

# LATEX



SOOCHOW UNIVERSITY

排版手記

吳昊昊

# 目录

<b>第一章 L<sup>A</sup>T<sub>E</sub>X 基础</b> . . . . .	<b>1</b>	5.7 列表 . . . . .	17
1.1 L <sup>A</sup> T <sub>E</sub> X 书写环境 . . . . .	1	5.7.1 常规列表 . . . . .	17
1.2 长度与间距 . . . . .	1	5.7.2 排序列表 . . . . .	17
1.2.1 正确处理单词间距 . . . . .	2	5.7.3 解说列表 . . . . .	19
1.2.2 数学符号中的间距 . . . . .	3	5.7.4 带圈数字列表 . . . . .	19
1.3 行、段落、页面 . . . . .	3	5.8 附录 . . . . .	20
1.4 页面 . . . . .	3	5.9 代码环境 . . . . .	20
1.5 字体 . . . . .	3	<b>第六章 数学排版</b> . . . . .	<b>22</b>
1.5.1 全局字体设置 . . . . .	5	6.1 数学模式 . . . . .	22
1.5.2 局部字体设置 . . . . .	5	6.2 数学宏包 . . . . .	23
1.5.3 在数学环境中使用中文 . . . . .	5	6.3 数学符号 . . . . .	23
1.5.4 汉字“斜体” . . . . .	5	6.3.1 上标与下标 . . . . .	23
1.6 字符 . . . . .	5	6.3.2 画线补充 . . . . .	23
<b>第二章 版面和格式</b> . . . . .	<b>7</b>	6.3.3 分式 . . . . .	23
2.1 文本格式 . . . . .	7	6.3.4 斜线分式和斜线除号 . . . . .	24
2.2 标题 . . . . .	7	6.3.5 根式 . . . . .	24
2.3 页眉页脚 . . . . .	7	6.3.6 嵌套 . . . . .	24
2.4 颜色 . . . . .	8	6.3.7 定界符 . . . . .	25
2.5 标题 . . . . .	8	6.3.8 数学字体 . . . . .	26
<b>第三章 表格</b> . . . . .	<b>8</b>	6.3.9 希腊字母 . . . . .	26
3.1 浮动体 . . . . .	8	6.3.10 符号 . . . . .	27
3.2 array 宏包 . . . . .	9	6.3.11 转置符号 . . . . .	27
3.3 booktabs 宏包 . . . . .	9	6.4 公式环境 . . . . .	27
3.4 表格 . . . . .	10	6.4.1 单行公式 equation . . . . .	27
3.4.1 跨行和跨列表格 . . . . .	12	6.4.2 公式组 align 和 alignat . . . . .	28
3.4.2 彩色表格 . . . . .	13	6.4.3 公式组 gather . . . . .	29
3.4.3 斜线表头 . . . . .	13	6.4.4 多行公式 multiline . . . . .	29
3.4.4 表格标题 . . . . .	14	6.4.5 多行公式 split . . . . .	29
<b>第四章 插图</b> . . . . .	<b>14</b>	6.4.6 breqn 宏包 . . . . .	30
<b>第五章 正文工具</b> . . . . .	<b>15</b>	6.4.7 公式块 . . . . .	30
5.1 目录 . . . . .	15	6.5 矩阵环境 . . . . .	30
5.2 脚注 . . . . .	15	6.6 定理环境 . . . . .	32
5.3 边注 . . . . .	15	<b>第七章 宏包</b> . . . . .	<b>32</b>
5.4 参考文献 . . . . .	16	7.1 mhchem . . . . .	32
5.5 链接 . . . . .	17	7.2 Chemfig . . . . .	35
5.6 引用功能 . . . . .	17	7.3 C <sub>T</sub> <sub>E</sub> Xzhnumber . . . . .	35
		7.4 siunitx . . . . .	36
		7.4.1 数字 . . . . .	36
		7.4.2 单位 . . . . .	37
		7.4.3 单位命令 . . . . .	38
		7.4.4 在表格中使用单位命令 . . . . .	38

7.5	pgfplots . . . . .	38	7.6.2	更好的矩阵环境 . . . . .	39
7.6	mathtools . . . . .	38	7.6.3	长分式 . . . . .	40
7.6.1	单花括号环境 . . . . .	38			

