

Zpravodaj Československého sdružení uživatelů TeXu

Robert Mařík

Spolupráce TeXu se systémem počítačové algebry Sage

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 22 (2012), No. 3, 163–175

Persistent URL: <http://dml.cz/dmlcz/150142>

Terms of use:

© Československé sdružení uživatelů TeXu, 2012

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* <http://dml.cz>

Spolupráce \TeX u se systémem počítačové algebry Sage

ROBERT MAŘÍK

Abstrakt

V tomto článku si popíšeme možnosti spolupráce \TeX u a volně šiřitelného systému počítačové algebry Sage. Zaměříme se zejména na převod \LaTeX ového dokumentu do formátu programu Sage a opačný převod z programu Sage do PDF prostřednictvím \PDFLaTeX u. Oba způsoby konverze jsou velmi mladé (závěr roku 2009), využívají však ve velké míře podrobně odzkoušené a odladěné programy, jako například program $\text{\TeX}4ht$. Dále popíšeme možnosti a výhody volání programu Sage uvnitř \LaTeX ového souboru prostřednictvím balíčku Sage \TeX .

Klíčová slova: \LaTeX , Sage, Python, HTML, $\text{\TeX}4ht$, matematika.

Představení programu Sage

Než přistoupíme k popisu způsobů spolupráce mezi programy \TeX a Sage, program Sage si stručně představíme. Tento program je na své domovské stránce umístěn na <http://www.sagemath.org/> uveden následujícím popisem:

Sage is a free open-source mathematics software system licensed under the GPL. It combines the power of many existing open-source packages into a common Python-based interface.

Mission: Creating a viable free open source alternative to Magma, Maple, Mathematica and Matlab.

Sage je velmi mladý systém počítačové algebry – je vyvíjen teprve od roku 2004. Přesto je již poměrně vyspělý a to zejména díky tomu, že v těch oblastech matematiky, které dosud nebyly do programu začleněny nativně, program pouze předá potřebné údaje jinému (třeba i úzce specializovanému) programu a zpracuje jeho výstup. Například řešení diferenciálních rovnic a řešení nerovnic je realizováno voláním programu Maxima¹. I výpočet integrálů je realizován programem Maxima. Pokud však tento výpočet selže, je možno pouhým doplněním volitelného parametru příkazu `integrate` použít integraci jiným programem, včetně

*Zkoumání možností programu Sage při výuce, jeho spolupráce s programem \LaTeX a spoluúčasť na vývoji programu `sws2tex` jsou podporovány grantem 131/2010 FRVŠ *Počítačová podpora výuky matematiky pomocí volně šiřitelného software.*

¹<http://maxima.sourceforge.net/>

programu Mathematica (prostřednictvím volně dostupné webové služby Wolfram Mathematica Online Integrator² – výstup této služby je programem Sage automaticky zpracován a uživatel vidí pouze příslušnou primitivní funkci).

Zajímavé je i pracovní prostředí programu Sage. S programem Sage je možno pracovat buď voláním ze skriptu Pythonu, interaktivně v textové konzoli, nebo pomocí Sage Notebooku v okně internetového prohlížeče. Pro potřeby začátečníků je nejzajímavější právě poslední možnost – práce s programem Sage v prostředí internetového prohlížeče.

Pro přístup k zápisníku programu Sage Notebook (zkráceně Sage) stačí otevřít okno prohlížeče a připojit se na příslušný server. Tímto serverem může být `localhost` v případě instalace na Linuxu, server spuštěný ve virtuálním počítači v případě instalace ve Windows nebo kterýkoliv Sage server na Internetu, ke kterému má uživatel přístup. Protože jsou k dispozici i volně přístupné Sage servery [5], je možno s programem pracovat i bez instalace na lokální počítač.

Vzhledem k tomu, že práce probíhá v internetovém prohlížeči, nečiní první seznámení začátečníkům zpravidla nijak velké potíže. Absence klikacího grafického uživatelského rozhraní známého z programů Maple, Mathematica nebo wxMaxima je vyvážena automatickým doplňováním příkazů, které funguje podobně jako doplňování příkazů v Linuxové konzoli a dále propracovaným systémem nápovědy. V zápisníku je přítomen plugin s populárním editorem TinyMCE³ pro vkládání textových komentářů.

