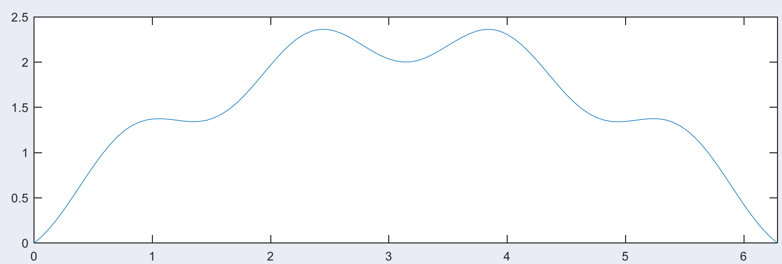
Programowanie w zastosowaniach inżynierskich

Obliczenia symboliczne

Obliczenia numeryczne i symboliczne

Obliczenia numeryczne – metody rozwiązywania problemów matematycznych wykorzystujące wyłącznie działania na liczbach. Uzyskiwane wyniki na ogół są przybliżone, jednak zazwyczaj dokładność obliczeń może być dobrana do potrzeb.

Obliczenia symboliczne – metody rozwiązywania problemów matematycznych wykorzystujące operacje na symbolach. Uzyskiwane rozwiązania mają postać równań lub funkcji opisujących dokładne rozwiązanie problemu.

Rozwiązanie numeryczne	*Rozwiązanie symboliczne*
równanie algebraiczne $x^2 + 2x + a = 0$ x niewiadoma, a parametr	
dla $a = 5$: $x_1 \approx -3{,}44949$, $x_2 \approx 1{,}44949$	$x = \pm\sqrt[2]{1-a} - 1$
równanie różniczkowe $\dot{y}(t) = sin(at) + cos(bt), \quad y(0) = 0$ $y(t)$ niewiadoma funkcja, a, b parametry	
dla $a = 4$ i $b = 0.5$ w czasie $t \in [0, 2\pi]$ 	$y(t) = \frac{1}{a} - \frac{\cos(at)}{a} + \frac{\sin(bt)}{b}$

Symbolic Math Toolbox udostępnia zestaw funkcji do przeprowadzania obliczeń symbolicznych. Najważniejsze zastosowania:

- algebra liniowa,
- rachunek różniczkowy i całkowy,
- rozwiązywanie równań algebraicznych i różniczkowych,
- upraszczanie i transformacja wyrażeń matematycznych.

Tworzenie obiektów symbolicznych

Instrukcja	*Przykład*	*Opis*
`syms` *`n1 n2 … nN`*	`syms a b c`	Tworzy skalarne zmienne symboliczne o nazwach określonych przez *`n1, n2`*, …, *`nN`*
`syms n1 n2 … nN set`	`syms q w real`	Tworzy skalarne zmienne symboliczne przypisując im ograniczenia wyspecyfikowane przez *`set`*, dozwolone wartości: `real, positive, integer, rational`
`syms` *`fun`*`(`*`v1,v2,…,vN`*`)`	`syms f(x,y)`	Tworzy symboliczną funkcję *`fun`* oraz symboliczne zmienne *`v1`*, *`v2`*, …, *`vN`* reprezentujące jej argumenty
`n` `= sym('`*`n`*`')`	`x=sym('x')`	Tworzy skalarną zmienną symboliczną o nazwie *`n`*
`n` `= sym('`*`n`*`',` *`set`*`)`	`q=sym('q',real)`	Tworzy skalarną zmienną symboliczną z ograniczeniami
`v = sym(`*`num`*`)`	`f=sym(1/7)`	Konwertuje wartość numeryczną *`num`* na symboliczną wartość numeryczną

Wyrażenie symboliczne jest dowolnym wyrażeniem zawierającym zmienne symboliczne. Do budowy takich wyrażeń mogą być wykorzystane standardowe operatory i wbudowane funkcje Matlab-a.

Manipulowanie wyrażeniami symbolicznymi

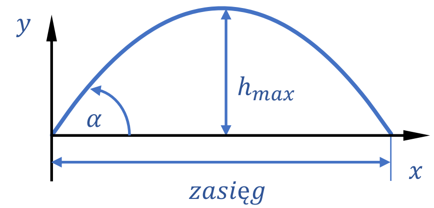
Funkcja	*Przykład*	*Opis*
`collect(`*exp*`,`*v*`)`	`collect((x-5)^2+(y-2)^2,x)` `x^2-10*x+(y-2)^2+25`	Grupuje współczynniki zmiennej `v`
`expand(`*exp*`)`	`expand(sin(x+y))` `cos(x)*sin(y)+cos(y)*sin(x)`	Rozwija wyrażenie `exp`
`fplot(`*exp*`,`*r*`)`	`fplot(2*x^2+5, [-3,3])`	Wykreśla funkcję jednej zmiennej opisanej przez `exp` w przedziale `r` (dwuelementowy wektor)
`poly2sym(`*c*`,`*var*`)`	`poly2sym([2,3,0,1], y)` `2*y^3 + 3*y^2 + 1`	Tworzy wielomian zmiennej `var` o współczynnikach określonych wektorem `c`
`simplify(`*exp*`)`	`simplify((x^2+5*x+6)/(x+2))` `x+3`	Upraszcza wyrażenie *exp*
`subs(`*exp*`,`*v1*`,`*v2*`)`	`subs(1/2+sin(y), y, pi/4)` `2^(1/2)/2 + 1/2`	Podstawia za zmienną symboliczną `v1` wartość `v2` (obiekt symboliczny, liczba) w wyrażeniu *exp*
`vpa(`*exp*`,`*d*`)`	`vpa(sin(x)+1/7, 5)` `sin(x) + 0.14286`	Wyznacza wartości stałych symbolicznych z dokładnością do d miejsc po kropce

