

tex2page 使用指南

王垠
清华大学计算机系

2003 年 6 月



Contents

1	简介	9
1.1	tex2page 是什么?	9
1.2	tex2page 与 TeX 的不同点	9
1.3	tex2page 的成功案例	10
1.4	需要注意的问题	11
2	安装	13
2.1	系统要求	13
2.2	安装	13
3	运行方法	15
3.1	usepackage	15
3.2	中文用户特别注意	15
3.3	我的第一个 tex2page 文档	15
3.4	从命令行运行	17
3.5	指定另外一个目录	17
3.6	从 Scheme 解释器执行	17
4	样式文件	19
4.1	CSS class 信息	19
4.2	导入 CSS 文件	20
4.3	css.t2p	20
4.4	为什么要使用 CSS?	20
5	颜色	23
6	章节, 段落, 分页	25
7	列表结构	27
7.1	enumerate	27
7.2	列表的样式	29
8	交叉引用	31
8.1	文件内部引用	31
8.2	连接外部 URL	31
8.2.1	简化的连接方式	31

9	目录, 脚注, 索引和参考文献	33
9.1	目录	33
9.2	脚注	33
9.3	索引	33
9.4	参考文献	33
10	插图	35
10.1	插入图片	35
10.2	图片缩放	35
10.3	直接在 HTML 里插入图片	35
10.4	插入 TeX 图片	35
10.5	图片复用	36
11	原封不动的文字 (verbatim)	39
11.1	verbatim	39
11.2	显示空格	39
11.3	verbatim 里的命令	39
11.3.1	如果要在 verb 环境里输入 “—”?	40
11.4	原封不动输入文件	40
11.5	改变 verb 的样式	40
11.6	verbwritefile	40
12	用 Scheme 作为 TeX 的扩展语言	41
13	排版 Scheme	43
13.1	scm	43
13.2	Scheme 的样式	43
13.3	定义关键字	44
13.4	slatex	44
13.4.1	schemedisplay 环境	44
13.4.2	插入段落的 Scheme 语句	44
13.4.3	插入 Scheme 程序文件	45
13.4.4	schemeresult	45
13.4.5	特殊符号	45
13.4.6	定义关键字	46
13.4.7	scmdribble	46
14	.t2p 文件的设定	47
14.1	操纵 HTML 分页方式	47
14.2	改变 TeX 图片比例	47
14.3	重新定义标题样式	47
14.4	写入 Scheme 语句	48
14.5	中文设置	48

15 中文设置	49
15.1 设置 HTML charset	49
15.2 设置标题	49
16 htmlonly 和 texonly	51
17 TeX hacker 请看	53

Contents

1 简介

1.1 tex2page 是什么?

tex2page 是一个可以把 plain TeX 或者 L^AT_EX 文档转换成 HTML 的工具。它是由 Dorai Sitaram 开发的一个 Scheme 程序。

虽然 tex2page 是 Scheme 写的, 但是大部分常规排版任务你不需要了解 Scheme 就能完成。你只需要知道一些 LaTeX 的基本命令。

如果你对 LaTeX 很熟悉, 你就会发现, LaTeX 里能使用的几乎所有基本结构 tex2page 都能支持。比如, 表格, 数学公式,

$$\left(\int_{-\infty}^{\infty} e^{-x^2} dx\right)^2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-(x^2+y^2)} dx dy \quad (1.1)$$

$$= \int_0^{2\pi} \int_0^{\infty} e^{-r^2} r dr d\theta \quad (1.2)$$

$$= \int_0^{2\pi} \left(-\frac{e^{-r^2}}{2}\right) \Big|_{r=0}^{r=\infty} d\theta \quad (1.3)$$

$$= \pi. \quad (1.4)$$

$$(1.5)$$

还有章节, 目录, 插图, 脚注, 定义命令, 交叉引用, 参考文献, 索引 你都可以在这篇文章里看到。

1.2 tex2page 与 TeX 的不同点

tex2page 是用来把 TeX 文档转化成 HTML 的工具, 所以它支持 HTML 的很多特性, 比如外部 URL 链接, tag 定位, CSS 样式表等等。

由于 HTML 与 TeX 的不同点, tex2page 缺少 TeX 宏语言的某些高级功能。但是这些功能是一般 LaTeX 用户在日常文档处理中遇不到的, 只有样式设计者才能遇到。


如果你需要更加强大的语言支持, 你可以在文档里使用嵌入的 Scheme. Scheme 语言比 TeX 的宏语言还要方便强大的多。如何使用嵌入的 Scheme, 请参考第 12 章。

其实 tex2page 可以轻松支持下面这段有点难度的 TeX 宏。

```
\def\x{\a}
\def\a{a\def\x{\b}}
\def\b{b\def\x{\c}}
\def\c{c\def\x{\a}}
\x\x\x\x\x
```

1 简介

Table 1.1: 一个简单的表格

A	B	C
		
10	hello	
2^{10}	$\sqrt{100}$	$\sum_{0 < j < m} \sum_{j < k < n} x_{jk}$

得到的输出是: abcab.

我自己定义了一个宏 `\ic`, 它可以把命令以命令本身的样子加入索引但是却按命令第一个字母排序。

```
\def\gobble#1{}
\def\ic#1{\expandafter\expandafter\expandafter%
\index\expandafter\expandafter\expandafter{%
\expandafter\gobble\string#1@\verb{#1}}}
```

比如 `\ic{\section}` 会展开成 `\index{section@p{\section}}`。这样你会在索引里“S”开头的字母那组看到 `\section`, 而不是在反斜杠开头那组。

当然你可以直接打 `\index{section@p{\section}}`, 但是我定义了 这个宏就可以少按很多键盘。

你可能从来没用过 `\expandafter`, `\csname`, `\endcsname` 之类的高级 TeX 命令吧。这些 `tex2page` 都支持! 我发现它似乎不支持 `\long\def` 这种定义方式。

1.3 tex2page 的成功案例

你可能不太相信这个小玩艺儿可以处理你的大作? 其实已经有很多人开始使用它。

其中成功的例子包括 MIT Press 的教科书 “How to design programs”

<http://www.htdp.org/>

Harold Abelson and Gerald Jay Sussman 的大作 “Structure and Interpretation of Computer Programs”

<http://mitpress.mit.edu/sicp/full-text/book/book.html>

R. Kent Dybvig 著:

“The Scheme Programming Language Second Edition”

<http://www.scheme.com/tspl2ed.html>

还有 PLT 小组的几乎所有 manual

<http://www.plt-scheme.org/>

1.4 需要注意的问题

tex2page 的语法分析基本上不依赖于 T_EX，不过它需要 T_EX来处理数学公式 等东西。所以你的系统必须配置好 T_EX。

另外 tex2page 不能理解所有的 T_EX宏指令，所以如果你有一个很 奇特的 LaTeX 宏包，里面提供了非常炫的功能，你别想让 HTML 也 具有这种功能。

所以你必须按照 LaTeX 对于作者的要求，把文档格式于样式分开。不要在文档里固定太多的样式。这样你的文档才具有灵活性。

tex2page 生成的 HTML 可以变得非常漂亮。因为它可以利用 CSS 和 Scheme 的强大力量。所以你不用担心它的能力。

这个手册的浮动导航栏，浮动章节号，和所有中文的“目录”，“第 x 章”的字样，本来 tex2page 是不支持的，但是作者看了 tex2page 的代码 2 个小时之后就解决了中文的问题。可见 Scheme 语言表达 能力之清晰！

1 简介

2 安装

2.1 系统要求

你首先要有一个 Scheme 解释器，你可以用 MzScheme, guile, bigloo, chez, chicken, gambit, kawa, GNU/MIT scheme, petite, plt, pscheme, scheme48, scm, scsh, stk, sxm, umbscheme, ... 都行。

我个人推荐 MzScheme. 因为它遵守 R5RS, 功能完善。而且因为它是 tex2page 的作者使用的，不会出现不兼容问题。

或者你可以不用 Scheme 而用 Common Lisp 解释器。不过估计除非 你是一个老牌 Common Lisp 用户，否则你还是用 Scheme 比较好。

2.2 安装

tex2page 的安装很简单。你只需要到 Dorai Sitaram 的主页下载一个 tex2page。

解压之后，把所有的 .tex 和 .sty 文件拷贝到你的 TeX 目录树中。比如我的拷贝到了

```
/usr/local/texlive/texmf-local/tex/latex/tex2page/
```

特别重要的是 tex2page.tex 和 tex2page.sty. 这两个文件是必不可少的。

然后把 tex2page 脚本拷贝到 PATH 中的一个目录，比如 /usr/local/bin.

如果你的 Scheme 解释器是 MzScheme, 安装就完成了。如果不是，或者你用 Common Lisp, 那就自己看看 INSTALL 文件吧。

2 安装

3 运行方法

3.1 usepackage

为了使用 tex2page 提供的一些特殊命令，比如在 HTML 文档里插入 URL 链接，你需要加载 tex2page 提供的宏文件：tex2page.tex 和 tex2page.sty.

