Experimentálna výučba numerickej matematiky pomocou programového systému Mathematica na SjF STU.

Kováčová Monika, Halada Ladislav

Katedra Matematiky, Strojnícka fakulta STU,
Nám. Slobody 17, 812 31 Bratislava,
email: kovacova_v@dekan.sjf.stuba.sk, halada@dekan.sjf.stuba.sk

Výučbu matematiky na našej fakulte možno rozdeliť do dvoch skupín (principiálne rozdielneho charakteru). Jednou skupinou je " základný kurz " matematicky (prvé štyri semestre), druhou sú voliteľné predmety v 3. - 5. ročníku (napr. Aplikovaná matematika, Matematická štatistika). Toto rozdelenie určuje nielen charakter predmetu, ale aj počet a vedomostnú úroveň študentov.

Základný kurz zahŕňa 3 semestre matematickej analýzy a lineárnej algebry a 1 semester numerickej matematiky. Experimentálna výučba pomocou programového systému Mathematica prebiehala v 2. až 4. semestri „základného kurzu"

DÔVODY EXPERIMENTU

Vývoj v oblasti výpočtovej techniky a tvorby programových produktov ide veľmi rýchlo dopredu a výrazne ovplyvňuje spôsob výučby matematiky na VŠ. Zatiaľ, čo v laboratórnych cvičeniach je používanie PC už samozrejmosťou, jeho použitie na prednáškach a teoretických cvičeniach je menej rozšírené. Hlavne preto, že ostáva nevyriešeným problémom, akým spôsobom a do akej miery používať programové produkty, ktoré človeku aj bez znalosti príslušného algoritmu poskytujú výsledky riešenia úloh pre zadané vstupné údaje. K otázke, čo absolvent inžinierského štúdia z matematiky potrebuje vedieť, sa pridáva ďalšia, akým spôsobom ho to čo najefektivnejšie naučiť? Rozsah vedomostí, ktoré si vyžadujú súčasné technologické a konštrukčné procesy sa zväčšuje, čo vytvára prirodzený tlak na znižovanie počtu hodín doposiaľ venovaných získaniu " matematickej zručnosti ". Vo všeobecnosti sa očakáva, že počítač pomôže zredukovať čas, doposiaľ venovaný technike výpočtu.

Spoločnou snahou vyučujúcich je využiť počítač aj na:
- zlepšenie učebného procesu pomocou modelových príkladov,
- podporenie kreativity študenta,
- podporenie samovzdelávacích aktivít študenta.

V tomto príspevku chceme ukázať, na konkrétnych príkladoch, ako v predmete numerická matematika využívame programový systém MATHEMATICA na hlbšie pochopenie problému a jednoduchšiu realizáciu niektorých výpočtových úloh.
V prechádzajúcich rokoch sme pri výučbe numerickej matematiky používali programovací jazyk Pascal. Študenti v rámci počítačových cvičení vypracovávali jednoduché programy v tomto jazyku (napr. hľadanie koreňov

rovnice, riešenie dif. rovnice ...). Vzhľadom k tomu, že študenti nastupujúci na VŠ majú rôznu úroveň znalostí a zručností z informatiky, ba niektorí z nich vôbec nie sú pripravení na prácu pomocou tradičného programovacieho jazyka PASCAL, domnievali sme sa, že pomocou programového systému Mathematica možno študenta pomerne rýchlo zaškoliť na samostatnú tvorbu výpočtových postupov, ktoré uľahčujú analýzu študovaných problémov. Ak by sa podarilo umožniť študentovi v priebehu jeho štúdia všetky výpočty vykonávať pomocou tohto systému, získal by tak nástroj, ktorý by nielen zefektívnil jeho prácu, ale pripravil ho aj na presné riešenie náročnejších výpočtov.

Vzhľadom k už uvedenému, na základe našich skúseností sa domnievame, že prácu s programovým systémom Mathematica, hlavne jeho metajazyk, zvládne študent lepšie a s menšími problémami, ako programovací jazyk Pascal. Naviac práve vďaka tomuto metajazyku môžeme riešiť na cvičeniach úlohy numerickej matematiky a nehodnotíme programátorskú zručnosť študenta.