V prostředí internetového prohlížeče je možno používat interaktivní prvky podobné známým `maplet`ům, kdy uživatel mění vstupní veličiny nikoliv přímo v zápisníku programu Sage, ale ve formulářovém okně či pomocí posuvného táhla. Výstup se automaticky přizpůsobuje změnám na vstupu. Nejen tato vlastnost činí program Sage zajímavý jak pro vlastní výpočty a matematické experimenty, tak i pro výuku matematiky na středních a vysokých školách.

Vzhledem k rozšířenosti programu \TeX pro psaní matematických textů se nabízí například otázka, jak je možno začlenit možnosti nabízené programem Sage do našich textů, ať se již jedná o učební materiály nebo výstupy výzkumu.

Formát souboru `sws`

Formát v jakém pracuje se soubory \TeX je čtenářům dostatečně znám. Abychom pochopili možnosti spolupráce programů \TeX a Sage, popíšeme si stručně, jak vypadá formát `sws` souborů, používaných programem Sage. Při práci v prostředí Sage Notebooku se veškerá data ukládají na serveru. Pomocí volby v horním menu lze tato data uložit na lokální počítač ve formě souboru `sws` a poté opět

²<http://integrals.wolfram.com/index.jsp>

³<http://tinymce.moxiecode.com/>

nahrát na libovolný server. Soubor `sws` je ve své podstatě komprimovaný archiv, po jehož rozbalení získáme zejména

- soubory `worksheet.html` a `worksheet.txt`, obsahující vstupy a výstupy programu Sage a případné komentáře vložené mezi tyto vstupy uživatelem,
- adresářovou strukturu, která obsahuje obrázky vytvořené programem Sage a případné další objekty.

Obsah souboru `worksheet.html` může vypadat například takto:

```
<p>Výsledek počtu derivací</p>
{{{id=1|
diff(1/x,x)
///
-1/x^2
}}}
{{{id=3|
(_.show()
plot(x^2,(x,-1,1))
///
<html><div class="math">-\frac{1}{x^2}</div></html>
<html><img src='cell://sage0.png'></html>
}}}
```

Porozumět tomuto textu je snadné, zejména porovnáme-li jej se snímkem obrazovky na Obrázku 1 na straně 4. Z uvedeného příkladu je zřejmé, že zápisník programu Sage obsahuje

- vlastní značky pro oddělení vstupních a výstupních políček,
- HTML značky (zpravidla pouze omezenou množinu značek, vložených buď programem Sage, nebo editorem TinyMCE).

Při převodu mezi programy Sage a \TeX se tedy jedná vlastně o konverzi mezi \TeX em a HTML.

Z \LaTeX u do Sage

Konverze z \LaTeX u do HTML (a Sage) je založena podobně jako [4] na programu $\text{\TeX}4\text{ht}$. Výsledný soubor je poté zpracován postprocesorem `tex2sws`⁴, který na uživatelem definovaných místech vloží značky pro vstupní a výstupní buňky programu Sage, vytvoří potřebnou adresářovou strukturu a vše zabalí do formy `sws` souboru. Tento soubor je pak možno otevřít v programu Sage.

Takto vytvořený Sage zápisník bohužel obsahuje množství HTML tagů vložených programem $\text{\TeX}4\text{ht}$. Například rovnice v display modu jsou zapsané jako tabulky

⁴<http://wiki.sagemath.org/develop/LatexToWorksheet>

Derivace (Sage) - Mozilla Firefox

Soubor Úpravy Zobrazit Historie Záložky Nástroje nápověda

http://um-bc107.mendelu.cz:8000/home/rc sage history

SAGE The Sage
 Notebook
 Version 4.3.1

robert.marik [Toggle](#) | [Home](#) | [Published](#) | [Log](#) | [Settings](#) | [Help](#) | [Report a Problem](#) | [Sign out](#)