# 第 1 节 L<sup>A</sup>T<sub>E</sub>X 基础

## 1.1. L<sup>A</sup>T<sub>E</sub>X 书写环境

T<sub>E</sub>X 环境是 TeX Live 2016，想要下载速度快可以到清华的镜像站下载。<https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/>

IDE 使用的是 TeXstudio，整个笔记的目录树如下所示，配有详细的注释。

Study-LaTeX	
├── fig .....	图片文件夹
│   └── (name.pdf/png or other suffix) .....	插入正文的图片
├── body .....	章节文件夹
│   ├── cover.pdf .....	封面
│   └── (chapter name.tex) .....	章节文件
├── main.tex .....	主编译文件
├── Zousiyu.cls .....	样式文件
├── Zousiyu.bib .....	参考文献数据库
├── gb7714-2015.bbx .....	biblatex 参考文献样式
└── gb7714-2015.cbx .....	biblatex 参考文献样式

编译使用的是脚本，xelatex.exe main.tex 能直接完成编译，加入 --synctex=-1 这个参数可以配置 TeXstudio 的反向搜索。

```
:: Copyright (c) 2012-2016 Zousiyu

@echo off
:: compile the tex file
xelatex.exe --synctex=-1 main.tex

::pause
biber main

::pause
xelatex.exe --synctex=-1 main.tex

:: clear aux files
call clear
```

clear 脚本用来清理编译时产生的辅助文件，视情况添加后缀。

```
@echo off
del /q *.aux *.bbl *.bcf *.blg *.listing *.log *.out *.xml *.toc
```

## 1.2. 长度与间距

## 通用长度单位

首先介绍一下 T<sub>E</sub>X 中几个通用的长度单位。其中 **ex**、**em** 是**相对长度单位**，其数值大小正比于字体尺寸，当字体尺寸改变，绝对长度会随之改变。其他单位是绝对长度单位。

表 1.1 T<sub>E</sub>X 中常用的长度单位

单位	名称	说明
pt	点	欧美传统排版的长度单位，1pt=0.351mm
pc	派卡	相当于四号字大小，1pc=12pt=4.218mm
in	英寸	inch 英寸，1in=72.27pt=25.4mm
bp	大点	big point，1in=72bp)
cm/mm	都学过	1cm=28.453pt，1mm=2.845pt
em	em	当前字体中 M 的宽度
ex	ex	当前字体中 x 的高度

## 专用长度单位

**fil**、**fill**、**filll** 这三个长度单位均表示任意长，伸展能力依次递增。这几种长度单位主要用在长度无法预知或不便计算的情况下，例如将一段文字两侧用空白填满或将版面所剩空间用空白填满。

## 刚性长度与弹性长度

**刚性长度** 不会随排版情况变化而变化的长度，典型的如 **pt**、**em** 等单位。

**弹性长度** 可根据排版长度有一定程度伸缩的长度，如:2mm plus 0.2mm minus 0.3mm，相当于工程标注： $2^{+0.2}_{-0.3}mm$ 。

可伸缩的弹性长度是 L<sup>A</sup>T<sub>E</sub>X 的重要排版理念之一。

## 长度命令

下面是一些常用的长度命令。

表 1.2 一些常用的产生水平间距的命令

命令	作用
<code>\quad</code>	产生一段宽度为 1em 的水平空白
<code>\qqquad</code>	<code>\quad</code> 的两倍
<code>\,</code>	大约为 <code>\quad</code> 的 3/18
<code>\hspace{length}</code>	产生指定宽度的水平空白
<code>\hspace*{length}</code>	若要在行首产生一定的空白，则需使用此命令
<code>\hfill</code>	产生撑满整行的空白
<code>\hphantom{text}</code>	幻影命令，产生的空白等于 <code>text</code> 的宽度

表 1.3 一些常用的产生垂直空白的命令

命令	作用
<code>\smallskip</code>	产生高度为 3pt plus 1pt minus 1pt 的垂直空白

<code>\medskip</code>	<code>\smallskip</code> 的两倍
<code>\bigskip</code>	<code>\smallskip</code> 的四倍
<code>\vspace{length}</code>	产生指定高度的垂直空白
<code>\vspace*{length}</code>	在页面顶部产生垂直空白
<code>\vfill</code>	插入指定高度的垂直空白
<code>\vphantom{text}</code>	幻影命令, 产生的空白等于 <code>text</code> 的高度, 和 <code>text</code> 的长度无关的