POPIS EXPERIMENTÁLNEJ VÝUČBY

Experimentálna výučba pomocou programového systému Mathematica© (Wolfram Research) prebiehala na SjF STU v šk. rokoch 1996/ 97 a 1997/ 98. V prvom roku bolo do experimentu zaradených 50 študentov, v druhom 120 študentov (najmenšia organizačná jednotka - paralelka, ktorá má samostatné prednášky). Využitie programového systému Mathematica vo výučbe základného calculu podrobnejšie popisuje práca [2]. V tomto článku popíšeme priebeh výučby v štvrtom semestri nášho experimentu. Obsahom predmetu sú základy numerickej matematiky. Študenti prvého roku experimentálnej výučby momentálne ukončili základný kurz skúškou z numerickej matematiky. Vyhodnotenie výsledkov experimentu uvádzame v nasledujúcich odstavcoch nášho článku.

Už v prípravnej fáze experimentu bola vypracovaná koncepcia výučby a spôsob hodnotenia, v priebehu rokov 1996, 1997 postupne vznikali pomocné študijné materiály, nootebooky a packages pre študentov, ktoré sme neskôr použili priamo vo výučbe. Niektoré z nich uvádzame aj v tomto článku.

Rozsah výučby sú 2 hodiny prednášok a 2 hodiny cvičení týždenne. V rámci prednášok sa prednášajúci venoval, okrem teórie, aj možnostiam pg. systému Mathematica. Cvičenia boli rozdelené na teoretické a počítačové. Jednotlivé cvičenia sa navzájom striedali. *Teoretické cvičenia* boli klasickými cvičeniami, ktorých hlavnou úlohou bolo precvičiť myšlienkové postupy a metódy riešení úloh na výpočtovo jednoduchých príkladoch. *Počítačové cvičenia* sa uskutočňovali v samostatnej učebni, pri počítači pracoval každý študent samostatne. Dosiahnuté výsledky sme tak mohli ľahko hodnotiť. Naviac táto skupina študentov pracovala s pg. systémom Mathematica už tretí semester, a tak hlavnou prioritou počítačových cvičení bola simulácia úloh k danej téme cvičenia. Metajazyk systému Mathematica už študentom nerobil problémy.

Na počítačových cvičeniach sme sa zaoberali témami:

- Numerické chyby, podmienenosť výpočtu, prenos numerickej chyby pri jednotlivých aritmetických operáciách.
- Metódy hľadania koreňov rovnice $f(x)=0$.

- Riešenie sústav rovníc, maticový počet, výpočet charakteristického polynómu, vlastných čísel a vektorov matice.
- Lagrangeov a Newtonov interpolačný polynóm.
- Numerické integrovanie a numerické derivovanie.
- Numerické metódy na riešenie dif. rovnice 1. rádu.

Skúška sa skladala z niekoľkých častí: dve písomné práce počas semestra, počítačový test z nadobudnutých zručností pri práci s pg. systémom Mathematica, test teoretických vedomostí z numerickej matematiky. Ak študent získal aspoň 70% bodov, získal známku (1, 1-, 2, 2-) podľa dosiahnutého výsledku. V prípade menšieho bodového zisku musel vykonať ešte ďalšiu ústnu a písomnú skúšku počas riadneho skúškového obdobia.
V experimentálnej skupine sme dosiahli tieto výsledky:
Zápočet z numerickej matematiky nezískal len jeden študent. 18% študentov nesplnilo podmienku 70% úspešnosti a známku nezískali na prvom termíne a 6,2% študentov skúšku nespravilo ani na 3. termín.
Pri porovnaní s výsledkami ostatných študijných skupín sa percentuálna úspešnosť vykonania skúšky u experimentálnej skupiny takmer nelíšila od ostatných skupín. Zmena nastala hlavne v kvalite. Kým pri klasickom spôsobe výučby (pomocou programovacieho jazyka Pascal) takmer 60% študentov predmet ukončilo s výsledkom 3- (najhorší možný výsledok), úspešnosť v experimentálnej skupine bola nasledovná:

známku 1	dosiahlo	5,8%	študentov
známku 1-	dosiahlo	17,6%	študentov
známku 2	dosiahlo	23,5%	študentov
známku 2-	dosiahlo	32,3%	študentov
známku 3	dosiahlo	5,8%	študentov
známku 3-	dosiahlo	8,8%	študentov

