -мірний вектор, створений згладжуванням $n$ -мірного вектора $vy$ за допомогою ковзної медіани. $m$ — ширина вікна, за яким відбувається згладжування, причому $m$ повинно бути непарним числом і $m < n$ .
$supsmooth(vx, vy)$	Повертає $n$ -мірний вектор, створений локальним використанням симетричної лінійної процедури згладжування МНК.

## Порядок виконання лабораторної роботи 4

**Завдання 1.** Створіть таблицю експериментальних даних:  $x_i = a + h_i$ ,  $i = 0, 1, \dots, 10$ ,  $h = (b - a)/10$  на відрізьку  $[a, b]$ .

### Варіанти завдання 1

№ варіанта	$y_i$	$[a, b]$
1	2.86; 2.21; 2.96; 3.27; 3.58; 3.76; 3.93; 3.67; 3.90; 3.64; 4.09	[0, 1]
2	1.14; 1.02; 1.64; 1.64; 1.96; 2.17; 2.64; 3.25; 3.47; 3.89; 3.36;	[-1, 1]
3	4.70; 4.64; 4.57; 4.45; 4.40; 4.34; 4.27; 4.37; 4.42; 4.50; 4.62	[2, 4]
4	0.43; 0.99; 2.07; 2.54; 1.67; 1.29; 1.24; 0.66; 0.43; 0.35; 0.70	[2, 4]
5	1.55; 1.97; 1.29; 0.94; 0.88; 0.09; 0.02; 0.84; 0.81; 0.09; 0.15	[1, 4]
6	3.24; 1.72; 1.95; 2.77; 2.47; 0.97; 1.75; 1.55; 0.12; 0.70; 1.19	[0, 4]
7	2.56; 1.92; 2.85; 2.94; 2.39; 2.16; 2.51; 2.10; 1.77; 2.28; 1.70	[-1, 2]
8	1.77; 0.92; 2.21; 1.50; 3.21; 3.46; 3.70; 4.02; 4.36; 4.82; 4.03	[-1, 3]
9	1.53; 0.45; 1.68; 0.12; 0.68; 2.36; 2.58; 2.53; 3.45; 2.70; 2.82	[4, 8]
10	2.50; 3.90; 3.54; 4.63; 3.87; 5.25; 4.83; 3.24; 3.08; 3.00; 4.70	[0, 5]
11	2.95; 3.38; 2.71; 2.37; 2.29; 2.75; 2.76; 2.74; 2.57; 2.40; 2.99	[1, 5]

<b>12</b>	-0.23; -0.03; -0.98; -0.97; -0.43; -0.91; -0.27; -0.19; 0.88; 1.06; 0.72	[2, 4]
<b>13</b>	2.36; 0.03; -0.38; -1.33; 0.25; -1.36; 0.95; 3.16; 4.03; 4.92; 4.20	[0, 2]
<b>14</b>	3.82; 4.07; 3.53; 4.83; 5.53; 5.04; 5.09; 5.87; 5.53; 4.72; 4.73	[3, 4]
<b>15</b>	2.35; 2.16; 2.39; 2.39; 2.18; 2.09; 2.44; 2.56; 3.35; 3.22; 2.65	[-3, 4]

**Завдання 2.** Апроксимувати многочленами 2-го і 6-го ступеня за методом найменших квадратів функцію, задану таблицею значень  $x_i$  та  $y_i$  і порівняти якість наближень. Побудувати графіки многочленів і відзначити вузлові точки  $(x_i, y_i)$ .

**Завдання 3.** Для наведених у таблиці експериментальних даних  $(x_i, y_i)$  визначити параметри лінійної регресії з використанням вбудованих функцій *Mathcad slope* і *intercept*. Відобразити графічно сукупність точок векторів  $x_i$  та  $y_i$  результати проведеної лінійної регресії.



MathCAD  
PLUS 6.0

**Завдання 4.** Апроксимувати дані з векторів  $x_i$  та  $y_i$  поліномом 4-го ступеня за допомогою функцій *regress* і *interp*; наборами поліномів другого порядку за допомогою функцій *loess* і *interp*, (при *span* рівному 0,5 і 2,5).

Відобразити графічно результати апроксимації.

**Завдання 5.** Апроксимувати експериментальні дані з таблиць значень  $x_i$  та  $y_i$  лінійною комбінацією функцій:

$$f(x) = a_1 f_1(x) + a_2 f_2(x) + a_3 f_3(x).$$

Коефіцієнти вектора  $a$  знайти за допомогою функції *linfit*. Відобразити графічно сукупність точок векторів  $x_i$  та  $y_i$  і результати проведеної лінійної регресії загального вигляду.

## Варіанти завдання 5

№ варіанта	$f_1(x)$	$f_2(x)$	$f_3(x)$
1	$e^x$	$1/\sqrt{1+2\cos^2 x}$	$\sin x$
2	$1/(1+x^2)$	$e^x$	$\sin(3x)$
3	$1/(1+x^2)$	$e^{\sin x}$	$x$
4	$\operatorname{arctg} x$	$\ln \ln x$	$\sin x$
5	$e^{-x^2/2}$	$1/x$	$e^{-x}$
6	$(1+x)/(2+x)$	$\cos(x/10)$	$\cos x$
7	$1/(1+e^{x^2})$	$\sqrt{1+x^2}$	$\cos x$
8	$\cos(x/2)$	$2 - \cos x$	$\sin(x/2)$
9	$1/(1+e^x)$	$\operatorname{arctg} \sqrt{x}$	$\sin(3x)$
10	$\ln(x+5)$	$\sqrt{1+x}$	$\sin x$
11	$1/x$	$\sqrt{1+x}$	$1/x^2$
12	$\cos x$	$1/(1+x+x^2)$	$1/(1+x)$
13	$e^x$	$\cos 4x$	$-e^{x/2}$
14	$\sqrt{1+e^{-x}}$	$e^x/3$	$\sin^2(3x)$
15	$1/(1+x+x^2)$	$\cos(x/10)$	$\cos(x/10)$

**Завдання 6.** Апроксимувати експериментальні дані з таблиць значень  $x_i$  та  $y_i$  функцією виду:

$$f(x) = e^{u_0 + u_1 x + u_2 x^2}$$

Параметри вектора  $u$  знайти за допомогою функції *genfit*. Відобразити графічно сукупність точок векторів  $x_i$  та  $y_i$  і результати проведеної *нелінійної регресії загального вигляду*.



MathCAD  
PLUS 6.0

**Завдання 7.** Виконати згладжування експериментальної функції, заданої таблицею значень  $x_i$  та  $y_i$  за допомогою вбудованих функцій MathCAD: *medsmooth*, *ksmooth* і *supsmooth*. Результати згладжування відобразити графічно.

## Лабораторна робота 5

### Чисельне інтегрування і диференціювання

#### Чисельне інтегрування~Чисельне диференціювання~Символьне інтегрування і диференціювання~Порядок виконання лабораторної роботи 5

#### Чисельне інтегрування

Формули для наближеного обчислення визначених інтегралів застосовуються дуже часто. Справа в тому, що для великого числа елементарних функцій первісні вже не виражаються через елементарні функції, в результаті чого не можна обчислити визначений інтеграл за допомогою формули Ньютона-Лейбніца.

Зустрічаються також і випадки, коли доводиться вдаватися до формул наближеного інтегрування навіть для таких інтегралів, які можуть бути знайдені в кінцевому вигляді, але такий вислів виявляється занадто складним. Особливо важливі формули наближеного інтегрування при вирішенні завдань, що містять функції, задані таблично.

#### *Квадратурні формули*

Найбільш поширеним підходом до чисельного обчислення інтеграла

$$I = \int_a^b f(x) dx \quad (1)$$

є розбиття від  $\frac{b-a}{n}$   $[a, b]$  на  $n$  рівних частин  $a = x_0 < x_1 < \dots < x_n = b$  з кроком  $h = \frac{b-a}{n}$ , інтерполювання функції  $y = f(x)$  на відрізку  $[a, b]$  (отримання інтерполяційного многочлена  $\varphi(x)$ ) і заміна в (1) інтеграла інтегральною сумою:

$$I_n = \int_a^b \varphi(x) dx = \sum_{i=0}^n A_i \cdot y_i, \quad I_n \approx I. \quad (2)$$

Співвідношення виду (2) називають квадратурними формулами.

У найпростіших випадках в якості інтерполяційного многочлена  $\varphi(x)$  беруть ступінчасту, кусочно-лінійну або кусочно-параболічну функції, а також поліном ступеня  $k = n$  ( $\varphi(x) = x^k$ ) для яких квадратурні формули приймають вид (див. Приклад 1 рис. 8):

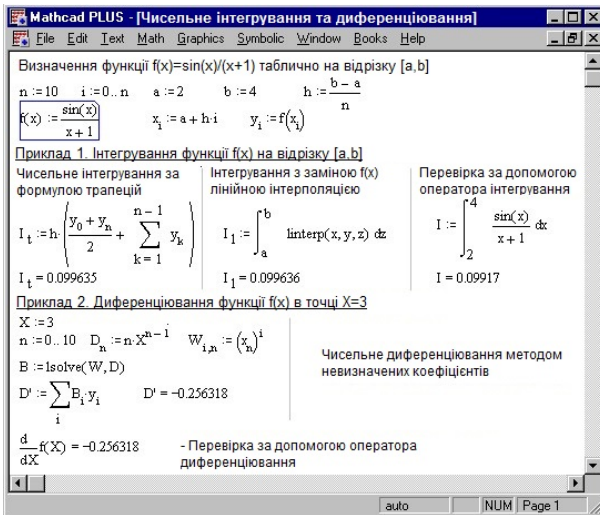


Рисунок 8. Чисельне інтегрування та диференціювання

формула прямокутників:

$$I_n = h \sum_{i=1}^n f(x_{i-1/2}), \quad x_{i-1/2} = x_{i-1} + \frac{1}{2}h, \quad i = 1, 2, \dots, N; \quad (3)$$

формула трапецій:

$$I_n = h \cdot \left( \frac{y_0 + y_n}{2} + \sum_{i=1}^{n-1} y_i \right); \quad (4)$$

формула Сімпсона ( $n$  — парне число):

$$I_n = \frac{h}{3} (y_0 + 4(y_1 + y_3 + \dots + y_{n-1}) + 2(y_2 + y_4 + \dots + y_{n-2}) + y_n); \quad (5)$$

метод невизначених коефіцієнтів полягає в обчисленні визначеного інтеграла (1) за допомогою формули (2) коефіцієнти  $A_i$ , якої знаходяться в результаті вирішення наступної системи рівнянь:

$$\begin{cases} I_0 = A_0 + A_1 + \dots + A_n \\ I_1 = A_0 x_0 + A_1 x_1 + \dots + A_n x_n, \\ \dots \\ I_n = A_0 x_0^n + A_1 x_1^n + \dots + A_n x_n^n, \end{cases} \quad (6)$$

де  $I_k = \int_a^b x^k dx = \frac{b^{k+1} - a^{k+1}}{k+1}, k = 0, 1, \dots, N.$

### Метод Монте-Карло

У багатьох задачах вихідні дані носять випадковий характер, тому для їх вирішення повинен застосовуватися статистико-імовірнісний підхід. На основі такого підходу і побудований метод статистичних випробувань, названий також методом Монте-Карло.

Нехай  $\eta$  - рівномірно розподілена на відрізьку  $[a, b]$  випадкова величина,:

$$\int_a^b f(x) dx = \frac{b-a}{n} \sum_{i=0}^n f(\eta_i) \quad (7)$$

Для генерування послідовності випадкових чисел з нормальним законом розподілу в Mathcad можливо використовувати функцію *rnd*

rnd (x)	Повертає рівномірно розподілене випадкове число між 0 і x.
---------	--

Для реалізації методу Монте-Карло зручно використовувати функцію *mean*

mean (A)	Повертає середнє арифметичне значення елементів масива A.
----------	---

### Чисельне диференціювання

Чисельне диференціювання аналітично або таблично заданої функції  $f(x)$  на відрізку  $[a, b]$  в точці  $x = X$  полягає в заміні  $f(x)$  інтерполяційним поліномом  $\varphi(X)$ , і?

$\frac{d^m \varphi(x)}{dx^m} \approx \frac{d^m f(x)}{dx^m}$  якого можна знайти аналітично за допомогою відповідних формул:

$$\frac{d^m f(x)}{dx^m} \approx \frac{d^m \varphi(x)}{dx^m} = \sum_{i=0}^n B_i \cdot y_i \quad (8)$$

Метод невизначених коефіцієнтів (Див. Приклад 2 рис. 8) припускає використання в якості інтерполяційного многочлена  $\varphi(x)$  поліном ступеня  $k = n$  ( $\varphi(x) = (X - x_i)^k$ ), а коефіцієнти  $B_i$  формули (8) знаходяться в результаті рішення наступної системи рівнянь:

$$\begin{cases} D_0 = B_0 + B_1 + \dots + B_n, \\ D_1 = B_0 x_0 + B_1 x_1 + \dots + B_n x_n, \\ \dots \\ D_n = B_0 x_0^n + B_1 x_1^n + \dots + B_n x_n^n. \end{cases} \quad (9)$$

де  $D_k = (X^k)^{\prime} = k \cdot X^{k-1}$ ,  $K = 0, 1, \dots, N$ .

## Символьне інтегрування і диференціювання

Для обчислення інтегралів (або знаходження первісних) аналітично заданої функції використовується команда **Symbolic**⇒**Integrate on Variable** (Інтегрувати по змінній). Вона повертає символьне значення невизначеного інтеграла за вказаною маркером введення змінної. Вираз, до складу якого входить змінна, є підінтегральною функцією.

Команда **Symbolic**⇒**Differentiate on Variable** (Диференціювати по змінній) повертає символьне значення похідної виразу по тій змінній, яка вказана курсором. Для обчислення похідних вищого порядку потрібно повторити обчислення необхідне число разів.

Результат символьного перетворення іноді містить *спеціальні функції*, які не є частиною списку *вбудованих функцій* Mathcad. Ось визначення деяких з них:

$\gamma$ - константа Ейлера,

$$\text{Ci}(x) = \gamma + \text{Ln}(x) + \int_0^x \frac{\cos(t) - 1}{t} dt, \quad \text{Si}(x) = \int_0^x \frac{\sin(t)}{t} dt,$$
$$\text{Chi}(x) = \gamma + \text{Ln}(x) + \int_0^x \frac{\cosh(t) - 1}{t} dt, \quad \text{Shi}(x) = \int_0^x \frac{\sinh(t)}{t} dt.$$

## Порядок виконання лабораторної роботи 5

**Завдання 1.** Визначити функцію  $f(x)$  таблично, обчисливши значення  $y_i = f(x_i)$  в точках  $x_i = a + hi$ ,  $i = 0, 1, \dots, 8$ ,  $h = (b - a) / 8$  на відрізьку  $[a, b]$ .