L^AT_EX 用户可以简单的输入：

```
\usepackage{tex2page}
```

plain T_EX 用户使用

```
\input tex2page
```

就行了。这些特殊命令以后我们会提到。现在你就可以按通常的方式输入 plain T_EX 或 L^AT_EX 文件了。

3.2 中文用户特别注意

tex2page 转换 L^AT_EX 文件到 HTML 时，不需要使用 CJK 宏包。

```
\usepackage{CJK}
```

```
\begin{CJK}{GBK}{song}
```

这些指令在这里不起任何作用。`\begin{CJK}{GBK}{song}` 会输出一行“GBKsong”的字样到文档，所以你最好把它用

```
\texonly
```

```
\begin{CJK}{GBK}{song}
```

```
\endtexonly
```

括起来。`\texonly` 和 `\htmlonly` 请参考第 16 章。

tex2page 直接就可以处理汉字，因为它输出的只是 HTML，排版实际上是由你的网页浏览器进行的，所以它不需要嵌入字体什么的，也不需要理解字体长度，计算大小，分页方式 因为这些都是浏览器的工作了。

好了，不废话了。你马上就可以编辑一个 L^AT_EX 文件来试试 tex2page。比如：

3.3 我的第一个 tex2page 文档

把这些存为一个文件，起名为 first.tex.

```
\documentclass{article}
```

```
\usepackage{tex2page}
```

3 运行方法

```
\title{我的第一个 tex2page 文档}
\author{无名氏}
\begin{document}

\section{我是谁? }
我是一个喜欢漂亮文档的人。我还喜欢:

\begin{enumerate}
\item 踢足球。
\item 打篮球。
\item 溜冰。
\item 听音乐。我喜欢这些音乐家和歌手:
    \begin{itemize}
\item Cielli Di Toscana
\item Sogno
\item Romero
\item BackStreet Boys
    \end{itemize}
\end{enumerate}

\section{我爱看的书}
\ldots

\section{我的数学作业}
\subsection{一道组合数学题}
一正立方体的六个面用g, r, b, y四种颜色涂染, 求其中两个面用色g, 两个面用色y, 其余一面用b, 一面用r的方案数。

\textbf{答:} 设G是对应正立方体全不可能的旋转的置换群。在这个群中有24个置换, 可分为下面5类:

\begin{enumerate}
\item 恒等变换。这个置换的轮换结构表示是 $x_1^6$ 。
\item 绕相对面的中点连线旋转 $180^\circ$ 的3个置换。这3个置换的轮换结构表示都是 $x_1^2x_2^2$ 。
\item 相对面终点的连线旋转 $90^\circ$ 的6个置换。这6个置换的轮换结构表示都是 $x_1^2x_4$ 。
\item 相对棱中点的连线旋转 $180^\circ$ 的6个置换。这6个置换的轮换结构表示都是 $x_2^3$ 。
\item 绕相对顶点旋转 $120^\circ$ 的8个置换。这8个置换的轮换结构表示都是 $x_3^2$ 。
\end{enumerate}
所以, 这个置换群的轮换示式是

$$\frac{1}{24}\left(x_1^6+3x_1^2x_2^2 +$$

```



```

6x_1^2x_4+6x_2^3+8x_3^2\right).$$
因而模式存储是
\begin{eqnarray*}
\{1\over24\}\big((g+r+b+y)^6&+3(g+r+b+y)^2(g^2+r^2+b^2+y^2)^2\backslash\backslash
&+6(g+r+b+y)^2(g^4+r^4+b^4+y^4)\backslash\backslash
&+6(g^2+r^2+b^2+y^2)^3+8(g^3+r^3+b^3+y^3)^2\big).
\end{eqnarray*}
其中, $g^2y^2br$的系数是
\${1\over24}\left(\{6\choose2\}\{4\choose2\}\{2\choose1\} +
3\{2\choose1\}\{2\choose1\}\right) = 8$. 所以方案数是8.

\end{document}

```

3.4 从命令行运行

假设你的 tex 文件名叫 first.tex. 那么在命令行输入

```
tex2page first
```

就行了。生成的图片，HTML 都会放在当前目录。

3.5 指定另外一个目录

如果你不喜欢所有文件都放到当前目录，那么可以编辑一个 jobname.hdir 文件，其中 jobname 是你的任务的名字，比如 first.hdir.

这个文件里有一行字，指明了生成的文件应该放到哪里。比如，我在 first.hdir 文件里写上：

```
myfirstpage
```

那么生成的 HTML 都放到 myfirstpage 目录里。

如果你在文件里使用 \jobname，这个宏会被展开成当前任务的名字。比如我现在的文件是 intro.tex，我的 \jobname 是 intro.

如果你编辑一个 /.tex2page.hdir 文件，里面有一行：

```
\jobname
```

那么所有的任务生成的文件都会放到对应它名字的目录里，如果没有这个目录就会自动建立它。

3.6 从 Scheme 解释器执行

如果不管什么原因，你不能从操作系统运行 tex2page，那么你可以在 Scheme 解释器里运行它：

3 运行方法

```
(load "tex2page") ;use appropriate pathname  
(tex2page filename)
```

4 样式文件

呵呵。看了上一章 tex2page 输出的网页是不是有点失望。怎么那么丑啊？那是因为你没有加入任何样式。你可以用一个 CSS 文件轻轻松松把你的 HTML 变得比这个小册子漂亮十倍，而你不需要再次运行 tex2page!

4.1 CSS class 信息

tex2page 输出的文字都是含有 class 信息的。它输出的 HTML 有各种 class, 比如我摘取的这段生成的 HTML, 就是这个网页右上角的导航栏。

```
<div align=right class=navigation><i>[<span><a href="intro.html"> 封面</a>, <a href="intro-Z-H-12.html">上一  
页</a></span><span>, <a href="intro-Z-H-14.html">下一页  
</a></span> <span>; &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</span><span><a href="intro-Z-H-1.html#node_toc_start">目录  
</a></span><span><span>; &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</span><a href="intro-Z-H-18.html#node_index_start"> 索引  
</a></span>]</i></div>
```

可以看到, 它是包含在一个 class 为 navigation 的 `div` 里的, 如果你在样式文件里指定:

```
div.navigation{  
    position: fixed;  
    width: 30%;  
    top: 96%;  
    right: 0;  
    bottom: auto;  
    margin:0;  
    padding:5px;  
    font: 28px;  
    padding: 2px;  
    margin: 0px;  
    margin-bottom: 8px;  
    background: white;  
    border-style: solid;  
    border-width: 0.6pt;  
    border-color: orange;  
    text-align: center;  
}
```

4 样式文件

就可以得到跟本文一样的浮动导航条了。另外，章节，代码，强调，`code` 都有自己的 class。以后我们还会看到其它结构的样式定义。

4.2 导入 CSS 文件

你可以用

```
\inputcss basic.css
```

导入你自己的样式文件。这样你的 HTML 文件的 `head` 中就会有

```
<link rel="stylesheet" type="text/css" href="tex2page.css" title=default>
```

这样的信息。你也可以用

```
\cssblock
h1 {color: navy}
\endcssblock
```

它会被 `tex2page` 插入到 HTML 文件的 `head` 中作为嵌入 CSS。

好了，你现在就可以把这个 `main.css` 和 `tex2page.css` 下载过去放到你的 `first.tex` 的目录里。然后在 `first.t2p` 文件开头加入：

```
\inputcss tex2page.css
```

再运行一次

```
tex2page first
```

因为先头你的 `first.t2p` 文件里没有插入对 `tex2page.css` 的引用，所以需要再次运行。以后修改 CSS 就不用再次运行了，只需要刷新浏览器。

好！刷新一下你的浏览器。你的文档已经基本上跟我一样了！出去 happy 一下，回来再继续看：)

4.3 css.t2p

随 `tex2page` 提供的 `css.t2p` 中有基本的样式和对 Scheme 进行语法加亮的样式。`t2p` 文件的配置请看第 14 章。Scheme 的语法加亮请看第 12 章。

4.4 为什么要使用 CSS?

HTML 本来没有 LaTeX 那样的页面表达能力，一个转换程序何必一定要让生成的 HTML 跟 DVI 看起来一样？

用户通过 CSS `stylesheet` 来修改样式，大大简化了 `tex2page` 的设计，它不需要理解很多 LaTeX 的样式定义，也大大方便了作者修改页面的样式。

作者可以在 LaTeX 文件外部修改转换后 HTML 的样式，这可能就是 `tex2page` 区别于其它转换程序的地方。

CSS 具有非常灵活的屏幕排版方式，比如，我在这个文档里定义了一个宏：

```
\def\tl{\htmlonly\rawhtml<div class=tl>\endrawhtml\endhtmlonly}  
\def\endt1{\htmlonly\rawhtml</div>\endrawhtml\endhtmlonly}
```

我的 CSS 里有一段定义:

```
div.tl{  
    position: fixed;  
    width: 3.5em;  
    top: 0;  
    right: 0;  
    left: 0;  
    bottom: auto;  
    margin:0;  
    padding:5px;  
    font: 28px;  
    color:#666;  
    border-bottom: 5px solid #000;  
    padding: 2px;  
    margin: 0px;  
    margin-bottom: 8px;  
}
```

我只要用 `\tl` 和 `\endt1` 把某些文字括起来, 就可以把它悬挂在浏览器页面左上角。

这本手册每一页的左上角那个标记就是用这个办法加上去的。不过我用了很程序化的手段, 请参考第 12 章。

4 样式文件

5 颜色

你可以用命令

```
\color[color-model]{color-spec}
```

改变字符的颜色。其中 color-model 可以是 rgb, cmyk, gray。举一些例子：

```
{\color[gray]{.17} light gray},  
{\color[rgb]{.69, .19, .38} maroon},  
{\color[cmyk]{0, .89, .94, .28} brick red},  
{\color[RGB]{220, 20, 60} crimson},  
{\color{red} red}, and  
{\color{blue} blue}.
```

显示出来就是这个样子：

light gray, maroon, brick red, crimson, red, and blue.

你可以用

```
\definecolor{new-color-name}{color-model}{color-spec}
```

定义颜色命令。比如：

```
\definecolor{BrickRed}{cmyk}{0, .89, .94, .28}
```

就定义了颜色 BrickRed。以后使用 `\color{BrickRed}` 就可以是字符的颜色变成这个砖红色。

tex2page 自带的 x11rgb.tex 文件里有 X11 各种颜色的 `\definecolor` 定义。你只要把它放到 TEXINPUTS 的一个目录里就可以用 `\input x11rgb`，然后就可以使用那些名字了。比如：DarkSlateGray, LightSkyBlue, OrangeRed, Fuchsia, Lime ...