Podobný výsledok bol zaznamenaný aj pri experimentálnej výučbe základného calculu a popisuje ho práca [2]

UKÁŽKA EXPERIMENTÁLNEJ VÝUČBY

CHYBA FUNKČNEJ HODNOTY

Nech $y = f(x)$ je dif. funkcia jednej premennej a nech $\bar{x}$ je aproximácia x. Je zrejmé, že pre odhad chyby približnej hodnoty $f(\bar{x})$ platí

$$| f(x) - f(\bar{x})| = | f'(z)| \, | x - \bar{x} | \le M | x - \bar{x} |$$

kde $M = \sup |f'(z)|$, pričom suprémum hľadáme v okolí čísla $\bar{x}$ obsahujúcom číslo x. Z uvedeného vzťahu vyplýva, že M predstavuje mieru citlivosti funkcie $f(x)$ pri perturbácii argumentu x. Pre relatívnu chybu potom platí

$$| f(x) - f(\bar{x})| \;/\; | f(x)| \le M | x - \bar{x} | / | f(x)|$$

Ak číslo podmienenosti úlohy definujeme ako pomer relatívnej chyby výstupu k relatívnej chybe vstupu, potom číslo podmienenosti výpočtu funkčnej hodnoty pre okolie bodu x je dané vzťahom

$$C_p \approx | x f'(x) | / | f(x) |$$

Je zrejmé, že ak $C_p \approx 1$, úloha je dobre podmienená, ale ak C_p je veľké číslo, potom je úloha zle podmienená, t.j. malá zmena vo vstupných údajoch vyvolá veľkú zmenu vo výstupných údajoch. Pri skúmaní podmienenosti takejto úlohy má učiteľ širokú škálu funkcií, ktoré sa pri výpočte hodnôt z niektorých intervalov správajú veľmi rozdielne. Ako príklad môžeme uviesť funkciu *sin(x)* a jej číslo podmienenosti v okolí bodu *0* a v okolí bodu é pri rovnakej perturbácii argumentu o hodnotu 0.01. Obdobne možno odvodiť vzťah pre číslo podmienenosti výpočtu funkcie viac premenných a skúmať stabilitu výpočtu takýchto funkcií v rôznych oblastiach. Pri týchto výpočtoch podporovaných grafickým znázornením funkcií, študent rýchlo pochopí, že oblasti v ktorých je gradient funkcie veľký môže dochádzať k strate presnosti výpočtu. Preto pri riešení konkrétnych aplikačných úloh nesmie stratiť zo zreteľa aj tento aspekt.

Poučný je aj príklad iného druhu, ktorý sa týka funkcionálnych radov. Uvažujme rozvoj funkcie e^x do radu, t.j

$$e^x = 1 + x + x^2 / 2> + x^3 / 3> + x^4 / 4> + \ldots\ldots$$

Ak použijeme na výpočet Taylorov polynóm 7. stupňa

In[1]:=

```
Series[Exp[x],{x,0,7}];
Normal[%]
```

Out[2]=

```
             2     3     4     5      6       7
            x     x     x     x      x       x
1 + x + -- + -- + -- + --- + --- + ----
            2     6     24    120    720    5040
```

Pre výpočet $e^{-5.5}$ dostávame alternujúci rad. Súčet prvých 8 členov nám poskytuje hodnotu -12.6786.

In[3]:=

```
%2/.x->-5.5
```

Out[3]=

```
-12.6786
```

Kedže presná hodnota $e^{-5.5}$ = 0.0040867 nemá predchádzajúci výsledok ani jednu platnú číslicu.