## Варіанти завдання 1

№ варіанта	$f(x)$	[a, b]	[c, d]
1	$1/(\lg 2x + 1)$	[0.4, 0.8]	[2, 2.1]
2	$\cos 3x / (1 - \cos 3x)^2$	[0.8, 1.6]	[-1, -0.9]
3	$1 / (x \sqrt{x^3 + 4})$	[0.18, 0.98]	[0.5, 0.6]
4	$\sin x / (1 + \sin x)$	[0.8, 1.6]	[2, 2.1]
5	$x^2 \lg(x + 2)$	[0, 0.4]	[1.5, 1.6]
6	$x^2 \operatorname{arctg}(x/3)$	[0.8, 1.6]	[1, 1.1]
7	$e^{2x} \sin 3x$	[0.4, 1.2]	[2, 2.1]
8	$\operatorname{ctg} 2x / (\sin 2x)^2$	[0.8, 1.2]	[1, 1.1]
9	$(X + 1) \sin x$	[1, 5]	[1, 1.1]
10	$5x + x \lg x$	[0.2, 1]	[1.3, 1.4]
11	$(2x + 3) \sin x$	[0.4, 1.2]	[0.5, 0.6]
12	$\cos x / (2x + 5)$	[0.4, 1.2]	[1, 1.1]
13	$1 / (1 + x + x^2)$	[0, 4]	[2, 2.1]
14	$(1 + x) / (2 + x)$	[0.4, 0.8]	[1.5, 1.6]
15	$\sqrt{1 + e^{-x}}$	[0.4, 1.2]	[0.5, 0.6]

**Завдання 2.** Обчислити інтеграл  $\int_a^b f(x) dx$ :

- за допомогою вбудованого оператора інтегрування;
- за формулою прямокутників;
- за формулою Сімсона;
- за допомогою вбудованого оператора інтегрування і інтерполяцією табличної функції кубічним сплайном (функції *cspline* і *interp*);
- методом невизначених коефіцієнтів для чисельного інтегрування.

**Завдання 3.** Обчислити інтеграл  $\int_a^b f(x)dx$  методом Монте-Карло. Для цього необхідно:

- визначити діапазон випадкових чисел, наприклад  $j: = 0..1000$ ;
- визначити за допомогою функції *rnd* рівномірно розподілену випадкову величину  $\eta_j$  на відрізку інтегрування  $[a, b]$ ;
- створити вектор  $F_j = f(\eta_j)$ ;
- за допомогою функції *mean* обчислити інтеграл.

**Завдання 4.** Знайти первісну аналітично заданої функції  $f(x)$ , використовуючи команду **Symbolic⇒Integrate on Variable**.

**Завдання 5.** Обчислити значення першої і другої похідних функції  $f(x)$  в точці  $X = c$ :

- за допомогою операторів диференціювання *Mathcad*;
- *методом невизначених коефіцієнтів* для чисельного диференціювання. Визначити функцію  $f(x)$  таблично, обчисливши значення  $y_i = f(x_i)$  в точках  $x_i = c + h \cdot i$ ,  $i = 0, 1, \dots, 10$ ,  $h = 0.01$  на відрізку  $[c, d]$ .

**Завдання 6.** Визначити символічне значення першої і другої похідних  $f'(x)$ , використовуючи команду **Symbolic⇒Differentiate on Variable**.

## Лабораторна робота 6

### Вирішення звичайних диференціальних рівнянь

#### Задача Коші~Символьний розв'язок лінійних диференціальних рівнянь~Порядок виконання лабораторної роботи 6

Рішення звичайних диференціальних рівнянь (ЗДР) широко застосовується в практиці науково-технічних розрахунків. Хоча лінійні ЗДР можуть мати рішення у вигляді спеціальних функцій, багато фізичних систем нелінійні і описуються нелінійними ЗДР, що не мають аналітичного рішення. У цьому випадку доводиться використовувати чисельні методи рішення ЗДР.

Щоб вирішити ЗДР, необхідно знати значення залежної змінної і (або) похідних при деяких значеннях незалежної змінної. Якщо ці додаткові умови задаються при одному значенні незалежної змінної, то така задача називається задачею Коші. Якщо ж умови задаються при двох або більше значеннях незалежної змінної, то задача називається *крайовою*.

#### Задача Коші

*Задачу Коші* можна сформулювати наступним чином: нехай дано ЗДР:

$$\frac{dy}{dx} = f(x, y) \quad y(x_0) = y_0 \quad (1)$$

і початкова умова  $y(x_0) = y_0$ . Потрібно знайти функцію  $y(x)$ , що задовольняє як вказане рівняння, так і початкову умову.

Чисельне рішення задачі Коші полягає в побудові таблиці наближених значень  $y_1, y_2, \dots, y_n$  вирішення рівняння  $y(x)$  в точках  $x_1, x_2, \dots, x_n$ . Найчастіше  $x_i = x_0 + ih, i = 0, 1, \dots, n$ , де  $h$  —

крок збільшення змінної  $x$ ,  $n$  — число інтервалів рішення з кроком  $h$ .

Розглянемо тут дві групи чисельних методів розв'язання задачі Коші: однокрокові і багатокрокові.

### **Однокрокові методи**

*Однокрокові методи* — це методи, в яких для знаходження наступної точки на кривій  $y = f(x)$  потрібна інформація лише про один попередній крок. Найпростішим з однокрокових методів є метод Ейлера:

$$y_{i+1} = y_i + f(x_i, y_i)h, \quad i = 0, 1, \dots, n - 1. \quad (2)$$

*Метод Ейлера* має невисоку точність (порядку  $h$ ).

Для досягнення більш високої точності (порядку  $h^4$ ) використовують метод *Рунге-Кутта* четвертого порядку:

$$y_{i+1} = y_i + \frac{k_0 + 2 \cdot k_1 + 2 \cdot k_2 + k_3}{6}, \quad \text{де}$$

$$\begin{aligned} k_0 &= h \cdot f(x_i, y_i), & k_2 &= h \cdot f(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}), \\ k_1 &= h \cdot f(x_i + \frac{h}{2}, y_i + \frac{k_0}{2}), & k_3 &= h \cdot f(x_i + h, y_i + k_2). \end{aligned} \quad (3)$$

### **Багатокрокові методи**

У багатокрокових методах для відшукування наступної точки кривої  $y = f(x)$  потрібна інформація більш ніж про одну з попередніх точок.

Нехай знайдені значення  $y_{i-3}$ ,  $y_{i-2}$ ,  $y_{i-1}$ ,  $y_i$  в чотирьох послідовних точках. При цьому є також обчислені раніше значення правої частини рівняння (1)  $f_{i-3}$ ,  $f_{i-2}$ ,  $f_{i-1}$ ,  $f_i$ . Тоді схему *методу Адамса* можна представити у вигляді:

$$y_{i+1} = y_i + h \cdot f_i + \frac{h}{2} \cdot \Delta f_i + \frac{5 \cdot h}{12} \cdot \Delta^2 f_i + \frac{3 \cdot h}{8} \cdot \Delta^3 f_i, \quad i = 3, 4, \dots, n - 1. \quad (4)$$

де кінцеві різниці в точці  $x_i$  мають вигляд:

$$\Delta f_i = f_i - f_{i-1}, \quad (5)$$

$$\Delta^2 f_i = f_i - 2f_{i-1} + f_{i-2},$$

$$\Delta^3 f_i = f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}.$$

### Рішення задачі Коші засобами Mathcad

Інструментарій для вирішення ЗДР (систем ЗДР) різного порядку в Mathcad представлений широким спектром вбудованих функцій, робота однієї з яких (*rkfixed* — метод Рунге-Кутта (*rk*) четвертого порядку з фіксованим (*fixed*) кроком інтегрування) показана на рисунку 9.

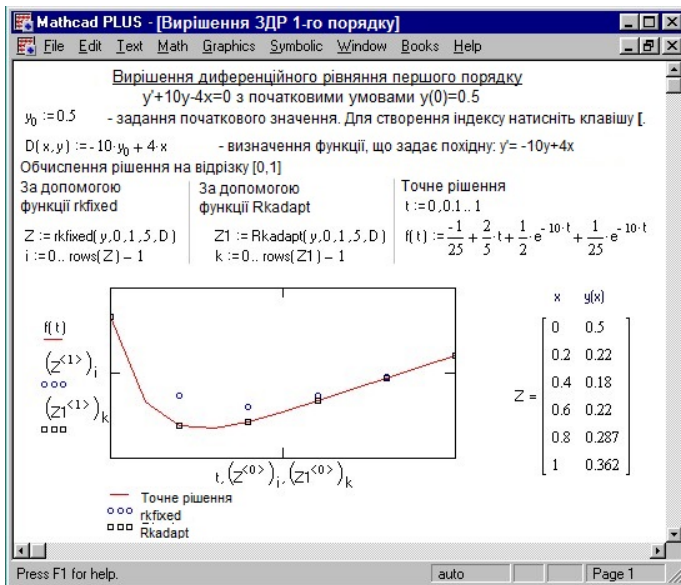


Рисунок 9. Рішення ЗДР 1-го порядку

$rkfixed(y, a, b, n, D)$	Повертає матрицю з $p + 1$ стовпцями і $n + 1$ рядками ( $p$ – кількість рівнянь або порядок рівняння, $n$ – число кроків на інтервалі $[a, b]$ ) – таблицю рішень системи: перший стовпчик –
--------------------------	---

	це значення аргументу $x$ , а наступні стовпці – значення ординат рішення. $y$ – вектор початкових умов розмірності $n$ . $D(x, y)$ – функція-вектор з $n$ елементів, що містить перші похідні невідомих функцій.
--	---

Можна вирішити задачу більш точно (швидше), якщо зменшити крок  $h$ там, де похідна змінюється швидко, і збільшити крок там, де вона веде себе більш спокійно. Для цього передбачена функція *Rkadapt* (*adaption* — адаптація). Аргументи і матриця, яка повертається функцією *Rkadapt*, такі ж, як при *rkfixed* (див. Малюнок 9). Рішення системи ЗДР показано на рисунку 11 (Приклад 2).

### ***Крайові задачі***

*Крайова задача* формулюється так: нехай на відрізку  $[a, b]$  потрібно знайти вирішення диференціального рівняння (для простоти виклад будемо вести на прикладі ЗДР другого порядку):

$$\frac{d^2 y}{dx^2} = f(x, y, y') \tag{6}$$

при граничних умовах  $y(a) = A, y(b) = B$ .

В цьому випадку Mathcad пропонує використовувати функцію *sbval*, щоб знайти відсутні початкові умови в точці  $a$ .

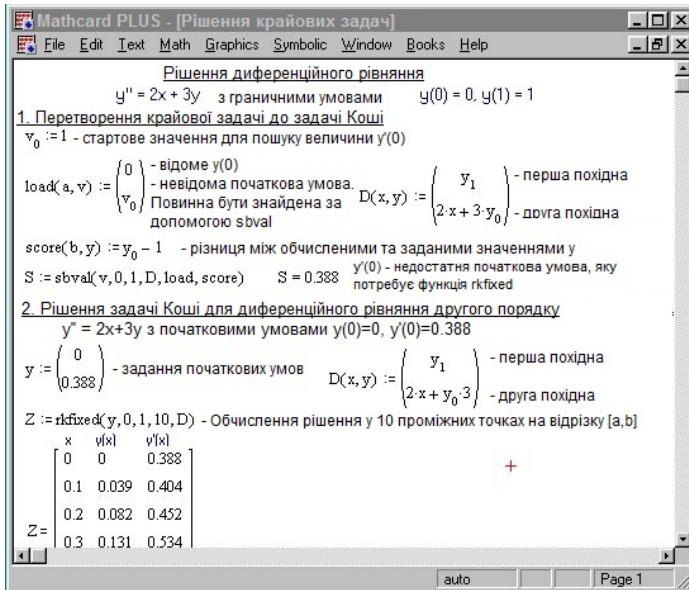


Рисунок 10. Рішення крайової задачі

<p><math>Sbval(v, a, b, D, load, score)</math></p>	<p>Повертає вектор, що містить відсутні початкові умови в точці <math>a</math>. Вектор <math>v</math> задає початкові наближення, <math>a, b</math> - граничні точки інтервалу рішень, <math>D(x, y)</math> - функція-вектор з першими похідними невідомих функцій. <math>load(a, v)</math> - функція-вектор, що повертає значення початкових умов в точці <math>a</math>. <math>score(b, y)</math> - функція-вектор, кожен елемент якого містить різницю між початковою умовою заданою в точці <math>b</math>, із значенням шуканого рішення в цій точці.</p>
--	--

Після того, як ці відсутні початкові умови будуть отримані, можна вирішувати звичайну задачу з початковими умовами — задачу Коші, використовуючи будь-яку з функцій, описаних

вище (Малюнок 9). Приклад рішення крайової задачі показаний на рисунку 10.

### **Символьний розв'язок лінійних диференціальних рівнянь**

Для отримання аналітичного рішення лінійних ЗДР в Mathcad необхідно виконати наступні дії (Приклад 1 рисунку 11):



Якщо ви працюєте з пакетом Mathcad 5.0, не забудьте попередньо виконати команду **Symbolic?Load Symbolic Processor** для завантаження символічного процесора..  
MathCAD PLUS 5.0 Пропустіть цей пункт, якщо ви працюєте з пакетом Mathcad 6.0.

- Надрукувати вихідне рівняння, використовуючи оператори диференціювання і комбінацію клавіш **[Ctrl] =** для друку символу  $=$ .
- Відзначивши незалежну змінну, виконати пряме перетворення Лапласа **Symbolic⇒Transforms⇒Laplace Transform (Перетворення Лапласа)**. Результат для ЗДР вище 1-го порядку буде розміщений в буфер обміну. Викличте його натиснувши клавішу **F4**.
- За результатами перетворення Лапласа "вручну" скласти алгебраїчне рівняння, прийнявши позначення  $L = \text{laplace}(y(t), t, s)$ ,  $C1 = y(0)$  і  $C2 = \text{diff}(y(0), 0)$ .
- Вирішити складене алгебраїчне рівняння щодо змінної  $L$ , використовуючи команду **Symbolic⇒Solve for Variable** (Вирішити щодо змінної).

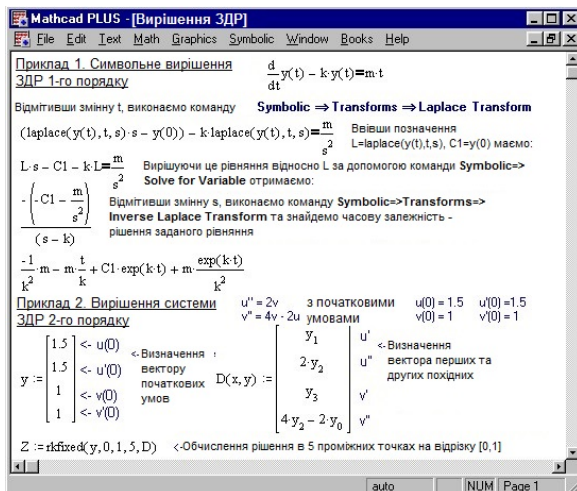


Рисунок 11. Деякі можливості вирішення ЗДР в Mathcad

- Відзначити змінну  $s$  і, зробивши зворотнє перетворення Лапласа **Symbolic** $\Rightarrow$ **Transforms** $\Rightarrow$ **Inverse Laplace Transform** (**Зворотнє перетворення Лапласа**), отримати рішення заданого ЗДР у вигляді часової залежності.

