Lucrarea 3

.1 Calcul simbolic în Matlab

Toolbox-ul *Symbolic Math*, care utilizează nucleul Maple ca şi motor de calcul algebric, permite calcule simbolice în MATLAB.

.1.1 Variabile simbolice

Declararea unei variabile simbolice se poate face utilizând comanda *syms*. Spre exemplu, *syms x* creează variabila simbolică x. Comanda *clear all* permite ştergerea tuturor variabilelor din MATLAB şi Maple (comenzile *clear* sau *clear x* şterg variabilele din MATLAB nu şi cele din Maple).
Variabilele simbolice pot fi construite din variabile numerice existente cu ajutorul funcţiei *sym*, spre exemplu: $z = 10$; $a = sym(z)$.

.1.2 Operaţii asupra expresiilor simbolice

Diferenţiere

Funcţia $diff$ poate fi utilizată pentru a diferenţia expresii simbolice şi pentru a aproxima derivata unei funcţii numerice:
$>> syms\ x;\ diff(x\hat{}3)$
$ans =$
$3 * x\hat{}2$
$>> f = inline('x\hat{}3','x');\ diff(f(x))$
Sintaxa pentru calculul derivatei de ordinul doi este $diff(f(x), 2)$, şi pentru calculul derivatei de ordin n este $diff(f(x), n)$.
Funcţia $diff$ permite de asemenea calculul derivatelor parţiale pentru expresii cu mai multe variabile:
$>> syms\ x\ y\ z;$

$>> diff(x\hat{}2 * y, y)$
$ans =$
$x\hat{}2$
$>> diff(diff(sin(x * y/z), x), y)$
$ans =$
$cos((x * y)/z)/z - (x * y * sin((x * y)/z))/z\hat{}2$
Dacă cel de al doilea argument este omis, MATLAB alege o variabilă simbolică implicită pentru expresie, care poate fi aflată utilizând funcţia $findsym$:
$>> syms\ x\ x1\ x2$
$>> F = x * (x1 * x2 + x1 - 2)$
$>> findsym(F, 1)$
$ans =$
x
Funcţia *dsolve* permite rezolvarea ecuaţiilor diferenţiale utilizând solver-ul ODE simbolic. În cazul utilizării acestei funcţii diferenţierea se va reprezenta prin litera D. Spre exemplu, pentru rezolvarea ecuaţiei $xy' + 1 = y$, se va utiliza sintaxa:
$>> dsolve('x * Dy + 1 = y', 'x')$
$ans =$
$C2 * x + 1$

Integrare

MATLAB permite calculul integralelor definite şi nedefinite.
$>> syms\ x;\ \ int(asin(x), 0, 1)$
$ans =$
$pi/2 - 1$
$>> syms\ x;\ \ int(x\hat{}2, 'x')$
$ans =$
$x\hat{}3/3$

Limite

Funcţia *limit* este utilizată pentru a calcula limitele diverselor expresii simbolice. Spre exemplu:
$>> syms\ h\ n\ x$
$>> limit((1 + x/n)\hat{}n, n, inf)$ % calculează limita $(1 + x/n)^n,\ n \to \infty$

$>> limit(sin(x), x, 0)$ % calculează limita expresiei $sin(x)$, $x \to 0$
$>> limit((sin(x + h) - sin(x))/h, h, 0)$

Sume şi produse

Funcţiile *sum* şi *prod* permit calculul sumelor şi produselor numerice finite. Funcţia *symsum* permite realizarea de sume simbolice finite şi infinite:
$>> syms\ k\ n;\ \ symsum(k, 1, n)$ %calcul sumă $\sum_{k=1}^{n} k$
$>> syms\ a\ k;\ \ symsum(a\hat{}k, 0, Inf)$ %calcul serie geometrică infinită

Substituire numerică şi simbolică

O expresie simbolică poate fi modificată sau evaluată numeric utilizând funcţia *subs*. Funcţia are rolul de a înlocui toate apariţiile variabilei simbolice într-o expresie cu o altă expresie specificată. Dacă toate variabilele expresiei sunt înlocuite cu valori numerice rezultatul va fi numeric.

```
>> syms x s t
>> subs(sin(x),x,pi/3)
ans =
0.8660
>> subs(sin(x),x,sym(pi)/3)
ans =
3^(1/2)/2
double(ans)
ans =
0.8660
>> subs(2*t^2/2,t,sqrt(2*s))
ans =
2*s
>> subs(sqrt(1-x^2),x,cos(x))
ans =
(1-cos(x)^2)^(1/2)
>> subs(sqrt(1-x^2),1-x^2,cos(x))
ans =
cos(x)^(1/2)
```

