

На головну ► Мої курси ► Учебная категория ► СО ► Тема 2 ►

Лекція 2. Системи алгебраїчних рівнянь і нерівностей

Лекція 2. Системи алгебраїчних рівнянь і нерівностей

Розв'язування рівнянь та їх систем

Для кращого засвоєння матеріалу лекції рекомендується повторити основні відомості з лінійної алгебри. Ви можете скористатись методичним посібником з лінійної алгебри, також корисним буде посібник з аналітичної геометрії.

Основна функція *solve*

В *Maple* існують потужні засоби розв'язування лінійних та нелінійних рівнянь та їх систем. Для розв'язування рівнянь та систем рівнянь в аналітичному вигляді використовується достатньо універсальна та гнучка функція ***solve(eq_n, var)*** або ***solve({eq_{n1}, eq_{n2}, ... }, {var₁, var₂, ... })***, де ***eq_n*** — рівняння, що містить функцію ряду змінних, ***var*** — змінна, по якій шукається розв'язок. Якщо не використовується знак рівності або відношення в запису ***eq_n***, тоді вважається, що ***solve*** шукає корні рівняння ***eq_n=0***.

Функція ***solve*** намагається надати розв'язок в аналітичному вигляді. Але її можна використовувати для знаходження коренів в числовому вигляді. Для цього доведеться використовувати функції ***evalf*** або ***convert***.

Приклади використання функції ***solve*** в рівняннях:

малюнок 2.1

```

> solve(x^3-x^2+3*x-1,x);

$$\frac{1}{3}(1+3\sqrt{57})^{1/3} - \frac{8}{3(1+3\sqrt{57})^{1/3}} + \frac{1}{3}, -\frac{1}{6}(1+3\sqrt{57})^{1/3} + \frac{4}{3(1+3\sqrt{57})^{1/3}} + \frac{1}{3} + \frac{1}{2}I\sqrt{3}\left(\frac{1}{3}(1+3\sqrt{57})^{1/3} + \frac{8}{3(1+3\sqrt{57})^{1/3}}\right),$$


$$-\frac{1}{6}(1+3\sqrt{57})^{1/3} + \frac{4}{3(1+3\sqrt{57})^{1/3}} + \frac{1}{3} - \frac{1}{2}I\sqrt{3}\left(\frac{1}{3}(1+3\sqrt{57})^{1/3} + \frac{8}{3(1+3\sqrt{57})^{1/3}}\right)$$

> solve(x^(3/2)=5,x);

$$5^{2/3}$$

> evalf(%);
2.924017738
> solve(ln(x)=6,x);

$$e^6$$

> evalf(%);
403.4287935
>

```

Часто рівняння та його розв'язки для зручності задають у вигляді окремих об'єктів:

```

> eq:=(x^3-2*x+1=0);

$$eq:=x^3-2x+1=0$$

> s:=solve(eq,x);

$$s:=\left[1, \frac{1}{2}\sqrt{5}-\frac{1}{2}, -\frac{1}{2}-\frac{1}{2}\sqrt{5}\right]$$

>

```

Для розв'язування систем лінійних рівнянь створені матричні методи. Але функція *solve* також успішно використовується в цих задачах.

Наприклад:

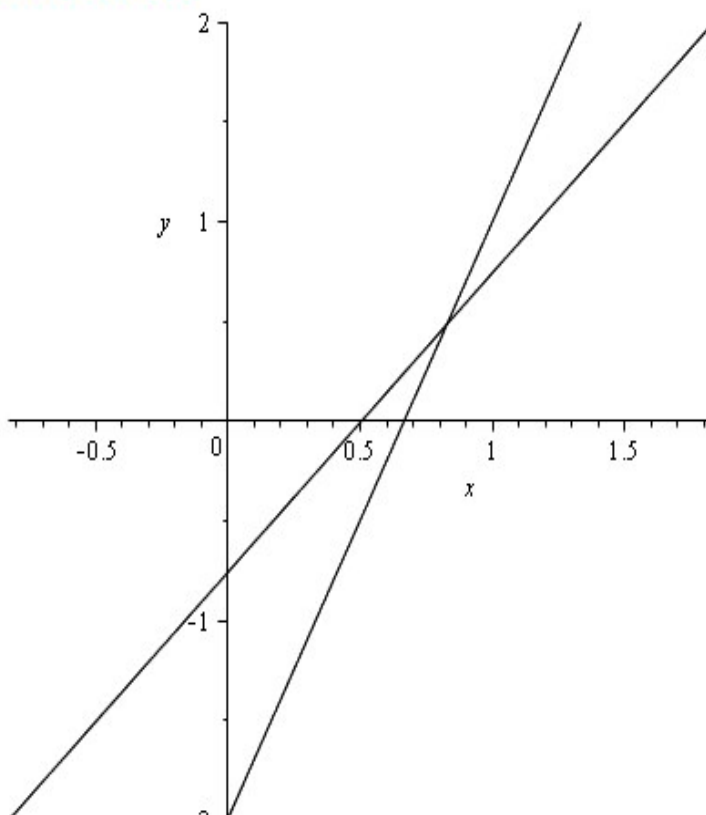
малюнок 2.2

```
> with(plots):
> sys:={3*x-y=2, 6*x-4*y=3};
> solve(sys, [x,y]);
```

```
sys := {3x - y = 2, 6x - 4y = 3}
```

```
[[x = 5/6, y = 1/2]]
```

```
> implicitplot(sys, x=-2...2, y=-2...2, color=black);
```



Пакет розширень **plots** надає можливість графічно проілюструвати розв'язки. Розв'язати систему із попереднього прикладу означає знайти точки перетину двох прямих.

В наступному прикладі система розв'язків не має, графіками рівнянь цієї системи будуть паралельні прямі.

Малюнок 2.3

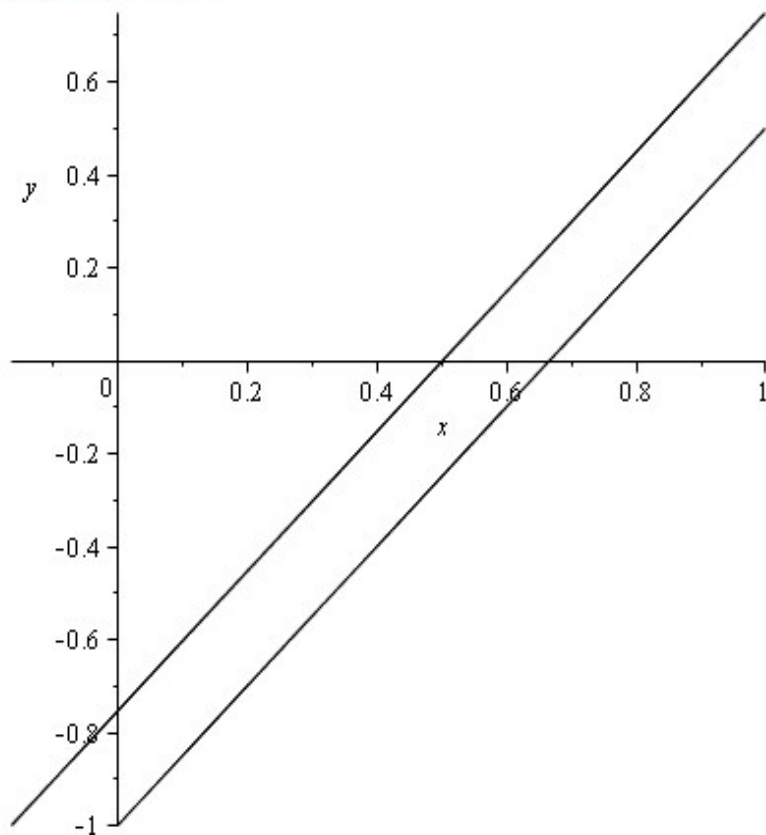
```
> with(plots):  
> sys:={3*x-2*y=2, 6*x-4*y=3};
```

