

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačů



Bakalářská práce

Demonstrační aplikace pro podporu kurzu neuronových sítí

Adam Činčura

Vedoucí práce: Ing. Zdeněk Buk

Studijní program: Softwarové technologie a management, Bakalářský

Obor: Softwarové inženýrství

3. května 2011

Poděkování

Děkuji Ing. Zdeňku Bukovi za užitečné rady a připomínky k tvorbě této bakalářské práci.

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 24. 5. 2011

.....

Abstract

Demonstrational application created in this bachelor thesis is supposed to complete electronic attachment of lecture notes for the "Neural networks and neurocomputers" course. Application is created in Mathematica program with use of NeuralNetworks library. Every file allows to experiment with specific neural network or its parameters. All experiments have enclosed explaining commentary. Final application was created by extension and completion of current demonstrational application.

Abstrakt

Demonstrační aplikace vytvořená v rámci této bakalářské práce doplňuje elektronickou přílohu skript pro předmět „Neuronové sítě a neuropočítače“. Aplikace je vytvořena v programu Mathematica s použitím knihovny NeuralNetworks. Každý soubor umožňuje snadno experimentovat s danou neuronovou sítí nebo s jejími parametry. Všechny experimenty jsou doprovázeny vysvětlujícím komentářem. Výsledná aplikace vznikla rozšířením a doplněním stávající demonstrační aplikace.

Obsah

1	Úvod	1
1.1	Vymezení cíle práce	1
1.2	Struktura práce	1
2	Umělé neuronové sítě	3
2.1	Historie	3
2.2	Aplikace	3
2.3	Využití	3
3	Umělý neuron	5
3.1	Struktura neuronu	5
3.2	Proces učení	5
3.3	Učící algoritmy	5
3.3.1	Levenberg-Marquardt algoritmus	5
3.3.2	Gauss-Newton algoritmus	5
3.3.3	Algoritmus nejvyššího poklesu	5
3.3.4	Algoritmus zpětné propagace	5
4	Neuronové sítě	7
4.1	Dopředná síť	7
4.2	RBF síť	7
4.3	Samoorganizující mapy	7
4.4	Hopfieldova síť	7
4.5	LVQ	7
5	Učení sítě	9
5.1	Rozdělení vstupních dat	9
5.2	Křížová validace	9
6	Implementační prostředí	11
6.1	Wolfram Mathematica	11
6.2	Knihovna NeuralNetworks	11
7	Experimenty s neuronovými sítěmi	13
7.1	Úvod	13
7.2	Učící algoritmy	13

7.3	Křížová validace	13
7.4	Dopředná síť	14
7.4.1	Jednoduchá data	14
7.4.2	Iris data	14
7.4.3	Různé přístupy k učení sítě	14
7.5	RBF síť	14
7.5.1	Ukázka výstupu skrytého neuronu	14
7.5.2	Ukázka výstupu tří skrytých neuronů	14
7.5.3	Jednoduchá data	14
7.5.4	Iris data	14
7.5.5	Různé přístupy k učení sítě	14
7.6	Hopfieldova síť	14
7.7	Samoorganizující mapy	14
7.7.1	Iris data	14
7.7.2	Shlukování Iris dat	14
7.8	LVQ	14
8	Závěr	15
A	Testování zaplnění stránky a odsazení odstavců	19
B	Pokyny a návody k formátování textu práce	23
B.1	Vkládání obrázků	23
B.2	Kreslení obrázků	24
B.3	Tabulky	24
B.4	Odkazy v textu	25
B.4.1	Odkazy na literaturu	25
B.4.2	Odkazy na obrázky, tabulky a kapitoly	27
B.5	Rovnice, centrovaná, číslovaná matematika	27
B.6	Kódy programu	28
B.7	Další poznámky	28
B.7.1	České uvozovky	28
C	Seznam použitých zkratk	29
D	UML diagramy	31
E	Instalační a uživatelská příručka	33
F	Obsah přiloženého CD	35