Simplificări algebrice

Funcţia *expand* extinde simbolic polinoamele şi funcţiile elementare, în timp ce funcţia *factor* funcţionează invers. Funcţia *collect* vede o expresie simbolică ca fiind un polinom în variabila sa simbolică (care poate fi specificată) şi colectează toţi termenii care au aceeaşi putere cu variabila.

```
>> syms a b x y z
>> expand((a + b)^5)
ans =
a^5 + 5 * a^4 * b + 10 * a^3 * b^2 + 10 * a^2 * b^3 + 5 * a * b^4 + b^5
>> factor(ans)
ans =
(a + b)^5
>> expand(exp(x + y))
ans =
exp(x) * exp(y)
>> expand(sin(x + y))
ans =
cos(x) * sin(y) + cos(y) * sin(x)
>> factor(x^3 - 1)
ans =
(x - 1) * (x^2 + x + 1)
>> collect(x * (x * (x + 3) + 5) + 1)
ans =
x^3 + 3 * x^2 + 5 * x + 1
>> horner(ans)
ans =
x * (x * (x + 3) + 5) + 1
>> collect((x + y + z) * (x - y - z))
ans =
x^2 - (y + z)^2
>> collect((x + y + z) * (x - y - z), y)
ans =
- y^2 + (-2 * z) * y + (x + z) * (x - z)
>> collect((x + y + z) * (x - y - z), z)
ans =
- z^2 + (-2 * y) * z + (x + y) * (x - y)
>> diff(x^3 * exp(x))
ans =
3 * x^2 * exp(x) + x^3 * exp(x)
```

$>> factor(ans)$
$ans =$
$exp(x) * x\hat{\ }2 * (x + 3)$
Funcţia $simplify$ încearcă să reducă expresia simbolică la o formă simplă.
$>> simplify(exp(5 * log(x) + 1))$
$ans =$
$x\hat{\ }5 * exp(1)$
$>> d = diff((x\hat{\ }2 + 1)/(x\hat{\ }2 - 1))$
$ans =$
$(2 * x)/(x\hat{\ }2 - 1) - (2 * x * (x\hat{\ }2 + 1))/(x\hat{\ }2 - 1)\hat{\ }2$
$>> simplify(d)$
$ans =$
$-(4 * x)/(x\hat{\ }2 - 1)\hat{\ }2$
Funcţia $simple$, alternativă a funcţiei $simplify$ calculează mai multe simplificări şi alege varianta mai compactă.
$simplify(cos(x) + (-sin(x)\hat{\ }2)\hat{\ }(1/2))$
$simple(cos(x) + (-sin(x)^2)\hat{\ }(1/2))$
$simplify((1/x^3 + 6/x^2 + 12/x + 8)\hat{\ }(1/3))$
$simple((1/x^3 + 6/x^2 + 12/x + 8)\hat{\ }(1/3))$

Exercitii

1. Derivaţi următoarele funcţii. Simplificaţi răspunsul dacă este posibil.
(a) $f(x) = arcsin(2x + 3)$
(b) $f(x) = arctan(x^2 + 1)$
2. Calculaţi următoarele integrale:
(a) $\int xsin(x^2)dx$
(b) $\int x^2\sqrt{x+4}dx$
(c) $\int_{-\infty}^{\infty} e^{-x^2}dx$

.1.3 Integrarea folosind descompunerea polinomială

Pentru integrarea numerică a unei funcţii, intervalul de integrare poate fi descompus în intervale mai mici şi se poate aproxima aria de sub curbă pe porţiuni.
Expresia unei curbe care trece prin trei puncte $(x_{j-1},\ f_{j-1})$, $(x_j,\ f_j)$ şi $(x_{j+1},\ f_{j+1})$ poate fi scrisă sub forma patratic:

$$f_q(x) = f_0 + \Delta f_0 \frac{x - x_0}{h} + \Delta^2 f_0 \frac{(x - x_0)(x - x_1)}{2h^2} \qquad (.1)$$

Pentru a obţine formula integralei pe intervalul $[x_0,\ x_2]$ este necesară integrarea funcţiei $f_q(x)$ de la $x = x_0$ la $x = x_2$.
Secvenţa de cod pentru a obţine expresia integralei în MATLAB:
$>> syms\ x\ f0\ f1\ f2\ h\ x0$
$>> q = f0 + (f1 - f0)/h * (x - x0) + (f2 - 2 * f1 + f0)/(2 * h^2) * (x - x0) * (x - (x0 + h));$
$>> iq = int(q, x0, x0 + 2 * h);$
$>> simplify(iq)$
$ans =$
$1/3 * h * (f0 + 4 * f1 + f2)$
Descrierea secvenţei de cod:

- linia 1: se declară variabilele x, $f0$, $f1$, $f2$, h şi $x0$ ca fiind simbolice;
- linia 2: se defineşte funcţia q ($f_q(x)$);
- linia 3: se integrează funcţia q între $x0$ şi $x2 = x0 + 2 * h$;
- linia 4: se simplifică rezultatul integrării.

Expresia rezultată este:

$$\int_{x=x_0}^{x=x_2} f_q(x)dx = \frac{h}{3}(f_0 + 4f_1 + f_2) \qquad (.2)$$

.1.4 Pendulul simplu

Figura .1 prezintă un pendul simplu (punctul de masă m) suspendat de un fir de lungime l, considerat inextensibil şi fără greutate.