HTML 的网页浏览器一般都能识别颜色名字，所以在转成 HTML 时，你不需要加载 x11rgb.tex。

5 颜色

6 章节，段落，分页

这些跟 LaTeX 的用法一样，你用

```
\section{我的某一小节}
```

就可以开始新的一节。

其它的章节命令还有 `\part`, `\chapter`, `\section`, `\subsection`, `\paragraph`, `\subparagraph`, ... 都是跟 LaTeX 一样的。

只有一点特殊，就是 `tex2page` 生成的是 HTML，如果你不主动使用 `\pagebreak` 分页，它就只在 `\chapter` 分页。

不过你可以在 `t2p` 文件里设定从 `\section` 分页。具体方法 请参考第 14 章。

6 章节, 段落, 分页

7 列表结构

7.1 enumerate

你可以用 `enumerate` 环境来得到有计数的列表。下面这个列表用一个数字列表列举了一些基本的 LaTeX 输入字符时的基本要素。

1. Hi, 今天是 June 7, 2003。你可以用 `\today` 输入今天的日期。
2. 左边的双引号要用两个反引号 ‘ 表示，右边的双引号用两个单引号表示 ’。
3. 这些字符 `$ & % # _ { } ~ ^` 要这样输入：

```
\$ \& \% \# \_ \{ \} \~{} \^{} 
```

4. 你可以用 `\verb|something|` 原封不动的输入某些特殊的 TeX 符号。比如 `\verb|\documentclass{article}` 得到 `\documentclass{article}`。
5. Use `\textbf{something}` to enter something with font **Bold Weight**.
6. Use `\textsf{something}` to enter something with font Sans Serif.
7. `\bfseries` can change **all the text follows to bold face, until the group ends**.
8. 你可以用改变字体大小的命令

```
\tiny tiny \scriptsize scriptsize \footnotesize  
footnotesize \small small \normalsize  
normalsize(default) \large large \Large Large \LARGE  
LARGE \huge huge \Huge Huge.}
```

结果是：

```
tiny scriptsize footnotesize small normalsize(default) large Large LARGE huge  
Huge.
```

9. TeX 里有三种不同的横线：“emdash”，“endash” 和 “hyphen”。
 - emdash 就是“破折号”，用3根“-”(减号)表示: He lay there—dead.
 - endash 用来连接一个范围，用2根减号表示: from 10–12

7 列表结构

- hyphen 就是连字号，就是一根减号: state-of-the-art.

10. 不要使用 “...” (三个点) 来表示省略号。因为它们挤得很紧，不好看。在正文中你应该用 `\ldots` 表示省略号。看看这个 ... 是不是好看多了？

其实 $\text{T}_{\text{E}}\text{X}$ 里有好几种不同的省略号，你应该在适当的地方使用适当的省略号，这样你的文档才会好看。

- a) `\ldots` (...): 用于位置较低的符号，比如 a_1, a_2, \dots, a_n .
- b) `\cdots` (\cdots): 用于位置较高的符号，比如 $x_1 + x_2 + \cdots + x_n$.
- c) `\vdots` (\vdots) 和 `\ddots` (\ddots), 主要用于矩阵里的省略表示，比如:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots \\ x_{21} & x_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

- d) `\dotfill` 用于填充一行，就像某些目录里的那些点线。

上面这个数字列表我是这样输入的:

```
\begin{enumerate}
\item Hi, 今天是 \today。你可以用 \verb+\today+ 输入今天的日期。
\item 左边的双引号要用两个反引号 \verb` 表示，右边的双引号用两个单引号表示 '。
\item 这些字符 \$ \% \# \_ \{ \} \~{} \^{} 要这样输入:

\verb{
\$ \% \# \_ \{ \} \~{} \^{}
}
.....
.....
\item \tex 里有三种不同的横线: ‘‘emdash’’, ‘‘endash’’ 和 ‘‘hyphen’’。

\begin{itemize}
\item emdash 就是“破折号”，用3根‘-’ (减号)表示: He lay there---dead.
\item endash 用来连接一个范围，用2根减号表示: from 10--12
.....
\end{itemize}

.....
```

`\item` `\tex` 里有几种不同的省略号:

```

\begin{enumerate}
\item \verb
\ldots (\ldots): 用于位置较低的符号, 比如  $a_1$ ,
 $a_2$ ,  $\ldots$ ,  $a_n$ .
\item \verb$\cdots$ ( $\cdots$ ): 用于位置较高的符号, 比如  $x_1+x_2+\cdots+x_n$ .
.....
\end{enumerate}

.....
\end{enumerate}

```

7.2 列表的样式

`enumerate` 的各种样式可以用 CSS 设定, 比如下面这些 CSS 样式 定义了: 第1级数字列表用阿拉伯数字表示, 第2级用小写字母, 第3 级用罗马数字, 第4级用大写字母。

```

ol {
    list-style-type: decimal;
}

ol ol {
    list-style-type: lower-alpha;
}

ol ol ol {
    list-style-type: lower-roman;
}

ol ol ol ol {
    list-style-type: upper-alpha;
}

```

回头看看我的第二层数字列表, 是不是显示为小写字母编号 a., b., c., ...? 你可以试试加入第 3 层列表是不是显示为 i., ii., iii, iv, ...

7 列表结构

8 交叉引用

8.1 文件内部引用

tex2page 的交叉引用就是 LaTeX 的 `\label{name}`, `\ref{name}`. LaTeX 用户应该十分清楚了。

这些引用只能跳转到章节, `caption` 的地方, 为了能够精确定位 HTML 文件里的每一处地方, 你需要使用

```
\tag{atag}{tagvalue}
```

其中 `tagvalue` 是用来显示在 `\ref{atag}` 处的文本。

现在来测试以下这个 测试连接, 它会把你带到第 3 章末尾。我在在第 3 章末尾插入了:

```
\tag{test:tag}{测试连接}
```

8.2 连接外部 URL

```
\urlhd{URL}{HTML text}{DVI text}
```

可以让你做连接到外部, 比如

```
\urlhd{http://www.tug.org}{访问 TeX 用户组的网站}{go to \TeX  
User Group}
```

你需要指定 HTML 和 DVI 两个版本里显示的字符。这样你可以依据 不同的版本显示不同的文字。

必须现在是 DVI 版本, 所以 上面那个链接显示为: `go to TEXUser Group`。

我怎么知道现在是用的 DVI 版本呢? 请看第 16 章。

8.2.1 简化的连接方式

每次都指定两个版本的连接不是很方便, 所以如果 HTML 和 DVI 里 连接符号一样, 就可以使用下面这种方式:

```
\urlh{URL}{text}
```

只需要两个参数, 它在 HTML 和 DVI 里显示是一样的。比如,

```
\urlh{http://www.tug.org}{TUG}
```

都显示为 TUG。

如果使用

8 交叉引用

```
\urlh{http://www.tug.org}{TUG\{ only in DVI}}
```

就可以使 \ 之后的部分只在 DVI 文件里显示。就像这样： TUG only in DVI。

9 目录，脚注，索引和参考文献

9.1 目录

只需要 `\tableofcontents` 就行了，LaTeX 用户都知道。不用多说。

9.2 脚注

9.3 索引

你只需要把需要索引的符号都用 `\index{...}` 做上标记，然后在文档某个地方放上一句 `\printindex`，索引就会放到那里，所有的索引都是可以点击的。

跟 LaTeX 一样，你需要运行 2 次 `tex2page` 才能更新索引。但是你不需要自己调用 `makeindex`，因为 `tex2page` 已经帮你调用了。

请看本文最后的索引。

9.4 参考文献

`tex2page` 可以利用 BibTeX 来处理参考文献。你不用自己调用 `bibtex`，只需要运行 `tex2page` 两次就行了。

你只需要

1. 选择一个文献数据库样式，比如 `alpha`。

```
\bibliographystyle{alpha}
```

可选的样式还很多，你可以用 `plain`, `alpha`, `unsrt`, `siam`, `apalike`, ... 举不胜举！你可以参考一下 <http://www.cs.stir.ac.uk/~kjt/software/latex/showbst.html>。

2. 在文章末尾(一般是末尾吧)，插入你的数据库。比如插入 `jacm.bib`:

```
\bibliography{jacm}
```

3. 在文章里使用 `\cite{...}` 引用数据库里的标识符。

文献数据库 `jacm.bib` 里被你引用过的文献，就会以 `[Knu86]` 这样的样式的样式出现在末尾。你还可以去下载更多的格式。如果你使用

```
\nocite{karpinski95new}
```

这样的字样，标记是“`karpinski95new`”的文献虽然没有被引用，也会出现在 `Bibliography` 里。你还可以使用

9 目录, 脚注, 索引和参考文献

`\nocite{*}`

把数据库里所有的文献全部列出。现在我引用一片论文吧, 我用 `apalike` 的样式。请看 [Li et al., 1996]。

10 插图

10.1 插入图片

插入图片很简单，你可以是用 LaTeX 的

```
\includegraphics{eps-file}
```

也可以是用 plain TeX 的

```
\epsfbox{eps-file}
```

10.2 图片缩放

使用

```
\imgpreamble
  \let\LaTeXdocument\document
  \def\document{\LaTeXdocument\Large}
\endingpreamble
```

可以把数学公式转化而来的图片放大一些。

10.3 直接在 HTML 里插入图片

```
\rawhtml
...
\endrawhtml
```

可以输入一段原封不动的 HTML，所以你可以使用

```
\rawhtml

\endrawhtml
```

来插入任何 HTML 可以支持的图片。

10.4 插入 TeX 图片

你可以让 TeX 把某些东西排版后作为一个图片插入 HTML:

```

\begin{figure}
\begin{center}
\makehtmlimage{
\newdimen\unit
\def\point#1 #2 {\rlap{\kern#1\unit
\raise#2\unit\hbox{$
\scriptstyle\bullet\;(#1,#2)$}}
\unit=\baselineskip
\centerline{\vtop{\hrule
\hbox{\vrule height10\unit depth9.4\unit \kern2\unit
\hbox{%
\point 0 0 % Alioth (Epsilon Ursae Majoris), mag 1.79
\point 0 8 % Dubhe (Alpha Ursae Majoris), mag 1.81
\point 0 -8 % Alkaid (Eta Ursae Majoris), mag 1.87
\point -1 -2.5 % Mizar (Zeta Ursae Majoris), mag 2.26
\point 4 7 % Merak (Beta Ursae Majoris), mag 2.37
\point 4 2 % Phekda (Gamma Ursae Majoris), mag 2.44
\point 1 1.5 % Megrez (Delta Ursae Majoris), mag 3.30
}% Src: Atlas of the Universe; Astronomy Data Book
\kern7\unit \vrule}\hrule}}
}
\caption{把 TeX 排版的输出作为图片}
\end{center}
\end{figure}

```

结果是:

10.5 图片复用

如果你使用数学公式，最好把复杂的，需要多次使用的数学符号用 `\imgdef` 定义为一个可以复用的图片，这样 `tex2page` 可以重复利用 这个图片，而不用生成重复的图片了。比如：

```

\imgdef\ohm{\$\Omega$}
The circuit uses two 10-\ohm\ resistors, three 50-\ohm\
resistors and one 1-k\ohm\ resistor.