Seznam obrázků

B.1	Popiska obrázku	24
F.1	Seznam přiloženého CD — příklad	35

Seznam tabulek

B.1 Ukázka tabulky	24
------------------------------	----

Kapitola 1

Úvod

Problematika neuronových sítí není triviální záležitostí. Pro pochopení principu fungování libovolné neuronové sítě je potřeba porozumět teoretickému fungování sítě. Tohoto porozumění se dosáhne snáze, pokud má student možnost si danou síť vlastnoručně osahat, nastavovat si její parametry, pozorovat výstupy a předkládat síti vlastní data. Z těchto důvodů má předmět „Neuronové sítě a neuropočítače“, v rámci kterého je na FEL ČVUT problematika neuronových sítí vyučována, rozsáhlou elektronickou přílohu skript tzv. Courseware[2].

Součástí Courseware je také demonstrační aplikace v programu Mathematica, kterou jsem v rámci této bakalářské práce přepracoval a rozšířil. Demonstrační aplikace nevyžaduje téměř žádné znalosti programování v systému Mathematica, umožňuje velmi snadno měnit parametry a strukturu zadaných sítí, vizualizovat průběh učení sítě, vizualizovat výstup sítě a upravovat vstupní data. Celá aplikace je doplněna vysvětlujícím komentářem. Věřím, že moje práce pomůže studentům snáze se v neuronových sítích zorientovat.

1.1 Vymezení cíle práce

Cílem práce je provést úpravy a rozšíření stávající demonstrační aplikace. Výstup této práce nahradí stávající aplikaci a bude používán při výuce předmětu „Neuronové sítě a neuropočítače“.

1.2 Struktura práce

TODO až bude hotovo

Kapitola 2

Umělé neuronové sítě

2.1 Historie

2.2 Aplikace

2.3 Využití

Kapitola 3

Umělý neuron

3.1 Struktura neuronu

3.2 Proces učení

3.3 Učicí algoritmy

3.3.1 Levenberg-Marquardt algoritmus

3.3.2 Gauss-Newton algoritmus

3.3.3 Algoritmus nejvyššího poklesu

3.3.4 Algoritmus zpětné propagace

Kapitola 4

Neuronové sítě

4.1 Dopředná síť

4.2 RBF síť

4.3 Samoorganizující mapy

4.4 Hopfieldova síť

4.5 LVQ

Kapitola 5

Učení sítě

5.1 Rozdělení vstupních dat

5.2 Křížová validace

Kapitola 6

Implementační prostředí

6.1 Wolfram Mathematica

6.2 Knihovna NeuralNetworks

Kapitola 7

Experimenty s neuronovými sítěmi

7.1 Úvod

Úvodní kapitola seznamuje studenty s používáním programu *Mathematica* od naprostých základů. Nejprve je probrána základní syntaxe, dále jsou studenti seznámeni s používáním proměnných, funkcí a s možnostmi vizualizace dat. V další části notebooku jsou probrány základy programování v systému *Mathematica*. Prostudování tohoto úvodu umožní rychleji se orientovat v experimentech s neuronovými sítěmi a také dá studentovi znalosti potřebné k úpravám jednotlivých jednotlivých experimentů. Tato kapitola se nachází v souboru *"01-uvod.nb"*.

Soubor vznikl úpravou souboru Petra Chlumského. Změnil jsem grafický vzhled notebooku, změnil a rozšířil jsem textový komentář k jednotlivým příkazům.