$$\text{sys} := \{3x - 2y = 2, 6x - 4y = 3\}$$

```
> solve(sys, [x,y]);
```

$$[]$$

```
> implicitplot(sys, x=-1...1, y=-1...1, color=black);
```



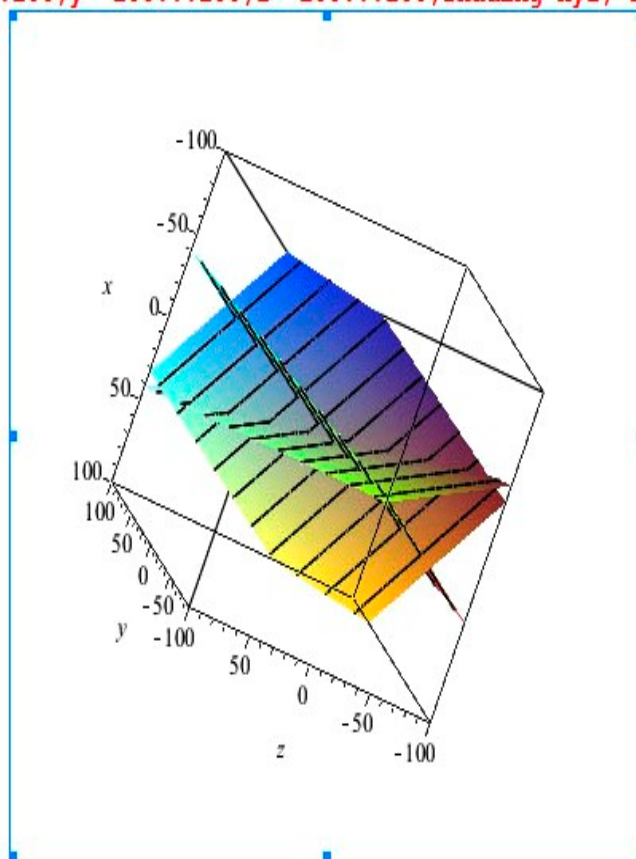
Приклад розв'язку системи з трьох лінійних рівнянь з графічною ілюстрацією розв'язку:

малюнок 2.4

```
> sys:={7*x-3*y=-4,6*x-5*y+3*z+1=0,5*x+2*z+3=0};
sys:={7x-3y=-4,5x+2z+3=0,6x-5y+3z+1=0}
> solve(sys,[x,y,z]);

$$\left[ \left[ x = -\frac{61}{79}, y = -\frac{37}{79}, z = \frac{34}{79} \right] \right]$$

> display(implicitplot3d(sys,x=-100...100,y=-100...100,z=-100...100,shading=xyz, style=patchcontour));
```



Пакет розв'язків задач лінійної алгебри linalg

В ядро Maple розробники ввели мінімально необхідні засоби для розв'язку задач лінійної алгебри. В основному, для їх реалізації використовується пакет розв'язування задач лінійної алгебри **Linalg**. Це один з потужних та об'ємних пакетів, він містить більш ніж сто функцій.

Малюнок 2.5

```
> with(linalg);
[BlockDiagonal, GramSchmidt, JordanBlock, LUdecomp, QRdecomp, Wronskian, addcol, addrow, adj, adjoint, angle, augment, backsub, band, basis, bezout, blockmatrix, charmat, (1)
charpoly, cholesky, col, coldim, colspace, colspan, companion, concat, cond, copyinto, crossprod, curl, definite, delcols, delrows, det, diag, diverge, dotprod, eigenvals,
eigenvalues, eigenvectors, eigenvecs, entermatrix, equal, exponential, extend, ffgausselim, fibonacci, forwardsub, frobenius, gausselim, gaussjord, genegns, genmatrix, grad,
hadamard, hermite, hessian, hilbert, htranspose, ihermite, indexfunc, innerprod, intbasis, inverse, ismith, issimilar, iszero, jacobian, jordan, kernel, laplacian, leastsqrs, linsolve,
matadd, matrix, minor, minpoly, mulcol, mulrow, multiply, norm, normalize, nullspace, orthog, permanent, pivot, potential, randmatrix, randvector, rank, ratform, row, rowdim,
rowspan, rowspan, rref, scalarmul, singularvals, smith, stackmatrix, submatrix, subvector, sumbasis, swapcol, swaprow, sylveste, toeplitz, trace, transpose, vandermonde,
vecpotent, vecdim, vector, wronskian]
```