## Порядок виконання лабораторної роботи 6

**Завдання 1.** Вирішити задачу Коші:  $\frac{dy}{dx} = f(x, y)$ ,  $y(0) = 1$  з кроком  $h = 0.1$  на відрізьку  $[0, 1]$ :

- методом *Ейлера*;
- методом *Рунге-Кутта* (коефіцієнти  $k_i$  задати як функції від  $x$  і  $y$ );
- методом *Адамса*;
- використовуючи функцію *rkfixed*.

## Варіанти завдання 1

№ варіанта	$f(x, y)$	№ варіанта	$f(x, y)$	№ варіанта	$f(x, y)$
1	$x^+ y$	6	$2 y - \cos 2 x$	11	$2 y + 3 e^{-x}$
2	$2x^2 + 2y$	7	$y - e^{x/2} + 2$	12	$y/2 - e^{-x}$
3	$e^x - 3 y$	8	$3 y - 2 \sin x$	13	$y + (\cos x) / 3$
4	$y - \sin x$	9	$e^{2x} - y$	14	$y - 4 x + 5$
5	$y/3 - x^2$	10	$2 \sin x + y$	15	$2 x - y/3 - e^x$

**Завдання 2.** Побудувати графіки рішень, отриманих методами *Ейлера*, *Рунге-Кутта*, *Адамса* і за допомогою функції *rkfixed*.

Обчислити в точці  $x = 1$  відносну похибку для кожного методу.

**Завдання 3.** Знайти аналітичне (точне) рішення ЗДР із завдання 1 за допомогою перетворень Лапласа (команди **Symbolic**⇒**Transforms**⇒**Laplace Transform** **Inverse Laplace Transform**).

**Завдання 4.** Вирішити задачу Коші для системи ЗДР при заданих початкових умовах на відрізку  $[0, 2]$  з кроком  $h = 0.2$ . Вирішувати за допомогою функції *rkfixed*. Побудувати графіки функцій  $u(t)$  і  $v(t)$ .

## Варіанти завдання 4

№ варіанта	Система ЗДР	Початкові умови				№ варіанта	Система ЗДР	Початкові умови			
		$u(0)$	$u'(0)$	$v(0)$	$v'(0)$			$u(0)$	$u'(0)$	$v(0)$	$v'(0)$
1	$\begin{cases} u'' = 2v + u \\ v'' = 4v - 2u \end{cases}$	1.5	1.5	1	1	9	$\begin{cases} u'' = 1/2 + v \\ v'' = 4 - u + f \end{cases}$	2	0	-1	1
2	$\begin{cases} u'' = -v + 3u \\ v'' = v - 2u \end{cases}$	-1	1	-1.5	3	10	$\begin{cases} u'' = -v + f \\ v'' = v + 3u \end{cases}$	-1	2	-1.5	0

3	$\begin{cases} u'' = 2v - u \\ v'' = 4v + u \end{cases}$	1.5	1.5	1	1	11	$\begin{cases} u'' = v - u - f \\ v'' = 2v + u \end{cases}$	1.5	1.5	-1	-1
4	$\begin{cases} u'' = 5v \\ v'' = v + 2u + f \end{cases}$	1	1.5	0	2	12	$\begin{cases} u'' = 5v + f \\ v'' = 3v + u \end{cases}$	-1	1.5	0	-2
5	$\begin{cases} u'' = v + u + f \\ v'' = v + 2u - f \end{cases}$	0.5	1.5	-1	2	13	$\begin{cases} u'' = v + u \\ v'' = v + u - f \end{cases}$	-0.5	1	-1	2
6	$\begin{cases} u'' = 2v + u + f \\ v'' = 4v \end{cases}$	0.5	2	1	2	14	$\begin{cases} u'' = 2v - u \\ v'' = 4v + f \end{cases}$	0	-2	0	2
7	$\begin{cases} u'' = -v + f \\ v'' = 5v - 7u \end{cases}$	5	5	-1	1	15	$\begin{cases} u'' = v - 2f \\ v'' = v + 3u \end{cases}$	3	3	-1	1
8	$\begin{cases} u'' = v - 5u \\ v'' = 2v + u + f \end{cases}$	1.5	1	3	1						

**Завдання 5.** На відрізку  $[a, b]$  з використанням функцій *load*, *score* і *sbal* перетворити крайову задачу:

$$\frac{d^2y}{dx^2} = f(x, y, y') \text{ при граничних умовах } y(a) = A, y(b) = B$$

до задачі Коші і знайти рішення заданого ЗДР в 10 проміжних точках за допомогою функції *rkfixed*.

**Варіанти завдання 5**

№ варіанта	$f(x, y, y')$	Граничні умови			
		$a$	$b$	$y(a)$	$y(b)$
1	$e^x y + \cos x$	1	2	0	0
2	$y \sin x + e^{-x}$	2	3	1	0
3	$y \cos x + \operatorname{tg} x$	0	1	0	0.45
4	$x^3 y + \cos x$	0	1	1	0
5	$x + e^x y / (1 - x)$	2	4	1	0.14
6	$x^2 y + 1 / (1 + x)$	1	3	0	0.17
7	$y \cos x + \cos^2 x$	1	2	0	0
8	$(2 + x) y + \operatorname{arctg} x$	0	3	0	0.22
9	$(5 - x) y + x$	2	4	0	-1.2
10	$e^{-x} y + 2 e^{-x}$	0	1.5	2.4	0

<b>11</b>	$e^{-x}y / x + x$	-3	-2	3	0
<b>12</b>	$(x^2 + 1 / x) y + 1 / x^2$	2	3	0	0
<b>13</b>	$(10 - x) y + x$	-1	0	2	0
<b>14</b>	$y / x^2 + x$	1	3	1.5	0
<b>15</b>	$y \ln x + 1 + x$	7	8	0	0

## Лабораторна робота 7

### Вирішення диференціальних рівнянь в частинних похідних

#### Метод кінцевих різниць~Гіперболічні рівняння в частинних похідних~Параболічні рівняння в частинних похідних~Еліптичні рівняння в частинних похідних~Порядок виконання лабораторної роботи 7

На практиці часто доводиться стикатися з завданнями, в яких шукана величина залежить від декількох змінних. В цьому випадку рівняння, що вирішуються, містять частинні похідні і називаються диференціальними рівняннями в частинних похідних. На жаль, дуже багато з таких рівнянь не мають аналітичного рішення, і щоб вирішити їх, доводиться вдаватися до чисельних методів. Для вирішення диференціальних рівнянь в частинних похідних чисельно використовується метод *кінцевих різниць*.

#### Метод кінцевих різниць

Чисельне рішення диференціальних рівнянь в частинних похідних методом кінцевих різниць полягає в наступному:

1. Побудова в області рішення рівномірної сітки, що містить  $n$  вузлових точок (рис. 12).

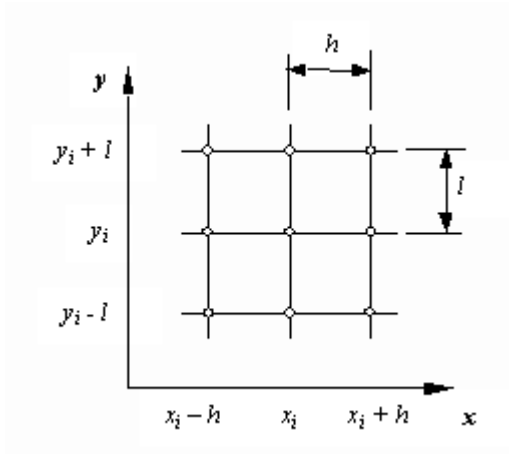


Рисунок 12. Двовимірна сітка

2. Подання похідних в кінцево-різницевій формі:

$$\frac{\partial f}{\partial x} \approx \frac{f_{i+1,j} - f_{i-1,j}}{2h}, \quad \frac{\partial f}{\partial y} \approx \frac{f_{i,j+1} - f_{i,j-1}}{2l}, \quad (1)$$

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{h^2}, \quad \frac{\partial^2 f}{\partial y^2} \approx \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{l^2} \quad \text{і т.д.,}$$

де  $f_{i,j}$ ,  $f_{i+1,j}$ ,  $f_{i-1,j}$ ,  $f_{i,j+1}$ ,  $f_{i,j-1}$  — значення функції  $f(x, y)$  в точках  $(x_i, y_j)$ ,  $(x_i + h, y_j)$ ,  $(x_i - h, y_j)$ ,  $(x_i, y_j + l)$ ,  $(x_i, y_j - l)$  відповідно.

Такі різницеві рівняння записують для всіх вузлів сітки і отримують в результаті систему з  $n$  рівнянь з  $n$  невідомими.

3. Вирішення отриманої системи з метою отримання наближеного рішення у вузлах сітки.

## Гіперболічні рівняння в частинних похідних

Найпростішим видом рівняння гіперболічного типу є *хвильове рівняння*. До дослідження хвильового рівняння приводить розгляд процесів поперечних коливань струни, поздовжніх коливань стержня, електричних коливань в проводі, крутильних коливань вала і т. п.

Розглянемо одномірне рівняння коливань струни. В області  $\{0 \leq x \leq l, 0 \leq t \leq T\}$  потрібно знайти рішення рівняння:

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad (2)$$

Шукана функція  $u(x, t)$  повинна задовольняти початковим умовам, які описують початкову ( $t = 0$ ) форму струни  $\varphi(x)$  і швидкість її точок  $\psi(x)$ :

$$u(x, 0) = \varphi(x), \quad \frac{\partial u}{\partial t}(x, 0) = \psi(x), \quad 0 \leq x \leq l \quad (3)$$

і граничним умовам, що вказують, що відбувається на кінцях струни ( $x = 0$  і  $x = l$ ):

$$u(0, t) = \mu_1(t), \quad u(l, t) = \mu_2(t), \quad 0 \leq t \leq T. \quad (4)$$

Сукупність початкових і граничних умов називається *крайовими умовами*.

Для побудови різницевої схеми вирішення завдання (2) - (4) побудуємо в області  $\{0 \leq x \leq l, 0 \leq t \leq T\}$  сітку  $x_i = ih, i = 0, 1, \dots, n, i = hn, t_j = j\tau, j = 0, 1, \dots, m, T = m\tau$  і апроксимуємо рівняння (2) в кожному внутрішньому вузлі сітки на шаблоні "хрест" (Малюнок 13).

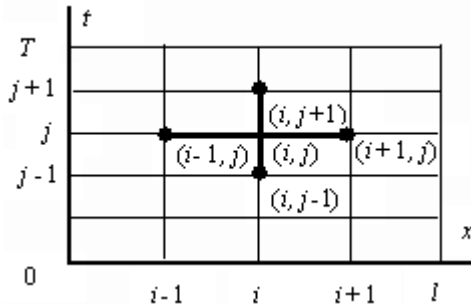


Рисунок 13. Шаблон для хвильового рівняння

Використовуючи для апроксимації приватних похідних вираження (1), отримуємо наступну різницеву апроксимацію рівняння (2):

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\tau^2} = a^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} \quad (5)$$

Вирішуючи рівняння (6) відносно єдиного невідомого значення  $u_{i,j+1}$ , отримуємо наступну схему:

$$u_{i,j+1} = 2(1 - \lambda)u_{i,j} + \lambda(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1}, \quad (6)$$

$$\lambda = a^2\tau^2/h^2, \quad i = 1, \dots, n-1, j = 1, \dots, m-1.$$

Схема (6) називається *тришаровою* тому, що пов'язує між собою значення  $u_{i,j}$  функції  $u(x, t)$  на трьох часових шарах з номерами:  $j-1, j, j+1$ . Схема (6) є *явною*, тобто дозволяє в явному вигляді висловити  $u_{i,j}$  через значення  $u$  з попередніх двох шарів.

Для початку рахунку за схемою (6) необхідні значення  $u_{i,j}$  функції  $u(x, t)$  на нульовому ( $j=0$ ) і першому ( $j=1$ ) часових шарах. Вони визначаються початковими умовами (3) і записуються в вигляді:

$$u_{i,0} = \varphi(x_i), \quad \frac{u_{i,1} - u_{i,0}}{\tau} \approx \psi(x_i) \Rightarrow u_{i,1} = u_{i,0} + \tau\psi(x_i), \quad I = 0, 1, \dots, n. \quad (7)$$

Граничні умови (4) також записуються в сітковому вигляді:

$$u_{0,j} = \mu_1(t_j), u_{n,j} = \mu_2(t_j), J = 0, 1, \dots, m. \quad (8)$$

Таким чином, рішення вихідної диференціальної задачі (2) - (4) зводиться до вирішення різницевої задачі (6) - (8).

Схема стійка, якщо виконано умову Куранта  $a\tau/h < 1$ .

## Параболічні рівняння в частинних похідних

Найпростішим видом рівняння параболічного типу є *рівняння теплопровідності*, або *рівняння Фур'є*. До дослідження рівняння теплопровідності, або рівняння Фур'є, призводить розгляд процесів поширення тепла, фільтрації рідини і газу в пористому середовищі, деякі питання теорії ймовірностей.

Розглянемо задачу про поширення тепла в однорідному стержні довжини  $l$ , на кінцях якого підтримується заданий температурний режим. Завдання полягає в знаходженні функції  $u(x, t)$ , що задовольняє в області  $\{0 \leq x \leq l, 0 \leq t \leq T\}$  рівнянню

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2}, \quad a > 0 \quad (9)$$

початковій умові

$$u(x, 0) = \varphi(x) \quad (10)$$

і граничним умовам

$$u(0, t) = \mu_1(t), u(l, t) = \mu_2(t) \quad (11)$$

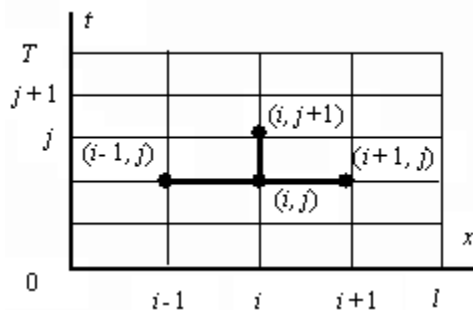


Рисунок 14. Шаблон для рівняння теплопровідності

Побудуємо в області  $\{0 \leq x \leq l, 0 \leq t \leq T\}$  рівномірну прямокутну сітку з кроком  $h$  в напрямку  $x$  і кроком  $\tau$  - в напрямку  $t$  (рис. 14). Тоді  $x_i = ih, i = 0, 1, \dots, n, h = l / n; t_j = j\tau, j = 0, 1, \dots, m, \tau = T / m$ .