7.2 Učící algoritmy

Notebook učící algoritmy demonstruje použití různých učících algoritmů. Pro každý učící algoritmus je graficky zobrazen jeho postup při učení. Jsou zde demonstrovány tyto algoritmy: *Levenberg-Marquardt algoritmus*, *Gauss-Newton algoritmus*, *Algoritmus nejvyššího poklesu (Steepest Descent)* a *Algoritmus zpětné propagace (Backpropagation)*. Učení je ukázáno na jednoduché dopředné síti s jedním neuronem, jedním vstupem a jedním výstupem. Nejprve je vytvořena potřebná síť, poté jsou pomocí této sítě vygenerovány data, na kterých budeme demonstrovat učení. Poté je pro každý algoritmus ukázáno jakým způsobem ho použít pro učení sítě a je graficky zobrazen průběh učení. U algoritmu zpětné propagace je potřeba nastavit délku kroku (`stepLength`) a momentum. Jak tyto parametry ovlivňují průběh učení je ilustrováno pomocí interaktivního grafu, kde je možné si pomocí posuvníků měnit hodnotu těchto parametrů, graf se automaticky překresluje podle aktuálně nastavených hodnot parametrů. Tato kapitola se nachází v souboru *02-algoritmy-uceni.nb*.

7.3 Křížová validace

V kapitole křížová validace se student seznámí s principem fungování křížové validace. Křížová validace je ukázána na Iris datech z databáze UCI[1]. Nejprve jsou načteny data,

poté je vytvořena neuronová síť, která je pomocí křížové validace otestována. Iris data jsou rozdělena na vstupní a výstupní vektory, výstupní vektory jsou zakódovány kódy 1 z N. Poté je detailně rozebráno předzpracování dat pro křížovou validaci. Následuje implementace samotné křížové validace, tato implementace vypíše pro každý fold úspěšnost RMSE jakou síť dosáhla. Po skončení všech kol křížové validace je zobrazen boxový graf s výsledky křížové validace. Po najetí myši na graf se k němu zobrazí legenda. Student má také možnost nechat si vypsat výsledky křížové validace v textovém formátu. Textový formát vypíše nejlepší výsledek, nejhorší výsledek a aritmetický průměr všech výsledků. Na závěr je umožněno studentovi nechat si provést křížovou validaci vlastní sítě s vlastními daty. Vlastní křížovou validaci nastaví parametry jednoduchým přiřazením do proměnných, je potřeba aby zadaná síť odpovídala zadaným vstupním a výstupním datům a data byla předzpracována pomocí postupu uvedeného v této kapitole.

7.4 Dopředná síť

7.4.1 Jednoduchá data

7.4.2 Iris data

7.4.3 Různé přístupy k učení sítě

7.5 RBF síť

7.5.1 Ukázka výstupu skrytého neuronu

7.5.2 Ukázka výstupu tří skrytých neuronů

7.5.3 Jednoduchá data

7.5.4 Iris data

7.5.5 Různé přístupy k učení sítě

7.6 Hopfieldova síť

7.7 Samoorganizující mapy

7.7.1 Iris data

7.7.2 Shlukování Iris dat

7.8 LVQ

Kapitola 8

Závěr

- Zhodnocení splnění cílů DP/BP a vlastního přínosu práce (při formulaci je třeba vzít v potaz zadání práce).
- Diskuse dalšího možného pokračování práce.

