

---

---

# Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach

Žilina 2.–5. júla 2009



---

## MATHEMATICAL ASSISTANT ON WEB SKLÁDANKA ZE SVOBODNÝCH MATEMATICKÝCH PROGRAMŮ

MAŘÍK, Robert, (CZ)

### 1 Úvod

Cílem tohoto příspěvku je představit základní myšlenky, na nichž je vybudována online aplikace Mathematical Assistant on Web (MAW, [4]). Tato aplikace automatizuje a sama provádí řešení těch úloh základních kurzů matematické analýzy, které jsou do jisté míry jen mechanickým sledem předem daných kroků. Vstupem jsou vždy data pro zadání jisté úlohy a výstupem řešení úlohy, včetně jednotlivých mezikroků. Například při úloze na výpočet derivace je zadáním funkce jedné nebo dvou proměnných a proměnná, podle které se má derivovat. Výstupem je PDF soubor, ve kterém jsou naznačeny jednotlivé kroky potřebné pro výpočet derivace zadané funkce a mezivýsledky po aplikaci těchto jednotlivých kroků.

Aplikace je vybudována se snahou využít v maximální možné míře již hotové Open-Source nástroje pro řešení jednotlivých dílčích úloh. Zpracování těchto úloh je realizováno vždy voláním externího programu, který dílčí úlohu umí vyřešit. Tento přístup je zcela v duchu unixové filozofie používání jednoúčelových nástrojů, které sice řeší třeba jenom jeden problém, řeší ho však efektivně a dobře.

Aplikace je volně přístupná na adrese <http://www.mendelu.cz/user/marik/maw> a její zdrojové kódy jsou k dispozici na <http://mathassistant.sourceforge.net>. Je v provozu od listopadu 2007 a postupně vylepšována, doplňována a opravována. Ve zkušebním období registrujeme cca 3000 přístupů denně.

Problematice výpočtů v prostředí webového prohlížeče jsou věnovány i internetové verze komerčních programů Mathematica a Maple. V naprosté většině výukových aplikací je však nasazení těchto produktů střelba dělovou koulí proti mouše. Proto v závěru uvedeme jednoduchý praktický příklad, který umožňuje vzít z parametru PHP skriptu zadání, zpracovat

toto zadání matematickým softwarem a výsledek zobrazit ve webovém prohlížeči. To vše s minimální znalostí programování. (Dobrá znalost systému počítačové algebry, který tyto výpočty provádí, je naopak nezbytná.)

## 2 Opičí dráha jménem MAW

Skutečnost, že výpočet některých typů úloh vysokoškolské matematiky je zcela mechanický, je jistě nepřehlédnutelná. Při řešení některých úloh pomocí systémů počítačové algebry a zejména při vhodně navržených zápisnicích často stačí pouze měnit vstupní údaje na začátku zápisníku a po přepočítání všech příkazů dostáváme odpovídající výstup. MAW je založen na podobné myšlence, přichází však ještě s jednou další a podstatnou vlastností: je možné jej používat pouze prostřednictvím webového prohlížeče, bez instalace jakéhokoli programu na lokální počítač a bez nutnosti učit se program ovládat nebo v něm zapisovat příkazy. Tím je systém zpřístupněn mnohem většímu počtu zájemců, než by tomu bylo v případě offline zápisníku pro některý ze systémů počítačové algebry. Díky tomu je systém otestován na velice rozsáhlém vzorku matematických problémů. Jistou nevýhodou je nutnost udržování serveru který provádí požadované výpočty a nutnost věnovat pozornost otázkám bezpečnosti a výkonu, aby jeden uživatel neomezil (ať již vědomě či nechtěně) v používání systému ostatní uživatele. V následujících odstavcích se zaměřím na popis procesu, kterým vstupní data procházejí při zpracování systémem MAW.

Cestu vstupních údajů MAWem je možno přirovnat k opičí dráze, kterou musí vstupní data proběhnout, aby se z nich na konci dráhy stal vzorově vyřešený příklad. Hlídač na vstupu provádí základní selekci a jeho úkolem je nepustit na dráhu zraněné, nemocné nebo celkově slabé opičky, u nichž je na první pohled zřejmé, že dráhu neproběhnou bez úhony. Jestliže se opička zdá být v pořádku, je vpuštěna na dráhu tvořenou řetězcem překážek – programů, které se podílejí na zpracování úlohy, které způsobují nejdelsí prodlevy ve zpracování a u kterých je největší riziko neúspěchu. Tyto překážky (OpenSource programy) byly získány zdarma a v některých případech navíc výrobcem přímo upraveny na míru pro naši dráhu<sup>1</sup>. Protože opička je po proběhnutí dráhy zadýchaná a rozčuchaná, je vhodné ji pro závěrečné defilé úspěšných běžců učesat a celkově poupravit (TeXem) případné nedostatky ve vzhledu.