In[4]:=

```
Exp[-5.5]
```

Out[4]=

```
0.00408677
```

Uvedený príklad je prípadom nevhodného postupu výpočtu, v ktorom zaokrúhľovacie chyby niektorých sčítancov sú rovnakého rádu ako je konečný výsledok. Opäť, na zoslabenie účinku tohto algoritmu buď použijeme väčší počet platných čísel a členov radu pri výpočte, alebo použijeme vhodnejšiu metódu výpočtu, napr.

$e^{-5.5} = 1 / e^{5.5} = 1 / (1 + 5.5 + \ldots\ldots)$

Približnú hodnotu funkcie v bode -5,5 vypočítame

In[5]:=

```
%2/.x->5.5
```

Out[5]=

```
198.075
```

In[6]:=

```
1/(%2/.x->5.5)
```

Out[6]=

```
0.0050486
```

CHYBY PRI MATICOVÝCH VÝPOČTOCH

Medzi základné úlohy maticového počtu na VŠ patrí riešenie systému lineárnych rovníc, výpočet charakteristického polynómu, vlastných čísel a vektorov matice. Pomocou programového systému MATHEMATICA možno ľahko modelovať rôzne príklady dokumentujúce nestabilitu výpočtu danej úlohy s poukázaním na hodnotu čísla podmienenosti matice. Ako príklad môžeme uviesť úlohu typu *A x* = *b*, pre ktorú číslo podmienenosti úlohy je úmerné vzťahu $\|A\|.\|A^{-1}\|$. Nech

$$A = \begin{pmatrix} 6.1 & 3.9 \\ 3.8 & 2.4 \end{pmatrix} \qquad b1 = \begin{pmatrix} 6.1 \\ 3.8 \end{pmatrix}$$

Riešením systému *A x* = *b1* je vektor *x* = $(1,0)^T$. Ak zmeníme pravú stranu systému na *b2=b1+{-0.02,0.02}* potom je riešením systému *A* $\bar{x}$ = *b2* vektor $\bar{x}=(1.7,-1.1)^T$. Pre číslo podmienenosti platí $C_p = \dfrac{\|x-\bar{x}\| / \|x\|}{\|b1-b2\| / \|b1\|} = 331.295$. Čo iba potvrdzuje, že úloha je nestabilná. Pomocou definície normy vieme rovnako jednoducho spočítať aj číslo podmienenosti C_p. Realizácia v pg. systéme Mathematica bude nasledujúca:

In[1]:=

```
A={{6.1,3.9},{3.8,2.4}};
b1={6.1,3.8};
b2={6.08,3.82};
```

In[4]:=

```
LinearSolve[A,b1]
LinearSolve[A,b2]
```

Out[4]=

```
{1., 0.}
{1.7, -1.1}
```

NUMERICKÉ RIEŠENIE OBYČAJNÝCH DIFERENCIÁLNYCH ROVNÍC

V mnohých prípadoch ODR, pomerne jednoduchých, nemožno nájsť riešenie vyjadrené pomocou elementárnych funkcií, ktoré by spĺňalo danú dif. rovnicu a dané počiatočné podmienky. Študenti tieto problémy poznajú, či už z fyziky, alebo z vlastných skúseností z predchádzajúceho semestra.

Numerické metódy na riešenie takýchto typov rovníc sa preto stretajú každoročne s veľkým záujmom.
V rámci prednášok sa stretnú s Eulerovou metódou a jej modifikáciami, metódou Runge-Kutta. Keďže ide o látku záveru semestra, niektorí študenti sa zoznámia aj s viackrokovými metódami. Postup výpočtu nie je ťažký, avšak realizovať ho len s pomocou kalkulačky, s požadovaným krokom a presnosťou je pomerne časovo náročné.

Výhoda grafických a programových možností systému Mathematica je v tomto prípade ľahko viditeľná.

Uvedieme ukážku krátkeho programu, vytvoreného v metajazyku programu Mathematica. Na cvičení tento program študenti postupne po jednotlivých krokoch tvoria sami. Prekvapilo nás, že realizácia výpočtu na experimentálnych cvičeniach pôsobila podstatne menšie problémy, ako programovanie rovnakého problému v PASCALE. Teraz uvedieme študentské riešenie úlohy $y' = 2x - \sin y$, $y(0) = 1$, na intervale $[0,1]$ s krokom $h = 0{,}1$. Modifikácia programu na inú metódu, alebo inú dif. rovnicu je veľmi jednoduchá.