Розглянемо деякі з цих функцій більш докладно.

Інтерактивне введення матриць

Для інтерактивного введення матриць можна використовувати функцію **entermatrix**. Для цього необхідно заздалегідь визначити розмір деякого масиву.

```
> M:=array(1..3,1..3);
```

$$M := \text{array}(1 \dots 3, 1 \dots 3, [])$$

```
> entermatrix(M);
```

Після цієї команди виникає наступний діалог:

```
enter element 1,1 > 3;
```

```
enter element 1,2 > 3;
```

```
enter element 1,3 > 0;
```

```
enter element 2,1 > -1;
```

```
enter element 2,2 > 1;
```

```
enter element 2,3 > 2;
```

```
enter element 3,1 > -3;
```

```
enter element 3,2 > 0;
```

```
enter element 3,3 > 4;
```

$$\begin{bmatrix} 3 & 3 & 0 \\ -1 & 1 & 2 \\ -3 & 0 & 4 \end{bmatrix}$$

```
> M:= (%);
```

$$M = \begin{bmatrix} 3 & 3 & 0 \\ -1 & 1 & 2 \\ -3 & 0 & 4 \end{bmatrix}$$

```
> A [1,1];
```

3

```
>A [2,2];
```

1

```
>A [3,3];
```

4

Основні функції для завдання векторів та матриць

В бібліотечному файлі Linalg існують наступні функції для завдання векторів та матриць:

- **vector(n,list)** - утворює вектор з n елементами, що задані в списку list;
- **matrix(n,m,list)** - утворює матрицю з кількістю рядків n і стовпців m з елементами зі списку list.

Малюнок 2.6

> A:=vector(5,[1,5,0,23,14]);	$A := \begin{bmatrix} 1 & 5 & 0 & 23 & 14 \end{bmatrix}$	(1)
> B:=matrix(3,2,[1,0,91,4,7]);	$B := \begin{bmatrix} 1 & 0 \\ 91 & 4 \\ 7 & B_{3,2} \end{bmatrix}$	(2)
> A[5];	14	(3)
> B[2,1];	91	(4)
> B[3,2];	$B_{3,2}$	(5)

Основні операції із структурою окремого вектора V та матриці M

- **coldim(A)** — повертає число стовпців матриці A;
- **rowdim(A)** — повертає число рядків матриці A;
- **vectdim(V)** — повертає вимірність V;
- **col(A,i)** — повертає i-й стовпчик матриці A;
- **row(A,i)** — повертає i-й рядок матриці A;
- **minor(A,i, j)** — повертає мінор матриці A для елемента з ідексами i та j;
- **delcols(A,i.. j)** — видаляє стовпці матриці A від i-го до j-го;
- **delrows (A,i..j)** — видаляє рядки матриці A від i-го до j-й;
- **extend (A, m, n,x)** — расширяє матрицю A на m рядків та n стовпців з використанням заповнювача x.