Апроксимуємо диференціальну задачу (9) – (11) на чотирихточковому шаблоні, в результаті отримуємо явну двошарову різницеву схему:

$$\begin{aligned}
 u_{i,j+1} &= \lambda u_{i+1,j} + (1 - 2\lambda) u_{i,j} + \lambda u_{i-1,j}, \\
 i &= 1, 2, \dots, n-1, j = 0, 1, \dots, m-1 \\
 u_{i,0} &= \varphi(x_i), i = 0, 1, \dots, n, \\
 u_{0,j} &= \mu_1(t_j), u_{n,j} = \mu_2(t_j), J = 0, 1, \dots, m, \\
 \lambda &= \frac{\alpha\tau}{h^2}.
 \end{aligned}
 \tag{12}$$

Схема стійка при  $\lambda \leq 1/2$ .

### Еліптичні рівняння в частинних похідних

До дослідження такого рівняння приводить розгляд задач про електричні і магнітні поля, про стаціонарний тепловий

стан, задач гідродинаміки, дифузії і т. д. Розглянемо рішення рівняння Пуассона і його однорідної форми - рівняння Лапласа.

Рішення рівняння Пуассона будемо шукати в деякій обмеженій області  $\Omega = \{0 \leq x \leq q_1, 0 \leq y \leq q_2\}$  зміни незалежних змінних  $x, y$ :

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y) \quad (13)$$

Граничні умови:

$$\begin{aligned} u(0, y) &= \mu_1(Y), \quad u(a, y) = \mu_2(Y), \quad y \in [0, b], \\ u(x, 0) &= \mu_3(X), \quad u(x, b) = \mu_4(X), \quad x \in [0, a], \end{aligned} \quad (14)$$

де  $f, \mu_1, \mu_2, \mu_3, \mu_4$  — задані функції (задача, що складається в рішенні еліптичного рівняння при заданих значеннях шуканої функції на кордоні розрахункової області, називається *задачею Дирихле*.).

Побудуємо в області  $\Omega$  рівномірну прямокутну сітку з кроками  $h$  і  $l$  по  $x$  і  $y$  відповідно:  $x_i = ih, i = 0, 1, \dots, n, h = q_1 / n; y_j = jl, j = 0, 1, \dots, m, l = q_2 / m$ .

Апроксимуємо диференціальну задачу (13) – (14) на шаблоні "хрест" (Малюнок 13), в результаті отримуємо *невну тришарову різницеву схему*:

$$a_{i,j} u_{i+1,j} + b_{i,j} u_{i-1,j} + c_{i,j} u_{i,j+1} + d_{i,j} u_{i,j-1} + e_{i,j} u_{i,j} = f_{i,j}, \quad (15)$$

где  $a_{i,j} = b_{i,j} = \frac{1}{h^2}, c_{i,j} = d_{i,j} = \frac{1}{l^2}, e_{i,j} = -2 \cdot \left( \frac{1}{h^2} + \frac{1}{l^2} \right)$

$$\begin{aligned} i &= 1, 2, \dots, n-1, \quad j = 1, 2, \dots, m-1, \\ u_{0,j} &= \mu_1(y_j), \quad u_{n,j} = \mu_2(y_j), \quad j = 0, 1, \dots, m, \\ u_{i,0} &= \mu_3(x_i), \quad u_{i,m} = \mu_4(x_i), \quad i = 0, 1, \dots, n \end{aligned}$$

Для вирішення рівняння Пуассона в Mathcad використовується функція *relax*

$relax(a, b, c, d, e, f, u, rjac)$	Повертає квадратну матрицю рішення рівняння Пуассона. Тут $a, b, c, d, e$ - квадратні матриці однакового розміру, що містять коефіцієнти рівняння (15); $f$ - квадратна матриця, що містить значення правої частини рівняння (15) в кожній точці по області $\Omega$ , в якій шукається рішення; $u$ - квадратна матриця, яка містить граничні значення рішення на кордоні області і початкове наближення для рішення всередині області; $rjac$ - число між 0 і 1, яке управляє сходимістю алгоритму.
------------------------------------	---

при  $f = 0$  отримуємо рівняння Лапласа:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (16)$$

Якщо для рівняння Лапласа в області  $\Omega$  ввести сітку з рівним кроком по осях  $x$  і  $y$ , то різницева схема (16) істотно спрощується

$$\begin{aligned}
 &u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = 0, \\
 &i = 1, 2 \dots n-1, \quad j = 1, 2 \dots m-1, \\
 &u_{0,j} = \mu_1(y_j), \quad u_{n,j} = \mu_2(y_j), \quad j = 0, 1 \dots m, \\
 &u_{i,0} = \mu_3(x_i), \quad u_{i,m} = \mu_4(x_i), \quad i = 0, 1 \dots n.
 \end{aligned} \quad (17)$$

Рішення рівняння Лапласа за допомогою функції *relax* показано на рисунку 15.

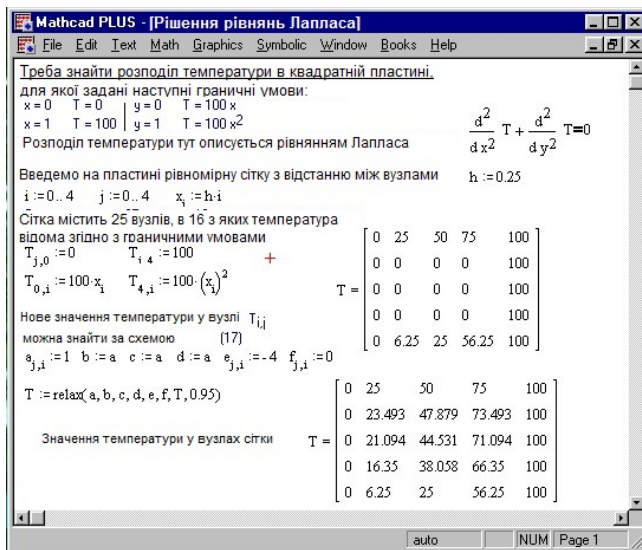


Рисунок 15. Рішення рівняння Лапласа

## Порядок виконання лабораторної роботи 7

**Завдання 1.** Вирішити задачу про коливання струни одиничної довжини із закріпленими кінцями:

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad a = 1$$

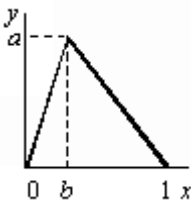
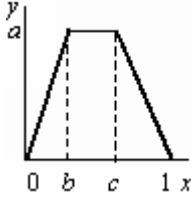
з початковими умовами

$$u(x, 0) = f(x), \quad \frac{\partial u}{\partial t}(x, 0) = 0, \quad 0 \leq x \leq 1$$

і нульовими граничними умовами

$$u(0, t) = u(1, t) = 0.$$

### Варіанти завдання 1

№ варіанта	$f(x)$	$a$	$b$	№ Варіанта	$f(x)$	$a$	$b$	$c$
1		1	0.1	9	$x \sin(2(x-1))$			
2		2	0.1	10	$4x^3(x-1)$			
3		4	0.2	11		1	0.1	0.2
4		6	0.3	12		3	0.2	0.4
5		8	0.4	13		5	0.4	0.6
6	$x(x^2 - 1)$			14		7	0.6	0.8
7	$\sin(\pi x^2)$			15		9	0.8	0.9
8	$\sin(\pi x) \cos x$							

Для вирішення завдання побудувати сітку з 11 вузлів по  $x$  ( $i = 0, 1, \dots, 10$ ) і провести обчислення для 16 шарів по  $t$  ( $j = 0, 1, \dots, 16$ ). Обчислення виконати з кроком  $h$  по  $x$ , рівним 0.1 і кроком  $\tau$  по  $t$ , рівним 0.05. Відобразити графічно рішення задачі на 0-му, 5-му, 10-му і 16-му часових шарах.

**Завдання 2.** Знайти рішення  $u(x, t)$  для рівняння теплопровідності з постійними коефіцієнтами:

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2}, \quad A = 1$$

з початковими умовами

$$u(x, 0) = f(x) \quad 0 \leq x \leq 1$$

і граничними умовами

$$u(0, t) = a, \quad u(1, t) = b.$$

Для вирішення завдання побудувати сітку з 11 вузлів по  $x$  ( $i = 0, 1, \dots, 10$ ) і провести обчислення для 12 шарів по  $t$  ( $j = 0, 1, \dots, 12$ ). Обчислення виконати з кроком  $h$  по  $x$ , рівним 0.1 і кроком  $\tau$  по  $t$ , рівним 0.005. Відобразити графічно рішення задачі на 0-му, 4-му, 8-му і 12-му шарах і побудувати

інтегральну поверхню розподілу температури в стержні за допомогою команди **Graphics**  $\Rightarrow$  **Create Surface Plot**.

### Варіанти завдання 2

№ варіанта	$f(x)$	$a$	$b$	№ варіанта	$f(x)$	$a$	$b$
1	$x(x - 1)$	0	0	9	$(x^2 + 0.5) \cos(2\pi x)$	0.5	1.5
2	$x^3 + x^2 - x$	0	1	10	$\sin(\pi x) \cos x$	0	0
3	$x^2(1 - x)$	0	0	11	$x \sin(2(x - 1))$	0	0
4	$1 - x^4$	1	0	12	$\ln(0.5 + x)(x - 1)$	0.7	0
5	$x \sin(2\pi x)$	0	-0.3	13	$x \sin(4(x - 1)) - x$	0	-1
6	$(x - 1) \sin^2 x$	0	0	14	$x \cos(2\pi x)$	0	1
7	$4x^2(x - 1)$	0	0.5	15	$x e^{-x}(x^4 - 2)$	0	-0.4
8	$10x^3(x - 1)$	0.5	0				

**Завдання 3.** Знайти стаціонарний розподіл температури в квадратній пластині зі стороною 1, що описується рівнянням Лапласа

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

з крайовими умовами вигляду

$$u(0, y) = f_1(y), (0 \leq y \leq 1), u(1, y) = f_2(y), (0 \leq y \leq 1),$$

$$u(x, 0) = f_3(x), (0 \leq x \leq 1), u(x, 1) = f_4(x), (0 \leq x \leq 1).$$

Вирішувати завдання за допомогою функції *relax*.

Для вирішення завдання побудувати сітку з 11 вузлів по  $x$  ( $i = 0, 1, \dots, 10$ ) і з 11 вузлів по  $y$  ( $j = 0, 1, \dots, 10$ ). Відобразити графічно за допомогою команди **Graphics**  $\Rightarrow$  **Create Contour Plot** стаціонарний розподіл температури в пластині.

### Варіанти завдання 3

№ варіанта	$f_1(y)$	$f_2(y)$	$f_3(x)$	$f_4(x)$
1	$y^2$	$\cos y + (2 - \cos 1)y$	$x^3$	$1 + x$
2	$e^y - e y^2$	$y$	$1 - x^3$	$x^2$
3	$1 - y^2$	$y$	$\frac{\sin x + 1 - x^3}{(1 + \sin 1)}$	$x$
4	0	$y$	$\sin x - x^3 \sin 1$	$x$
5	$e^y + y^2 (1 - e) - 1$	$y$	0	$x$
6	$y^2$	$\cos y + (3 - \cos 1)y$	$x^3$	$1 + 2x$
7	0	$y$	$\sin x - x^3 \sin 1$	$x^2$
8	$\frac{2ey - (1 + 2e)y^2}{1}$	$-y$	$1 - x^3$	$x - 2$
9	$-10y^2 - 8y + 6$	$-10y^2 - 30y + 22$	$9x^2 + 7x + 6$	$9x^2 - 15x - 12$
10	$-7y^2 - 5y + 3$	$-7y^2 - 21y + 13$	$6x^2 + 4x + 3$	$6x^2 - 12x - 9$
11	1	$y + 1$	1	$1 + x$
12	1	$e^y$	1	$e^x$
13	$-y^2 - 5y$	$4 + 5y - y^2$	$x^2 + 3x$	$x^2 + 3x + 4$
14	$3 - 7y$	$7 - 6y$	$4x + 3$	$5x - 4$
15	0	$\sin y$	0	$\sin x$

## Лабораторна робота 8

### Спектральний аналіз і синтез

#### Гармонійний аналіз і синтез~Класичний спектральний аналіз~Чисельний спектральний аналіз~Спектральний аналіз на основі швидкого перетворення Фур'є~Фільтрація аналогових сигналів~Порядок виконання лабораторної роботи 8

Одним з фундаментальних положень математики, що знайшов широке застосування в багатьох прикладних задачах (процеси передачі інформації, в теорії електротехніки, в дослідженні руху машин, в теорії корабля та ін.), є можливість опису будь-якої періодичної функції  $f(t)$  з періодом  $T$ , що задовольняє умовами Дирихле (згідно з теоремою Дирихле періодична функція повинна мати кінцеве число розривів і безперервність похідних між ними.), за допомогою тригонометричного ряду Фур'є:

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos k\omega_1 t + b_k \sin k\omega_1 t) \quad (1)$$

Де  $\omega_1 = 2\pi/T$  — частота повторення (або частота першої гармоніки);  $k$  — номер гармоніки. Цей ряд містить нескінченне число косинусних або синусних складових — *гармонік*, причому амплітуди цих складових  $a_k$  і  $b_k$  є коефіцієнтами Фур'є, обумовленими інтегральними виразами:

$$a_k = \frac{2}{T} \int_0^T f(t) \cos k\omega_1 t dt, \quad (2)$$

$$b_k = \frac{2}{T} \int_0^T f(t) \sin k\omega_1 t dt. \quad (3)$$

Крім згаданої форми ряд Фур'є можна представити у вигляді

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} A_k \cos(k\omega_1 t + \varphi_k) \quad (4)$$

де амплітуда  $A_k$  і фаза  $\varphi_k$  гармонік визначаються виразами:

$$A_k = \sqrt{a_k^2 + b_k^2}, \quad (5)$$

$$\varphi_k = -\arctg \frac{b_k}{a_k}. \quad (6)$$

## Гармонійний аналіз і синтез

*Гармонійним аналізом* називають розкладання функції  $f(t)$ , заданої на відрізку  $[0, T]$  в ряд Фур'є або в обчисленні коефіцієнтів Фур'є  $a_k$  і  $b_k$  за формулами (2) і (3).

*Гармонійним синтезом* називають отримання коливань складної форми шляхом підсумовування їх гармонійних складових (гармонік) (рис. 16).

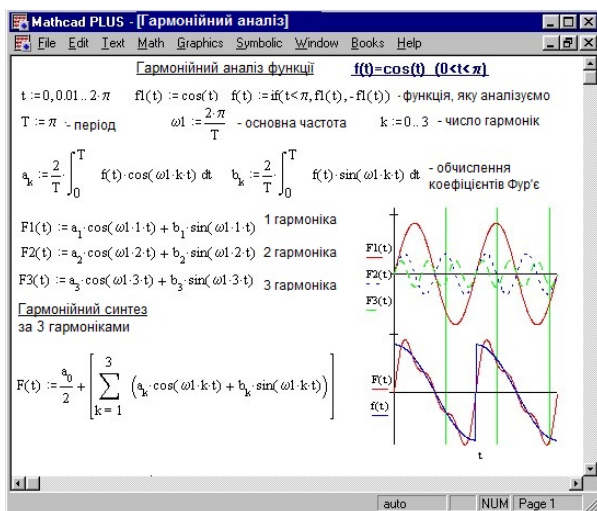


Рисунок 16. Гармонійний синтез

## Класичний спектральний аналіз

Спектротчасової залежності (функції)  $f(t)$  називається сукупність її гармонійних складових, що утворюють ряд Фур'є. Спектр можна характеризувати деякої залежністю  $A_k$  (спектр амплітуд) і  $\phi_k$  (спектр фаз) від частоти  $\omega_k = k\omega_1$ .