```

结果是:

The circuit uses two 10-Ω resistors, three 50-Ω resistors and one 1-kΩ resistor.

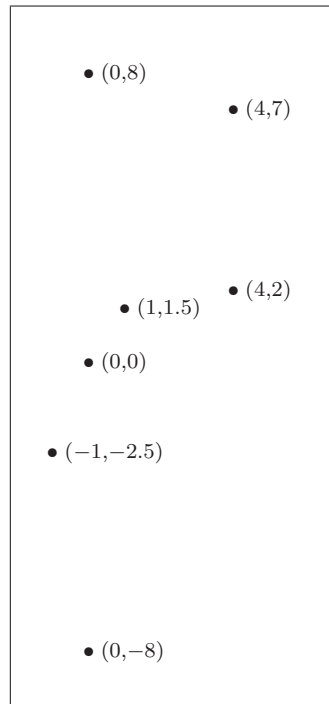


Figure 10.1: 把 TeX 排版的输出作为图片

11 原封不动的文字 (verbatim)

11.1 verbatim

使用 `\verb` 就可以把你的 TeX 宏原封不动显示在页面里。其中的 “—”, 竖线可以用其它的任何非字母数字的符号代替, 比如 `+`, `-`, `&`, ...

所以

```
A \verb|cons|-cell has two components: a \verb+car+ and a \verb&cdr&.
```

显示为:

```
A cons-cell has two components: a car and a cdr.
```

它实际上在文档里插入了 class 为 `verbatim` 的 `icode`。

使用

```
\verb{
....
....
}
```

就可以在 HTML 文件里插入一段 class 为 `verbatim` 的 `ipre`。

11.2 显示空格

使用 `\verb*{three spaces}` 就可以把中间的3个空格显示 出来成为:

```
“three spaces”
```

11.3 verbatim 里的命令

你还可以在 `\verb{...}` 的环境里使用 TeX 命令, 你只需要 使用竖线 “—”, 作为 `escapechar`. 比如:

```
\def\evalsto{::==}

\verb{
(cons 1 2) |evalsto (1 . 2)
}
```

显示为:

```
(cons 1 2) ::= (1 . 2)
```

这种竖线命令还可以让你可以在 `verb` 的区域中插入不匹配的花括号, 比如:

11 原封不动的文字 (*verbatim*)

```
\verb{
This is wrong: \def\a{\bc}||}
}
```

11.3.1 如果要在 `verb` 环境里输入 “—”?

如果你经常需要在 `verb` 的环境里使用竖线，那么最好的办法就是使用

```
\verbescapechar\@
```

把 `escapechar` 换成 `@`, 或者其它什么符号。刚才为了行文方便, 我就转换过两次 `escapechar`.

11.4 原封不动输入文件

使用

```
\verbatiminput{program.scm}
```

就可以原封不动插入 `program.scm`. 不过这样不能进行语法加亮, 所以最好使用 `slatex`.

11.5 改变 `verb` 的样式

段落里的 `verb` 被转换为 class `verbatim` 的 `verbatim`, 单独分行的 `verb` 是作为 `verbatim` 的 `verbatim`. 所以你可以通过它来修改 `verb` 区域的样式。比如你可以把它改成用 `darkgreen` 颜色显示。

```
.verbatim {color: darkgreen}
```

11.6 `verbwritefile`

这个命令可以指定一个文件作为 `\verbwrite` 的输出文件。这样你在 `\verbwrite` 里写的内容, 就会转存到一个文件。

比如:

```
\verbwritefile test1.txt
```

就会把

```
\verbwrite{
\tag{sec:verbwritefile}{\verb+\verbwritefile+}
}
```

里的内容送到 `test1.txt`, 但是并不把这些内容插入文档。

12 用 Scheme 作为 TeX 的扩展语言

用 `\eval` 就可以在 TeX 文件里插入一段 Scheme. 当运行第二次 `tex2page` 的时候, 代码的运行结果就会被插入 TeX 文件. 你可以用这些代码来改变 `tex2page` 的行为。

必须下面的代码将会打印 $\sqrt{2}$ 的值: 1.4142135623730951

```
\eval{ (display (sqrt 2)) }
```

你可以想象一下, 你能用 Scheme 在 TeX 里做什么吗? 你几乎无所不能!

看看第 15 章, 里面有一些我自己定义的函数, 可以随意操纵文档的排版方式。我看了一会儿 `tex2page` 的代码就想出了几乎所有我用的着的中文解决方案。

现在我把一个 `coroutine` 的测试代码放在这里, 看看是什么结果:

```
(require (lib "defmacro.ss"))
(define-macro coroutine
  (lambda (x . body)
    '(letrec ((+local-control-state
              (lambda (,x) ,@body))
              (resume
               (lambda (c v)
                 (call/cc
                  (lambda (k)
                    (set! +local-control-state k)
                    (c v)))))))
      (lambda (v)
        (+local-control-state v))))))

(define make-matcher-coroutine
  (lambda (tree-cor-1 tree-cor-2)
    (coroutine dont-need-an-init-arg
      (let loop ()
        (let ((leaf1 (resume tree-cor-1 'get-a-leaf))
              (leaf2 (resume tree-cor-2 'get-a-leaf)))
          (if (eqv? leaf1 leaf2)
              (if (null? leaf1) #t (loop))
              #f))))))

(define make-leaf-gen-coroutine
  (lambda (tree matcher-cor)
    (coroutine dont-need-an-init-arg
      (let loop ((tree tree))
```

12 用 Scheme 作为 TeX 的扩展语言

```
(cond ((null? tree) 'skip)
      ((pair? tree)
       (loop (car tree)
             (loop (cdr tree))))
      (else
       (resume matcher-cor tree))))
(resume matcher-cor '()))

(define tree-match?
  (lambda (tree1 tree2)
    (letrec ((tree-cor-1
              (make-leaf-gen-coroutine
               tree1
               (lambda (v) (matcher-cor v))))
             (tree-cor-2
              (make-leaf-gen-coroutine
               tree2
               (lambda (v) (matcher-cor v))))
             (matcher-cor
              (make-matcher-coroutine
               (lambda (v) (tree-cor-1 v))
               (lambda (v) (tree-cor-2 v)))))
      (matcher-cor 'start-ball-rolling))))

(define tree1 '((a b) (y z) (3 4)))
(define tree2 '((a b) (t z) (3 4)))
(define tree3 '((a (b y) z) (3 4)))

(define (display-line str . rest)
  (for-each display
            (cons str rest))
  (newline))

(if (tree-match? tree1 tree2)
    (display-line "tree1 and tree2 match. ")
    (display-line "tree1 and tree2 don't match. "))

(if (tree-match? tree1 tree3)
    (display-line "tree1 and tree3 match. ")
    (display-line "tree1 and tree3 don't match. "))
}

tree1 and tree2 don't match. tree1 and tree3 match.
```

13 排版 Scheme

13.1 scm

用 `\scm` 把 Scheme 程序括起来就可以进行显示了，并且含有语法加亮信息。你只要试用 `css.t2p` 里提供的 CSS 样式就可以得到很漂亮的 Scheme 语法加亮。比如：

```
\scm{
(define fact
  "The factorial function"
  (lambda (n)
    (if (= n 0) 1 ;the base case
        (* n (fact (- n 1))))))
}

(define fact
  "The factorial function"
  (lambda (n)
    (if (= n 0) 1 ;the base case
        (* n (fact (- n 1))))))
```

13.2 Scheme 的样式

有了语法加亮的信息还不能把各种关键字用不同的颜色显示。你必须在 CSS 文件里定义样式，或者把这些样式使用 `\cssblock ...\endcssblock` 的形式嵌入 HTML 中。比如：

```
\cssblock
.scheme          {color: brown} /* background punctuation */
.scheme .keyword {color: #990000; font-weight: bold}
.scheme .builtin {color: #990000}
.scheme .variable {color: navy}
.scheme .global  {color: purple}
.scheme .selfeval {color: green}
.scheme .comment {color: teal}
\endcssblock
```

就可以对 Scheme 语句的各种关键字，变量等进行语法加亮了。

13.3 定义关键字

你可以使用

```
\scmbuiltin{cons car cdr}
```

这样的形式定义内部函数。

```
\scmkeyword{define-class unwind-protect}
```

这样的形式定义关键字。

使用

```
\scmvariable{and 42 +i}
```

的形式定义变量。这样它们就能被加上特殊的颜色了。

13.4 slatex

以上都是 tex2page 自己提供的排版 Scheme 的方式。对 Scheme 排版，最好使用 Slatex，它可以在

<http://www.ccs.neu.edu/home/dorai/slatex/slatxdoc.html>

下载。

在装好 slatex 之后。只要

```
\usepackage{slatex}
```

就可以使用 slatex 提供的各种功能了。

13.4.1 schemedisplay 环境

使用 tex2page 提供的 `\scm{...}` 如果内容太长容易出错，所以用

```
\begin{schemedisplay}
```

```
...
```

```
\end{schemedisplay}
```

就好多了。

13.4.2 插入段落的 Scheme 语句

段落中的 Scheme 语句，比如

```
...
```

, 可以这样输入:

```
\scheme|(cons 'a '(b c))|
```

13.4.3 插入 Scheme 程序文件

如果要读入一个 Scheme 程序文件就使用

```
\schemeinput{evenodd.scm}
% the .scm is optional!
```

上面介绍的这些这些基本够用了。如果你还不满足就继续看下一节。

13.4.4 schemeresult

Scheme 解释器的输出结果可以用一种特殊的颜色显示：

```
\scheme|((lambda () (cons 'lambda
'cons)))| yields
\schemeresult|(lambda . cons)|.
```