Derivace Save Save & quit Discard & quit

last edited on February 02, 2010 04:46 PM by robert.marik

File... Action... Data... sage Typeset

[Print](#) [Worksheet](#) [Edit](#) [Text](#) [Undo](#) [Share](#) [Publish](#)

Výpočet derivace

```
diff(1/x, x)
```

$-1/x^2$

```
(_).show()
plot(x^2, (x, -1, 1))
```

Hotovo

jsMath

Obrázek 1: Práce se zápisníkem programu Sage.

a editace a další práce s tímto textem není příliš pohodlná. Tato vlastnost je však jednak odstranitelná vhodnou konfigurací programu $\text{T}_{\text{E}}\text{X}4\text{ht}$ nebo vhodným rozšířením postprocesoru a není příliš omezující. Je totiž přirozené předpokládat, že čtenář takto vytvořeného materiálu či učebního textu bude převážně experimentovat s příkazy programu Sage a do textu nejvýše doplňovat své vlastní komentáře. Případné úpravy původního textu bude dělat autor v primárním $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ovém zdrojovém souboru. Příkazy programu Sage jsou v $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ovém zdrojovém souboru uloženy v prostředí `sageverbatim`⁵. Vložíme-li do $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ového zdrojového souboru například text

```
Graf funkce nakreslíme příkazem \texttt{plot}:
\begin{sageverbatim}
  plot(x^2, (x, -1, 1))
\end{sageverbatim}
```

Pro nastavení os a dalších parametrů obrázku použijte příkaz `\texttt{show}`.

a zpracujeme-li jej následně programy $\text{T}_{\text{E}}\text{X}4\text{ht}$ a `tex2sws`, obdržíme soubor `sws`, který po načtení do programu Sage obsahuje náš text. Tento text je v místě prostředí `sageverbatim` přerušen a jsou sem vloženy vstupní a výstupní políčka programu Sage. Uživatel může pracovat klasicky jako v každém jiném zápisníku programu Sage, tj. může zejména

- editovat vstupní část políčka a spustit výpočet s jinými parametry (např. nakreslit graf jiné funkce),
- dvojklikem na textovou část vyvolat editor TinyMCE a upravit nebo doplnit doprovodný text dle svých vlastních potřeb.

V případě, že program $\text{T}_{\text{E}}\text{X}4\text{ht}$ rozdělí text do více souborů, obdržíme na výstupu programu `tex2sws` odpovídající počet vzájemně provázaných zápisníků.

Ze Sage do $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u

Další možnost spolupráce programů Sage a $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ocení především uživatelé, pro které je primárním dokumentem zápisník programu Sage a chtějí tento zápisník převést do $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u a následně do PDF. Tento převod je možno realizovat programem `sws2tex`⁶. Ve srovnání s klasickým tiskem do PDF souboru se tento způsob tvorby PDF vyznačuje

- vyšší kvalitou sazby, zejména u matematických výrazů⁷,
- automatickým barevným zvýrazněním programového kódu pro lepší čitelnost,

⁵Viz například ukázka umístěna na <http://bitbucket.org/rbeezer/tex2sws/src/tip/example/example.tex>.

⁶<http://bitbucket.org/whuss/sws2tex/>

⁷Viz např. ukázky na <http://user.mendelu.cz/marik/sage/>.

- přibalením `sws` souboru k výslednému PDF, pro možnost snadno načíst text opět do programu Sage.

Protože `sws` soubor obsahuje HTML kód, není možné očekávat, že takto může být zpracován libovolný vstupní soubor. Nicméně na výstupu současné verze programu `sws2tex` obdržíme správné formátování textu (barva, typ písma, zarovnání odstavců, výčtová prostředí, jednotlivé úrovně nadpisů), obrázky v souborech `pdf`, `png` a `jpg`, kotvy a odkazy a také jednoduché tabulky.