Rozwiązywanie równań algebraicznych

`S = solve(eqn, var)`

funkcja rozwiązuje równanie opisane wyrażeniem `eqn` ze względu na zmienną `var`.

Przykład. Rzut ukośny



$$x(t) = v_0\, t \cos\alpha$$
$$y(t) = v_0\, t \sin\alpha - \frac{g\, t^2}{2}$$

```
>> syms x(t) y(t) v0 a g

>> x(t) = v0*t*cos(a);
>> y(t) = v0*t*sin(a)-g*t^2/2;

>> T = solve(y==0,t)

   T =
           0
          (2*v0*sin(a))/g

>> T = T(2);

>> z = subs(x, t, T)

   z =
          (2*v0^2*cos(a)*sin(a))/g

>> h = subs(y, t, T/2)

   h =
          (v0^2*sin(a)^2)/(2*g)
```

```
% uproszczenie wyrażenia na "z"
>> z = simplify(z)

   z =
        (v0^2*sin(2*a))/g

% wyrażenie w czytelnej formie

>> pretty(z)
     2
   v0  sin(2 a)
   ------------
         g
```

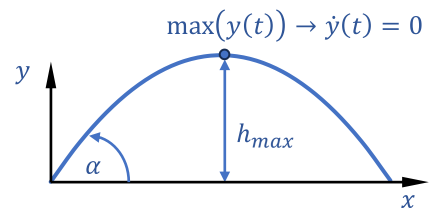
Pochodna funkcji

`Df = diff(f, var, n)`

wyznacza pochodną funkcji `f` rzędu `n` ze względu na zmienną `var`

Przykład. Rzut ukośny (alternatywne wyznaczenie h_{max})

```
>> syms y(t) v0 a g
>> y(t) = v0*t*sin(a)-g*t^2/2;
>> T = solve(diff(y,t)==0,t)
   T =
       (v0*sin(a))/g
>> h = y(T)
   h =
       (v0^2*sin(a)^2)/(2*g)
>> pretty(h)
      2        2
   v0   sin(a)
   -----------
       2 g
```



$$y(t) = v_0\, t \sin\alpha - \frac{g\, t^2}{2}$$

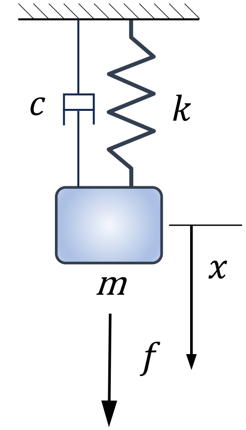
Rozwiązywanie równań różniczkowych

`S = dsolve(eqn, cond)`

Funkcja rozwiązuje równanie różniczkowe `eqn` z warunkami początkowymi `cond`

Przykład. Układ masa-sprężyna

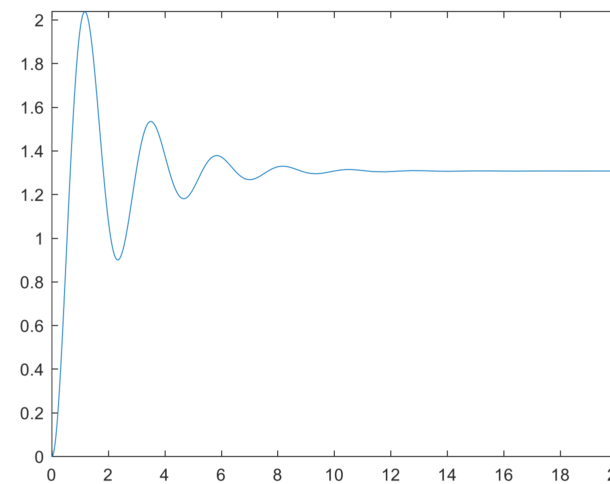
```
>> syms x(t)
>> syms m c k g positive
>> E = m*diff(x,t,2)+c*diff(x,t)+k*x == m*g
>> Dx(t) = diff(x(t),t);
>> X(t) = dsolve(E, [x(0)==0, Dx(0)==0]);
>> X(t) = subs(X,[m,c,k,g], [2,2,15,9.81]);
>> fplot(X(t),[0,20])
```



$$m\ddot{x} + c\dot{x} + kx = f(t)$$

$$f = m\,g$$

$$x = 0,\ \dot{x} = 0$$



Obliczenia symboliczne w Live Script