结果是：

```
(cons 'a '(b c)) yields
\verb{ \scheme|(cons 'a '(b c))| .
```

13.4.5 特殊符号

你不想把 ((**lambda** () (cons 'lambda 'cons))) 都用 (lambda . cons) 来显示？用

```
\setsspecialsymbol{lambda}{\lambda$}
```

就可以达到你的目的。我们还可以花哨一点：

```
\setsspecialsymbol{lambda}{\lambda$}
\setsspecialsymbol{and}{\land$}
\setsspecialsymbol{equiv?}{\equiv$}
\setsspecialsymbol{below?}{\subseteq$}
\setsspecialsymbol{above?}{\supseteq$}
\setsspecialsymbol{a1}{a1$}
\setsspecialsymbol{a2}{a2$}
```

```
\begin{schemedisplay}
(define equiv?
  (lambda (a1 a2)
    (and (below? a1 a2)
         (above? a1 a2))))
\end{schemedisplay}
```

显示为：

```
((lambda () (cons 'lambda 'cons)))
```

13.4.6 定义关键字

```
\setkeyword
  \setconstant
  \setvariable
  \setdata
```

可以定义你自己需要的关键字，常数，变量，数据，它们会被正确的语法加亮。比如，通常 `cons` 不是一个关键字，它一般显示为：

```
(cons 'a '(b c))
```

我们用了

```
\setkeyword{cons}
```

之后，`cons` 就用跟 `define` 一样的方式显示了：

```
(cons 'a '(b c))
```

13.4.7 `scmdribble`

这个命令可以把 `\scm` 区域里的内容送到最近的一个 `\verbwritefile` 指定的文件里，同时把这些代码语法加亮在文档中显示。

这可以被作为一种 Scheme “文学编程”的方式。我来试试：

```
\verbwritefile test1.scm
\scmdribble{
(define factorial
  (lambda (n)
    (if (= n 0)
        ;\scheme{(zero? n)}
        ;also possible
        1 (* n (factorial
              (- n 1))))))
}
```

嗯。还真是生成了一个文件叫做 `test1.scm`。里面是这些代码。

```
(define factorial
  (lambda (n)
    (if (= n 0)
        ;\scheme{(zero? n)}
        ;also possible
        1 (* n (factorial
              (- n 1))))))
```

14 .t2p 文件的设定

你可以在一个跟源文件同名的 .t2p 文件里进行针对 HTML 的配置，比如定义一些专门输出指定格式 HTML 的 TeX 宏，定义 CSS，等等。

14.1 操纵 HTML 分页方式

tex2page 的分页缺省是每章分成一个 HTML 文件。

我可以把下面这些加到 jobname.t2p, 使得我的每一个 \section 都成为单独一个网页。

```
\let\ORIGsection\section
\def\section{\vfill\eject\ORIGsection}
```

14.2 改变 TeX 图片比例

这些可以使你的图片和公式都放大一些：

```
\imgpreamble
\let\LaTeXdocument\document
\def\document{\LaTeXdocument\Large}
\endingpreamble
```

14.3 重新定义标题样式

这些可以重新定义标题一的样式：

```
\def\maketitle{\begingroup
\let\\break
\rawhtml<h1>\endrawhtml
\TIIPtitle
\rawhtml</h1>\endrawhtml
\medskip
\leftline{\SAVEedition}
\bigskip\bigskip
\leftline{\TIIPauthor}
\medskip
\leftline{foreword by \SAVEforewordauthor}
\bigskip\bigskip\bigskip}
```

```
\leftline{\SAVEpublisher}  
\endgroup}
```

14.4 写入 Scheme 语句

这恐怕是最强大功能了。我可以在 t2p 文件里定义我需要的函数，改变程序运行的方式。t2p 文件里的内容会在输出标题页之后执行，我认为这是不对的。我把代码修改为让 t2p 在最开头就执行。这样我可以更大程度的修改 tex2page 的行为。

比如，我的每一章左上角的浮动标题就是这段 Scheme 代码加进去的。代码被包含在了对 TeX 的命令 `\chapter` 的重新定义当中。

```
\let\ORIGchapter\chapter  
\def\chapter{  
  
\evalh{  
(let ((chno (string->number (section-counter-value 0))))  
  (when (> chno 0)  
    (tex2page-string "\\t1")  
    (emit-chapter-heading chno)  
    (tex2page-string "\\endtl"))  
  )  
\ORIGchapter  
}
```

它的含义就是，如果当前的章编号大于0，那么就在章标题前面插入

```
\t1  
这章的编号  
\endtl
```

其中 `\t1` 和 `\endtl` 是我定义的宏，用来浮动文字到浏览器左上角。它们的定义也在 t2p 文件里，因为它们是直接跟 HTML 有关系的东西。这两个宏实际上是输出了一个含有 CSS 信息的 `¡div¡` 标记。详细情况请看第 4 章。

这些都是跟 HTML 相关的，如果你把它们定义在 .t2p 文件里，使用 TeX 处理这个文档的时候就不会出现不能识别命令的问题了。这样的文档既可以转成 HTML 屏幕阅读，也可以转成 PDF, PS, 打印在纸上。

14.5 中文设置

请看下一章。

15 中文设置

tex2page 本身是一个不认识人类语言的 Scheme 程序，所以它理论上支持任何人类语言。

但是如果你不知道怎么设置，你不会得到很好的中文效果。

15.1 设置 HTML charset

如果你使用缺省的 charset 是 iso5559-1, 这样你的中文网页无论在中国人还是外国人的浏览器里都不能正确显示。

为了改变 charset, 你可以在你的 jobname.t2p 文件里设定:

```
\evalh{
(set! *html-head* (append *html-head*
  '("<meta http-equiv=\"Content-Type\" content=\"text/html;\
charset=gb2312\">")))
}
```

html-head 是一个表，它里面的内容会被依次加入到 HTML 的 i head i 里。

15.2 设置标题

缺省的 tex2page 会有一些页面显示为 “Contents”, “Chapter”, “Part”, “Index”, “Bibliography”, ...

如何使这些字符都显示为中文?

在 LaTeX 里，你可以定义:

```
\renewcomammd{contentsname}{目录}
```

但是似乎 tex2page 没有提供多少让你改变这些条目的东西。所以我对 tex2page 的代码做了一些修改，让它支持这种命令。改动以后的 tex2page 代码可以在这里 [下载](#)。可能只支持 Scheme, 不像原来的程序可以在 Common Lisp 下运行。

注意这个包里只有修改过的 tex2page 程序和我的一个 chinese.t2p. 我不提供其它的必须的 LaTeX 样式文件，请到 tex2page 的主页去下载那些。

我的 intro.t2p 文件里有这些设定:

```
\evalh{
(set! *html-head* (append *html-head*
  '("<meta http-equiv=\"Content-Type\" content=\"text/html;\
charset=gb2312\">")))
}
```

```

\eval {
  (set! *navigation-sentence-begin* "")
  (set! *navigation-first-name* "首页")
  (set! *navigation-previous-name* "上一页")
  (set! *navigation-next-name* "下一页")
  (set! *navigation-page-name* "")
  (set! *navigation-contents-name* "目录 ")
  (set! *navigation-index-name* "索引 ")
  (set! *last-modified* "最后更新")
  (set! *html-conversion-by* "HTML 转换程序: ")
}

(define emit-chapter-heading
  (lambda (lbl-val)
    (if *inside-appendix?*
        (tex2page-string "\\appendixname"))
        (tex2page-string "第 ")
        (emit lbl-val)
        (tex2page-string " 章"))))

(define emit-part-heading
  (lambda (lbl-val)
    (emit "第 ")
    (emit lbl-val)
    (emit " 部分")))

}

\renewcommand\contentsname{目录}
\renewcommand{\bibname}{参考书目}
\renewcommand{\bibname}{参考文献}
\renewcommand{\chaptername}{章}
\renewcommand{\appendixname}{附录}
\renewcommand\indexname{索引}
\renewcommand\figurename{图}
\renewcommand\tablename{表}

```

其中 *navigation-...* 是一些 Scheme 变量，它们定义了导航栏显示的文字。后面几个 \renewcommand 改变了很多条目用中文显示。

注意其中很多都是我修改过的版本才有的设置。

16 htmlonly 和 texonly

`\htmlonly` 和 `\texonly` 可以分别设定某些内容只在 HTML 或者只在 DVI 文件出现。比如下面这些内容只会出现在 HTML 文件里：

```
\htmlonly
  \def\evalsto{\rawhtml<font
    color=blue><b>=&gt;</b></font>\endrawhtml}
\endhtmlonly
```

```
\texonly
  \def\evalsto{\leavevmode\hbox{${\Rightarrow$}}
\endtexonly
```

之后，

```
\scm{
(cons 1 2) |evalsto (1 . 2)
}
```

就会显示为：

```
(cons 1 2) ⇒ (1 . 2)
```

在 HTML 里显示为蓝色的 而在 DVI 文件里显示为一个向右的箭头。

17 TeX hacker 请看

tex2page 最初对于 TeX 的 primitive 支持有问题，对于国际化有一些不好配置的地方。经过我和 Dorai 的共同努力，tex2page 已经可以正确处理以下这些奇怪的 TeX 命令。

```
\documentclass{article}
\begin{document}
\tracingcommands=1
\tracingmacros=1

\def\gobble#1{}

\section{Done}
\string OK.

\section{Done}
\string\TeX

\section{Done}
\expandafter\gobble\string\TeX

\section{Done}

\def\OK{Haha}
\expandafter\gobble\string\OK hahahaha

\section{Done}
\expandafter\csname\gobble Tsubsection\endcsname{This is OK.}

\section{Done}
\csname\expandafter\gobble\string\section\endcsname{OK!}

\section{Done}

\def\TeXxxx{hello}

\def\appendroman#1#2#3{\expandafter\def\expandafter#1\expandafter{%
\csname\expandafter\gobble\string#2\romannumeral#3\endcsname}}
```

```
\appendroman\a\TeX{30}  
\a  
  
\end{document}
```