Спектральний аналіз періодичних функцій полягає в знаходженні амплітуди  $A_k$  і фази  $\phi_k$  гармонік (косинусоїд) ряду Фур'є (4). Задача, зворотна спектральному аналізу, називається спектральним синтезом (рис. 17 - продовження рисунка 16).

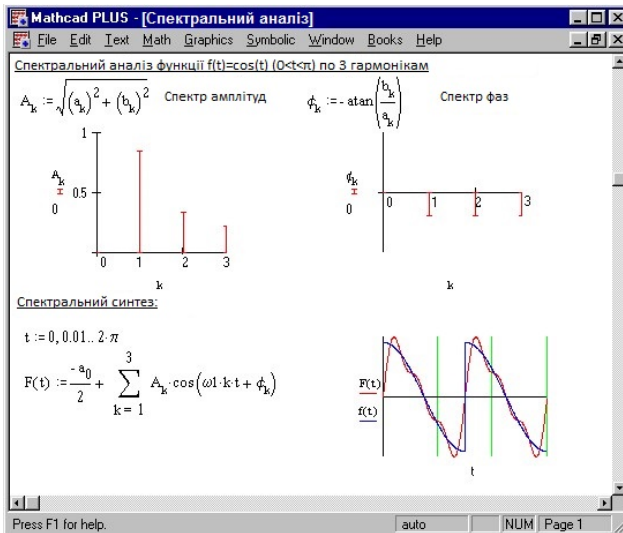


Рисунок 17. Класичний спектральний аналіз і синтез

Слово "класичний" тут означає, що коефіцієнти Фур'є обчислюються прямим інтегруванням тим методом, який використовується в Mathcad.

## Чисельний спектральний аналіз

Чисельний спектральний аналіз полягає в знаходженні коефіцієнтів  $a_0, a_1, \dots, a_k, b_1, b_2, \dots, b_k$  (або  $A_1, A_2, \dots, A_k, \varphi_1, \varphi_2, \dots, \varphi_k$ ) для періодичної функції  $y = f(t)$ , заданої на відрізку  $[0, T]$  дискретними відліками. Він зводиться до обчислення коефіцієнтів Фур'є за формулами чисельного інтегрування для методу прямокутників (див. Лабораторну роботу 5)

$$a_k = \frac{2}{N} \sum_{i=0}^{N-1} y_i \cos k \omega_1 i \Delta t, \quad (7)$$

$$b_k = \frac{2}{N} \sum_{i=0}^{N-1} y_i \sin k \omega_1 i \Delta t, \quad (8)$$

де  $\Delta t = T/N$  — крок, з яким розташовані абсциси  $y = f(t)$ .

## Спектральний аналіз на основі швидкого перетворення Фур'є

Вбудовані в Mathcad засоби швидкого перетворення Фур'є (ШПФ) істотно спрощують процедуру наближеного спектрального аналізу. ШПФ — швидкий алгоритм перенесення відомостей про функцію, задану  $2^m$  ( $m$  — ціле число) відліками в часовій області, в частотну область. Якщо мова йде про функції  $f(t)$ , заданої дійсними відліками, слід використовувати функцію *fft*.

fft (v)	Повертає пряме ШПФ $2^m$ -мірного речовозначного вектора $v$ , де $v$ — вектор, елементи якого зберігають відліки функції $f(t)$ .
---------	--

Результатом буде вектор  $A$  розмірності  $1 + 2^{m-1}$  з комплексними елементами — відліками в частотній області. Фактично дійсна і уявна частини вектора є коефіцієнти Фур'є  $a_k$  і  $b_k$ , що істотно спрощує їх отримання (див. Додаток 3).

Функція *ifft* реалізує зворотне ШПФ:

iff(t) (v)	Повертає зворотне ШПФ для вектора v з комплексними елементами. Вектор v має $1 + 2^{m-1}$ елементів.
------------	--

Результатом буде вектор A розмірності  $2^m$  з дійсними елементами.

На рисунку 18 показано застосування ШПФ для спектрального аналізу і синтезу імпульсу.

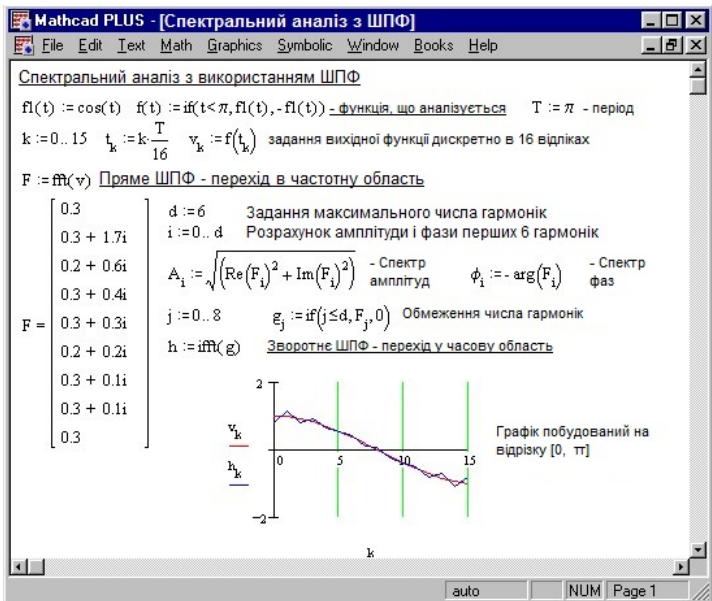


Рисунок 18. Спектральний аналіз з використанням ШПФ

## Фільтрація аналогових сигналів

Під фільтрацією мається на увазі виділення корисного сигналу з його суміші з заважаючим сигналом — шумом. Найбільш поширений тип фільтрації — частотна фільтрація. Якщо відома область частот, що зайняті корисним сигналом,

досить виділити цю область і подавити ті області, які зайняті шумом.

Малюнок 19 ілюструє техніку фільтрації із застосуванням ШПФ. Спочатку синтезується вихідний сигнал, представлений 128 отсчетами вектора  $v$ . Потім до цього сигналу приєднується шум за допомогою генератора випадкових чисел (функція  $rnd$ ) і формується вектор зі 128 відліків на зашумленого сигналу.

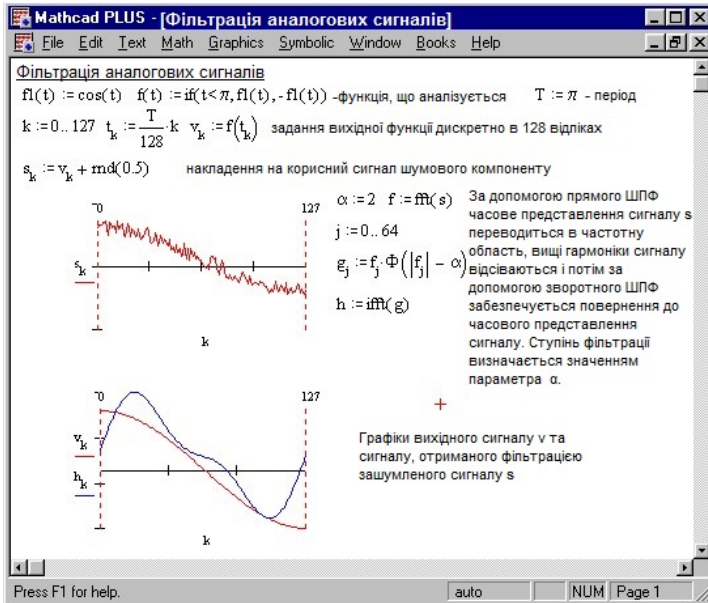


Рисунок 19. Фільтрація аналогових сигналів

Використовуючи пряме ШПФ, сигнал з шумом перетворюється з часової області в частотну, що створює вектор  $f$  з 64 частотних складових. Потім виконується фільтруюче перетворення, ефективність якого оцінюється параметром  $\alpha$ . Фільтруюче перетворення зручно виконувати за допомогою функції Хевісайда

$\Phi(x)$	Ступінчаста функція Хевісайда. Повертає 1, якщо $x \geq 0$ ; інакше 0.
-----------	--

Відфільтрований сигнал (вектор  $g$ ) має зворотний ШПФ і створює вектор вихідного сигналу  $h$ .

Порівняння часових залежностей вихідного і вхідного сигналів, показує, що вихідний сигнал майже повністю повторює вхідний і в значній мірі позбавлений високочастотних шумових перешкод, що маскують корисний сигнал.

## Порядок виконання лабораторної роботи 8

**Завдання 1.** Обчислити перші шість пар коефіцієнтів розкладання в ряд Фур'є функції  $f(t)$  на відрізку  $[0, 2\pi]$ .

Побудувати графіки 1, 2 і 3 гармонік.

Виконати гармонійний синтез функції  $f(t)$  по 1, 2 і 3 гармонікам. Результати синтезу відобразити графічно.

### Варіанти завдання 1

№ вар	$f(t)$	№ вар	$f(t)$	№ вар	$f(t)$
1	$\frac{\cos t}{1 + \cos^2 2t}$	6	$\cos t \cos \sqrt{t}$	11	$\sin(\sqrt{1+t^2})$
2	$\frac{\sin t}{1 + \cos^2 2t}$	7	$\arctg\left(\cos \frac{1}{2}t\right)$	12	$\cos(\sqrt{1+t^2})$
3	$\frac{\sin 2t + \sin^2 3t}{3 + \sin t + \cos 2t}$	8	$e^{\sin \frac{1}{3}t}$	13	$e^{-10(t-\pi)^2}$
4	$\frac{\sin 3t}{ \sin t  +  \cos t }$	9	$ \sin t  +  \sin 2t $	14	$e^{\cos \frac{1}{3}t}$
5	$\cos e^{ \sin 3t }$	10	$\sin\left(\frac{1}{2}t\right)^2$	15	$e^{-\cos \frac{1}{2}t} \cos(\sin t)$

**Завдання 2.** Виконати класичний спектральний аналіз і синтез функції  $f(t)$ . Відобразити графічно спектри амплітуд і фаз, результат спектрального синтезу функції  $f(t)$ .

**Завдання 3.** Виконати чисельний спектральний аналіз і синтез функції  $f(t)$ . Для цього необхідно задати вихідну функцію  $f(t)$  дискретно в 32 відліках. Відобразити графічно спектри амплітуд і фаз, результат спектрального синтезу функції  $f(t)$ .

**Завдання 4.** Виконати спектральний аналіз і синтез функції  $f(t)$  за допомогою ШПФ. Для цього необхідно:

- задати вихідну функцію  $f(t)$  дискретно в 128 відліках;
- виконати пряме ШПФ за допомогою функції  $fft$  і відобразити графічно знайдені спектри амплітуд і фаз перших шести гармонік;
- виконати зворотне ШПФ за допомогою функції  $ifft$  і відобразити графічно результат спектрального синтезу функції  $f(t)$ .

**Завдання 5.** Виконати фільтрацію функції  $f(t)$  за допомогою ШПФ:

- синтезувати функцію  $f(t)$  у вигляді корисного сигналу, представленого 128 відліками вектора  $v$ ;
- до корисного сигналу  $v$  приєднати шум за допомогою функції  $rnd$  ( $rnd(2) - 1$ ) і сформувати вектор зі 128 відліків зашумленого сигналу  $s$ ;
- перетворити сигнал з шумом  $s$  з часової області в частотну, використовуючи пряме ШПФ (функція  $fft$ ). В результаті вийде сигнал  $f$  з 64 частотних складових;
- виконати фільтруюче перетворення за допомогою функції Хевісайда (параметр фільтрації  $\alpha=2$ );
- за допомогою функції  $ifft$  виконати зворотне ШПФ і отримати вектор вихідного сигналу  $h$ ;
- побудувати графіки корисного сигналу  $v$  і сигналу, отриманого фільтрацією зашумленого сигналу  $s$ .

## Лабораторна робота 9

### Безперервно-детерміновані схеми математичного моделювання (D-схеми). Імітаційне моделювання

Прикладами використання цих схем можуть служити моделі еволюції біологічних систем, статичні та динамічні моделі математичної фізики, а також моделі процесів в конкретних предметних областях і деякі моделі оптимізації (наприклад, оптимального управління).

**Приклад 1.** Найпростішою еволюційною моделлю є модель динаміки популяції (модель Мальтуса).

#### Математична модель

Математичною моделлю є задача Коші для лінійного диференціального рівняння першого порядку

$$\frac{dN(t)}{dt} = (\alpha(t) - \beta(t))N(t), N(0) = N_0,$$

де  $N(t)$  — чисельність популяції,  $\alpha(t)$  — коефіцієнт розмноження,  $\beta(t)$  — коефіцієнт смертності. Дане рівняння допускає поділ змінних.

#### Рішення

Розділимо змінні  $\frac{dN(t)}{dt} = (\alpha(t) - \beta(t))dt$ .

Інтегруючи праву і ліву частини рівняння, отримуємо загальний інтеграл  $\int \frac{dN(t)}{N(t)} = \ln|N(t)| = \int (\alpha(t) - \beta(t))dt + \ln|C|, C \neq 0$ .

Таким чином, загальне рішення має вигляд

$$N(t) = C e^{\int(\alpha(t)-\beta(t))dt}.$$

З початкової умови слідує

$$N_0 = C e^{\int(\alpha(t)-\beta(t))dt} |_{t=0} = 0$$

Якщо коефіцієнти  $\alpha(t)$  і  $\beta(t)$  є константами, то рішення має вигляд

$$N(t) = N_0 e^{(\alpha-\beta)t}.$$

### Аналіз результату

Якщо  $\alpha > \beta$ , то чисельність популяції необмежено зростає. При  $\alpha < \beta$  чисельність знижується, і при  $\alpha = \beta$  чисельність залишається незмінною. Якщо коефіцієнти  $\alpha(t)$  і  $\beta(t)$  залежать від часу, то картина може бути суттєво іншою. Наприклад, якщо  $\alpha(t) = \cos t, \beta(t) = \sin\left(\frac{\pi}{4}\right)$ , то рішення має вигляд

$$N(t) = N_0 e^{\sin t - \cos\left(t + \frac{\pi}{4}\right) + \cos\left(\frac{\pi}{4}\right)}.$$

На рис.1 представлений характер цієї залежності при  $N_0 = 50$ .