Основні векторні та матричні операції

- **dotprod(A, B)** — повертає скалярний добуток векторів A та B;
- **crossprod(A, B)** — повертає векторний добуток векторів A та B;
- **concat(A1,A2)** — повертає об'єднану матрицю з горизонтальним злиттям матриць A1 та A2;
- **stack(A1,A2)** — повертає об'єднану матрицю з вертикальним злиттям матриць A1 та A2;
- **matadd(A,B)** та **evalm(A+B)** — повертає суму матриць A и B;
- **multiply(A,B)** и **evalm(A*B)** — повертає добуток матриць A и B;
- **evalm(C)** - деякій вказаній матриці повертає значення матриці C;
- **adjoint (M)** или **adj(M)** — повертає приєднану матрицю, таку що $\text{multiply}(M, \text{adj}(M)) = A$, причому A - діагональна, а $A_{ii} = \det(M)$;
- **Charpoly(M, x)** — повертає характеристичний поліном матриці M відносно заданої змінної x;
- **det(M)** — повертає визначник матриці M;
- **Eigenvals(M,vecs)** — повертає власні значення матриці M та (якщо вказаний параметр vecs) відповідні власні вектори;
- **trace(A)** — повертає слід матриці A;
- **rank(A)** — повертає ранг матриці A;
- **transpose(A)** — повертає транспоновану матрицю A;
- **inverse(A)** или **evalm(I/A)** — повертає матрицю, обернену до A;
- **copyinto(A,B,i, j)** — копіює матрицю A в B для елементів послідовно від i до j;

Приклади:

малюнок 2.7

<code>> A:=matrix(3,3,[1,5,0,2,3,1,4,7,5]);</code>	$A := \begin{bmatrix} 1 & 5 & 0 \\ 2 & 3 & 1 \\ 4 & 7 & 5 \end{bmatrix}$
<code>> B:=matrix(3,2,[1,0,91,4,7,9]);</code>	$B := \begin{bmatrix} 1 & 0 \\ 91 & 4 \\ 7 & 9 \end{bmatrix}$
<code>> evalm(A*B);</code>	$\begin{bmatrix} 456 & 20 \\ 282 & 21 \\ 676 & 73 \end{bmatrix}$
<code>> trace(A);</code>	9
<code>> rank(A);</code>	3
<code>> transpose(A);</code>	$\begin{bmatrix} 1 & 2 & 4 \\ 5 & 3 & 7 \\ 0 & 1 & 5 \end{bmatrix}$
<code>> adj(A);</code>	$\begin{bmatrix} 8 & -25 & 5 \\ -6 & 5 & -1 \\ 2 & 13 & -7 \end{bmatrix}$

малюнок 2.8

<code>> C:=multiply(A,adj(A));</code>	$C := \begin{bmatrix} -22 & 0 & 0 \\ 0 & -22 & 0 \\ 0 & 0 & -22 \end{bmatrix}$
<code>> det(A);</code>	-22
<code>> minor(B,3,2);</code>	$\begin{bmatrix} 1 \\ 91 \end{bmatrix}$
<code>> Charpoly(A,x) mod 5;</code>	$2 + x^3 + x^2 + x$
<code>> evalf(Eigenvals(A,vecs));</code>	$\begin{bmatrix} 4.030627507 & -1.259652896 & 0.4333104150 \end{bmatrix}$

Розв'язування систем лінійних рівнянь

Приклад 1. Розв'язати систему лінійних рівнянь методом Крамера (дивись опис метода Крамера в вікіпедії або в методичці Визначник квадратної матриці, стор.37)

$$\begin{cases} 2x + 3y + 5z = 10, \\ 3x + 7y + 4z = 3, \\ x + 2y + 2z = 3. \end{cases}$$

Далі надається розв'язок системи за допомогою функцій з пакету Linalg.