如果你看不懂这些是怎么回事，那么你就可以放心的使用 `tex2page`. 因为你不会使它为难 :P

如果你还能想出比最后一个 `\appendroman` 更奇怪的宏，那么你最好小心点。这是我们考虑过的最怪的一个宏。你可以参考

<http://learn.tsinghua.edu.cn/homepage/015450/wiki/TeXProblems.html>The TeXbook 练习题 7.10 的勘误。

Bibliography

- [Abrahamson et al., 1994] Abrahamson, K., Adler, A., Higham, L., and Kirkpatrick, D. (1994). Tight lower bounds for probabilistic solitude verification on anonymous rings. *Journal of the ACM*, 41(2):277–310.
- [Afek et al., 1993] Afek, Y., Attiya, H., Dolev, D., Gafni, E., Merritt, M., and Shavit, N. (1993). Atomic snapshots of shared memory. *Journal of the ACM*, 40(4):873–890.
- [Afek et al., 1994] Afek, Y., Attiya, H., Fekete, A., Fischer, M., Lynch, N., Mansour, Y., Wang, D.-W., and Zuck, L. (1994). Reliable communication over unreliable channels. *Journal of the ACM*, 41(6):1267–1297.
- [Afek et al., 1996] Afek, Y., Awerbuch, B., Plotkin, S., and Saks, M. (1996). Local management of a global resource in a communication network. *Journal of the ACM*, 43(1):1–19.
- [Afek et al., 1995] Afek, Y., Greenberg, D. S., Merritt, M., and Taubenfeld, G. (1995). Computing with faulty shared objects. *Journal of the ACM*, 42(6):1231–1274.
- [Afrati and Papadimitriou, 1993] Afrati, F. and Papadimitriou, C. H. (1993). The parallel complexity of simple logic programs. *Journal of the ACM*, 40(4):891–916.
- [Alon and Megiddo, 1994] Alon, N. and Megiddo, N. (1994). Parallel linear programming in fixed dimension almost surely in constant time. *Journal of the ACM*, 41(2):422–434.
- [Alon et al., 1995] Alon, N., Yuster, R., and Zwick, U. (1995). Color-coding. *Journal of the ACM*, 42(4):844–856.
- [Alur et al., 1996] Alur, R., Feder, T., and Henzinger, T. A. (1996). The benefits of relaxing punctuality. *Journal of the ACM*, 43(1):116–146.
- [Alur and Henzinger, 1994] Alur, R. and Henzinger, T. A. (1994). A really temporal logic. *Journal of the ACM*, 41(1):181–204.
- [Angluin et al., 1993] Angluin, D., Hellerstein, L., and Karpinski, M. (1993). Learning read-once formulas with queries. *Journal of the ACM*, 40(1):185–210.
- [Arnborg et al., 1993] Arnborg, S., Courcelle, B., Proskurowski, A., and Seese, D. (1993). An algebraic theory of graph reduction. *Journal of the ACM*, 40(5):1134–1164.

Bibliography

- [Aspnes et al., 1994] Aspnes, J., Herlihy, M., and Shavit, N. (1994). Counting networks. *Journal of the ACM*, 41(5):1020–1048.
- [Atallah et al., 1994] Atallah, M. J., Goodrich, M. T., and Kosaraju, S. R. (1994). Parallel algorithms for evaluating sequences of set-manipulation operations. *Journal of the ACM*, 41(6):1049–1088.
- [Attiya et al., 1995] Attiya, H., Bar-Noy, A., and Dolev, D. (1995). Sharing memory robustly in message-passing systems. *Journal of the ACM*, 42(1):124–142.
- [Attiya et al., 1994] Attiya, H., Dwork, C., Lynch, N., and Stockmeyer, L. (1994). Bounds on the time to reach agreement in the presence of timing uncertainty. *Journal of the ACM*, 41(1):122–152.
- [Awerbuch and Peleg, 1995] Awerbuch, B. and Peleg, D. (1995). Online tracking of mobile users. *Journal of the ACM*, 42(5):1021–1058.
- [Baader, 1993] Baader, F. (1993). Unification in commutative theories, Hilbert’s basis theorem and Gröbner bases. *Journal of the ACM*, 40(3):477–503.
- [Baccelli et al., 1993] Baccelli, F., Liu, Z., and Towsley, D. (1993). Extremal scheduling of parallel processing with and without real-time constraints. *Journal of the ACM*, 40(5):1209–1237.
- [Bachmair and Dershowitz, 1994] Bachmair, L. and Dershowitz, N. (1994). Equational inference, canonical proofs, and proof orderings. *Journal of the ACM*, 41(2):236–276.
- [Baeten et al., 1993] Baeten, J. C. M., Bergstra, J. A., and Klop, J. W. (1993). Decidability of bisimulation equivalence for processes generating context-free languages. *Journal of the ACM*, 40(3):653–682.
- [Baker, 1994] Baker, B. S. (1994). Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM*, 41(1):153–180.
- [Bay and Bilardi, 1995] Bay, P. and Bilardi, G. (1995). Deterministic on-line routing on area-universal networks. *Journal of the ACM*, 42(3):614–640.
- [Bell and Witten, 1994] Bell, T. C. and Witten, I. H. (1994). The relationship between greedy parsing and symbolwise text compression. *Journal of the ACM*, 41(4):708–724.
- [Berger et al., 1995] Berger, B., Brady, M., Brown, D., and Leighton, T. (1995). Nearly optimal algorithms and bounds for multilayer channel routing. *Journal of the ACM*, 42(2):500–542.
- [Bergstra and Tucker, 1995] Bergstra, J. A. and Tucker, J. V. (1995). Equational specifications, complete term rewriting systems, and computable and semicomputable algebras. *Journal of the ACM*, 42(6):1194–1230.

- [Bhatt and Cai, 1993] Bhatt, S. and Cai, J. (1993). Taking random walks to grow trees in hypercubes. *Journal of the ACM*, 40(3):741–764.
- [Bhatt et al., 1996] Bhatt, S. N., Chung, F. R. K., Hong, J.-W., Leighton, F. T., Obrenić, B., Rosenberg, A. L., and Schwabe, E. J. (1996). Optimal emulations by butterfly-like networks. *Journal of the ACM*, 43(2):293–330.
- [Birman et al., 1995] Birman, A., Gail, H. R., Hantler, S. L., Rosberg, Z., and Sidi, M. (1995). An optimal service policy for buffer systems. *Journal of the ACM*, 42(3):641–657.
- [Bloom et al., 1995] Bloom, B., Istrail, S., and Meyer, A. R. (1995). Bisimulation can't be traced. *Journal of the ACM*, 42(1):232–268.
- [Blum, 1994] Blum, A. (1994). New approximation algorithms for graph coloring. *Journal of the ACM*, 41(3):470–516.
- [Blum et al., 1994] Blum, A., Jiang, T., Li, M., Tromp, J., and Yannakakis, M. (1994). Linear approximation of shortest superstrings. *Journal of the ACM*, 41(4):630–647.
- [Blum and Kannan, 1995] Blum, M. and Kannan, S. (1995). Designing programs that check their work. *Journal of the ACM*, 42(1):269–291.
- [Bol and Groote, 1996] Bol, R. and Groote, J. F. (1996). The meaning of negative premises in transition system specifications. *Journal of the ACM*, 43(5):863–914.
- [Boyar et al., 1995] Boyar, J., Brassard, G., and Peralta, R. (1995). Subquadratic zero-knowledge. *Journal of the ACM*, 42(6):1169–1193.
- [Boyer and Yu, 1996] Boyer, R. S. and Yu, Y. (1996). Automated proofs of object code for a widely used microprocessor. *Journal of the ACM*, 43(1):166–192.
- [Bshouty, 1993] Bshouty, N. H. (1993). On the complexity of functions for random access machines. *Journal of the ACM*, 40(2):211–223.
- [Bshouty and Tamon, 1996] Bshouty, N. H. and Tamon, C. (1996). On the Fourier spectrum of monotone functions. *Journal of the ACM*, 43(4):747–770.
- [Buntine and Bürckert, 1994] Buntine, W. L. and Bürckert, H.-J. (1994). On solving equations and disequations. *Journal of the ACM*, 41(4):591–629.
- [Busch and Mavronicolas, 1996] Busch, C. and Mavronicolas, M. (1996). A combinatorial treatment of balancing networks. *Journal of the ACM*, 43(5):794–839.
- [Callahan and Kosaraju, 1995] Callahan, P. B. and Kosaraju, S. R. (1995). A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. *Journal of the ACM*, 42(1):67–90.
- [Chandra et al., 1996] Chandra, T. D., Hadzilacos, V., and Toueg, S. (1996). The weakest failure detector for solving Consensus. *Journal of the ACM*, 43(4):685–722.

Bibliography

- [Chang and Nelson, 1995] Chang, C. S. and Nelson, R. (1995). Bounds on the speedup and efficiency of partial synchronization in parallel processing systems. *Journal of the ACM*, 42(1):204–231.
- [Chen and Warren, 1996] Chen, W. and Warren, D. S. (1996). Tabled evaluation with delaying for general logic programs. *Journal of the ACM*, 43(1):20–74.
- [Cheng and Muntz, 1996] Cheng, W. C. and Muntz, R. R. (1996). Bounding errors introduced by clustering of customers in closed product-form queueing networks. *Journal of the ACM*, 43(4):641–669.
- [Chien and Kim, 1995] Chien, A. A. and Kim, J. H. (1995). Planar-adaptive routing: Low-cost adaptive networks for multiprocessors. *Journal of the ACM*, 42(1):91–123.
- [Choudhury et al., 1995] Choudhury, G. L., Leung, K. K., and Whitt, W. (1995). Calculating normalization constants of closed queueing networks by numerically inverting their generating functions. *Journal of the ACM*, 42(5):935–970.
- [Clarkson, 1995] Clarkson, K. L. (1995). Las Vegas algorithms for linear and integer programming when the dimension is small. *Journal of the ACM*, 42(2):488–499.
- [Coffman, Jr. and Garey, 1993] Coffman, Jr., E. G. and Garey, M. R. (1993). Proof of the $4/3$ conjecture for preemptive vs. nonpreemptive two-processor scheduling. *Journal of the ACM*, 40(5):991–1018.
- [Cohen and Megiddo, 1993] Cohen, E. and Megiddo, N. (1993). Strongly polynomial-time and NC algorithms for detecting cycles in periodic graphs. *Journal of the ACM*, 40(4):791–830.
- [Compton and Ravishankar, 1995] Compton, K. J. and Ravishankar, C. (1995). Expected deadlock time in a multiprocessing system. *Journal of the ACM*, 42(3):562–583.
- [Conforti and Cornuéjols, 1995] Conforti, M. and Cornuéjols, G. (1995). A class of logic problems solvable by linear programming. *Journal of the ACM*, 42(5):1107–1113.
- [Conway et al., 1994] Conway, A. E., Pinsky, E., and Tridandapani, S. (1994). Efficient decomposition methods for the analysis of multi-facility blocking models. *Journal of the ACM*, 41(4):648–675.
- [Cosnard and Daoudi, 1994] Cosnard, M. and Daoudi, E. M. (1994). Optimal algorithms for parallel givens factorization on a coarse-grained PRAM. *Journal of the ACM*, 41(2):399–421.
- [Courcoubetis and Yannakakis, 1995] Courcoubetis, C. and Yannakakis, M. (1995). The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907.

