Počítačová podpora aplikácií numerickej kvadratúry

Alena Vagaská
Technická univerzita v Košiciach, Fakulta výrobných technológií, Katedra matematiky, informatiky a kybernetiky
e-mail: vagaska.alena@fvt.sk

Abstrakt

In this paper will be described some possibilities of using the program system Matlab to applications of numeric quadrature.

1. Úvod

Na technických univerzitách by malo byť samozrejmosťou, že učiteľ matematiky má záujem prehĺbiť aplikačný charakter matematických predmetov zavádzaním technických aplikácií matematiky do výučby. Taktiež by malo byť samozrejmé, že naučí študentov zvládnuť zvyšujúce sa nároky na rýchlosť a presnosť matematických výpočtov v takýchto aplikáciach pomocou využitia tzv. CAS systémov. Článok prezentuje možnosti Matlabu v aplikáciách numerickej kvadratúry pri riešení aplikačných diferenciálnych rovníc.

2. Matlab a M-funkcie v aplikáciach numerickej kvadratúry

Uveďme si možnosti Matlabu pri numerickej aproximácii určitého integrálu, ku ktorému sa dostaneme po aplikácii teórie diferenciálnych rovníc pri riešení nasledovného problému.
Každý tepelný proces, teda aj ohrev vody, môžeme popísať diferenciálnou rovnicou. Ak by sme mali určiť čas t, za ktorý elektrickou špirálou ohrejeme 1 kg vody z izbovej teploty $T = 20\,°\mathrm{C}$ na teplotu varu $T = 100\,°\mathrm{C}$, ak vieme, že elektrické napätie je $120\,\mathrm{V}$, odpor špirály $14{,}4\,\Omega$ a je známe, že 1 kg vody ochladne zo $40\,°\mathrm{C}$ na $30\,°\mathrm{C}$ za 10 minút; tak počas riešenia sa dopracujeme k diferenciálnej rovnici ohrevu vody [1], ktorá za daných podmienok vyzerá nasledovne:

$$\frac{dT}{dt} = 0{,}24 - \frac{ln\,2}{600}.T + \frac{ln\,2}{30}. \tag{1}$$

Separáciou premenných v rovnici (1) a aplikáciou určitého integrálu nájdeme hľadaný čas t:

$$t = 600\int_{20}^{100}\frac{dT}{144 + 20\,ln\,2 - T\,ln\,2} = -\frac{600}{ln\,2}\Big[ln\left|144 + 20\,ln\,2 - T\,ln\,2\right|\Big]_{20}^{100} \approx 421\mathrm{s}. \tag{2}$$

Je zrejmé, že určitý integrál zo vzťahu (2) sa dá vypočítať aj analyticky, pri numerickej aproximácii nám Matlab ponúka viaceré možnosti. Buď editujeme príkazy podľa jednotlivých Newton-Cotesových vzocov rovno do promptov, čo je zdĺhavejšie, alebo môžeme použiť M-fukcie. V článku je uvedená ukážka numerickej integrácie v Matlabe bez použitia M-funkcií pomocou lichobežníkovej a Simpsonovej metódy s vopred daným počtom podintervalov. Ak by sme sa rozhodli zvyšovať presnosť aproximácie zvyšovaním počtu podintervalov, či využiť Booleov alebo Milneho vzorec, museli by sme vždy nanovo editovať zmenené zložené N-C vzorce do príkazových riadkov, čo by bolo pomerne pracné. Tomu možno predísť využitím M-funkcií, ktoré umožňujú modifikovať kolekciu príkazov (meniť N-C vzorce či počet podintervalov) bez pracného vpisovania zmien do promptov. V článku je uvedený postup vytvárania M-funkcií ako aj ukážky M-funkcií „NCvahy“, „NCqelem“ a „IDRohrevu“, ktoré sú potrebné pri aproximácii integrálu (2). Nasledujúca ukážka použitia spomenutých M-funkcií ilustruje, že pri kvadratúre elementárnymi vzorcami zvyšovaním počtu uzlov interpolácie sa zvyšuje presnosť integrovania. Prvý výsledok sme získali pomocou elementárneho lichobežníkového vzorca (2 uzly interpolácie), druhý elementárnym Simpsonovym vzorcom (3 uzly), tretí Booleovym (4 uzly) a štvrtý výsledok pomocou elementárneho Milneho vzorca (5 uzlov).

```
>> format long
>> for i=2:5, aprox(i)=NCqelem('IDRohrevu',20,100,i); end, aprox(2:5)
ans =
  1.0e+002 *
   4.37705412497180    4.21113536472869    4.21008046716879    4.20922133759089
```

Zvyšovanie počtu uzlov interpolácie nie vždy vedie k zvyšovaniu presnosti výsledku, preto je v článku uvedená aj ukážka tzv. zloženej kvadratúry pomocou M-funkcie „NCqzloz“.

```
function y=NCqzloz(maf,a,b,m,n)
% Newton-Cotesova kvadratura na [a,b]
% m - pocet uzlov elem.vzorca na podintervale
% n - pocet podintervalov na [a,b]
w=NCvahy(m);                %vektor vah
x=linspace(a,b,n*(m-1)+1);  %vektor uzlov na [a,b]
fx=feval(maf,x);
fx=fx(:);                   %fx bude stlpcovy vektor f-hodnot
y=0;
index1=1;
index2=m;
for i=1:n
    y=y + w*fx(index1:index2);
    index1=index2;
    index2=index2 + m-1;
end
y=y*(b-a)/n;
```

Ak sa rozhodneme, že počet podintervalov bude napr. $n = 8$ a na každom z nich použijeme Booleov vzorec, jednoduchým volaním funkcie získame výsledok:

```
>> NCqzloz('IDRohrevu', 20, 100, 4, 8)
ans =
   4.209201316352159e+002
```

3. Záver

Matlab nám pri numerickej kvadratúre ponúka ešte aj M-funkciu „quadl“, ktorej podstatou sú moderné metódy numerického integrovania používajúce adaptívne algoritmy. Pre náš prípad:

```
>> [nI,k]=quadl('IDRohrevu', 20, 100, 5e-8)
nI =
   4.209201059354103e+002
```

Na základe skúsenosti s výučbou numerickej matematiky môžeme vysloviť presvedčenie, že vhodnou kombináciou klasických metód vyučovania s novými, využívajúcimi softwérovu podporu, docielime skvalitnenie a zefektívnenie edukačného procesu.

Literatúra

[1] Пономарев, К. К.: Составление дифференциальных уравнений. Минск: Выщзйщая школа, Минск, 1973

[2] Volauf, P.: Numerické a štatistické výpočty v Matlabe. Bratislava: STU, 2005, ISBN 80-227-2259-6