Literatura

- [1] *UCI - Machine Learning Repository* [online]. 2011. [cit. 22. 4. 2011]. Dostupné z: <<http://archive.ics.uci.edu/ml/>>.
- [2] *Courseware* [online]. 2011. [cit. 19. 4. 2011]. Dostupné z: <<http://neuron.felk.cvut.cz/courseware/>>.
- [3] HAINDL, M. – KMENT, Ľ. – SLAVÍK, P. Virtual Information Systems. In *WSCG'2000 – Short communication papers*, s. 22–27. University of West Bohemia, Pilsen, 2000.
- [4] Příspěvatelé Wikipedie. *Framework* [online]. 2009. [cit. 10. 9. 2009]. Dostupné z: <<http://cs.wikipedia.org/wiki/Framework>>.
- [5] Příspěvatelé Wikipedie. *Object-relational mapping* [online]. 2009. [cit. 6. 12. 2009]. Dostupné z: <http://en.wikipedia.org/wiki/Object-relational_mapping>.
- [6] ŽÁRA, J. – BENEŠ, B. – FELKEL, P. *Moderní počítačová grafika*. Computer Press s.r.o, Brno, 1st edition, 1998. In Czech.
- [7] SLAVÍK, P. Grammars and Rewriting Systems as Models for Graphical User Interfaces. *Cognitive Systems*. 1997, 4, 3/4, s. 381–399.
- [8] web:cstug. CSTUG — $\mathcal{C}\mathcal{S}\mathcal{T}\mathcal{E}\mathcal{X}$ Users Group — hlavní stránka. <http://www.cstug.cz/>, stav z 2. 3. 2009.
- [9] web:infodp. K336 Info — pokyny pro psaní diplomových prací. <https://info336.felk.cvut.cz/clanek.php?id=400>, stav ze 4. 5. 2009.
- [10] web:infogs. Knihovna Grafické skupiny. <http://www.cgg.cvut.cz/Bib/library/>, stav z 30. 8. 2001.
- [11] web:ipe. Grafický vektorový editor pro práce vhodný pro práci $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ em. <http://tclab.kaist.ac.kr/ipe/>, stav z 4. 5. 2009.
- [12] web:latexdocweb. $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ — online manuál. <http://www.cstug.cz/latex/lm/frames.html>, stav ze 4. 5. 2009.
- [13] web:latexwiki. Wiki Books $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$. <http://en.wikibooks.org/wiki/LaTeX/>, stav z 3. 4. 2009.

Příloha A

Testování zaplnění stránky a odsazení odstavců

Tato příloha nebude součástí vaší práce. Slouží pouze jako příklad formátování textu.

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Příloha B

Pokyny a návody k formátování textu práce

Tato příloha samozřejmě nebude součástí vaší práce. Slouží pouze jako příklad formátování textu.

Používat se dají všechny příkazy systému L^AT_EX. Existuje velké množství volně přístupné dokumentace, tutoriálů, příruček a dalších materiálů v elektronické podobě. Výchozím bodem, kromě Googlu, může být stránka CSTUG (Czech Tech Users Group) [8]. Tam najdete odkazy na další materiály. Většinou dostačující a přehledně organizovanou elektronikou dokumentaci najdete například na [12] nebo [13].

Existují i různé nadstavby nad systémy T_EX a L^AT_EX, které výrazně usnadní psaní textu zejména začátečníkům. Velmi rozšířený v Linuxovém prostředí je systém Kile.

B.1 Vkládání obrázků

Obrázky se umísťují do plovoucího prostředí **figure**. Každý obrázek by měl obsahovat **název** (`\caption`) a **návěští** (`\label`). Použití příkazu pro vložení obrázku `\includegraphics` je podmíněno aktivací (načtením) balíku `graphicx` příkazem `\usepackage{graphicx}`.

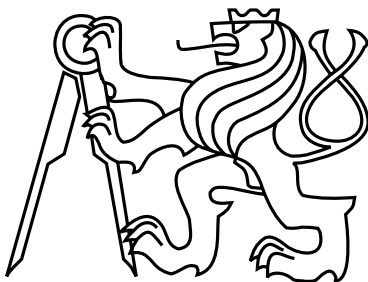
Budete-li zdrojový text zpracovávat pomocí programu `pdflatex`, očekávají se obrázky s příponou `*.pdf`¹, použijete-li k formátování `latex`, očekávají se obrázky s příponou `*.eps`.²

Příklad vložení obrázku:

```
\begin{figure}[h]
\begin{center}
\includegraphics[width=5cm]{figures/LogoCVUT}
\caption{Popiska obrazku}
\label{fig:logo}
```

¹pdflatex umí také formáty PNG a JPG.

²Vzájemnou konverzi mezi snad všemi typy obrázku včetně změn velikostí a dalších vymožeností vám může zajistit balík ImageMagick (<http://www.imagemagick.org/script/index.php>). Je dostupný pod Linuxem, Mac OS i MS Windows. Důležité jsou zejména příkazy `convert` a `identify`.