- [Curien et al., 1996] Curien, P.-L., Hardin, T., and Lévy, J.-J. (1996). Confluence properties of weak and strong calculi of explicit substitutions. *Journal of the ACM*, 43(2):362–397.
- [Dallery et al., 1994] Dallery, Y., Liu, Z., and Towsley, D. (1994). Equivalence, reversibility, symmetry and concavity properties in fork-join queuing networks with blocking. *Journal of the ACM*, 41(5):903–942.
- [Dolev et al., 1993] Dolev, D., Dwork, C., Waarts, O., and Yung, M. (1993). Perfectly secure message transmission. *Journal of the ACM*, 40(1):17–47.
- [Dolev et al., 1995] Dolev, D., Halpern, J. Y., Simons, B., and Strong, R. (1995). Dynamic fault-tolerant clock synchronization. *Journal of the ACM*, 42(1):143–185.
- [Donald et al., 1993] Donald, B., Xavier, P., Canny, J., and Reif, J. (1993). Kinodynamic motion planning. *Journal of the ACM*, 40(5):1048–1066.
- [Driscoll et al., 1994] Driscoll, J. R., Sleator, D. D. K., and Tarjan, R. E. (1994). Fully persistent lists with catenation. *Journal of the ACM*, 41(5):943–959.
- [Drusinsky and Harel, 1994] Drusinsky, D. and Harel, D. (1994). On the power of bounded concurrency I: Finite automata. *Journal of the ACM*, 41(3):517–539.
- [Dubiner et al., 1994] Dubiner, M., Galil, Z., and Magen, E. (1994). Faster tree pattern matching. *Journal of the ACM*, 41(2):205–213.
- [Eiter and Gottlob, 1995] Eiter, T. and Gottlob, G. (1995). The complexity of logic-based abduction. *Journal of the ACM*, 42(1):3–42.
- [Fagin and Halpern, 1994] Fagin, R. and Halpern, J. Y. (1994). Reasoning about knowledge and probability. *Journal of the ACM*, 41(2):340–367.
- [Feige et al., 1996] Feige, U., Goldwasser, S., Lovász, L., Safra, S., and Szegedy, M. (1996). Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292.
- [Fekete et al., 1993] Fekete, A., Lynch, N., Mansour, Y., and Spinelli, J. (1993). The impossibility of implementing reliable communication in the face of crashes. *Journal of the ACM*, 40(5):1087–1107.
- [Fischer and Paterson, 1994] Fischer, M. J. and Paterson, M. S. (1994). Fishspear: A priority queue algorithm. *Journal of the ACM*, 41(1):3–30.
- [Fonlupt and Nacheff, 1993] Fonlupt, J. and Nacheff, A. (1993). Dynamic programming and the graphical traveling salesman problem. *Journal of the ACM*, 40(5):1165–1187.
- [Freivalds et al., 1995] Freivalds, R., Kinber, E., and Smith, C. H. (1995). On the impact of forgetting on learning machines. *Journal of the ACM*, 42(6):1146–1168.

Bibliography

- [Galil, 1995] Galil, Z. (1995). A constant-time optimal parallel string-matching algorithm. *Journal of the ACM*, 42(4):908–918.
- [Gallier et al., 1993] Gallier, J., Narendran, P., Plaisted, D., Raatz, S., and Snyder, W. (1993). An algorithm for finding canonical sets of ground rewrite rules in polynomial time. *Journal of the ACM*, 40(1):1–16.
- [Gao and Kaufmann, 1994] Gao, S. and Kaufmann, M. (1994). Channel routing of multiterminal nets. *Journal of the ACM*, 41(4):791–818.
- [Glass and Ni, 1994] Glass, C. J. and Ni, L. M. (1994). The turn model for adaptive routing. *Journal of the ACM*, 41(5):874–902.
- [Goemans and Williamson, 1995] Goemans, M. X. and Williamson, D. P. (1995). Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145.
- [Goldreich and Ostrovsky, 1996] Goldreich, O. and Ostrovsky, R. (1996). Software protection and simulation on oblivious RAMs. *Journal of the ACM*, 43(3):431–473.
- [Golumbic and Shamir, 1993] Golumbic, M. C. and Shamir, R. (1993). Complexity and algorithms for reasoning about time: A graph-theoretic approach. *Journal of the ACM*, 40(5):1108–1133.
- [Goodrich and Kosaraju, 1996] Goodrich, M. T. and Kosaraju, S. R. (1996). Sorting on a parallel pointer machine with applications to set expression evaluation. *Journal of the ACM*, 43(2):331–361.
- [Gottlob, 1995a] Gottlob, G. (1995a). NP trees and Carnap’s modal logic. *Journal of the ACM*, 42(2):421–457.
- [Gottlob, 1995b] Gottlob, G. (1995b). Translating default logic into standard autoepistemic logic. *Journal of the ACM*, 42(4):711–740.
- [Haldar and Vidyasankar, 1995] Haldar, S. and Vidyasankar, K. (1995). Constructing 1-writer multireader multivalued atomic variable from regular variables. *Journal of the ACM*, 42(1):186–203.
- [Halpern and Tuttle, 1993] Halpern, J. Y. and Tuttle, M. R. (1993). Knowledge, probability, and adversaries. *Journal of the ACM*, 40(4):917–962.
- [Harper et al., 1993] Harper, R., Honsell, F., and Plotkin, G. (1993). A framework for defining logics. *Journal of the ACM*, 40(1):143–184.
- [Hellerstein et al., 1996] Hellerstein, L., Pillaipakkamnatt, K., Raghavan, V., and Wilkins, D. (1996). How many queries are needed to learn? *Journal of the ACM*, 43(5):840–862.

- [Hirst and Harel, 1994] Hirst, T. and Harel, D. (1994). On the power of bounded concurrency II: Pushdown automata. *Journal of the ACM*, 41(3):540–554.
- [Hobby, 1993] Hobby, J. D. (1993). Generating automatically tuned bitmaps from outlines. *Journal of the ACM*, 40(1):48–94.
- [Jean-Marie and Gün, 1993] Jean-Marie, A. and Gün, L. (1993). Parallel queues with resequencing. *Journal of the ACM*, 40(5):1188–1208.
- [Johnson, 1993] Johnson, C. A. (1993). Factorization and circuit in the connection method. *Journal of the ACM*, 40(3):536–557.
- [Joung and Smolka, 1996] Joung, Y.-J. and Smolka, S. A. (1996). A comprehensive study of the complexity of multiparty interaction. *Journal of the ACM*, 43(1):75–115.
- [Kahale, 1995] Kahale, N. (1995). Eigenvalues and expansion of regular graphs. *Journal of the ACM*, 42(5):1091–1106.
- [Karger et al., 1995] Karger, D. R., Klein, P. N., and Tarjan, R. E. (1995). A randomized linear-time algorithm to find minimum spanning trees. *Journal of the ACM*, 42(2):321–328.
- [Karger and Stein, 1996] Karger, D. R. and Stein, C. (1996). A new approach to the minimum cut problem. *Journal of the ACM*, 43(4):601–640.
- [Karloff and Raghavan, 1993] Karloff, H. J. and Raghavan, P. (1993). Randomized algorithms and pseudorandom numbers. *Journal of the ACM*, 40(3):454–476.
- [Karp, 1994] Karp, R. M. (1994). Probabilistic recurrence relations. *Journal of the ACM*, 41(6):1136–1150.
- [Karp and Zhang, 1993] Karp, R. M. and Zhang, Y. (1993). Randomized parallel algorithms for backtrack search and branch-and-bound computation. *Journal of the ACM*, 40(3):765–789.
- [Kearns and Valiant, 1994] Kearns, M. and Valiant, L. (1994). Cryptographic limitations on learning Boolean formulae and finite automata. *Journal of the ACM*, 41(1):67–95.
- [Kfoury et al., 1994] Kfoury, A. J., Tiuryn, J., and Urzyczyn, P. (1994). An analysis of ML typability. *Journal of the ACM*, 41(2):368–398.
- [Khuller and Vishkin, 1994] Khuller, S. and Vishkin, U. (1994). Biconnectivity approximations and graph carvings. *Journal of the ACM*, 41(2):214–235.
- [Kifer et al., 1995] Kifer, M., Lausen, G., and Wu, J. (1995). Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42(4):741–843.
- [Knessl, 1993] Knessl, C. (1993). On the sojourn time distribution in a finite capacity processor shared queue. *Journal of the ACM*, 40(5):1238–1301.