Vstupní pole jsou formátovány přímo prostředím `verbatim`. Pro barevné zvýraznění programového kódu je použita knihovna `pygments`⁸, v případě potřeby však je možno upravit program tak, aby spolupracoval s jiným zvýrazňovačem kódu, například s programem `Highlight` popsaném v jednom z nedávných Zpravodajů [3]. Výstupní pole a případná textová políčka jsou čtena funkcemi modulu `HTMLParser`⁹ a použité tagy jsou nahrazovány odpovídajícím \LaTeX kódem. Program `sws2tex` podporuje i konfigurační soubory, kde je možno definovat vlastní hlavičku vygenerovaného \LaTeX souboru, nastavit jazyk dokumentu, autora a další parametry.

\LaTeX uvnitř Sage a Sage uvnitř \LaTeX

Systém Sage umožňuje přímé použití řady dalších programů. Uvedeme-li na prvním řádku vstupního pole příkaz `%octave`, je vstup zpracován programem `Octave`. Použijeme-li `%latex` následovaný kódem v \LaTeX u, je tento vstupní kód předán programu \LaTeX a následně programu `dvipng`. Sage zobrazí výsledek ve formátu `png`. Pokud řádku se specifikací programu určeného pro zpracování vstupu předradíme ještě řádek `%hide`, zůstane vstup skrytý a vidíme jenom zpracovaný výstup.

Je možno pracovat i naopak a zapisovat příkazy programu Sage přímo do \LaTeX ového zdrojového souboru. Pro co největší pohodlí při tomto způsobu práce slouží balíček `SageTeX`¹⁰. Při kompilaci takového souboru \TeX em je vytvořen pomocný soubor příkazů pro Sage. Tento soubor následně zpracujeme programem Sage a při další kompilaci \TeX em již jsou výsledky těchto výpočtů vloženy na své místo. Práce je tedy podobná začleňování obrázků pomocí `mfpic`. Navíc `SageTeX` umí pracovat nejen s instalací Sage na lokálním počítači, ale je možno výpočty provádět i na některém z veřejných serverů [5]. Je dále vhodné poznamenat, že při použití `SageTeX`u má uživatel k dispozici interpreter programovacího jazyka Python. Tento jazyk je možno použít například k výrobě tabulek, což je demonstrováno v dokumentaci šířené se `SageTeX`em na příkladu Pascalova trojúhelníku. Tím se otevírá možnost poněkud odlišného přístupu k problému

⁸<http://pygments.org/>

⁹<http://docs.python.org/library/htmlparser.html>

¹⁰<http://www.ctan.org/tex-archive/help/Catalogue/entries/sagetex.html>

tvorby tabulek [2]. Je například také možno v L^AT_EXovém dokumentu použít Pythonovský kód uvedený v [1] k tvorbě Hornerova schématu. Autorem zmíněného kódu je pan Michal Kaukič. Tímto způsobem lze do dokumentu vkládat i obrázky, pro jejichž tvorbu Sage používá program `matplotlib`¹¹.

Praktické ukázky

Není bez zajímavosti ukázat použití představených nástrojů v praxi.

Konverzi programem `sws2tex` si ukážeme na příkladu zápisníku programu Sage s ukázkami řešení rovnic a nerovnic. Na Obrázcích 2 až 4 na str. 9 až 11 vidíme nejprve část zápisníku vytvořeného v programu Sage a poté výsledek po zpracování programem `sws2tex`.

Pozn. Špendlík s připojeným `sws` souborem na těchto obrázcích není vidět, protože připojené soubory jsou příkazem `\includegraphics` ignorovány.

Soubor pro SageT_EX s podobnou problematikou může vypadat například následovně. Všimněte si, že se jedná o klasický L^AT_EXový text obohacený o příkazy `\sage` (příkaz programu Sage), `\sageplot` (obrázek vytvořený v programu Sage) a prostředí `sagesilent` (příkazy programu Sage, které se však netisknou).

```
\documentclass{article}
\usepackage{sagetex}
\usepackage[margin=1in]{geometry}
\usepackage[T1]{fontenc}
\usepackage[latin2]{inputenc}
\usepackage[czech]{babel}
\begin{document}
\title{Hrátky s programem Sage}
\maketitle

\section{Řešení rovnic}
\begin{sagesilent}
g(x) = x-cos(x)
\end{sagesilent}
```

Program Sage umožňuje řešení rovnic. Pro řešení rovnice $\text{\sage{g(x) == 0}}$ můžeme použít přímo vestavěné algoritmy a obdržíme $\text{\$x\approx\sage{g.find_root(0,1)}\$}$.