In[1]:=
```
Clear[f,x,y,obrazok]
```

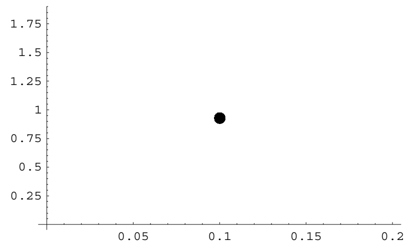
In[2]:=
```
f[x_,y_]:=2x-Sin[y]
x=0;y=1;h=0.1;
```

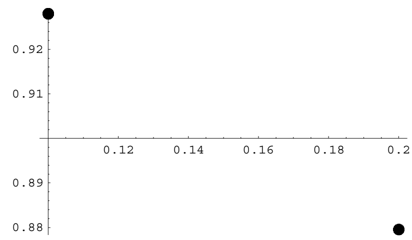
In[3]:=
```
obrazok:=ListPlot[{{x,y}},PlotStyle->PointSize[0.03]]
```

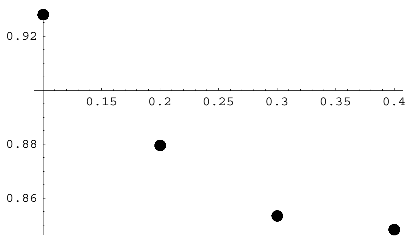
In[4]:=
```
Do[k1=h*f[x,y];
    k2=h*f[x+h/2,y+k1/2];
    k3=h*f[x+h/2,y+k2/2];
    k4=h*f[x+h,y+k3];
    lambda=1/6*(k1+2k2+2k3+k4);
    y=y+lambda;x=x+h;
Print["x= ",x," y= ",y];
obrazok=Show[obrazok,
ListPlot[{{x,y}}, PlotStyle->PointSize[0.03],
DisplayFunction->Identity],
DisplayFunction->$DisplayFunction],{10}]
```

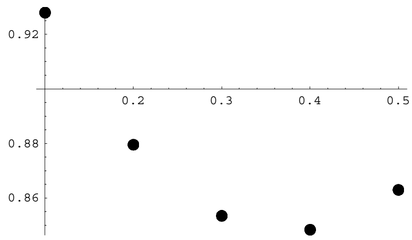
Získame aj vypočítané hodnoty riešenia v krokových bodoch, aj jeho grafickú interpretáciu

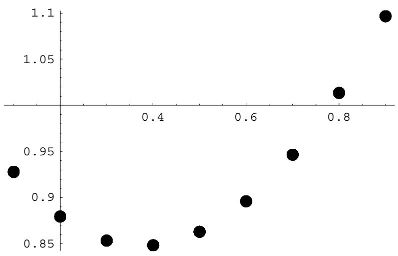
x= 0.1	y= 0.927985	x= 0.6	y= 0.896041
x= 0.2	y= 0.879546	x= 0.7	y= 0.946498
x= 0.3	y= 0.853459	x= 0.8	y= 1.01354
x= 0.4	y= 0.848382	x= 0.9	y= 1.09664
x= 0.5	y= 0.862985	x= 1.	y= 1.19561

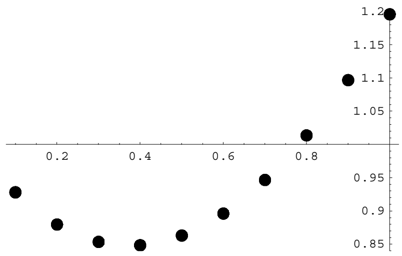












Z grafického výstupu uvádzame len niektoré jeho časti, pre názornejšiu predstavu.

Ďalšou variantou výučby je použitie preddefinovaných príkazov na riešenie konkrétneho problému.