### 1.2.1. 正确处理单词间距

英文排版时 T<sub>E</sub>X 通常默认句号 . 表示一句话的结束, 因此 T<sub>E</sub>X 在处理句号时会留出稍宽一点的水平间距。但是有些情况下, 句号并不代表句子的结尾, 比如「i.e. a word」和「e.g. a word」。按照 T<sub>E</sub>X 默认规则, 排版出的宽度会比正常句中单词之间的间隔稍大一些, 因此我们需要使用使用 \ (即一个反斜杠 + 空格) 来消除这个过大的间距: `i.e.\ a word` 以及 `e.g.\ a word`。仔细观察下面例子的排版效果。

<code>i.e. a word\par</code>	i.e. a word
<code>i.e.\ a word</code>	i.e. a word

句号跟在一个大写字母的后面, 此时 T<sub>E</sub>X 会认为这个句号表示人名缩写的间隔符, 因此仍然按照正常间距来排版, 比如「A. Einstein」。然而这个看似贴心的规则在一些情况下会适得其反, 比如一句话明明以缩略语结尾, T<sub>E</sub>X 反而认为这并不是一句话的结尾: 「... in NBA. He...」。此时, 排版出的「He」之前的空格会小于正常的句间间距。这种情况下, 需要使用 \@. (反斜杠 + @ + 句号 + 空格), 来强制告诉 T<sub>E</sub>X 这里的的确确是一个句子的结尾。

<code>... played in NBA. He was ...\par</code>	... played in NBA. He was ...
<code>... played in NBA\@. He was ...</code>	... played in NBA. He was ...

以上规则除句号外, 同样适用于感叹号和问号等其他符号。

### 1.2.2. 数学符号中的间距

数学公式中, 积分符号  $dx$ <sup>①</sup> 前应该加入一个间距 !, , 同时在公式结尾的标定符号与公式之间也应该插入一个间距 !, 。此外, 积分符号  $\int$  与被积分项之间的间距在默认情况下过大, 完美的排版需要利用 \! 来缩小这个间距。如下  $\int$  与  $f(x)$ ,  $f(x)$  与  $dx$ ,  $\alpha$  与  $.$  之间的间距都值得注意。

<code>\[ \int_a^b f(x)dx = \alpha.</code>	$\int_a^b f(x)dx = \alpha.$
<code>\]\par</code>	
<code>\[ \int_a^b \! f(x)\!, dx = \alpha</code>	$\int_a^b f(x) dx = \alpha.$
<code>\[, . \]</code>	

$$\int_a^b f(x)dx = \alpha.$$

$$\int_a^b f(x) dx = \alpha.$$

## 1.3. 行、段落、页面

### 避免数字出现在行首

使用 ~ 来代替空格可以避免交叉引用或者输入人名时尴尬地被打破成两行, 例如 ... 如图 ~\ref{Fig1} 或者 ...A.~Einstein said...。

<sup>①</sup>积分符号是直立还是斜体尚有争论

## 中英混排时空格的使用

中英文混排时, X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X 能在中文与英文(或数字)之间, 没有必要手动敲入一个空格, 编译时会自动为中文与英文(或数字)之间添加合适的间距。但是有一个情况比较特殊, 就是在交叉引用时, 这个空格是需要手动敲入的, 否则这个间距会消失。

- 换行** L<sup>A</sup>T<sub>E</sub>X 会自动换行, 若需强制换行, 可使用`\`或`\newline`。`\`后面可以带长度, 以增加当前行与新行之间的距离, 参数可正可负, 如: `\[3mm]`, `\[-5pt]`。
- 分段** 两个连续回车(即一个空行)或`\par`。
- 分页** L<sup>A</sup>T<sub>E</sub>X 会自动分页若需强制分页, 可用命令`\newpage`或`\clearpage`。

### 1.4. 页面

### 1.5. 字体

- 等宽字体** Typewriter Family 英文的 a 和 i 在非等宽字体里面肯定宽度不一样, 这样在大段文本里就不好辨认, 等宽字体的所以字母宽度一样, 笔画的起止还有装饰衬线(所以等宽字体多数属于衬线字体), 易读性高
- 等线字体** 无字头字脚, 笔画圆润, 粗细均匀, 例如 Windows 自带的 Arial、黑体和幼圆
- 衬线字体** serif 在字的笔画开始、结束的地方有额外的装饰, 而且笔画的粗细会有所不同, 宋体就是一种最标准的 serif 字体
- 无衬线字体** sans serif 在字的笔画开始、结束的地方没有这些额外的装饰, 而且笔画的粗细差不多

