Algebra Symboliczna
Wykład XII

Andrzej Odrzywolek

Instytut Fizyki, Zakład Teorii Względności i Astrofizyki

10.10.2007, środa, 13:15

Dane kontaktowe

dr Andrzej Odrzywołek

pokój 447, IV piętro

E-mail: `odrzywolek@th.if.uj.edu.pl`

Wykład: środy 13.15-15.00 s. 128

Ćwiczenia: piątki 10.30-12.00

Konsultacje: środy ∼11-13, czwartki 10-12

WWW: http://ribes.if.uj.edu.pl/alsymb/

Maple - informacje podstawowe

Notacja „tradycyjna”	Notacja **Maple**
$\sin^2 x$	$\sin(x)^2$
$\sin^{-1} x$	$\arcsin(x)$ czy $\sin(x)\hat{}(-1)$
$\dfrac{1}{\sin x}$	$1/\sin(x)$
$2x+3$	$2*x+1$
$\mathrm{tg}\,2x+1/\mathrm{tg}\,3x-1$	$\mathrm{tg}(2*x+1)/\mathrm{tg}(3*x-1)$

Notacja „tradycyjna”	Notacja **CAS**
π	Pi
$\imath, i, \jmath, j$	I
e	$\exp(1)$

Najprostsze obliczenia

Maple jako kalkulator

Wykonajmy działanie:

$$\frac{1024}{16} + 128 \cdot 8 - 2^{16}$$

Po uruchomieniu dysponujemy otwartym oknem z tzw. „Maple prompt”:

$>$

gdzie wpisujemy nasze komendy, kończąc je średnikiem (;)

$> 1024/16 + 128 * 8 - 2\hat{}16;$

naciśnięcie klawisza **Enter** daje wynik:

-64448

Uwaga! *Dokładnie na odwrót niż w* MATHEMATICE !

- naciśnięcie **Enter** powoduje obliczenie wyrażenia
- naciśnięcie **Shift+Enter** powoduje przejście do następnej linii

Wartość numeryczna wyrażeń

Obliczanie wartości numerycznej można wymusić dodając kropkę rozwinięcia dziesiętnego:

$> 1024.0/16 + 128 * 8 - 2\hat{\ }16;$

-64448.00000 Nie zadziała to jeżeli w wyrażeniu są składniki „symboliczne”:

$> 1024.0/16 + 128 * \mathit{sin}(1) - 2\hat{\ }16;$

$-65472.00000 + 128\sin(1)$ MAPLE „stara się” zachować „symboliczną” informację w postaci funkcji $\sin(1)$.

Możemy jednak zażądać obliczenia wartośći numerycznej *explicite* za pomocę funkcji evalf:

$> \mathit{evalf}(1024.0/16 + 128 * \mathit{sin}(1) - 2\hat{\ }16);$

-65364.29171

Funkcja evalf

Instrukcja evalf służy do obliczania numerycznej wartości wyrażeń oraz wywoływania numerycznych metod np. całkowania

Praca interaktywna

Odwoływanie się do wcześniejszych wyników

- Operator % wskazuje na ostatni obliczony wynik.
- Operator %% wskazuje na przedostatni obliczony wynik.
- Operator %%% wskazuje na wynik obliczony 3 kroki wstecz.

Przykład:

$> 10!;$

$$3628800$$

$> \mathit{evalf}(\%)$

$$0.3628800$$

$> \mathit{ifactor}(\%\%)$

$$(2)^8(3)^4(5)^2(7)$$

Uwaga:

- Operatory składające się z większej ilości % niż 3 nie istnieją!

Całkowanie

Funkcja obliczająca wartość całki jest dobrym przykładem pokazującym ogólne zasady obowiązujące w MAPLE.
Nazwa funkcji: integrate lub krócej int.

Przykłady

- Całka nieoznaczona funkcji

$$\int f(x)\, dx$$

```
> int( f(x), x );
```

- Całka oznaczona funkcji

$$\int_a^b f(x)\, dx$$

```
> int( f(x) , x = a..b );
```

Bardzo częsty schemat w MAPLE

instrukcja(funkcja, zmienna=początek..koniec);

Np: przedział na osi liczb rzeczywistych $(0, 1)$ zapiszemy w Maple jako 0..1, a ciąg liczb naturalnych $0, 1, 2, \ldots, 100$ jako 0..100.

Duża ilość funkcji uzywa tego schematu np: int, sum, plot, seq, ...

UWAGA:

Zapis a..b może dotyczyć również innych uporządkowanych zmiennych np. funkcja seq(i, i="a".."z"); wygeneruje alfabet.