### 2.1 Vstup jen pro zvané

Systém MAW je určen zejména pro slabší studenty matematiky. Je proto nutno očekávat, že uživatelé budou značně neohrabaní při zápisu matematických výrazů. Aby se co největší procento chyb odhalilo ještě před vypuštěním opičky na dráhu, jsou vstupní data testována pomocí regulárních výrazů a seznamu nezvaných hostů, aby byly odchyceny například

- nebezpečné příkazy, zneužitelné pro útok na server;

<sup>1</sup>Například díky ochotě autorů programu `formconv` je MAW v současnosti jediný program, který umí nakreslit graf funkce  $y = \sqrt[3]{1-x^2}$  na množině všech reálných čísel.

- zapomenuté závorky vyznačující rozsah působení funkcí (např.  $\sin x$ );
- zapomenutý znak pro násobení (např.  $2x+8x^3$ ) nebo nepárová závorka;
- použití jiných než explicitně povolených funkcí (erf, hyperbolické funkce), proměnných a parametrů.

Při tvorbě těchto vstupních filtrů bylo mnoho myšlenek převzato z projektu MaximaPHP [7]. Pokud vstupní data sítím těchto filtrů neprojdou, MAW věnuje úsilí k zajištění nápravy. Toto je realizováno jednak opět pomocí regulárních výrazů (například konverze  $2x+8x^3$  na  $2*x+8*x^3$ ), jednak pomocí programu formconv (například konverze výrazu  $\sin^3 8x$  na  $(\sin((8*x))^3)$ ).

## 2.2 Typy překážek

Opičí dráha je sestavena z překážek typu Maxima ([6], systém počítačové algebry, zajišťuje veškeré matematické výpočty), GNUplot ([3], kresba většiny grafů a obrázků), formconv ([1], převod matematických výrazů mezi formáty pro program Maxima, GNUplot a  $\text{\TeX}$ ) a mfpic (kreslení některých obrázků). Jedná se o kvalitní a velkým počtem uživatelů prověřené programy, které patří k nejkvalitnějším nástrojům svého druhu i ve srovnání se svými komerčními alternativami.

Překážka Maxima je jedna z nejtěžších překážek. Protože její zdolání je časově náročné<sup>2</sup>, je většina opičích drah postavena tak, aby se na nich těchto překážek vyskytovalo co nejméně, pokud možno pouze jedna (Lagrangeův polynom, Taylorův polynom, výpočet derivací, . . .). V případě obzvláště obtížných překážek je hlavní překážce předřazena ještě jedna nebo více lehčích překážek Maxima, které provedou podrobnější selekci než byla provedena vstupními filtry. Například při výpočtu dvojného integrálu  $\int_a^b \int_{f(x)}^{g(x)} \phi(x, y) dy dx$  je nutno zkontrolovat, že platí nerovnost  $a < b$  a na intervalu  $(a, b)$  platí  $f(x) \leq g(x)$ . Vstupní data, která touto kontrolou neprojdou, nejsou vpuštěna na další překážku, protože úloha není korektně zadána a případné další výpočty by postrádaly smysluplnou interpretaci.

Další záležitost překážky Maxima spočívá v tom, že opička v překážce někdy zabloudí a ptá se na cestu. Tato situace může nastat například při výpočtu integrálu  $\int \frac{1}{x^2+a} dx$ . Pro další pokračování výpočtu program Maxima potřebuje informaci, jestli je  $a$  kladné, záporné nebo nula. Protože bez další interakce s uživatelem MAWu není možno tuto informaci zjistit, opička by zůstala na překážce viset do nekonečna. Aby k této situaci nedocházelo, má každá překážka typu Maxima svého hlídače<sup>3</sup>, který hlídá aby opička při pokusu o zdolání nepřekročila jistý předem daný časový limit. Pokud se po uplynutí limitu opička z překážky nedostane, hlídač opičku sesadí a oznámí neúspěch jejímu majiteli.

<sup>2</sup>Například pouhé spuštění programu Maxima 5.13 (GNU Common Lisp) trvá více než 0.1 s. Spuštění poslední verze programu Maxima 5.18 (CMU Common Lisp) dokonce 0.7 s.

<sup>3</sup>OpenSource skript timeout, <http://www.shelldorado.com/scripts/cmds/timeout>

Další překážkou je překážka GNUplot. Protože program GNUplot má poněkud neobvyklou formu zápisu matematických výrazů<sup>4</sup>, je této překážce vždy předřazen program pro konverzi matematických výrazů formconv. V některých případech je tato překážka dokonce trojitá a po dvojici formconv+GNUplot ještě následuje jednoduchý filtr v Perlu, jehož úkolem je odstranit z grafů nespojitých funkcí nežádoucí svislé čáry, které se zde někdy objevují.