Obrázek B.1: Popiska obrázku

DTD	construction	elimination
	in1 A B a:sum A B in1 A B b:sum A B	case([_:A]a)([_:B]a)ab:A case([_:A]b)([_:B]b)ba:B
+	do_reg:A -> reg A	undo_reg:reg A -> A
*, ?	the same like and + with empty_el:empty	the same like and + with empty_el:empty
R(a,b)	make_R:A->B->R	a: R -> A b: R -> B

Tabulka B.1: Ukázka tabulky

```
\end{center}
\end{figure}
```

B.2 Kreslení obrázků

Zřejmě každý z vás má nějaký oblíbený nástroj pro tvorbu obrázků. Jde jen o to, abyste dokázali obrázek uložit v požadovaném formátu nebo jej do něj konvertovat (viz předchozí kapitola). Je zřejmě vhodné kreslit obrázky vektorově. Celkem oblíbený, na ovládání celkem jednoduchý a přitom dostatečně mocný je například program Inkscape.

Zde stojí za to upozornit na kreslicí programe Ipe [11], který dokáže do obrázku vkládat komentáře přímo v latexovském formátu (vzroce, stejné fonty atd.). Podobné věci umí na Linuxové platformě nástroj Xfig.

Za pozornost ještě stojí schopnost editoru Ipe importovat obrázek (jpg nebo bitmap) a krelit do něj latexovské popisky a komentáře. Výsledek pak umí exportovat přímo do pdf.

B.3 Tabulky

Existuje více způsobů, jak sázet tabulky. Například je možno použít prostředí `table`, které je velmi podobné prostředí `figure`.

Zdrojový text tabulky B.1 vypadá takto:

```

\begin{table}
\begin{center}
\begin{tabular}{|c|l|l|l|}
\hline
\textbf{DTD} & \textbf{construction} & \textbf{elimination} & \\
\hline
 $\mid$  &  $\verb+in1|A|B$  a:sum A B+ &  $\verb+case([_:A]a)([_:B]a)ab:A+\backslash\backslash$  & 
 $\verb+in1|A|B$  b:sum A B+ &  $\verb+case([_:A]b)([_:B]b)ba:B+\backslash\backslash$  \\
\hline
 $\$$  &  $\verb+do\_reg:A \rightarrow reg A+$  &  $\verb+undo\_reg:reg A \rightarrow A+\backslash\backslash$  & \\
\hline
 $\$,? \$$  & the same like  $\mid$  & and  $\$$  & the same like  $\mid$  and  $\$$  \\
& with  $\verb+empty\_el:empty+$  & with  $\verb+empty\_el:empty+\backslash\backslash$  & \\
\hline
 $R(a,b)$  &  $\verb+make\_R:A\rightarrow B\rightarrow R+$  &  $\verb+a: R \rightarrow A+\backslash\backslash$  & 
&  $\verb+b: R \rightarrow B+\backslash\backslash$  \\
\hline
\end{tabular}
\end{center}
\caption{Ukázka tabulky}
\label{tab:tab1}
\end{table}
\begin{table}

```

B.4 Odkazy v textu

B.4.1 Odkazy na literaturu

Jsou realizovány příkazem `\cite{odkaz}`.

Seznam literatury je dobré zapsat do samostatného souboru a ten pak zpracovat programem bibtex (viz soubor `reference.bib`). Zdrojový soubor pro bibtex vypadá například takto:

```

@Article{Chen01,
  author   = "Yong-Sheng Chen and Yi-Ping Hung and Chiou-Shann Fuh",
  title    = "Fast Block Matching Algorithm Based on
              the Winner-Update Strategy",
  journal  = "IEEE Transactions On Image Processing",
  pages    = "1212--1222",
  volume   = 10,
  number   = 8,
  year     = 2001,
}

```

```

@Misc{latexdocweb,

```

```

author = "",
title = "{\LaTeX} --- online manuál",
note = "\verb|http://www.cstug.cz/latex/lm/frames.html|",
year = "",
}
...