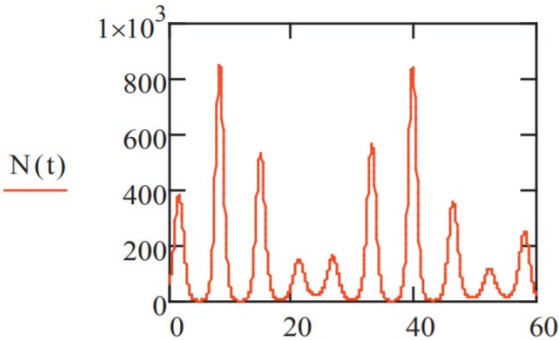


Рисунок. 1

## Обмеження моделі

Модель не враховує обмеженість ресурсів і фактори конкуренції з іншими популяціями, хоча частково ці фактори можна врахувати, вибираючи функції  $\alpha(t)$  і  $\beta(t)$ . Більш точна модель була запропонована Ферхюльста (див. Задачу № 1 в кінці розділу).

**ПРИКЛАД 2.** Модель конкуренції популяцій хижаків і жертв (модель Лотки - Вольтерра).

## Математична модель

Математичною моделлю є система нелінійних диференціальних рівнянь

$$\begin{aligned}\frac{dN(t)}{dt} &= (\alpha(t) - \beta(t) M(t)) N(t), \\ \frac{dM(t)}{dt} &= (-\gamma(t) + \delta(t) N(t)) M(t), \\ N(0) &= N_0, M(0) = M_0 \quad (1)\end{aligned}$$

$N(t) \geq 0, M(t) \geq 0$  - число жертв і хижаків, відповідно,  
 $\alpha(t) \geq 0, \beta(t) \geq 0$  - коефіцієнти народжуваності і зменшення жертв за рахунок хижаків,  $\gamma(t) \geq 0, \delta(t) \geq 0$  - коефіцієнти убування і зростання хижаків. Далі будемо вважати всі коефіцієнти постійними.

## Аналіз моделі

З системи слідує

$$\frac{dN(t)}{(\alpha - \beta M(t))N(t)} = \frac{dM(t)}{(-\gamma + \delta N(t))M(t)}$$

або

$$\frac{(-\gamma + \delta N(t))dN(t)}{N(t)} = \frac{(\alpha - \beta M(t))dM(t)}{M(t)}$$

Інтегруючи обидві частини, отримаємо

$$-\gamma \ln N(t) + \delta N(t) = \alpha \ln M(t) - \beta M(t) + C$$

або

$$\ln(M^\alpha(t)N^\gamma(t)) = \delta N(t) + \beta M(t) + \ln|C|, \quad C \neq 0.$$

Таким чином, маємо перший інтеграл у вигляді

$$(M^\alpha(t)N^\gamma(t)) = Ce^{\delta N(t) + \beta M(t)}, \quad C > 0.$$

З початкових умов отримуємо

$$M_0^\alpha N_0^\gamma = Ce^{\delta N_0 + \beta M_0}, \text{ т. е. } C = M_0^\alpha N_0^\gamma e^{-\delta N_0 - \beta M_0}.$$

Система (9.1) має стаціонарні рішення. Якщо коефіцієнти постійні, то з умов  $\frac{dN(t)}{dt} = 0$ ,  $\frac{dM(t)}{dt} = 0$  випливає, що рівноважні значення хижаків і жертв рівні  $N = \frac{\gamma}{\delta}$  и  $M = \frac{\alpha}{\beta}$ .

## Рішення

Так як система (1) нелінійна, то  $N(t)$  і  $M(t)$  знаходимо, вирішуючи систему чисельно з використанням процедури *rkfixed*. У наведеному нижче лістингу програми використані позначення  $N(t) = N_0(t)$  и  $M(t) = N_1(t)$ .

Лістинг програми.

$\alpha := 2$   $\beta := 0.02$   $\gamma := 3$   $\delta := 0.05$   $N_0 := 80$   $N_1 := 30$   $t_{\max} := 10$   
 $i := 100$

$$D(t, N) := \begin{pmatrix} \alpha \cdot N_0 - \beta \cdot N_0 N_1 \\ -\gamma \cdot N_1 + \delta \cdot N_0 N_1 \end{pmatrix} R$$

$$:= \text{rkfixed} \left( \begin{pmatrix} N_0 \\ N_1 \end{pmatrix}, 0, t_{\max}, D \right)$$

$$t := R^{(0)}N := R^{(1)}M := R^{(2)}.$$

Графіки отриманих рішень представлені на рис.2.

На рис.3 приведена фазова траєкторія.

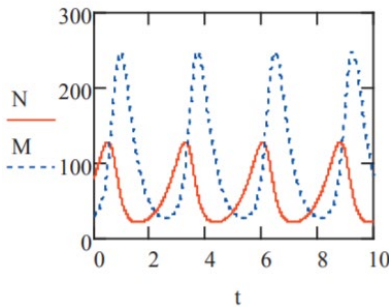


Рисунок.2

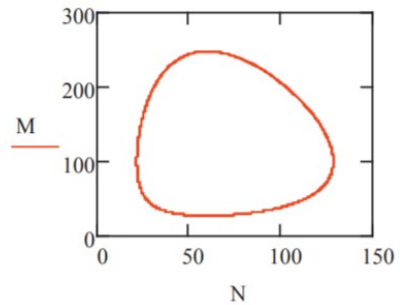


Рисунок.3

### Аналіз результатів

При вибраних значеннях коефіцієнтів рівноважні значення рівні  $N = 60$ ,  $M = 100$ . Рішення мають періодичний характер. Коливання відбуваються щодо рівноважних значень.

Аналогічна модель використовується і для опису інших процесів суперництва.

### Деякі моделі математичної фізики

Розглянемо найпростішу гідродинамічну модель очистки води методом осадження. В основі моделі лежить опис руху твердої частинки в рідині.

### ПРИКЛАД 3.

#### Математична модель

Диференціальне рівняння руху твердої частинки в рідині отримують з балансу діючих на неї сил тяжкості, тертя та підйомної сили. Рівняння руху має вигляд

$$g(\rho_T - \rho_{ж})V - \zeta\rho_{ж}\frac{U^2(t)}{2}\Omega - (\rho_T - \xi\rho_{ж})V\frac{dU}{dt} = 0, \quad (2)$$

де  $\rho_{ж}$  — щільність рідини;  $V$  — об'єм частинки;  $\sigma$  — коефіцієнт гідравлічного опору;  $\zeta$  — коефіцієнт приєднаної маси;  $\Omega$  — Міделевий переріз;  $U(t)$  — швидкість частинки.

Для цього рівняння ставиться задача Коші з  $U(t_0)=U_0$ .

#### Аналіз моделі

Запишемо рівняння у вигляді

$$A\frac{dU(t)}{dt} + BU^2(t) = C,$$

де  $A = (\rho_T - \xi\rho_{ж})V > 0$ ,  $B = \frac{\zeta\rho_{ж}\Omega}{2} > 0$ ,  $C = g(\rho_T - \rho_{ж})V > 0$ .

Розділивши на  $A$ , отримаємо

$$\frac{dU(t)}{dt} + bU^2(t) = c, \quad b = \frac{B}{A}, \quad c = \frac{C}{A}.$$

Якщо коефіцієнти  $b$  і  $c$  не залежать від часу, то це рівняння із перемінними, в іншому випадку це спеціальне рівняння Ріккати.

#### Рішення

Вважаючи  $b$  і  $c$  постійними, розділимо змінні

$\frac{dU}{c-bU^2} = dt$  або  $\frac{dU}{\frac{c}{b}-U^2} = bdt$ . Інтегруючи, отримаємо

$$\frac{1}{2} \sqrt{\frac{b}{c}} \ln \left| \frac{U + \sqrt{\frac{c}{b}}}{U - \sqrt{\frac{c}{b}}} \right| = bt + \sqrt{\frac{b}{c}} \ln|C|.$$

Таким чином маємо

$$\frac{U + \sqrt{\frac{c}{b}}}{U - \sqrt{\frac{c}{b}}} = Ce^{2\sqrt{cb}t}.$$

З початкових умов  $U(0)=U_0$  слідує

$$C = \frac{U_0 + \sqrt{\frac{c}{b}}}{U_0 - \sqrt{\frac{c}{b}}}.$$

Остаточно для вирішення отримуємо вираз

$$U(t) = - \frac{\sqrt{\frac{c}{b}} \left( U_0 + \sqrt{\frac{c}{b}} \right) e^{2\sqrt{cb}t} + U_0 - \sqrt{\frac{c}{b}}}{U_0 - \sqrt{\frac{c}{b}} - \left( U_0 + \sqrt{\frac{c}{b}} \right) e^{2\sqrt{cb}t}}.$$

Полученное выражение можно преобразовать, если разделить числитель и знаменатель сначала на  $e^{\sqrt{cb}t}$  и затем на  $e^{\sqrt{cb}t} + e^{-\sqrt{cb}t}$ . В результате получим

$$U(t) = \frac{U_0 \sqrt{cb} + c \cdot \text{th}(\sqrt{cb}t)}{\sqrt{cb} + bU_0 \text{th}(\sqrt{cb}t)}.$$

## Аналіз рішення

Так як  $\lim_{t \rightarrow +\infty} \operatorname{th}(\sqrt{cbt}) = \lim_{t \rightarrow +\infty} \frac{e^{\sqrt{cbt}} - e^{-\sqrt{cbt}}}{e^{\sqrt{cbt}} + e^{-\sqrt{cbt}}} = 1$ , то швидкість

частинки прагне до постійного значення  $U(t) = \frac{U_0 \sqrt{cb} + c}{\sqrt{cb} + bU_0}$ .

Конкретні значення параметрів, що входять в рівняння (2.2), мають розмірності  $g \left(\frac{\text{м}}{\text{с}^2}\right)$ ,  $\rho_T, \rho_{\text{ж}} \left(\frac{\text{кг}}{\text{м}^3}\right)$ ,  $\zeta, \xi$  - безрозмірні величини. Значення  $\zeta$  варіюються в межах:

1) тонка пластинка  $\zeta \approx 1,1$ ; 2) відкрита півсфера  $\zeta \approx 0,35$ ;

3) сфера  $\zeta \approx 0,1-0,4$ ; тіло обтічної форми  $\zeta \approx 0,04$ . Для сфери  $\xi \approx 0,5$ . Розмірності  $\Omega$  и  $V \text{ м}^3$ , відповідно. Якщо частка має форму еліпсоїда обертання, то  $\Omega = \pi a^2$ ,  $V = \frac{4}{3} \pi a^2 c$ .

На рис. 2.4 представлений графік вирішення  $U(t)$ , отриманого при наступних значеннях параметрів:

$$V = 0,0000001, \Omega = 0,00003, \zeta = 0,4, \xi = 0,5, \rho_T = 2400, \\ \rho_{\text{ж}} = 998,2, g = 9,8.$$

## Аналіз моделі

Модель не враховує деякі важливі ефекти, які виникають при осадженні, зокрема, можливе збільшення маси частинки через приєднання інших частинок, наявність в рідині забруднень, а також виникнення градієнта концентрації, що змінює умови осадження. Математична модель, що враховує ці ефекти, має форму загального рівняння Риккати зі змінними коефіцієнтами, яке допускає тільки чисельне рішення (див. Задачу № 4 в кінці цього розділу).

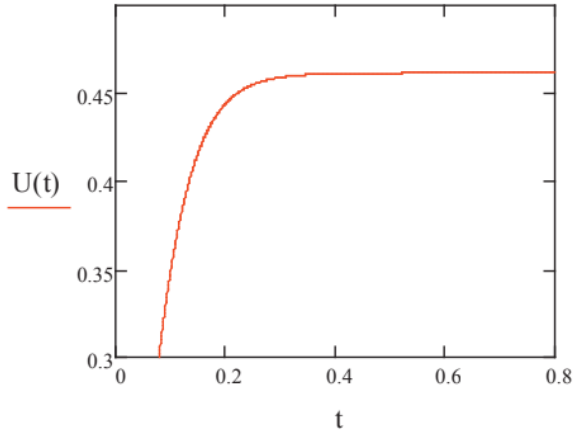


Рисунок. 4

**ПРИКЛАД 4.** Аналогічні моделі у D-схемі.

Ці математичні моделі описують процеси різної фізичної природи при тому, що математичні моделі аналогічні. Цей факт є наслідком того, що різні фізичні закони, що лежать в основі математичних моделей, мають однакове математичний опис:

1. Закон Фур'є поширення тепла

$$\bar{Q} = -\lambda \text{grad}T,$$

де  $\bar{Q}$  — потік тепла; T - температура;  $\lambda$  - коефіцієнт теплопровідності.

2. Закон Ома

$$\bar{J} = -\frac{1}{\rho} \text{grad}U,$$

де  $\bar{J}$  — потік струму; U — потенціал;  $\rho$  - коефіцієнт електропровідності.

### 3. Дарсі – Вейсбаха

$$\bar{V} = -k \text{grad} P,$$

де  $\bar{V}$  — вектор швидкості;  $P$  - тиск;  $k$  - коефіцієнт фільтрації.

### 4. Закон Фіка

$$\bar{M} = -D \text{grad} C,$$

де  $\bar{M}$  – потік маси;  $C$  - концентрація;  $D$  - коефіцієнт дифузії.

Як приклад розглянемо рівняння теплопровідності

$$c\rho \frac{\partial U(M, t)}{\partial t} = \text{div}(\lambda \text{grad} U) + F(M, t),$$

де  $M(x, y, z)$  — точка в просторі;  $U(M, t)$  - температура;  $c$  — питома теплоємність;  $\rho$  — щільність речовини;  $\lambda$  — коефіцієнт теплопровідності;  $F(M, t)$  щільність джерел тепла.

Далі будемо вважати, що  $c$ ,  $\rho$  и  $\lambda$  — постійні, і перетворимо рівняння до виду

$$\frac{\partial U(M, t)}{\partial t} = a \Delta U(M, t) + \frac{1}{c\rho} F(M, t),$$

де  $a = \frac{\lambda}{c\rho}$ ,  $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$  — оператор Лапласа.

Це тривимірне рівняння в приватних похідних, рішення якого перебуває при певних початкових і крайових умовах.

Розглянемо першу крайову задачу для одновимірного рівняння теплопровідності

$$\frac{\partial U(x, t)}{\partial t} = \frac{\partial^2 U(x, t)}{\partial x^2} + f(x, t), 0 < x < 1$$

$$U(x, t) = \varphi(x), U(0, t) = \mu_1(t), U(1, t) = \mu_2(t). \quad (3)$$

Це завдання є моделлю поширення тепла в тонкому теплоізолированном стрижні довжиною  $l$  з початковим

розподілом температури  $\varphi(x)$ , і на кінцях якого заданий режим зміни температури, що визначається функціями  $\mu_1(t)$  і  $\mu_2(t)$ .