Při postupném zdolávání překážek opička sbírá certifikáty o jejich zdolání – T<sub>E</sub>Xovské vzorce u překážek Maxima a obrázky u překážek GNUplot.

### 2.3 Příprava na závěrečné defilé

Po zdolání všech překážek čeká na běžce na konci opičí dráhy vizážista L<sup>A</sup>T<sub>E</sub>X. Jeho úkolem je upravit pocuchanou a udýchanou opičku do podoby vhodné pro prezentaci na veřejnosti. Jedná se opět o program s otevřeným zdrojovým kódem, který je velice podrobně prověřen obrovskou uživatelskou základnou, sahající od jednotlivců píšících seminární práce až po velká nakladatelství zabývající se tiskem matematických monografií. Pokud při postupném zdolávání překážek opička získala všechny certifikáty (všechny kroky proběhly bez problémů), je tento poslední úkol poměrně snadný a rychle vyřešený. Výsledek může vypadat jako na obrázku 3.

### 2.4 Zkušenosti provozovatele opičí dráhy

Záznamy o jednotlivých bězcích, snažících se opičí dráhu zdolat, jsou pečlivě evidovány a pokud to čas dovolí, jsou záznamy o jednotlivých výkonech a zejména selháních procházeny a analyzovány. Protože je aplikace volně dostupná na internetu, je v záznamech vždy dostatek materiálu pro testování a pro inspiraci k dalším vylepšením a opravám případných chyb. Vzhledem k velkému počtu zpracování většinou triviálních a stále se opakujících úloh však vzala za své představa autorů, že provozem získají mimo jiné zdroj zajímavých úloh, které budou poté moci použít ve vlastní pedagogické praxi.

Zajímavou a překvapivou zkušeností je, že někteří uživatelé opakovaně vypouštějí na dráhu opičky značně nemocné (například  $\text{odmocnina}*(x+y/2)$ ) a ignorují jakékoliv instrukce týkající se zápisu matematických výrazů. Tímto dáváme zcela za pravdu jiné studii o volně dostupném rozhraní pro online výpočty [2]. V této studii autor mimo jiné dospěl k závěru, že od většiny uživatelů nelze očekávat, že si prostudují jakkoliv krátké instrukce nebo návod k použití programu.

## 3 Praktická ukázka

Jako praktickou ukázkou si uvedme program na řešení kvadratické rovnice pomocí diskriminantu. PHP skript [5] převezme z URL adresy kvadratický výraz nebo kvadratickou rovnici

<sup>4</sup>Například lomítko pro celočíselné dělení.

a zobrazí postupně převod rovnice na základní tvar  $ax^2 + bx + c = 0$ , výpočet diskriminantu a kořenů rovnice a převod levé strany rovnice na čtverec. Vstup je zpracován programem `formconv`, což uživateli umožní například vynechávat znak `*` pro násobení. Veškeré výpočty jsou provedeny programem Maxima, jehož výstup je zobrazen v prohlížeči. Pro snazší čitelnost jsou v tomto výstupu všechny matematické výrazy zapsány v  $\text{T}_\text{E}_\text{X}$  a při zobrazení jsou automaticky nahrazeny obrázky generovanými programem `mathTEX`. Skript je možno vyzkoušet na adrese <http://wood.mendelu.cz/math/zilina/kvadr.php>. Rovnici je nutno předat jako parametr, například `.../kvadr.php?x^2+5x=15`.

Potřebné matematické výpočty jsou prováděny programem Maxima při dávkovém zpracování souboru `kvadr.mac` [5]. Protože je program Maxima „příliš aktivní“ při zpracování výrazu na vstupu, je v některých místech nutno nastavit proměnnou `simp` na hodnotu `false`. Toto nastavení zajistí například, že výraz `2*4*5` nebude automaticky upraven na číslo 40. Skript není příliš sofistikovaný, protože pomocí diskriminantu řeší například i rovnice jako  $x^2 = 3$  nebo  $x(x - 1) = 0$ , což by v praxi použil jenom naprostý matematický zoufalec. Je proto nutné ho chápat pouze jako ukázkou. Tyto speciální případy kvadratické rovnice se ostatně dají ošetřit samostatně a případné pokusy tímto směrem jsou ponechány čtenáři jako cvičení.