Bibliography

- [Korilis and Lazar, 1995] Korilis, Y. A. and Lazar, A. A. (1995). On the existence of equilibria in noncooperative optimal flow control. *Journal of the ACM*, 42(3):584–613.
- [Kościelski and Pacholski, 1996] Kościelski, A. and Pacholski, L. (1996). Complexity of Makanin’s algorithm. *Journal of the ACM*, 43(4):670–684.
- [Koutsoupias and Papadimitriou, 1995] Koutsoupias, E. and Papadimitriou, C. H. (1995). On the k-server conjecture. *Journal of the ACM*, 42(5):971–983.
- [Kurtz et al., 1995] Kurtz, S. A., Mahaney, S. R., and Royer, J. S. (1995). The isomorphism conjecture fails relative to a random oracle. *Journal of the ACM*, 42(2):401–420.
- [Ladkin and Maddux, 1994] Ladkin, P. B. and Maddux, R. D. (1994). On binary constraint problems. *Journal of the ACM*, 41(3):435–469.
- [LaPaugh, 1993] LaPaugh, A. S. (1993). Recontamination does not help to search a graph. *Journal of the ACM*, 40(2):224–245.
- [Lengauer and Wanke, 1993] Lengauer, T. and Wanke, E. (1993). Efficient decision procedures for graph properties on context-free graph languages. *Journal of the ACM*, 40(2):368–393.
- [Leung, 1993] Leung, K. K. (1993). An execution/sleep scheduling policy for serving an additional job in priority queueing systems. *Journal of the ACM*, 40(2):394–417.
- [Li et al., 1996] Li, M., Tromp, J., and Vitányi, P. M. B. (1996). How to share concurrent wait-free variables. *Journal of the ACM*, 43(4):723–746.
- [Lin and Shoham, 1995] Lin, F. and Shoham, Y. (1995). Provably correct theories of action. *Journal of the ACM*, 42(2):293–320.
- [Linial et al., 1993] Linial, N., Mansour, Y., and Nisan, N. (1993). Constant depth circuits, Fourier transform and learnability. *Journal of the ACM*, 40(3):607–620.
- [Lui and Muntz, 1994] Lui, J. C. S. and Muntz, R. R. (1994). Computing bounds on steady state availability of repairable computer systems. *Journal of the ACM*, 41(4):676–707.
- [Lund and Yannakakis, 1994] Lund, C. and Yannakakis, M. (1994). On the hardness of approximating minimization problems. *Journal of the ACM*, 41(5):960–981.
- [Luo and Tsitsiklis, 1993] Luo, Z. and Tsitsiklis, J. N. (1993). On the communication complexity of distributed algebraic computation. *Journal of the ACM*, 40(5):1019–1047.
- [Marcus and Subrahmanian, 1996] Marcus, S. and Subrahmanian, V. S. (1996). Foundations of multimedia database systems. *Journal of the ACM*, 43(3):474–523.
- [McAllester and Givan, 1993] McAllester, D. and Givan, R. (1993). Taxonomic syntax for first order inference. *Journal of the ACM*, 40(2):246–283.

- [McAllester, 1993] McAllester, D. A. (1993). Automatic recognition of tractability in inference relations. *Journal of the ACM*, 40(2):284–303.
- [Mehlhorn and Tsakalidis, 1993] Mehlhorn, K. and Tsakalidis, A. (1993). Dynamic interpolation search. *Journal of the ACM*, 40(3):621–634.
- [Miltersen et al., 1996] Miltersen, P. B., Paterson, M., and Tarui, J. (1996). The asymptotic complexity of merging networks. *Journal of the ACM*, 43(1):147–165.
- [Motwani, 1994] Motwani, R. (1994). Average-case analysis of algorithms for matchings and related problems. *Journal of the ACM*, 41(6):1329–1356.
- [Murray and Rosenthal, 1993] Murray, N. V. and Rosenthal, E. (1993). Dissolution: Making paths vanish. *Journal of the ACM*, 40(3):504–535.
- [Naughton and Ramakrishnan, 1994] Naughton, J. F. and Ramakrishnan, R. (1994). How to forget the past without repeating it. *Journal of the ACM*, 41(6):1151–1177.
- [Nebel and Bürckert, 1995] Nebel, B. and Bürckert, H.-J. (1995). Reasoning about temporal relations: A maximal tractable subclass of Allen’s interval algebra. *Journal of the ACM*, 42(1):43–66.
- [Nederhof and Bertsch, 1996] Nederhof, M.-J. and Bertsch, E. (1996). Linear-time suffix parsing for deterministic languages. *Journal of the ACM*, 43(3):524–554.
- [Neiger and Toueg, 1993] Neiger, G. and Toueg, S. (1993). Simulating synchronized clocks and common knowledge in distributed systems. *Journal of the ACM*, 40(2):334–367.
- [Nelson and Towsley, 1993] Nelson, R. and Towsley, D. (1993). A performance evaluation of several priority policies for parallel processing systems. *Journal of the ACM*, 40(3):714–740.
- [Nicol, 1993] Nicol, D. M. (1993). The cost of conservative synchronization in parallel discrete event simulations. *Journal of the ACM*, 40(2):304–333.
- [Nicola and Vaandrager, 1995] Nicola, R. D. and Vaandrager, F. (1995). Three logics for branching bisimulation. *Journal of the ACM*, 42(2):458–487.
- [Nodine and Vitter, 1995] Nodine, M. H. and Vitter, J. S. (1995). Greed sort: Optimal deterministic sorting on parallel disks. *Journal of the ACM*, 42(4):919–933.
- [O’Hearn and Tennent, 1995] O’Hearn, P. W. and Tennent, R. D. (1995). Parametricity and local variables. *Journal of the ACM*, 42(3):658–709.
- [Orponen et al., 1994] Orponen, P., Ko, K., Schöning, U., and Watanabe, O. (1994). Instance complexity. *Journal of the ACM*, 41(1):96–121.

Bibliography

- [Pitt and Warmuth, 1993] Pitt, L. and Warmuth, M. K. (1993). The minimum consistent DFA problem cannot be approximated within any polynomial. *Journal of the ACM*, 40(1):95–142.
- [Rabin, 1994] Rabin, T. (1994). Robust sharing of secrets when the dealer is honest or cheating. *Journal of the ACM*, 41(6):1089–1109.
- [Rathmann et al., 1994] Rathmann, P. K., Winslett, M., and Manasse, M. (1994). Circumscription with homomorphisms: Solving the equality and counterexample problems. *Journal of the ACM*, 41(5):819–873.
- [Reif and Sharir, 1994] Reif, J. and Sharir, M. (1994). Motion planning in the presence of moving obstacles. *Journal of the ACM*, 41(4):764–790.
- [Reif and Storer, 1994] Reif, J. H. and Storer, J. A. (1994). A single-exponential upper bounds for finding shortest paths in three dimensions. *Journal of the ACM*, 41(5):1013–1019.
- [Rivest and Schapire, 1994] Rivest, R. L. and Schapire, R. E. (1994). Diversity-based inference of finite automata. *Journal of the ACM*, 41(3):555–589.
- [Rockmore, 1994] Rockmore, D. N. (1994). Efficient computation of Fourier inversion for finite groups. *Journal of the ACM*, 41(1):31–66.
- [Ross, 1994] Ross, K. A. (1994). Modular stratification and magic sets for datalog programs with negation. *Journal of the ACM*, 41(6):1216–1266.
- [Ross et al., 1994] Ross, K. W., Tsang, D. H. K., and Wang, J. (1994). Monte Carlo summation and integration applied to multiclass queuing networks. *Journal of the ACM*, 41(6):1110–1135.
- [Santos, 1996] Santos, Jr., E. (1996). On linear potential functions for approximating Bayesian computations. *Journal of the ACM*, 43(3):399–430.
- [Selman and Kautz, 1996] Selman, B. and Kautz, H. (1996). Knowledge compilation and theory approximation. *Journal of the ACM*, 43(2):193–224.
- [Singh et al., 1994] Singh, A. K., Anderson, J. H., and Gouda, M. G. (1994). The elusive atomic register. *Journal of the ACM*, 41(2):311–339.
- [Storer and Reif, 1994] Storer, J. A. and Reif, J. H. (1994). Shortest paths in the plane with polygonal obstacles. *Journal of the ACM*, 41(5):982–1012.
- [Tay, 1993] Tay, Y. C. (1993). On the optimality of strategies for multiple joins. *Journal of the ACM*, 40(5):1067–1086.
- [Tempero and Ladner, 1995] Tempero, E. D. and Ladner, R. E. (1995). Recoverable sequence transmission protocols. *Journal of the ACM*, 42(5):1059–1090.

- [Toyama et al., 1995] Toyama, Y., Klop, J. W., and Barendregt, H. P. (1995). Termination for direct sums of left-linear complete term rewriting systems. *Journal of the ACM*, 42(6):1275–1304.
- [Tsitsiklis and Stamoulis, 1995] Tsitsiklis, J. N. and Stamoulis, G. D. (1995). On the average communication complexity of asynchronous distributed algorithms. *Journal of the ACM*, 42(2):382–400.
- [van Beek and Dechter, 1995] van Beek, P. and Dechter, R. (1995). On the minimality and global consistency of row-convex constraint networks. *Journal of the ACM*, 42(3):543–561.
- [van Glabbeek and Weijland, 1996] van Glabbeek, R. J. and Weijland, W. P. (1996). Branching time and abstraction in bisimulation semantics. *Journal of the ACM*, 43(3):555–600.
- [van Kreveld and Overmars, 1993] van Kreveld, M. J. and Overmars, M. H. (1993). Union-copy structures and dynamic segment trees. *Journal of the ACM*, 40(3):635–652.
- [Verma, 1995] Verma, R. M. (1995). A theory of using history for equational systems with applications. *Journal of the ACM*, 42(5):984–1020.
- [Vitter and Krishnan, 1996] Vitter, J. S. and Krishnan, P. (1996). Optimal prefetching via data compression. *Journal of the ACM*, 43(5):771–793.
- [Wang et al., 1995] Wang, K., Zhang, W., and Chau, S.-C. (1995). Decomposition of magic rewriting. *Journal of the ACM*, 42(2):329–381.
- [Wang, 1993] Wang, T. (1993). Z-module reasoning: An equality-oriented proving method with built-in ring axioms. *Journal of the ACM*, 40(3):558–606.
- [Yu et al., 1993] Yu, P. S., Dias, D. M., and Lavenberg, S. S. (1993). On the analytical modeling of database concurrency control. *Journal of the ACM*, 40(4):831–872.

Bibliography