```

Pozor: Sazba názvů odkazů je dána BibTeX stylem (`\bibliographystyle{abbrv}`). BibTeX tedy obvykle vysází velké pouze počáteční písmeno z názvu zdroje, ostatní písmena zůstanou malá bez ohledu na to, jak je napíšete. Přesněji řečeno, styl může zvolit pro každý typ publikace jiné konverze. Pro časopisecké články třeba výše uvedené, jiné pro monografie (u nich často bývá naopak velikost písmen zachována).

Pokud chcete BibTeXu napovědět, která písmena nechat bez konverzí (viz `title = "{\LaTeX} --- online manuál"` v předchozím příkladu), je nutné příslušné písmeno (zde celé makro) uzavřít do složených závorek. Pro přehlednost je proto vhodné celé parametry uzavírat do uvozovek (`author = "..."`), nikoliv do složených závorek.

Odkazy na literaturu ve zdrojovém textu se pak zapisují:

```

Podívejte se na \cite{Chen01},
další detaily najdete na \cite{latexdocweb}

```

Vazbu mezi soubory `*.tex` a `*.bib` zajistíte příkazem `\bibliography{}` v souboru `*.tex`. V našem případě tedy zdrojový dokument `thesis.tex` obsahuje příkaz `\bibliography{reference}`.

Zpracování zdrojového textu s odkazy se provede postupným voláním programů `pdflatex <soubor>` (případně `latex <soubor>`), `bibtex <soubor>` a opět `pdflatex <soubor>`.³

Níže uvedený příklad je převzat z dříve existujících pokynů studentům, kteří dělají svou diplomovou nebo bakalářskou práci v Grafické skupině.⁴ Zde se praví:

```

...
j) Seznam literatury a dalších použitých pramenů, odkazy na WWW stránky, ...
Pozor na to, že na veškeré uvedené prameny se musíte v textu práce
odkazovat -- [1].
Pramen, na který neodkazujete, vypadá, že jste ho vlastně nepotřebovali
a je uveden jen do počtu. Příklad citace knihy [1], článku v časopise [2],
statí ve sborníku [3] a html odkazu [4]:
[1] J. Žára, B. Beneš;, and P. Felkel.
    Moderní počítačová grafika. Computer Press s.r.o, Brno, 1 edition, 1998.
    (in Czech).

```

³První volání `pdflatex` vytvoří soubor s koncovkou `*.aux`, který je vstupem pro program `bibtex`, pak je potřeba znovu zavolat program `pdflatex` (`latex`), který tentokrát zpracuje soubory s příponami `.aux` a `.tex`. Informaci o případných nevyřešených odkazech (cross-reference) vidíte přímo při zpracovávání zdrojového souboru příkazem `pdflatex`. Program `pdflatex` (`latex`) lze volat vícekrát, pokud stále vidíte nevyřešené závislosti.

⁴Několikrát jsem byl upozorněn, že web s těmito pokyny byl zrušen, proto jej zde přímo necituji. Nicméně příklad sám o sobě dokumentuje obecně přijímaný konsensus ohledně citací v bakalářských a diplomových pracích na KP.

- [2] P. Slavík. Grammars and Rewriting Systems as Models for Graphical User Interfaces. *Cognitive Systems*, 4(4--3):381--399, 1997.
- [3] M. Haindl, Š. Kment, and P. Slavík. Virtual Information Systems. In *WSCG'2000 -- Short communication papers*, pages 22--27, Pilsen, 2000. University of West Bohemia.
- [4] Knihovna grafické skupiny katedry počítačů:
<http://www.cgg.cvut.cz/Bib/library/>

... abychom výše citované odkazy skutečně našli v (automaticky generovaném) seznamu literatury tohoto textu, musíme je nyní alespoň jednou citovat: Kniha [6], článek v časopisu [7], příspěvek na konferenci [3], [www odkaz](#) [10].

Ještě přidáme další ukázkou citací online zdrojů podle české normy. Odkaz na wiki o frameworkch [4] a ORM [5]. Použití viz soubor `reference.bib`. V seznamu literatury by nyní měly být živé odkazy na zdroje. V `reference.bib` je zcela nový typ publikace. Detaily dohledal a dodal Petr Dlouhý v dubnu 2010. Podrobnosti najdete ve zdrojovém souboru tohoto textu v komentáři u příkazu `\thebibliography`.