Funkcje „bezwładne”

Wiele funkcji MAPLE wystepuje w dwóch formach: „aktywnej” i „bezwładnej” *ang. inert function*.
Podczas gdy funkcja int (z małej litery) oblicza całkę
– jej bezwładna wersja Int (z dużej litery) pozostawia ja nieobliczoną.

Przykład:

```
> int((x-1)/ln(abs(x)),x=0..1);
```

$$\ln(2)$$

```
> Int((x-1)/ln(abs(x)),x=0..1);
```

$$\int_0^1 \frac{x-1}{\ln(|x|)}\,dx$$

Działanie funkcji „bezwładnej” wygląda jak gdyby MAPLE nie umiał obliczyć całki:
> int(abs(x)^ x, x=0..1);

$$\int (|x|)^x \, dx$$

Przykłady funkcji posiadających wersję bezwładną (w nawiasie):

- int (Int) – całkowanie
- diff (Diff) – rózniczkowanie
- limit (Limit) – granica
- product (Product) – iloczyn
- sum (Sum) – suma

Zastosowanie funkcji „bezwładnych”

Zapis z którego bardzo jasno wynika co liczymy i jaki jest wynik:

```
> Int(exp(-x)*ln(x),x=0..infinity)=int(exp(-x)*ln(x),x=0..infinity);
```

$$\int_0^\infty e^{-x} \ln(x)\, dx = -\gamma$$

Obliczenia numeryczno/analityczne:

```
> evalf(int(ln(tan(x)),x=0..Pi/2));
```

$$0.$$

Obliczenia czysto numeryczne:

```
> evalf(Int(ln(tan(x)),x=0..Pi/2));
```

$$0.7508019027\ 10^{-12}$$

Wykresy funkcji

Wykres funkcji $y = f(x)$ w przedziale (a, b): plot(f(x),x=a..b);
Kilka funkcji równocześnie: plot([f1(x),f2(x),f3(x)],x=a..b);
W postaci parametrycznej $x = x(t), y = y(t)$:
plot([x(t),y(t),t=t1..t2]);
Podstawowe opcje:

- Zakres na osi pionowej jako trzeci argument: y_min..y_max lub jako jeden z dalszych argumentow jako: view=y_min..y_max
- Kolor linii: color=nazwa_coloru lub color=COLOR(RGB,0.2,0.3,0.9)
- Więcej opcji: > ?plot,options

Wykres funkcji dwóch zmiennych $F(x, y)$:
plot3d(F(x,y),x=-x_min..x_max,y=y_min..y_max);
Cały wykład na temat grafiki w Maple na ribes.if.uj.edu.pl/alsymb

Upraszczanie i przekształcanie wyrażeń

- zapisanie wyrażenia do zmiennej:
  > w:=expand($(x+a)$^7);
- upraszczanie wyrażeń:
  > simplify(w);
- rozkład na czynniki:
  > factor(w);
- podstawienie wyrażenia:
  > subs(x=-2*a+y, w);factor(%);

Rozwiązywanie równań i układów równań

Rozwiązania analityczne

Równanie:

$>$ solve(exp(x)=1/x^2,x);

$$2\,\mathrm{LambertW}\left(\frac{1}{2}\right),2\,\mathrm{LambertW}\left(\frac{-1}{2}\right)$$

Układ równań:

$> \mathrm{solve}(\{x\hat{}2+y\hat{}2=1,x+y=0\},\{x,y\});$

$>$ convert(%,radical);

$$\{x=-\frac{\sqrt{2}}{2},y=\frac{\sqrt{2}}{2}\}$$

Rozwiązania numeryczne

Równanie:

$>$ fsolve(exp(x)=1/x^2,x=1);

0.7034674225

Układ równań:

$> \mathrm{fsolve}(\{x\hat{}2+y\hat{}2=1,x+y=0\},\{x,y\});$

$$\{y=-0.7071067812,x=0.7071067812\}$$

Rozwiązywanie równań różniczkowych

Równania różniczowe zwyczajne rozwiązujemy komendą dsolve.
Najprostszy przykład:

$$\frac{dy}{dx} = e^x$$

> dsolve(diff(y(x),x)=exp(x));

$$y(x) = e^x + \_C1$$

Rozwiazanie zawiera jedną stałą dowolną _C1.
Aby uzyskać jednoznaczne rozwiązanie możemy np. zarządać aby rozwiązanie przechodziło przez zadany punkt $P(x_0, y_0)$:

> dsolve({ diff(y(x),x)=exp(x), y(x0)=y0 });

$$y(x) = e^x - e^{x0} + y0$$

1. zapisujemy równania jako EQ_x,EQ_y, EQ_z:
   EQ_x:=m*diff(x(t),t,t)=0;
   EQ_y:=m*diff(y(t),t,t)=0;
   EQ_z:=m*diff(z(t),t,t)=- m * g;
2. Układ równań stanowi zbiór pojedynczych równań:
   SYS: = { EQ_x,EQ_y,EQ_z };
3. Podajemy warunki początkowe:
   Ics:= { x(0)=0,y(0)=0,z(0)=0,
   D(x)(0)=V0*cos(alpha), D(y)(0)=0,D(z)(0)=V0*sin(alpha)
   };
4. Pełny opis problemu jest zawarty w sumie zbiorów (union) zawierajacych równania i warunki początkowe: SYS union Ics
5. Funkcja dsolve rozwiązuje zadany problem:
   dsolve(SYS union Ics, [x(t),y(t),z(t)]); i otrzymujemy znany wynik:

$$\{\ x(t) = V0 * \cos(\alpha) * t, y(t) = 0, z(t) = -1/2 * g * t\hat{\ }2 + V0 * \sin(\alpha) * t\}$$

- Numeryczne rozwiązanie równania rózniczkowego otrzymujemy poprzez dodanie opcji typy=numeric do dsolve
- wynikiem działania dsolve jest wtedy *procedura numeryczna* wyliczająca rozwiazanie równania różniczkowego

Przykład

> rozw:= dsolve({ diff(y(t),t,t)=sin(t)/t, y(0)=0, D(y)(0)=1 }, type=numeric, output=listprocedure);

rozw:=[t=(**proc**(t) . . . **end proc**), y(t)=(**proc**(t) . . . **end proc**), $\frac{d}{dt}$ y(t)=(**proc**(t) . . . **end proc**)]

s:=subs(rozw,y(t));

s:=**proc**(t) . . . **end proc**

plot(s(x),x=-10..10); lub plot(s); lub plot(s,-1..1,-1..1);

Równania różniczkowe *cząstkowe* rozwiązujemy instrukcją pdsolve.

Przykład

- Szukamy rozwiązania ogólnego równania:

$$\frac{\partial F(x,y)}{\partial y} + x\,\frac{\partial F(x,y)}{\partial x} = 0$$

```
> pdsolve(diff(F(x,y),y)+x*diff(F(x,y),x), F(x,y));
```

$$\mathrm{F(x,y) = \_F1(y\text{-}ln(x))}$$

Wynik ten oznacza, że szukanym rozwiazaniem równania jest dowolna funkcja (_F1) argumentu $y - ln(x)$.

Szeregi, granice

Rozwinięcie w szereg

- Szereg potęgowy w otoczeniu $x = 0$:
  ```
  > series(AiryAi(x),x,1);
  ```
- Szereg asymptotyczny (w otoczeniu ∞) :
  ```
  > asympt(AiryAi(x),x,1);
  ```

Granica

Granica w nieskończoności:

```
> limit(AiryAi(x),x=infinity);
```

Oznaczając:

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

Macierz konstruujemy poleceniem Matrix, np. zdefiniowaną poprzednio macierz A:

A:=Matrix([[1,1,1,1],[2,4,0,-1],[1,-1,0,2],[1,1,-1,-1]]);

Jest wiele sposobów podania „zawartości" tej macierzy:

A:=Matrix($<< 1, 2, 1, 1 > | < 1, 4, -1, 1 > | < 1, 0, 0, -1 > | < 1, -1, 2, -1 >>$);

A:=Matrix($<< 1|1|1|1 >, < 2|4|0| - 1 >, < 1| - 1|0|2 >, < 1|1| - 1| - 1 >>$);

pierwszy opisuje składanie z kolumn, drugi składanie z wierszy.

Wektory macierzowe

Specjalnie wyróżnione są macierze o wymiarze $1 \times N$ lub $N \times 1$, czyli wektory. Oto kilka przykładów:

Wektor kolumnowy (pionowy) B:

B:=Vector([1,2,0,-1]);

B:=Vector($< 1, 2, 0, -1 >$);

B:=Vector[column]($< 1|2|0|-1 >$);

Jeżeli nie podamy jednej z opcji [column], [row] to zostanie utworzony wektor kolumnowy w przypadku 1 i 2 (rozdzielamy przecinkiem) , natomiast wierszowy (poziomy – rozdzielamy pionową kreską) w przypadku 3.