等宽字体一般用来书写代码, 特别是使用缩进控制语法的 *python* 语言, 更需要等宽字体来书写代码了。

科学书写中文文档的第一步应该是调用 C<sub>T</sub><sub>E</sub>X 宏包, 其提供四种命令来调用在中文文档中常用的四种字体。

```
{\songti 爆竹声中一岁除, 春风送暖入屠苏。}
{\fangsong 家家乞巧望秋月, 穿尽红丝几万条。}
{\heiti 黄沙百战穿金甲, 不破楼兰终不还。}
{\kaishu 君不见走马川行雪海边, 平沙莽莽黄入天。}
```

效果如下:

爆竹声中一岁除, 春风送暖入屠苏。  
家家乞巧望秋月, 穿尽红丝几万条。  
黄沙百战穿金甲, 不破楼兰终不还。  
君不见走马川行雪海边, 平沙莽莽黄入天。

汉字很少使用粗体和斜体字形, 中文文献中的粗体一般用黑体代替, 斜体一般用楷书代替。L<sup>A</sup>T<sub>E</sub>X 可以自动做到这一点, 当你使用`\textbf{文本}`或者`\bfseries`这两种粗体命令来强调汉字时, 会自动使用黑体汉字做为强调; 同样, 使用`\textit{}`或者`\itshape`这两种斜体命令来强调汉字时, 会自动使用楷书汉字做为强调。由于 x<sub>e</sub>C<sub>T</sub><sub>E</sub>X 宏包提供了设置备用字体的功能, 所以代码实现比较简单, 如下所示:

```
\setCJKmainfont[BoldFont={SimHei},ItalicFont={KaiTi}]{SimSun}
```

其中, 汉字字体名称可以使用如下命令查找, 将列出所有的中文字体的字体族名。

```
fc-list -f "%{family}\n" :lang=zh > zhfont.txt
```

```
%常见的中文字体字体族名
Microsoft YaHei, 微软雅黑
KaiTi, 楷体
SimHei, 黑体
LiSu, 隶书
YouYuan, 幼圆
FangSong, 仿宋
SimSun, 宋体

STLiti, 华文隶书
STSong, 华文宋体
STKaiti, 华文楷体
STFangsong, 华文仿宋
STXingkai, 华文行楷
STXihei, 华文细黑
STZhongsong, 华文中宋
```

fontspec 和 xeCJK 也可以使用字体的文件名访问字体。例如 Windows 下的宋体也可以使用命令：

```
\setCJKmainfont{simSun.ttc}
```

来设置。前提是字体已经被安装或者存在与 T<sub>E</sub>X 索引的目录内，否则需要另行指定路径，这里不再讨论，毕竟学术论文的写作所需字体很少，研究太多并无太大益处。

分全局和局部字体设置。

### 1.5.1. 全局字体设置

中文的文档都要调用 *ctex* 宏包，该宏包提供一个简单的参数可以设置全部正文的字体。

```
\setmainfont{Times New Roman} %设置主字体，仅对西文起作用
\setCJKmainfont{SimSun} %设置主字体，仅对中文起作用
```

有时候需要改变 L<sup>A</sup>T<sub>E</sub>X 默认的等宽字体，如本文档的等宽字体设置。更改等宽字体之后，将会影响 `\texttt{}`, `\ttfamily`, `\tt` 这些命令所作用的字体，还会影响默认使用等宽字体（如脚注，抄录环境）的环境。

```
\setmonofont{Source Code Pro} %英文等宽
\setCJKmonofont{simfang.ttf} %中文等宽，仿宋
```

### 1.5.2. 局部字体设置

```
\newfontfamily\daima{Consolas} %使用\daima直接调用
```

### 1.5.3. 在数学环境中使用中文

默认情况下，数学环境中是不允许输入汉字的。当我们需要输入汉字作为变量的标识时，可以使用 `\text{要输入的汉字字符}` 来完成这项工作。

```
$t_{\text{高温}}$
```