Můžeme také použít vlastní proceduru, například Newtonovu metodu.

¹¹<http://matplotlib.sourceforge.net/>


```
\begin{sagesilent}
gder(x) = diff(g(x),x)
x, pocet = 1, 10
for i in range(pocet):
    x = n(x-g(x)/gder(x),digits=50)
\end{sagesilent}
Výsledkem takového výpočtu po  $\text{\sage{pocet}}$  iterací
je hodnota  $\text{\sage{x}}$ .
```

```
\section{Řešení nerovnic}
Řešení nerovnic využijeme například při hledání intervalů,
kde funkce roste:
\begin{sagesilent}
x = var('x')
f(x) = (x^4+1)/(x-2)^2
sol = solve(diff(f(x),x)>0,x)
\end{sagesilent}
Funkce  $f(x)=\text{\sage{f(x)}}$  roste na následujících
intervalech  $\text{\sage{sol}}$ .
```

Pro kontrolu si můžeme nakreslit graf funkce $y=\text{\sage{f(x)}}$:

```
\begin{sagesilent}
P = plot(f(x), (x, -5, 8), detect_poles=True)
\end{sagesilent}

\begin{center}
\sageplot{P,ymax=100}
\end{center}
```

Hraniční body oddělující intervaly monotonie jsme získali ve tvaru desetinných čísel. Nulové body derivace ale můžeme vypočítat i přesně. Položíme-li v oboru reálných čísel derivaci rovnu nule,

```
\begin{sagesilent}
sol_c = solve(diff(f(x),x)==0,x)
sol_r=[i.rhs() for i in sol_c if i.rhs().imag() == 0]
\end{sagesilent}
obdržíme body  $\text{\sage{sol_r}}$ . Numerickou aproximací obdržíme
následující body:  $\text{\sage{[i.n() for i in sol_r]}}$ .
\end{document}
```

Hrátky s programem Sage (Sage) - Mozilla Firefox

Šoubor Úpravy Zobrazit Historie Záložky Nástroje Nápořádá

http://um-bc107.mendelu.cz:8000/home/robe

Hrátky s programem Sage Save Save & quit Discard & quit

last edited on February 18, 2010 12:47 PM by admin

File... Action... Data... sage Typeset

[Print](#) [Worksheet](#) [Edit](#) [Text](#) [Undo](#) [Share](#) [Publish](#)

Řešení rovnic

Program Sage umožňuje řešení rovnic. Pro řešení rovnice $\cos(x) - x = 0$ můžeme použít přímo vestavěné algoritmy.

```
g(x)=cos(x)-x
g.find_root(0,1)
```

0.739085133215

Můžeme také naprogramovat vlastní proceduru, například Newtonovu metodu.

```
gder(x) = diff(g(x), x)
x, pocet = 1, 10
for i in range(pocet):
    x = n(x-g(x)/gder(x), digits=50)
x
```

0.73908513321516064165531208767387340401341175890076

Řešení nerovnic

Řešení nerovnic využijeme například při hledání intervalů, kde funkce roste:

```
x = var('x')
f(x) = (x^4+1)/(x-2)^2
sol = solve(diff(f(x), x)>0, x)
sol
```

$[x > (-0.601231838282), x < 2], [x > 4.0154454023]$

Tento výsledek můžeme ověřit na grafu funkce

Hotovo jsMath

Obrázek 2: Zápisník programu Sage.