### Решение

Рішення завдання (2.3) можна отримати аналітично у вигляді ряду Фур'є і чисельно, використовуючи метод кінцевих різниць. Аналітичне рішення має вигляд

$$\begin{aligned}
 U(x, t) = & \mu_1(t) + \frac{x}{l} (\mu_2(t) - \mu_1(t)) \\
 + \sum_{n=1}^{\infty} e^{-\left(\frac{n\pi}{l}\right)^2 t} \sin \frac{n\pi x}{l} & \left( \varphi_n - \frac{2}{\pi n} (\mu_1(0) - (-1)^n \mu_2(0)) \right) + \\
 + \sum_{n=1}^{\infty} e^{-\left(\frac{n\pi}{l}\right)^2 t} \sin \frac{n\pi x}{l} & \int_0^t e^{\left(\frac{n\pi}{l}\right)^2 \tau} \left( f_n(\tau) \right. \\
 & \left. - \frac{2}{\pi n} (\dot{\mu}_1(\tau) - (-1)^n \dot{\mu}_2(\tau)) \right) d\tau,
 \end{aligned}$$

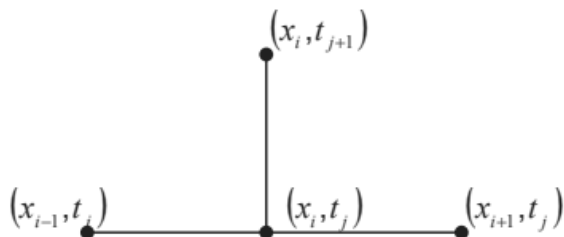
где  $\varphi_n(t) = \frac{2}{l} \int_0^l \varphi(x) \sin \frac{n\pi x}{l} dx$ ,  $f_n(t) = \frac{2}{l} \int_0^l f(x, t) \sin \frac{n\pi x}{l} dx$ .

Чисельне рішення методом кінцевих різниць отримують, замінюючи саме рівняння, а також початкові і крайові умови різницевиими аналогами. Для побудови різницевої схеми скористаємося рівномірної сіткою з кроком  $h$  по  $x$  і кроком  $\tau$  по  $t$ :

$$x_i = h \cdot i, \quad t_j = \tau \cdot j, \quad i = 0, 1, \dots, n, j = 0, 1, \dots, m.$$

В цьому випадку  $h = \frac{l}{n}$  и  $\tau m = T$ .

В явних схемах значення сіткової функції на наступному часовому шарі сітки явно виражається через значення на попередньому шарі. Це має місце, наприклад, при використанні шаблону



Для диференціальних операторів в точці  $(x_i, t_j)$  використовуємо різницеві відносини

$$\begin{aligned} \frac{\partial U}{\partial t} &\Rightarrow \frac{U(x_i, t_{j+1}) - U(x_i, t_j)}{\tau}, \quad \frac{\partial^2 U}{\partial x^2} \Rightarrow \\ &\Rightarrow \frac{U(x_{i+1}, t_j) - 2 \cdot U(x_i, t_j) + U(x_{i-1}, t_j)}{h^2} \end{aligned}$$

Різницеве рівняння має вигляд

$$\begin{aligned} U_i^{j+1} &= U_i^j + \left(\frac{\tau}{h^2}\right) (U_{i+1}^j - 2 \cdot U_i^j + U_{i-1}^j) + \tau \cdot \tilde{f}_i^j, \\ &i=1, 2, \dots, n-1, j=0, 1, \dots, m-1; \end{aligned} \quad (4)$$

$$U_0^j = \mu_1(t_j) = \mu_1^j, \quad U_n^j = \mu_2(t_j) = \mu_2^j, \quad j = 0, 1, \dots, m,$$

$$U_i^0 = \varphi(x_i) = \varphi_i \quad i = 1, \dots, n-1,$$

де  $\tilde{f}_i^j$  може визначатися за формулою  $\tilde{f}_i^j = f(x_i, t_j)$ .

Розглянемо рішення першої крайової задачі з початковою умовою  $\varphi = x(1-x)$  і крайовими умовами вигляду

$\mu_1(t) = t, \mu_2(t) = \sin t$ . На рис. 5 представлений графік вирішення, отриманого з аналітичного виразу з урахуванням в сумі перших п'яти членів.

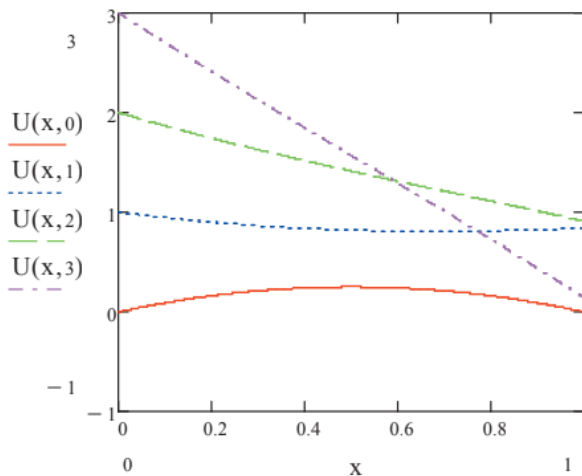


Рисунок 5

Лістинг програми чисельного рішення.

$$n := 9, \quad d := 1, \quad h := \frac{d}{n}, \quad \tau := 0.005, \mu_1(t) := t, \\ \mu_2(t) := \sin t, \quad \varphi := 0, \quad f(x, t) := 0$$



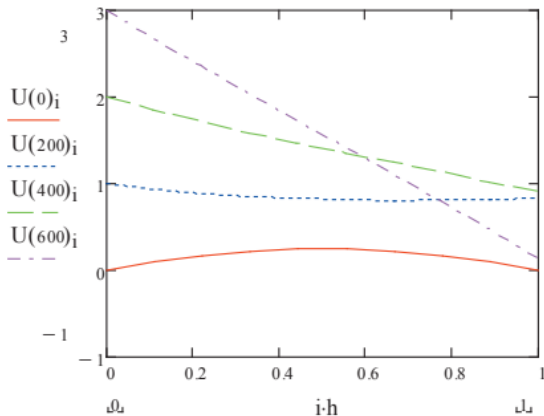


Рисунок. 6

## Завдання

Зверненням моделі Мальтуса є модель Ферхюльста, в якій вводиться обмеження на зростання популяції, і коефіцієнт приросту зменшується при наближенні до максимально допустимого значення чисельності популяції. Модель має вигляд:

$$\frac{dN(t)}{dt} = k \left( \frac{N_{max} - N(t)}{N_{max}} \right) N(t), N(0) = N_0,$$

де  $N_{max}$  — максимальна чисельність:

- досліджувати стаціонарні стану;
- отримати аналітичне та чисельне рішення при різних значеннях коефіцієнта  $k$ ;
- провести аналіз отриманих результатів.

2. Отримати рішення в моделі Лотки - Вольтерра зі змінним коефіцієнтом  $\alpha(t) = A \cos(\omega t)$  при різних значеннях параметрів  $A$  і  $\omega$ .

3. Отримати рішення і провести аналіз моделі

$$\begin{aligned}\frac{dN(t)}{dt} &= (\alpha_1(t) - \beta_{12}(t)M(t) - \beta_{11}(t)N(t))N(t) \\ \frac{dM(t)}{dt} &= (\alpha_2(t) - \beta_{21}(t)N(t) - \beta_{22}(t)M(t))M(t)\end{aligned}$$

за певних початкових умов.

4. Отримати рішення в моделі осадження при збільшенні маси частинок в процесі осадження. Модель має вигляд

$$g(\rho_T - \rho_{ж})V - 3c\rho_{ж}\frac{U^2(t)}{4D(t)} - (\rho_T + \xi\rho_{ж})V\frac{dU}{dt} - \frac{3\psi\rho_T}{D(t)}\frac{dD}{dt} = 0,$$

де  $\psi$  — коефіцієнт нерівномірності нарощування обсягу, а

$c$  — коефіцієнт опору частки в несучому шарі.

а Розглянути випадок  $D(t) = D_0(1 + t)$ .

5. Отримати чисельне рішення першої крайової задачі для рівняння теплопровідності при  $\varphi(x) = 0, f(x, t) = \sin\left(\frac{\pi x}{l}\right)$ .

## Лабораторна робота 10

### Методи рішень графічних задач в Maple.

#### 10.1. Побудова графіків. Двовимірні графіки.

Команда plot та її параметри

Для побудови графіків функції  $f(x)$  однієї змінної (в інтервалі  $a \leq x \leq b$  по осі  $Ox$  та в інтервалі  $c \leq y \leq d$  по осі  $Oy$ ) використовується команда **plot (f(x), x=a..b, y=c..d, parameters)**, де **parameters** — параметри управління зображенням. Якщо їх не вказувати, то будуть використані

установки за замовчуванням. Налаштування зображення також може здійснюватися з панелі інструментів.

Основні параметри команди **plot**:

- 1) **title="text"**, де **text** – заголовок рисунка (текст можна залишати без лапок, якщо він складається лише з латинських літер без пробілів).
- 2) **coords=polar** – установка полярних координат (за замовчуванням встановлено декартові).
- 3) **axes** – установка типу координатних осей: **axes=NORMAL** – звичайні осі; **axes=BOXED** – графік в рамці зі шкалою; **axes=FRAME** – осі з центром в лівому нижньому кутку рисунка; **axes=NONE** – без осей.
- 4) **scaling** – установка масштабу рисунка: **scaling=CONSTRAINED** – однаковий масштаб по осях; **scaling=UNCONSTRAINED** – графік масштабується за розмірами вікна.
- 5) **style=LINE(POINT)** – виведення лініями (або точками).
- 6) **numpoints=n** – число обчислюваних точок графіка (за замовчуванням n=49).
- 7) **color** – установка кольору лінії: англійська назва кольору, наприклад, **yellow** – жовтий і т.д.
- 8) **xtickmarks=nx** и **ytickmarks=ny** – число міток по осі Ox і осі Oy, відповідно.
- 9) **thickness=n**, де **n=1,2,3...** - товщина лінії (за замовчуванням n=1)
- 10) **linestyle=n** – тип лінії: безперервна, пунктирна і т.д. (n=1 – безперервна, встановлено за замовчуванням).
- 11) **symbol=s** – тип символу, яким позначають точки: **BOX, CROSS, CIRCLE, POINT, DIAMOND**.
- 12) **font=[f,style,size]** – установка типу шрифту для виведення тексту: **f** задає назву шрифтів: **TIMES, COURIER, HELVETICA, SYMBOL**; **style** задає стиль шрифту: **BOLD, ITALIC, UNDERLINE**; **size** – розмір шрифту в pt.

13) **labels=[tx,ty]** – написи по осях координат: **tx** – по осі  $Ox$  і **ty** – по осі  $Oy$ .

14) **discont=true** – вказівка для побудови нескінченних розривів.

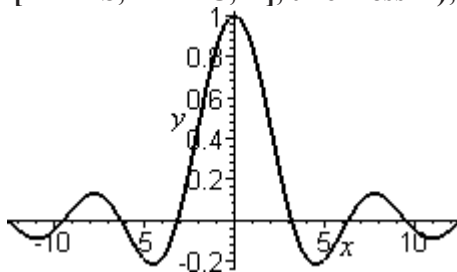
За допомогою команди **plot** можна будувати окрім графіків функцій  $y=f(x)$ , заданої явно, також графіки функцій, заданих параметрично  $y=y(t)$ ,  $x=x(t)$ , якщо записати команду **plot([y=y(t), x=x(t), t=a..b], parameters)**.

### Завдання 10.1.

1. Побудувати графік  $v = \frac{\sin x}{x}$  жирною лінією в інтервалі від  $-4\pi$  до  $4\pi$ .

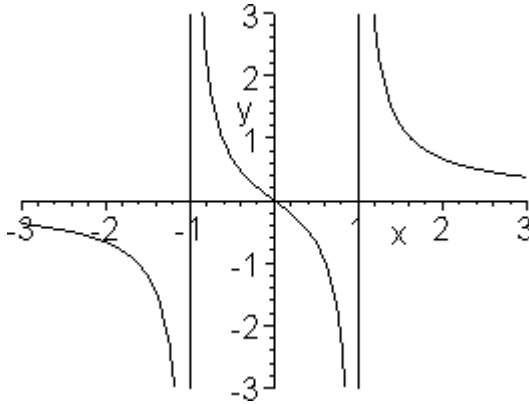
Наберіть:

```
> plot(sin(x)/x, x=-4*Pi..4*Pi, labels=[x,y],  
labelfont=[TIMES,ITALIC,12], thickness=2);
```



2. Побудувати графік розривної функції  $v = \frac{x}{x^2-1}$

```
> plot(x/(x^2-1),x=-3..3,y=-3..3,color=magenta);
```

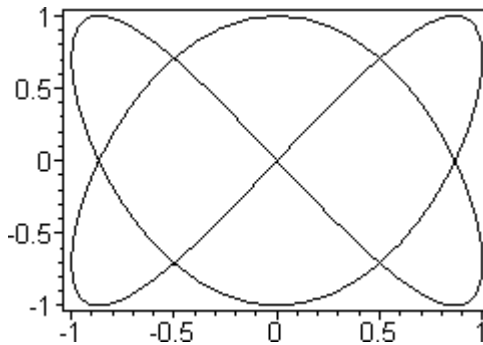


*Зауваження:* на рисунку автоматично з'являються вертикальні асимптоти.

3. Побудувати графік параметричної кривої  $y = \sin 2t$  ,  
 $x = \cos 3t$  ,

$0 \leq t \leq 2\pi$  в рамці. Наберіть:

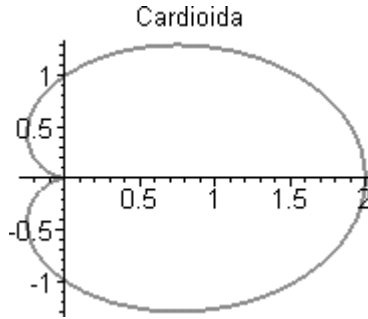
> **plot([sin(2\*t),cos(3\*t),t=0..2\*Pi], axes=BOXED, color=blue);**



4. Побудувати в полярних координатах графік кардіоїди  
 $\rho = 1 + \cos \varphi$

з назвою. Наберіть:

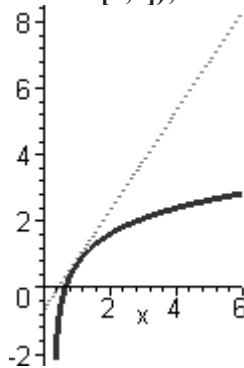
> **plot(1+cos(x), x=0..2\*Pi, title="Cardioida", coords=polar, color=coral, thickness=2);**



5. Побудувати два графіки на одному рисунку:  
 графік функції  $y = \ln(3x - 1)$  і дотичну до неї в  $y = \frac{3x}{2} - \ln 2$ .

Наберіть:

```
> plot([ln(3*x-1), 3*x/2-ln(2)], x=0..6,
scaling=CONSTRAINED, color=[violet,gold],
linestyle=[1,2], thickness=[3,2]);
```



Побудова графіку функції, заданої неявно.