Podstawowe operacje

Operacje mnożenia przez liczbę oraz dodawania (odejmowania) macierzy wykonujemy uzywając standardowych symboli $+$, $-$, $*$ W prosty sposób możemy również obliczyć n-tą potęgę macierzy kwadratowej ($N \times N$) jako A^n oraz macierz odwrotną A^(-1).

Przykład

$$3 \cdot A^2 - A + A^{-1}$$

obliczamy po prostu jako:
$3 * A^\wedge 2 - A + A^\wedge(-1)$

Operator *mnożenia macierzy*: kropka

Operator ten wykonuje mnożenie elementów (przede wszystkim macierzy) w sytuacji kiedy jest to możliwe, np. mnożenie $A \cdot B$ wykonujemy przez A.B gdy liczba kolumn A jest równa liczbie wierszy B.

Typowe zadania związane z macierzami

Bez problemu wykonamy wszystkie typowe obliczenia związane z macierzami np:

- Wyznaczenie wielomianu charakterystycznego z definicji: Determinant(A-x . IdentityMatrix(4)); lub korzystając z odpowiedniej instrukcji CharacteristicPolynomial(A,x);
- Wartości własne macierzy A możemy obliczyć znajdując pierwiaski wielomianu charakterystycznego W: fsolve(W,x,complex); lub funkcją Eigenvalues(A)
- Wektory własne: Eigenvectors(A)
- Sprowadzenie do postaci diagonalnej: JordanForm(A)

Pakiet VectorCalculus

Pakiet ładujemy instrukcją:
with(VectorCalculus);
Pierwszą czynnością powinno być zdefiniowanie układu współrzędnych (lista ukł. wsp: > ?VectorCalculus,Coordinates), np. zwykły kartezjański układ (x, y, z) definiujemy przez:
SetCoordinates('cartesian'[x,y,z]);
Teraz możemy zdefiniować pole wektorowe jako:
V:=VectorField(<Vx(x,y,z),Vy(x,y,z),Vz(x,y,z)>);

Iloczyn wektorowy i skalarny

- Iloczyn skalarny wektorów obliczamy jako DotProduct(V1,V2): lub korzystając z operatora . (kropka). Np. długość wektora V to sqrt(V.V)
- Iloczyn wektorowy obliczamy jako CrossProduct(V1,V2) lub za pomocą operatora &x jako V &x V

Operator nabla ∇

Operacje na polach wektorowych

Podstawowe operacje wektorowe obliczamy w następujący sposób:

- Dywergencja:
  Divergence(V); lub Nabla.V;
- Rotacja:
  Curl(V); lub Nabla &x V ;
- Gradient:
  Gradient(f(x,y,z)); lub Nabla(f(x,y,z));
- Laplacian:
  Laplacian(f(x,y,z)); lub Nabla.Nabla(f(x,y,z));

Pierszy ze sposobów używa odpowiedniej funkcji MAPLE, drugi oblicza odpowiednie wyrażenia z operatorem wektorowym Nabla (inaczej Del):

$$\nabla \equiv \frac{\partial}{\partial x}\mathbf{e}_x + \frac{\partial}{\partial y}\mathbf{e}_y + \frac{\partial}{\partial z}\mathbf{e}_z$$

Przykład: obracająca się czarna dziura

$$g_{ik} = \begin{bmatrix} \frac{\Delta - a^2 \sin^2\theta}{\rho^2} & 0 & 2a\frac{Mr\sin^2\theta}{\rho^2} & 0 \\ 0 & -\frac{\rho^2}{\Delta} & 0 & 0 \\ 2a\frac{Mr\sin^2\theta}{\rho^2} & 0 & -\frac{(r^2+a^2)^2 - a^2\Delta\sin^2\theta}{\rho^2}\sin^2\theta & 0 \\ 0 & 0 & 0 & \rho^2 \end{bmatrix}$$

$$\Delta = r^2 - 2Mr + a^2, \quad \rho^2 = r^2 + a^2\cos^2\theta$$

gdzie M to masa czarnej dziury, $a = J/M$, J – jej moment pędu.

„W literaturze nie ma konstruktywnego i analitycznego wyprowadzenia metryki [Kerra] które by odzwierciedlało jej sens fizyczny, a nawet proste sprawdzenie tego rozwiązania równań Einsteina jest związane z ogromnymi obliczeniami. Landau & Lifszyc „Teoria pola", 1972.

Zobaczmy ile czasu MAPLE potrzebuje na wykonanie tych „ogromnych obliczeń".