Hrátky s programem Sage

Robert Mařík

18. února 2010

1 Řešení rovnic

Program Sage umožňuje řešení rovnic. Pro řešení rovnice $\cos(x) - x = 0$ můžeme použít přímo vestavěné algoritmy.

```
_____ Sage code _____  
g(x)=cos(x)-x  
g.find_root(0,1)
```

[0.739085133215](#)

Můžeme také naprogramovat vlastní proceduru, například Newtonovu metodu.

```
_____ Sage code _____  
gder(x) = diff(g(x),x)  
x, pocet = 1, 10  
for i in range(pocet):  
    x = n(x-g(x)/gder(x),digits=50)  
x
```

[0.73908513321516064165531208767387340401341175890076](#)

2 Řešení nerovnic

Řešení nerovnic využijeme například při hledání intervalů, kde funkce roste:

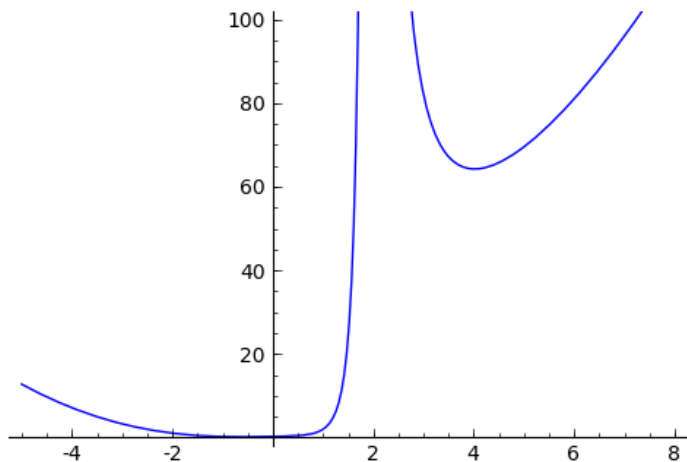
```
_____ Sage code _____  
x = var('x')  
f(x) = (x^4+1)/(x-2)^2  
sol = solve(diff(f(x),x)>0,x)  
sol
```

[\[\[x > \(-0.601231838282\), x < 2\], \[x > 4.0154454023\]\]](#)

Tento výsledek můžeme ověřit na grafu funkce

```
_____ Sage code _____  
plot(f(x), (x,-5,8), detect_poles=True).show(ymax=100)
```

Obrázek 3: Výstup programu `sws2tex`, strana 1.



Pro kontrolu můžeme ještě vypočítat nulové body derivace přesně. Položíme-li v oboru reálných čísel derivaci rovnu nule, obdržíme dva stacionární body:

```
Sage code
sol_c = solve(diff(f(x),x)==0,x)
sol_r=[i.rhs() for i in sol_c if i.rhs().imag() == 0]
sol_r
```

$$\left[-\frac{1}{2}\sqrt{8\sqrt{2}+10} + \frac{1}{2}\sqrt{2} + 1, \frac{1}{2}\sqrt{8\sqrt{2}+10} + \frac{1}{2}\sqrt{2} + 1 \right]$$

```
Sage code
n(sol_r[0]),n(sol_r[1])
```

(-0.601231825852331, 4.01544538822543)

Obrázek 4: Výstup programu `sws2tex`, strana 2.

Dokument na Obrázku 5 na straně 12 obdržíme po zpracování (například) touto sekvencí příkazů:

```
pdflatex file.tex
sage file.sage
pdflatex file.tex
```

Závěr

V článku byly popsány možnosti spolupráce typografického systému $\text{T}_{\text{E}}\text{X}$ se systémem počítačové algebry Sage, zejména způsoby konverze dat a volání jednoho programu uvnitř druhého.

Hrátky s programem Sage

18. února 2010

1 Řešení rovnic

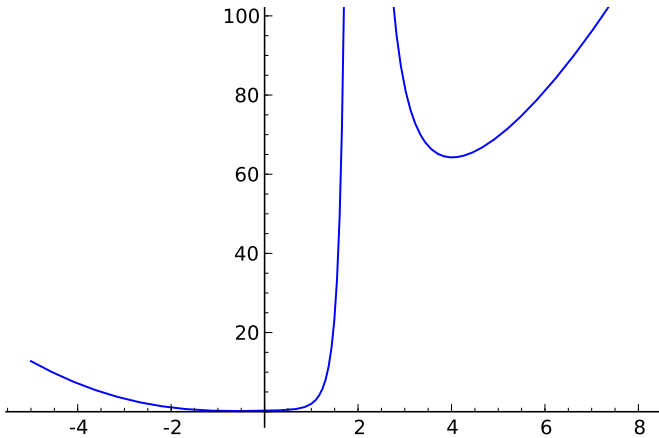
Program Sage umožňuje řešení rovnic. Pro řešení rovnice $x - \cos(x) = 0$ můžeme použít přímo vestavěné algoritmy a obdržíme $x \approx 0.739085133215$.