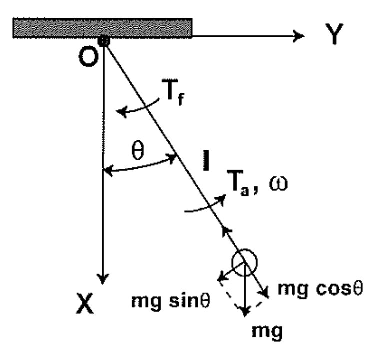

Figure .1: Pendulul simplu

Derivarea ecuaţiilor de mişcare presupunând că forţa de frecare este o funcţie liniară de viteza unghiulară ($T_f = B_m\omega$), B_m este coeficientul de frecare vâscoasă):

$$F_{rest} = -mg\sin\theta \tag{.3}$$

$$\sum M = T_{rest} + T_a - T_f = -mgl\sin(\theta) + T_a - B_m\omega \tag{.4}$$

Ecuaţiile diferenţiale de ordinul doi pentru modelarea mişcării pendulului

$$\begin{aligned} J\alpha = J\frac{d^2\theta}{dt^2} &= -mgl\sin\theta + T_a - B_m\omega \quad \text{, sau} \\ \frac{d^2\theta}{dt^2} &= \frac{1}{J}(-mgl\sin\theta + T_a - B_m\omega) \end{aligned} \tag{.5}$$

Sistemul de două ecuaţii diferenţiale:

$$\begin{aligned} \frac{d\omega}{dt} &= \frac{1}{J}(-mgl\sin\theta + T_a - B_m\omega) \quad , \\ \frac{d\theta}{dt} &= \omega \end{aligned} \tag{.6}$$

Momentul de inerţie este $J = ml^2$, prin urmare ecuaţiile pot fi scrise sub forma:

$$\begin{aligned} \frac{d\omega}{dt} &= -\frac{B_m}{ml^2}\omega - \frac{g}{l}\sin\theta + \frac{1}{ml^2}T_a \quad , \\ \frac{d\theta}{dt} &= \omega \end{aligned} \tag{.7}$$

Rezultatele simulării modelului pendulului pentru unghiuri mici (care permit aproximarea: $\sin\theta = \theta$) şi parametrii $l = 4$, $B_m/J = 0.05$, $T_a = 0$ sunt prezentate în Figura .2. Secvenţa de cod utilizată este

```
syms u w l g
uw = dsolve('Du = w','Dw = -0.05 * w - g/l * u','u(0) = 4','w(0) = 0')
theta = subs(uw.u, l, g, 4, 9.8);
omega = subs(uw.w, l, g, 4, 9.8);
hold on
ezplot(theta, [0, 30]);
ezplot(omega, [0, 30]);
```

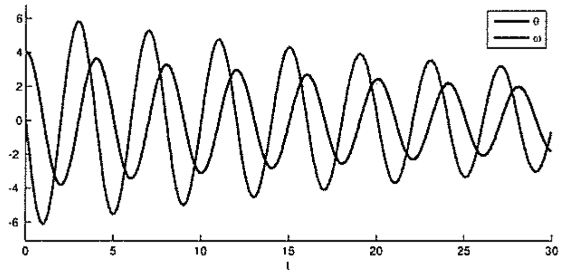

Figure .2: Simulare pendul simplu pentru unghiuri mici

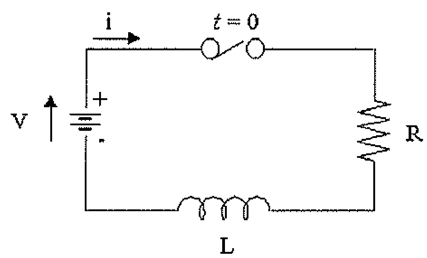

Figure .3:

.1.5 Dinamica unui circuit electric RL

Considerăm un model pentru un circuit electric care conţine un rezistor, un inductor, o sursă de tensiune continuă şi un întrerupător (Figura .3).

Dacă la momentul $t = 0$ se închide întrerupătorul (anterior acestui moment curentul prin inductor este nul) avem ecuaţia:

$$L\frac{di}{dt} + Ri = V \tag{.8}$$

Pornind de la ecuaţia diferenţială (.8), se deduce următoarea expresie pentru i:

$$i = \frac{V}{R}\left[1 - \exp\{-Rt/L\}\right] \tag{.9}$$

Pentru verificarea acestei soluţii folosind toolbox-ul simbolic al Matlab-ului, se poate utiliza următoarea secvenţă de cod:

% Script pentru a verifica că
% i = (V/R) * (1 - exp(-R*t/L))

```
% este soluţia ecuţiei
% di/dt + (R/L) * i - V/L = 0
%
% Pas 1: Se definesc variabilele simbolice
%
syms i V R L t
%
% Pas 2: Se construieşte soluţia pentru i
%
i = (V/R) * ( 1 - exp(-R*t/L) );
%
% Pas 3: Se calculează derivata lui i
%
didt = diff(i,t);
%
% Pas 4: Însumarea termenilor în ODE
%
didt + (R/L) * i - V/L;
%
% Pas 5: Răspunsul este ZERO?
%
simple(ans)
%
% Pas 6: Care este valoarea lui i la momentul t = 0?
%
subs(i,t,0)
%
```