In[1]:=

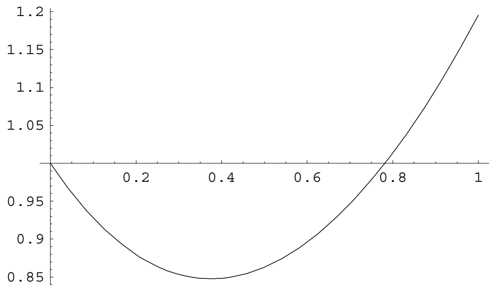
```
NDSolve[{y'[x]==2x-Sin[y[x]],y[0]==1},y,{x,0,1}]
```

Out[1]=

```
{{y -> InterpolatingFunction[{{0., 1.}}, <>]}}
```

In[2]:=

```
Plot[y[x]/.%,{x,0,1}]
```



Ďalšie skúsenosti s vyučovaním popisujú aj práce [1,3,5].

Program Mathematica pri výučbe umožňuje postupovať dvoma základnými cestami.

1.Postupovať po jednotlivých krokoch výpočtu spôsobom jednoriadkový vstup a následný výstup. Tento postup študenti používajú hlavne v prvom semestri experimentu a je výhodný, pri prvom stretnutí s daným problémom. Akýkoľvek problém si študent takpovediac “ohmatá”, zoznámi sa s problematickými miestami. Neskôr im táto mechanická práca s “lepšou kalkulačkou” prestáva stačiť a začnú sa zaujímať o programovanie v dostupnom metajazyku Mathematice.

2. Použiť preddefinované algoritmy: napr. DSolve, NDSolve, alebo iné pomocné balíky vytvorené pedagógom s presne stanoveným cieľom. Príprava výučby je natoľko pohodlná, že skúsenému pedagógovi (s minimálnou programovacou zručnosťou) nezaberie veľa času. Tento postup naviac umožňuje predkladať a riešiť aj ťažšie fyzikálne problémy, či problémy stability úloh. Výrazným spôsobom posúva hranicu, ktorú zvažuje pedagóg pri zaradení príkladu: význam $\leftrightarrow$ časová náročnosť. Programové balíky nemusí vytvárať len pedagóg. Študenti nášho experimentu prvé takéto balíky začali vytvárať už koncom druhého semestra ich práce s Mathematicou, niektorí sa počas experimentu naučili dokonca využívať možnosti dynamického programovania.

ZÁVER

V súčasnosti existuje veľký počet softwareových balíkov, dosť podobného charakteru, ktoré by mali pomáhať, či už priamo vo výučbe, alebo uľahčiť zložité výpočtové procesy. Niektoré sú voľne šíriteľné, väčšina sú však komerčné produkty. U voľne šíriteľných programov je obyčajne kvalita obmedzená na základné počtové úkony calculu, ktoré sú možno postačujúce pre výučbu v základnom kurze matematiky, ale sú celkom nevhodné pre neskoršie použitie v technickej praxi. Komerčné programy majú cielenejší charakter. Spomeňme aspoň niektoré z nich: Mathematica, Derive, X-math, MatLab, MathCad. Každý z nich má mnohé výhody, aj mnohé nevýhody. Preto je veľmi dôležité poznať požiadavky výučby, a hlavne neskoršej praxe, aby bolo možné vybrať vhodný komerčný produkt.

Komerčné produkty možno rozdeliť do dvoch základných skupín. Do jednej skupiny patria tie, ktoré podporujú hlavne numerickú prácu s dátami (napr. MatLab), do druhej tie, ktoré podporujú symbolickú prácu (napr. Derive, Mathematica, MathCad).

V našom experimente do úvahy prichádzali produkty podporujúce len numerickú prácu, alebo aj numerickú aj symbolickú prácu. Pokúsme sa teraz vysvetliť dôvody, ktoré nás viedli ku rozhodnutiu používať práve program Mathematica. Na základe ratingov softwarových produktov, ktoré sú dostupné v sieti a na základe dostupnosti do úvahy prichádzali tri produkty: Derive, Mathematica, MathCad.