Můžeme také použít vlastní proceduru, například Newtonovu metodu. Výsledkem takového výpočtu po 10 iteracích je hodnota $x \approx 0.73908513321516064165531208767387340401341175890076$.

2 Řešení nerovnic

Řešení nerovnic využijeme například při hledání intervalů, kde funkce roste: Funkce $f(x) = \frac{(x^4+1)}{(x-2)^2}$ roste na následujících intervalech $[x > (-0.601231838282), x < 2], [x > 4.0154454023]$.

Pro kontrolu si můžeme nakreslit graf funkce $y = \frac{(x^4+1)}{(x-2)^2}$:



Hraniční body oddělující intervaly monotonie jsme získali ve tvaru desetinných čísel. Nulové body derivace ale můžeme vypočítat i přesně. Položíme-li v oboru reálných čísel derivaci rovnu nule, obdržíme body $\left[-\frac{1}{2}\sqrt{8\sqrt{2}+10} + \frac{1}{2}\sqrt{2}+1, \frac{1}{2}\sqrt{8\sqrt{2}+10} + \frac{1}{2}\sqrt{2}+1\right]$. Numerickou aproximací obdržíme následující body: $[-0.601231825852331, 4.01544538822543]$.

Obrázek 5: Výstup programu SageTeX.

Seznam literatury

- [1] Blaško, Rudolf. \LaTeX a neobvyklé výpočty v pevné rádo­vej čiarke. [\LaTeX and Unusual Fixed Point Computations.] *7th International Conference Aplimat* (2008) [7. mezinárodní konference Aplimat], Part IV – Open Source Software in Research and Education, pp. 9–18. ISBN 978-80-89313-04-4. Též online: <http://www.sospreskoly.org/latex-a-neobvykle-vypocty-v-pevnej-radovej-ciarke>
- [2] Hlávka, Zdeněk. Velkovýroba tabulek pomocí AWK. [Large-scale Production of Tables in AWK.] *Zpravodaj Československého sdružení uživatelů \TeX* [The Bulletin of the Czechoslovak \TeX Users Group], Vol. 18 (1–2), pp. 93–95, 2008. ISSN 1211–6661.
- [3] Simon, André. The Highlight programme: code & syntax highlighting. [Program Highlight a jeho využití.] *Zpravodaj Československého sdružení uživatelů \TeX* [The Bulletin of the Czechoslovak \TeX Users Group], Vol. 19 (4), pp. 222–239, 2009. ISSN 1211–6661.
- [4] Sojka, Petr; Růžička, Michal. Publikování z jednoho zdroje v odlišných formátech pro různá vstupní zařízení. [Parallel Electronic Publications.] *Zpravodaj Československého sdružení uživatelů \TeX* [The Bulletin of the Czechoslovak \TeX Users Group], Vol. 18 (3), pp. 116–129, 2008. ISSN 1211–6661.
- [5] Stein, William et al. *The Sage notebook*. [online cit. 16. 2. 2010]
URL: <http://www.sagenb.org/>, <http://sagenb.kaist.ac.kr/>

Summary: Cooperation between \TeX and computer algebra system Sage

In this paper we describe recent progress in cooperation between \TeX and open-source computer algebra system Sage. The paper is focused on possibilities to call one program inside the other one and on data conversion.

Keywords: \LaTeX , Sage, Python, HTML, \TeX 4ht, mathematics.

*Robert Mařík, marik@mendelu.cz
Ústav matematiky, Lesnická a dřevařská fakulta
Mendelova univerzita v Brně, Zemědělská 1
CZ-621 00 Brno, Czech Republic*