## 4 Závěr

V příspěvku byly představeny myšlenky, na nichž je vybudována online aplikace Mathematical Assistant on Web. Tato aplikace automaticky provádí výpočty derivací, určitých a dvojných integrálů, Lagrangeova a Taylorova polynomu, lineárního modelu metodou nejmenších čtverců, definičních oborů funkcí jedné a dvou proměnných, umí řešit základní diferenciální rovnice prvního a druhého řádu a je silnou pomůckou při výpočtu neurčitých integrálů a vyšetřování průběhu funkce. Všechna řešení jsou prezentována včetně mezivýpočtů. Ve většině případů je možno s trochou nadsázky říct, že řešení jsou prezentována tak, jak by je pravděpodobně napsal šikovný student.

Protože přes svou jednoduchost má MAW příznivý ohlas mezi vysokoškolskými studenty, je v závěrečné části uveden praktický příklad sestavení podobné aplikace, která v jednotlivých krocích zobrazuje postup řešení kvadratické rovnice a doplňování výrazu na čtverec.

Aplikace MAW byla sestavena částečně jako pomůcka pro kombinované studium (pro kontrolu výsledků), částečně ze zvědavosti jak se současná výpočetní technika vypořádá s úlohami počítanými na technických školách a univerzitách v základních kurzech matematiky, částečně jako pomůcka pro matematiky kteří věnují svůj čas odpovídání matematických dotazů na internetových fórech, částečně jako nástroj pro pohodlné vymýšlení příkladů do zkouškových písemek, částečně i jako občasný únik od vědecké práce a zpestření života vysokoškolského učitele. Vždy však byla sestavována s myšlenkou, že role živého učitele ve výuce matematiky je zcela nezastupitelná.

## Lineární diferenciální rovnice druhého řádu

<http://user.mendelu.cz/marik/maw>

Řešíme počáteční úlohu  $y'' + 2y' + y = 0$ ;  $y(0) = 1$ ,  $y'(0) = -1$

Nejprve nalezneme obecné řešení rovnice a potom použijeme počáteční podmínky pro nalezení řešení partikulárního.

Řešíme homogenní rovnici

$$y'' + 2y' + y = 0$$

Charakteristická rovnice je  $\lambda^2 + 2\lambda + 1 = 0$

Kořeny charakteristické rovnice:  $\lambda_{1,2} = \frac{-(2) \pm \sqrt{(2)^2 - 4(1)}}{2} = \frac{-2 \pm \sqrt{0}}{2} = -1$

Charakteristická rovnice má jeden dvojnásobný kořen.

Dvě lineárně nezávislá řešení jsou  $y_1 = e^{-x}$  a  $y_2 = xe^{-x}$ .

Obecné řešení je  $y = C_1 e^{-x} + C_2 x e^{-x}$ .

Nyní řešíme počáteční úlohu  $y'' + 2y' + y = 0$ ;  $y(0) = 1$ ,  $y'(0) = -1$

Obecné řešení je  $y(x) = (C_2 x + C_1) e^{-x}$

Derivace obecného řešení je  $y'(x) = C_2 e^{-x} - (C_2 x + C_1) e^{-x}$

Dosazením počátečních podmínek dostáváme soustavu lineárních rovnic

$$\begin{cases} C_1 = 1 \\ C_2 - C_1 = -1 \end{cases}$$

Řešení soustavy lineárních rovnic

$$[C_1 = 1, C_2 = 0]$$

Dosazením vypočtených hodnot do obecného řešení získáme partikulární řešení

$$y(x) = e^{-x}$$

Obr. 1: Ukázka řešení počáteční úlohy pro lineární homogenní LDR s konstantními koeficienty

## Poděkování

Tato práce vznikla s podporou FRVŠ v rámci řešení projektu 99/2008.

## Literatura

- [1] BAKOS, G. – KOVACS, Z.: formconv, <http://particio.com/english/LightNEasy.php?page=formconv>
- [2] FATEMAN, R.: Analysis of a web user interface for mathematics: experiences with integral queries for TILU (Table of Integrals Look Up), <http://www.eecs.berkeley.edu/~fateman/papers/tjames.pdf>
- [3] GNUplot, <http://www.gnuplot.info/>
- [4] MAŘÍK, R. – TIHLAŘÍKOVÁ, M.: Mathematical Assistant on Web. <http://user.mendelu.cz/marik/maw/>
- [5] MAŘÍK, R.: <http://wood.mendelu.cz/math/zilina/kvadr.zip>
- [6] Maxima, <http://maxima.sourceforge.net/>
- [7] Maxima PHP, <http://maximaphp.sourceforge.net/>, <http://www.my-tool.com/mathematics/maximaphp/>

## Kontaktní adresa

**Robert MAŘÍK (doc., Mgr., PhD.),**  
Ústav matematiky MZLU v Brně, Zemědělská 3,  
613 00 Brno,  
[marik@mendelu.cz](mailto:marik@mendelu.cz)