Derive je produkt lokalizovaný pod DOS-om, veľmi jednoducho ovládateľný. Nemá síce komfortnosť prostredia Windows a jeho grafické možnosti sú obmedzené, ale má minimálne hardwareové nároky, čo vzhľadom na vybavenosť nášho školstva je dosť významný ukazovateľ.[1] Za veľkú nevýhodu však považujeme obmedzené schopnosti tohto programu. Je to vlastne len veľmi výkonná, veľká kalkulačka, ktorá síce takmer stačí v základnom kurze, ale tým sú jej schopnosti plne vyčerpané. Naviac pri riešení numerickej matematiky je potrebné hľadať a veľmi kombinovať aj neštandartné matematické postupy tak, aby sme dospeli ku výsledku. Hlavným dôvodom, pre ktorý sme zamietli možnosť použiť tento produkt, bola nemožnosť jeho použitia v ďalšom štúdiu, a tým viac v následnej praxi.

MathCad v súčasnej verzii je produkt porovnateľnej úrovne ako Mathematica, ale len pri použití zakladného balíka. Doplnkové možnosti pg. Mathematica (tzv. packages) tento program posúvajú na podstatne vyššiu úroveň. Nemôžeme však porovnávať súčasný stav. Výučbu sme začali pripravovať pred tromi rokmi, kedy program MathCad mal ešte veľmi obmedzené možnosti.

Poslednou možnosťou bolo použitie programu Matlab. Má mnohé výhody a pre výučbu numeriky je priamo stvorený. Teší sa aj veľkej obľube učiteľov na odborných katedrách. Hlavný dôvod, prečo sme v našom experimente nepoužili túto možnosť bol časový faktor. Na počítačové cvičenia z numeriky, tak ako sme už spomínali, je vyhradených 6 x 2 hodiny za semester. V tomto čase sa nedá stihnúť precvičenie hlavných tém z numeriky, aj naučiť študentov pracovať s úplne novým softwarom. Naviac študenti experimentálnej skupiny mali už dva semestre aktívnej práce s Mathematicou absolvované a práca v nej im nespôsobovala technické problémy. Z týchto dôvodov sme sa rozhodli pre program Mathematica. Nebudeme tu teraz uvádzať všetky jeho možnosti a prednosti, nie je to cieľom nášho článku. Stručne uvedieme len tie, ktoré považujeme za najdôležitejšie.

- grafické možnosti programu
- možnosť vytvárania uzavretých programových balíkov
- možnosť dymanického programovania
- prepojenie s programovým jazykom C, resp. C++ ,link na Fortranovské knižnice, objektové programovanie
- animačné možnosti programu a prepojenie aj so systémom MathLab.

Záujemcom môžeme len odporučiť www.wri.com (resp. jej zrkadlá), alebo základnú publikáciu [4].

LITERATÚRA

[1] *Halada, L.:* Stabilita úloh a algoritmov vo výučbe numerickej matematiky na SjF STU, *Informatika a Algoritmy, Prešov, 3. -4. September 1998.*

[2] *Kolesárová, A ., Kováčová, M., Záhonová, V.:* Použitie programového systému Mathematica pri výučbe základov matematickej analýzy na SjF STU, *Matematická štatistika a Numerická matematika, Kálnica, 1.-5. Júna 1998.*

[1] Existuje už aj windows verzia, ale nemali sme možnosť testovať ju a tak nemôžeme posúdiť jej parametre, ani zmeny oproti DOS verzii.

[3] *Kováčová, M.:* Výučba diferenciálneho počtu funkcie jednej reálnej premennej s podporou programového systému Mathematica, *In.:25 VŠTEP-Z Matematika v inžinierskom vzdelávaní, Trnava, 7. - 10. September 1998, p 173 - 180.*
[4] *Wolfram Research:* The Mathematica Book 3rd ed.,*Wolfram Media/Cambridge University Press, 1996.*
[5] *Záhonová, V.:* Výučba integrálneho počtu funkcie jednej reálnej premennej s podporou programového systému Mathematica, *In.:25 VŠTEP-Z Matematika v inžinierskom vzdelávaní, Trnava, 7. - 10. September 1998, p 192 - 197.*