$t_{\text{高温}}$

### 1.5.4. 汉字“斜体”

汉字没有加斜体。平常我们看到的加斜汉字，通常是几何变换得到的结果，非常的粗糙，并不严格满足排版要求；而真正的字形是需要精细的设计的。同时，汉字字体里面也很少有加



粗体的设计。但是，有时候却又有所谓的“斜体”需求。L<sup>A</sup>T<sub>E</sub>X 也是可以实现这种伪斜体的。虽然可以实现，但排版规范并不推荐我们使用斜体来强调某个元素。如果想要强调某个元素，可以使用黑体。

## 汉字伪斜体

```
{\CJKfontspec[FakeSlant=0.4]{SimSun}\zihao{1}汉字伪斜体}
```

### 1.6. 字符

在 L<sup>A</sup>T<sub>E</sub>X 的文本内容中，大部分字符都可以直接输入，但是 #, \$, %, &, {, }, \_, ^, ~, <, >, |, \ 这几个字符由于有特殊用途不能直接输入。

```
\#, \$, \%, \&, \{, \}, \_,  
\^{}, \~{}, \textless, \  
\textgreater, \textbar, \  
\textbackslash
```

#, \$, %, &, {, }, \_, ^, ~, <, >, |, \

英文的单引号并不是两个' 符号，双引号也并不是两个" 组成的。英文下的引号嵌套需要借助 \texttt{thinspace} 命令分隔。另外，双引号的右半边用" 和" 的效果是一样的。同样，还可以使用 Unicode 字符来输入引号，输入方法麻烦，但是更加标准。

```
`\`thinspace`Max' is here.'\  
par  
Pumas are ``large, cat-like  
animals'' which are `found in  
America'.\par  
\textquotedblleft Unicode \  
\textquotedblright \par  
\textquoteleft Unicode \  
\textquoteright
```

“ ‘Max’ is here.”

Pumas are “large, cat-like animals” which are ‘found in America’.

“Unicode ”

‘Unicode ’

用一个例子解释一下为什么英文的引号需要这样输入。能看出' 打出的都是右引号！

```
'wrong'\\  
'right'
```

'wrong'

'right'

英文引号的  
错误用法

### 短横

英文的短横可以产生三种符号：

**连字符** 通常用来连接复合词，输入一个短横，-，效果如 daughter-in-law

**数学起止符** 通常用来表示范围，输入两个短横，--，效果如 page 1-2，如果真的希望连续输入两个连字符，使用 {-}{-}

**英文破折号** 是一个正规的标点符号，用来表示转折或者承上启下。破折号与其前后的单词之间不应该存在空格，输入三个短横：---，效果如 Listen—I'm serious

注意：排版中的减号应该比连字符要长，因此用来表示减号或者负号时，请严格使用数学模式而不要使用文字模式。

以上符号区别如下，注意前面讲过的数学符号中的间距这个小细节：

```
daughter-in-law\par
page 1-2\par
Listen---I'm serious\par
The temperature is $ -5\,^{\circ}\mathrm{C} $
```

```
daughter-in-law
page 1-2
Listen—I'm serious
The temperature is −5 °C
```

## 省略号

中文破折号，省略号一般直接用中文输入法输入，英文的省略号一般使用`\ldots`或者`\dots`来输入。

```
hello\ldots\par
Thanks\dots
```

```
hello...
Thanks...
```

## 摄氏度

这两个符号需要借助数学模式`$ ... $`来输入：

```
$30\,^{\circ}$\\
$37\,^{\circ}\mathrm{C}$
```

```
30°
37 °C
```

角度符号, 摄氏度符号

# 第 2 节 版面和格式

## 2.1. 文本格式

$\text{\LaTeX}$  将多个空格视为一个，多个换行也会被视为一个。一般习惯使用`~`产生一个空格，使用`\mbox{}`产生一个空白段落（实际上就是一个空白行），使用`\par`产生一个带缩进的新段，使用`\\`来强制换行，但下一段的缩进会消失。

段落之间的距离一般这样控制：

```
| \setlength{\parskip}{0pt plus 1pt}%默认值
```

用`\newpage`命令开始新的一页。

用`\clearpage`命令清空浮动体队列 5，并开始新的一页。

用`\cleardoublepage`命令清空浮动体队列，并在偶数页上开始新的一页。注意：以上命令都是基于`\vfill`的。如果要连续新开两页，请在中间加上一个空的箱，如：

```
| \newpage\mbox{}\newpage
```