Функція задана неявно, якщо вона задана рівнянням  $F(x, y) = 0$ . Для побудови графіку неявної функції використовується команда **implicitplot** з графічного пакету **plots**: **implicitplot(F(x,y)=0, x=x1..x2, y=y1..y2)**.

Виведення текстових коментарів на рисунок.

В пакеті **plots** є команда **textplot** для виведення текстових коментарів на рисунок: **textplot([x0,y0,'text'], options)**, де **x0**, **y0** – координати точки, з якої починається виведення тексту **'text'**.

Виведення декількох графічних об'єктів на один рисунок.

Часто буває необхідно поєднати на одному рисунку кілька графічних об'єктів, отриманих за допомогою різних команд, наприклад, додати графіку, намальовану командою **plot**, текстові написи, отримані командою **textplot**. Для цього результат дії команди присвоюється деякій змінній:

> **p:=plot(...): t:=textplot(...):**

При цьому на монітор виведення не виконується. Для виведення графічних зображень необхідно виконати команду з пакета **plots**:

> **with(plots): display([p,t], options)**

Побудова двовимірної області, заданої нерівностями.

Якщо необхідно побудувати двовимірну область, задану системою нерівностей  $f_1(x, y) > c_1, f_2(x, y) > c_2, \dots, f_n(x, y) > c_n$ , то можна використовувати команду **inequal** з пакета **plots**.

У команді **inequals ({f1 (x, y)>c1, ..., fn (x, y)>cn}, x = x1 ... x2, y = y1..y2, options)** в фігурних дужках вказується система нерівностей, яка визначає область, потім розміри координатних осей і параметри. Параметри регулюють кольори відкритих і закритих кордонів, кольори зовнішньої і внутрішньої областей, а також товщину ліній кордонів:

- **optionsfeasible = (color = red)** - установка кольору внутрішньої області;
- **optionsexcluded = (color = yellow)** - установка кольору зовнішньої області;
- **optionsopen (color = blue, thickness = 2)** - установка кольору і товщини лінії відкритої кордону;
- **optionsclosed (color = green, thickness = 3)** - установка кольору і товщини лінії закритого кордону.

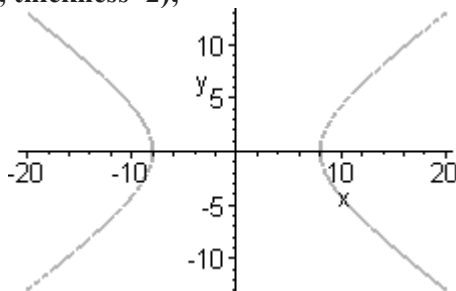
## Завдання 10.2.

1. Побудувати графік неявної функції (гіперболи):  $\frac{x^2}{4} - \frac{y^2}{2} = 16$

Наберіть:

**>with (plots):**

**implicitplot(x^2/4-y^2/2=16, x=-20..20, y=-16..16, color=green, thickness=2);**



2. Побудувати на одному рисунку графік астроїди  $4\cos^3 t$ ,  $x = 2\sin^3 t (0 \leq t \leq 2\pi)$  вписаної в еліпс  $\frac{x^2}{16} + \frac{y^2}{4} = 1$ . Виведіть назви ліній *Astroida* і *Ellips* жирним шрифтом, рівняння – курсивом. Для цього наберіть наступні рядки:

**> with (plots):**

**> eq:=x^2/16+y^2/4=1:**

**>el:=implicitplot(eq, x=-4..4, y=-2..2,**

**scaling=CONSTRAINED,**

**color=green, thickness=3):**

**>as:=plot([4\*cos(t)^3,2\*sin(t)^3, t=0..2\*Pi],**

**color=blue,**

**scaling=CONSTRAINED, thickness=2):**

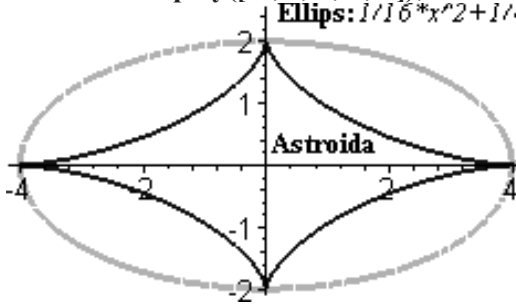
**> eq1:=convert(eq,string):**

**> t1:=textplot([1.5,2.5,eq1], font=[TIMES, ITALIC, 10], align=RIGHT):**

```

>t2:=textplot([0.2,2.5,"Ellips:"],
font=[TIMES, BOLD,10], align=RIGHT):
>t3:=textplot([1.8,0.4,Astroida],
font=[TIMES, BOLD,10], align=LEFT):
> display([as,el,t1,t2,t3]);

```

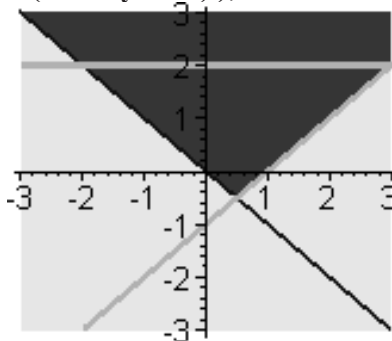


3. Побудувати область, обмежену лініями  $x + y > 0$ ,  $x - y \leq 1$ ,  $y = 2$ .

```

> with(plots):
>inequal({x+y>0, x-y<=1, y=2}, x=-3..3, y=-3..3,
optionsfeasible=(color=red),
optionsopen=(color=blue,thickness=2),
optionsclosed=(color=green, thickness=3),
optionsexcluded=(color=yellow) );

```



## 10.2. Тривимірні графіки. Анімація.

Графік поверхні, заданої явною функцією.

Графік функції  $z = f(x, y)$  можна намалювати, використовуючи команду **plot3d** (**f** (**x**, **y**), **x** = **x1** ... **x2**, **y** = **y1** ... **y2**, **options**). Параметри цієї команди частково збігаються з параметрами команди **plot**. До часто використовуваних параметрів команди **plot3d** відноситься **light** = [**angl1**, **angl2**, **c1**, **c2**, **c3**] – задання підсвічування поверхні, створеної джерелом світла з точки зі сферичними координатами (**angl1**, **angl2**). Колір визначається частками червоного (**c1**), зеленого (**c2**) і синього (**c3**) кольорів, які знаходяться в інтервалі [0,1]. Параметр **style** = **opt** задає стиль рисунка: **POINT** - точки, **LINE** - лінії, **HIDDEN** - сітка з видаленням невидимих ліній, **PATCH** - заповнювач (встановлено за замовчуванням), **WIREFRAME** - сітка з виведенням невидимих ліній, **CONTOUR** - лінії рівня, **PATCHCONTOUR** - заповнювач і лінії рівня. Параметр **shading** = **opt** задає функцію інтенсивності заповнювача, його значення дорівнює **xyz** - за замовчуванням, **NONE** - без розмальовки.

### Графік поверхні, заданої параметрично.

Якщо потрібно побудувати поверхню, задану параметрично:  $x = x(u, v)$ ,  $y = y(u, v)$ ,  $z = z(u, v)$ , то ці функції перераховуються в квадратних дужках в команді: **plot3d** [**x** (**u**, **v**), **y** (**u**, **v**), **z** (**u**, **v**)], **u** = **u1**..**u2**, **v** = **v1**..**v2**).

### Графік поверхні, заданої неявно.

Тривимірний графік поверхні, заданої неявно рівнянням  $F(x, y, z) = c$  будується за допомогою команди пакета **plot**: **implicitplot3d**  $F(x, y, z) = c$ , **x** = **x1**..**x2**, **y** = **y1**..**y2**, **z** = **z1**..**z2**, де вказується рівняння поверхні  $F(x, y, z) = c$  і розміри рисунка по координатним осям.

### Графік просторових кривих.

У пакеті **plot** є команда **spacescurve** для побудови просторової кривої, заданої параметрично:  $x = x(t)$ ,  $y = y(t)$ ,  $z = z(t)$ . Параметри команди:

> `spacecurve([x(t),y(t),z(t)],t=t1..t2)`, де змінна **t** змінюється від **t1** до **t2**.

#### Анімація.

Maple дозволяє виводити на екран зображення, що рухаються за допомогою команд **animate** (двовимірні) і **animate3d** (тривимірні) з пакету **plot**. Серед параметрів команди **animate3d** є **frames** - число кадрів анімації (за замовчуванням `frames = 8`).

Тривимірні зображення зручніше налаштовувати не за допомогою опцій команди **plot3d**, а використовуючи контекстне меню програми. Для цього слід натиснути правою кнопкою миші по зображенню. З'явиться контекстне меню настройки зображення. Команди цього меню дозволяють змінювати колір зображення, режими підсвічування, встановлювати потрібний тип осей, тип ліній і управляти рухомим зображенням.

Контекстне меню настройки зображення:



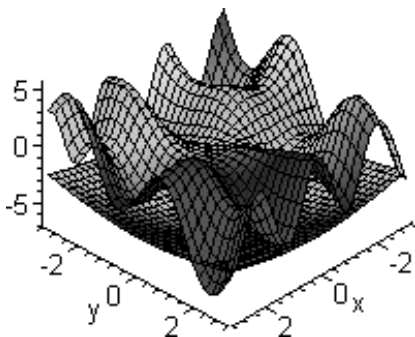
### **Завдання 10.3.**

1. Виконати побудову двох поверхонь  $z = x \sin 2y + y \cos 3x$  та

$z = \sqrt{x^2 + y^2} - 7$  в межах  $(x, y) \in [-\pi, \pi]$ . Встановіть змінний колір поверхонь як функцію  $x + y$ .

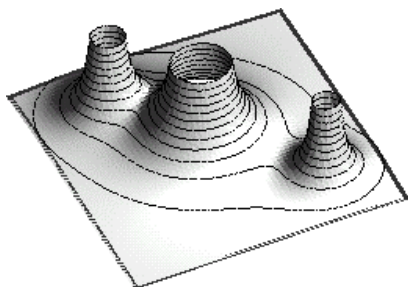
> `Plot3d ({x * sin (2 * y) + y * cos (3 * x), sqrt (x ^ 2 + y ^ 2) - 7},`

`x = -Pi..Pi, y = -Pi .. Pi, grid = [30,30], axes = FRAMED, color = x + y);`



2. Побудувати поверхню  $z = \frac{1}{x^2+y^2} + \frac{0,2}{(x+1,2)^2+(y-1,5)^2} + \frac{0,3}{(x-0,9)^2+(y+1,1)^2}$  разом з лініями рівня:

```
>plot3d(1/(x^2+y^2)+0.2/((x+1.2)^2+(y-1.5)^2)+0.3/((x-0.9)^2+(y+1.1)^2), x=-2..2, y=-2..2.5, view=[-2..2, -2..2.5, 0..6], grid=[60,60], shading=NONE, light=[100,30,1,1,1], axes=NONE, orientation=[65,20], style=PATCHCONTOUR);
```



3. Побудувати приблизну форму електронної хмари атома. Форма електронної хмари визначається двома квантовими числами: число  $l$  - визначає тип орбіталі, число  $m$  -

визначає магнітний момент електрона. При  $m = 0$  форма електронної хмари задається поліномами Лежандра першого роду:  $P(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n$ . Слід побудувати параметрично задану поверхню:  $x(\theta, \varphi) = Y(\varphi) \sin \varphi \cos \theta$ ,  $y(\theta, \varphi) = Y(\varphi) \sin \varphi \sin \theta$ ,  $z(\theta, \varphi) = Y(\varphi) \cos \varphi$ , де  $Y(\varphi) = \sqrt{\frac{2l+1}{4\pi}} P(\cos \varphi)$ . Спочатку прокладіть  $l = 3$ . Наберіть команди:

```
> l:=3;
> P:=(x,n)->1/(2^n*n!)*diff((x^2-1)^n,x$n);
> Y:=(phi)->abs(sqrt((2*l+1)/(4*Pi))*
subs(x=cos(phi),P(x,l)));
> X0:=Y(phi)*sin(phi)*cos(theta);
> Y0:=Y(phi)*sin(phi)*sin(theta);
> Z0:=Y(phi)*cos(phi);
> plot3d([X0,Y0,Z0],phi=0..Pi,theta=0..2*Pi,
scaling=CONSTRAINED, title="Електронна хмара");
```

Електронна хмара



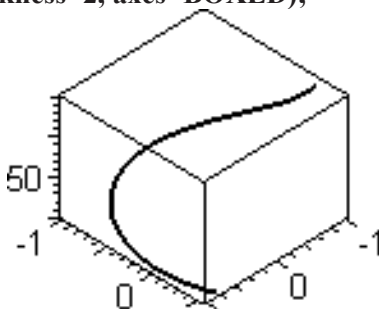
Після цього побудуйте форми електронної хмари для  $l = 1$  і  $l = 2$ .

4. Побудувати кулю  $x^2 + y^2 + z^2 = 4$  :  
**> with(plots): implicitplot 3d(x^2+y^2+z^2=4, x=-2..2, y=-2..2, z=-2..2, scaling=CONSTRAINED);**



5. Побудувати просторову криву:  $x = \sin(t)$ ,  $y = \cos(t)$ ,  $z = e^t$

**> with(plots):  
> spacecurve([sin(t),cos(t),exp(t)], t=1..5, color=blue, thickness=2, axes=BOXED);**



6. Намалювати об'єкт, що рухається. Спочатку наберіть в командному рядку:

**> animate3d(cos(t\*x)\*sin(t\*y), x=-Pi..Pi, y=-Pi..Pi, t=1..2);**

Клацніть правою кнопкою миші по зображенню, яке з'явилося. У контекстному меню виконайте команду *Animation*→*Continuous*. Потім знову викличте контекстне меню і виконайте команду *Animation*→*Play*. Для того, щоб зупинити

рух, виконайте команду *Animation*→*Stop*. За допомогою миші поверніть рисунок під іншим кутом і зробіть його рухомим.

7. Виконайте всі контрольні завдання. Результати покажіть викладачеві. Збережіть файл з усіма виконаними завданнями. Дайте відповідь на всі контрольні питання.







Навчальне видання

## МЕТОДИЧНІ ВКАЗІВКИ

### ПРОБЛЕМИ КОНСТРУЮВАННЯ ОБ'ЄКТІВ У КОМП'ЮТЕРНИХ СИСТЕМАХ

для підготовки докторів філософії зі спеціальностей:  
122 — Комп'ютерні науки, 131 — Прикладна механіка

Укладач АДАШЕВСЬКА Ірина Юріївна,

За авторською редакцією

План 2019 р., поз. 144 .

Підписано до друку 17.05.2019 .Формат 60×84 1/16.

Папір офсетний. Друк — цифровий.

Гарнітура Times New Roman Ум. друк. арк. 6,375

Наклад 20 прим. Зам. № 1275/19. Ціна договірна.

---

Надруковано: видавництво «НТМТ»  
61103, м. Харків, вул. Дерев'янка, 16  
E-mail: [jornal\\_2016@ukr.net](mailto:jornal_2016@ukr.net)

Свідоцтво суб'єкта видавничої справи ДК № 1748 від 15.04.2005 р..