малюнок 2.9-2.10

<pre>> with(linalg): > C:=matrix(3,3,[[2,3,5],[3,7,4],[1,2,2]]);</pre>	$C := \begin{bmatrix} 2 & 3 & 5 \\ 3 & 7 & 4 \\ 1 & 2 & 2 \end{bmatrix}$
<pre>> B:=matrix(3,1,[10,3,3]);</pre>	$B := \begin{bmatrix} 10 \\ 3 \\ 3 \end{bmatrix}$
<pre>> A:=evalm(C);</pre>	$A := \begin{bmatrix} 2 & 3 & 5 \\ 3 & 7 & 4 \\ 1 & 2 & 2 \end{bmatrix}$
<pre>> A1:=copyinto(B,C,1,1);</pre>	$A1 := \begin{bmatrix} 10 & 3 & 5 \\ 3 & 7 & 4 \\ 3 & 2 & 2 \end{bmatrix}$
<pre>> C:=evalm(A):A2:=copyinto(B,C,1,2);</pre>	$A2 := \begin{bmatrix} 2 & 10 & 5 \\ 3 & 3 & 4 \\ 1 & 3 & 2 \end{bmatrix}$
<pre>> C:=evalm(A):A3:=copyinto(B,C,1,3);</pre>	$A3 := \begin{bmatrix} 2 & 3 & 10 \\ 3 & 7 & 3 \\ 1 & 2 & 3 \end{bmatrix}$
<pre>> x1:=det(A1)/det(A);</pre>	$x1 := 3$
<pre>> x2:=det(A2)/det(A);</pre>	$x2 := -2$
<pre>> x3:=det(A3)/det(A);</pre>	$x3 := 2$

А з використанням команди **linsolve** це виглядатиме так:

малюнок 2.11

```
> with(linalg):
> C:=matrix(3,3,[[2,3,5],[3,7,4],[1,2,2]]);
```

$$C := \begin{bmatrix} 2 & 3 & 5 \\ 3 & 7 & 4 \\ 1 & 2 & 2 \end{bmatrix}$$

```
> B:=matrix(3,1,[10,3,3]);
```

$$B := \begin{bmatrix} 10 \\ 3 \\ 3 \end{bmatrix}$$

```
> X:=linsolve(C,B);
```

$$X := \begin{bmatrix} 3 \\ -2 \\ 2 \end{bmatrix}$$

В наступному прикладі показаний розв'язок системи $\begin{cases} ax + by = e \\ cx + dy = f \end{cases}$ в символьному вигляді:

малюнок 2.12

```
> with(linalg):
> C:=matrix(2,2,[[a,b],[c,d]]);
```

$$C := \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

```
> B:=matrix(2,1,[e,f]);
```

$$B := \begin{bmatrix} e \\ f \end{bmatrix}$$

```
> X:=linsolve(C,B);
```

$$X := \begin{bmatrix} -\frac{bf-ed}{ad-cb} \\ \frac{af-ec}{ad-cb} \end{bmatrix}$$

В наступному прикладі використовується функція множення матриць та обчислення оберненої матриці для знаходження розв'язків:

$$A\vec{x} = \vec{b}, \vec{x} = A^{-1}\vec{b}$$

Малюнок 2.13

```
> with(linalg):  
> A:=matrix(3,3,[[5,2,3],[2,-2,5],[3,4,2]]);
```

$$A := \begin{bmatrix} 5 & 2 & 3 \\ 2 & -2 & 5 \\ 3 & 4 & 2 \end{bmatrix}$$

```
> B:=matrix(3,1,[-2,0,-10]);
```

$$B := \begin{bmatrix} -2 \\ 0 \\ -10 \end{bmatrix}$$

```
> X:=multiply(inverse(A),B);
```

$$X := \begin{bmatrix} 2 \\ -3 \\ -2 \end{bmatrix}$$

```
> A:=evalm(C);
```

Остання зміна: понеділок, 15 грудень 2014 13:33

УСТАНОВКИ



Керування модулем сторінок

- Редагувати параметри
- Фільтри
- Події
- Резервна копія
- Відновлення

Керування курсом

Перемикнути на роль...

Мій профіль

Центр електронного навчання ХНУ (Харківський національний університет) імені В.Н. Каразіна © 2011 – 2015

Харківський національний університет імені В. Н. Каразіна

❗ Документація для цієї сторінки

Ми у соціальних мережах