$\text{\LaTeX}$  默认使用两端对齐来排版，我们可以用`\flushleft`, `\flushright`, `\center`这三个环境来构造居左，居右，居中三种版式。特殊情况可以使用`\centering`, `\raggedleft`, `\raggedright`来实现居中，居右，居左。

## 2.2. 标题

## 2.3. 页眉页脚

一般来说，设置页眉页脚需要调用使用比较广泛的 *fancyhdr* 宏包。我习惯使用如下代码先清空默认定义，然后自己重新定义。页眉页脚线的粗细也可以重新定义。

```
| \usepackage{fancyhdr}
| \pagestyle{fancy}
```

```
\fancyhf{}%清空当前设置
%单页文档
\lhead{}%l, r, c, 左中右
\cfoot{}
%双页文档
\fancyhead[R0,LE]{}%E, O, 左、右页
\fancyfoot[LE,R0]{\thepage}
\renewcommand{\headrulewidth}{0.4 pt}
\renewcommand{\footrulewidth}{0.4 pt}
```

我们可以将章节标题和序号插入到页眉或者页脚中去，其格式与正文中章节标题的定义一样。如果需要更改，要重新定义。例如，可以使用如下代码重新定义页眉内的章标题样式，用在在本书中，这将会使页眉的“第 X 章版式”更改为“X 版式”。

具体更改页眉页脚区域章节显示样式的代码如下。

```
\renewcommand{\chaptermark}[1]{\markleft{\thesection.\#1}}
%两种一样，\markleft影响\leftmark，而\makeboth影响两着，需要选一
\renewcommand{\chaptermark}[1]{\markboth{\thechapter.\#1}{节样式空置表示修改章样式}}
\renewcommand{\chaptermark}[1]{\markboth{章样式}{节样式}}
```

在 book 文件类别下，目录自动存录各章之章名，第二章 版面和格式记录节标题。所以，想要在页眉上显示章节标题是很容易实现的。

```
\lhead{\leftmark}%左页眉显示章
\rhead{\rightmark}%右页眉显示节
```

## 2.4. 颜色

一般来说，我们调用下 `xcolor` 这个宏包。如果对内置的颜色了解，或者现有 RGB 颜色值，一般使用如下代码直接调用颜色。

Color Text 中文测试

```
\color{RGB}{204, 128, 92}{Color Text 中文测试}
```

但是每次调用颜色都写颜色代码似乎不方便，我们可以先定义，基本定义形式如下。

```
\usepackage{xcolor}%颜色宏包
\definecolor{backcolor}{RGB}{242,242,242}%背景色
\definecolor{comment}{RGB}{0,128,0}%注释
\definecolor{keyword}{RGB}{0,0,255}%关键词
\definecolor{name名字随意}{model色值类型}{color-spec色值范围}
```

然后，我们就可以直接调用我们定义的颜色名称来设定颜色了。

*function, return, if, true, false*

```
\color{keyword}{\slshape function, return, if, true, false}
```

## 2.5. 标题

CT<sub>E</sub>X 宏包提供标题修改功能，所以中文文档很容易实现标题的修改。如果是直接使用 CT<sub>E</sub>X 提供的文类，标题是可以直接修改的；如果仅仅只是调用了 CT<sub>E</sub>X 宏包，需要给宏包加

上调用参数才能修改标题。

```
\usepackage[
  heading=true,%启用修改章节标题的接口
]{ctex}
```

## 第 3 节 表格

### 3.1. 浮动体

在学习 L<sup>A</sup>T<sub>E</sub>X 表格和图片的编排之前，了解一下什么是浮动体。图片和表格有时会很大，在插入的位置不一定放得下，因此需要浮动调整，这样一个浮动调整的环境就成为浮动体。

注意：因为有浮动体的存在，图片编排的位置是不确定的，所以要避免在文中使用「下图」、「上图」的说法，而是使用 *ref* 命令生成图表的编号。

浮动体将图或表与其标题定义为整体，然后动态排版，以解决图、表卡在换页处造成的过长的垂直空白的问题。但有时它也会打乱你的排版意图，因此使用与否需要根据情况决定。图片的浮动体是 `figure` 环境，而表格的浮动体是 `table` 环境。