B.4.2 Odkazy na obrázky, tabulky a kapitoly

- Označení místa v textu, na které chcete později čtenáře práce odkázat, se provede příkazem `\label{navesti}`. Lze použít v prostředích `figure` a `table`, ale též za názvem kapitoly nebo podkapitoly.
- Na návěští se odkážeme příkazem `\ref{navesti}` nebo `\pageref{navesti}`.

B.5 Rovnice, centrováná, číslovaná matematika

Jednoduchý matematický výraz zapsaný přímo do textu se vysází pomocí prostředí `math`, resp. zkrácený zápis pomocí uzavření textu rovnice mezi znaky `$`.

Kód `$ S = \pi * r^2 $` bude vysázen takto: $S = \pi * r^2$.

Pokud chcete nečíslované rovnice, ale umístěné centrovane na samostatné řádky, pak lze použít prostředí `displaymath`, resp. zkrácený zápis pomocí uzavření textu rovnice mezi znaky `$$`. Zdrojový kód: `$$$ S = \pi * r^2 $$$` bude pak vysázen takto:

$$S = \pi * r^2$$

Chcete-li mít rovnice číslované, je třeba použít prostředí `equation`. Kód:

```
\begin{equation}
S = \pi * r^2
\end{equation}
```

```
\begin{equation}
V = \pi * r^3
\end{equation}
```

je potom vysázen takto:

$$S = \pi * r^2 \quad (\text{B.1})$$

$$V = \pi * r^3 \quad (\text{B.2})$$

B.6 Kódy programu

Chceme-li vysázet například část zdrojového kódu programu (bez formátování), hodí se prostředí *verbatim*:

```
(* nickname2 *)
Lego> Refine in1
      (do_reg (nickname1 h));
Refine by in1 (do_reg (nickname1 h))
  ?4 : pcddata
  ?5 : pcddata
      (* surname2 *)
Lego> Refine surname1 h;
Refine by surname1 h
  ?5 : pcddata
      (* email2 *)
Lego> Refine undo_reg (email1 h);
Refine by undo_reg (email1 h)
*** QED ***
```

B.7 Další poznámky

B.7.1 České uvozovky

V souboru `k336_thesis_macros.tex` je příkaz `\uv{}` pro sázení českých uvozovek. „Text uzavřený do českých uvozovek.“

Příloha C

Seznam použitých zkratk

2D Two-Dimensional

ABN Abstract Boolean Networks

ASIC Application-Specific Integrated Circuit

⋮

Příloha D

UML diagramy

Tato příloha není povinná a zřejmě se neobjeví v každé práci. Máte-li ale větší množství podobných diagramů popisujících systém, není nutné všechny umísťovat do hlavního textu, zvláště pokud by to snižovalo jeho čitelnost.

Příloha E

Instalační a uživatelská příručka

Tato příloha velmi žádoucí zejména u softwarových implementačních prací.

Příloha F

Obsah přiloženého CD

Tato příloha je povinná pro každou práci. Každá práce musí totiž obsahovat přiložené CD. Viz dále.

Může vypadat například takto. Váš seznam samozřejmě bude odpovídat typu vaší práce. (viz [9]):



Obrázek F.1: Seznam přiloženého CD — příklad

Na GNU/Linuxu si strukturu přiloženého CD můžete snadno vyrobit příkazem:

```
$ tree . >tree.txt
```

Ve vzniklém souboru pak stačí pouze doplnit komentáře.

Z **README.TXT** (případně index.html apod.) musí být rovněž zřejmé, jak programy instalovat, spouštět a jaké požadavky mají tyto programy na hardware.

Adresář **text** musí obsahovat soubor s vlastním textem práce v PDF nebo PS formátu, který bude později použit pro prezentaci diplomové práce na WWW.