对表格来说，输出表格内容的是 `tabular` 环境，`table` 只是一个会浮动体（到处乱跑盒子）而已。没有 `tabular` 环境，`table` 环境一样会乱跑；没有 `table` 环境，`tabular` 环境一样会输出表格内容。图片浮动体与表格是一样的。图片和表格的浮动体环境如下所示：

```
\begin{table}[!htbp]
  表 格
\end{table}
%%%%%%%%%%%%%%
\begin{figure}[!htbp]
  图 片
\end{figure}
```

! 表示忽略内部参数（比如内部参数对一页中浮动体数量的限制）；h 当前位置 (here)，t 顶部 (top)，b 底部 (bottom)，p 单独成页 (p)。L<sup>A</sup>T<sub>E</sub>X 的默认参数是 `tbp`。另外需要注意的是 `label` 命令写在 `caption` 命令下方，否则交叉引用会出现问题。

### 3.2. array 宏包

数组宏包 *array* 改进和扩展了 L<sup>A</sup>T<sub>E</sub>X 的 *tabular*、*tabular\**、*array* 环境的功能，增强了列格式的功能和一些其他表格参数的调整功能。

### 3.3. booktabs 宏包

这个宏包是用来专门排版三线表的。其形式简洁、功能分明、阅读方便，广泛用在科技论文写作中排版实验测量和计算数据。`booktabs` 宏包就是一个非常适合用来排版三线表的宏包。用法非常简单，代码如下，表 3.2 是一个三线表示例。

```
\begin{tabular}{lll}
  \toprule[2pt]
  表 格 内 容
  \midrule[0.5pt]
  表 格 内 容
  \bottomrule[0.5pt]
```

表 3.1 array 宏包基本参数

选项	说明
l	左对齐
c	居中
r	右对齐
p{列宽}	顶对齐
m{列宽}	居中对齐
b{列宽}	底对齐
@{声明}	该列每行都插入声明中的文本
>{声明}	命令或需要插入列元素前的文本
<{声明}	命令或需要插入列元素后的文本
	在列边或列间插入垂直线
!{声明}	在列间插入声明要求的样式

`\end{tabular}`

其中，每个表格只有一条 `toprule` 和 `bottomrule`，但 `midrule` 可以添加任意多。

表 3.2 Ozone decomposition of SHB mechanism

State	Equation	Reaction rate constant
Chain initiation	$\text{O}_3 + \text{OH}^- \longrightarrow \text{HO}_2 \cdot + \text{O}_2 \cdot$	$k_1 = 70 \text{ L/mol} \cdot \text{s}$
Chain transfer	$\text{HO}_2 \cdot \longrightarrow \text{O}_2^- \cdot + \text{H}^+$	$k_2 = 7.9 \times 10^5 \text{ L/(mol} \cdot \text{s)}^{25}$
	$\text{O}_2^- \cdot + \text{H}^+ \longrightarrow \text{HO}_2 \cdot$	$k_3 = 5 \times 10^{10} \text{ L/(mol} \cdot \text{s)}^{25}$
	$\text{O}_3 + \text{O}_2^- \cdot \longrightarrow \text{O}_3^- \cdot + \text{O}_2$	$k_4 = 1.6 \times 10^9 \text{ L/(mol} \cdot \text{s)}$
	$\text{O}_3^- \cdot + \text{H}^+ \longrightarrow \text{HO}_3 \cdot$	$k_5 = 5.2 \times 10^{10} \text{ L/(mol} \cdot \text{s)}$
	$\text{HO}_3 \cdot \longrightarrow \text{O}_3^- + \text{H}^+$	$k_6 = 3.3 \times 10^2 \text{ s}^{-1}$
	...	...
Chain termination	$\text{HO}_4 \cdot + \text{HO}_4 \cdot \longrightarrow \text{H}_2\text{O}_2 \cdot + 2 \text{ O}_3$	$k_{10} = 5 \times 10^9 \text{ L/(mol} \cdot \text{s)}^{25}$
	$\text{HO}_4 \cdot + \text{HO}_3 \cdot \longrightarrow \text{H}_2\text{O}_2 \cdot + \text{O}_2 + \text{O}_3$	$k_{11} = 5 \times 10^9 \text{ L/(mol} \cdot \text{s)}^{25}$

`\cmidrule`能用来画局部水平线，可以用来制作跨列表格。局部水平线可以有多条，但需要在其他 `cmidrule` 前添加`\morecmidrules`命令，否则多条局部水平线重叠为一条。

表 3.3 Weather statistics

	weather		
	rain	sunny	cloudy
months			
1	2	1	0
2	3	2	1

3.4. 表格

LaTeX 原生的表格功能非常有限,甚至不支持单元格跨行和表格跨页,我们必须通过宏包来解决。如有需求,可在 *tabular* 环境外定义全部表格线的粗细,例如,`\setlength{\arrayrulewidth}{2pt}`或者直接写`\arrayrulewidth=2pt`。

```
\centering
\arrayrulewidth=1pt%表格线宽度
\begin{tabular}
  {|c|c|c|}
  \hline
  \multicolumn{3}{|c|}{整体
  表格线宽}\\
  \hline
  7 & 5 & 3 \\
  \hline
  6 & 1 & 8 \\
  \hline
\end{tabular}
```

整体表格线宽		
7	5	3
6	1	8

如果需要单独定义某一条表格线的粗细,必须要做额外的设置。比如我们要更改垂直表格线的粗细,可以利用 `array` 宏包提供的新列格式选项定义命令。其中的新选项名只能用一个字母来表示。使用该命令更改中间两条垂直线粗细为 `2pt`。

```
\newcolumntype{新选项名称}[参数数量]{列格式}
\newcolumntype{I}{!{\vrule width 4pt}}
```

```
\centering
\newcolumntype{I}{!{\vrule
width 2pt}}
\begin{tabular}
  {|c|c|c|}
  \hline
  \multicolumn{3}{|c|}{垂直
  线粗细}\\
  \hline
  7&5&3\\
  \hline
  6&1&8\\
  \hline
\end{tabular}
```

垂直线粗细		
7	5	3
6	1	8

水平表格线的粗细较难修改,需要使用 `booktabs` 宏包,该宏包可以任意修改水平线粗细,还可以在其上、下方附加一段垂直空白。

```
\centering
\begin{tabular}
  {|c|c|c|}
  \hline
  \multicolumn{3}{|c|}{水平
  线宽}\\
  \specialrule{2pt}{0pt}{0pt}
  }
  7&5&3\\
  \hline
  6&1&8\\
  \hline
\end{tabular}
```

水平线宽		
7	5	3
6	1	8

`array` 包重新实现了 `tabular` 环境，加了不少新选项进去。比如我们可以定义 `F` 为一个居中且在数学环境中的列类型。然后在 `tabular` 中调用 `F` 即可在表格环境中排出数学样式。

```
\newcolumntype{F}{>{$}c<{$}}
\centering
\begin{tabular}{FFF}
  \alpha & \beta & \gamma \\
  \\\
  \delta & \epsilon & \\\
  \upsilon & \\\
  \sigma & \tau & \phi \\
  \\\
\end{tabular}
```

$\alpha$	$\beta$	$\gamma$
$\delta$	$\epsilon$	$\upsilon$
$\sigma$	$\tau$	$\phi$

3.4.1. 跨行和跨列表格

既跨行又跨列时，必须把 `multirow` 命令放在 `multicolumn` 内部，始终记住跨列享受最高的优先级。

```
\centering
\begin{center}
  \begin{tabular}{|c|c|c|}
    \hline
    \multirow{2}{2cm}{A
    Text!}
    & ABC & DEF \\
    \cline{2-3}
    & abc & def \\
    \hline
    \multicolumn{2}{|c|}
    {\multirow{2}{*}{Nothing
    }} & XYZ \\
    \multicolumn{2}{|c|}
    & xyz \\
    \hline
  \end{tabular}
\end{center}
```

A Text!	ABC	DEF
	abc	def
Nothing		XYZ
		xyz

```
\centering
\begin{tabular}{|ccc|}
  \hline
  2 & 9 & 4 \\
  7 & \multicolumn{2}{c|}
  {\multirow{2}{*}{?}} \\
  6 & & \\
  \hline
\end{tabular}
```

2	9	4
7	?	
6		

3.4.2. 彩色表格

利用 xcolor 宏包的颜色功能，colortbl 宏包上给表格上色。提前定义好所需要的颜色。

```
%表格颜色
\definecolor{oddrows}{RGB}{243,246,246}%奇数行
\definecolor{evenrows}{RGB}{228,228,228}%偶数行
\definecolor{header}{RGB}{0,104,183}%表头
```

使用\rowcolor{color}单独给某一行行上色，使用 xcolor 宏包提供的\rowcolors{start}{oddrows}{evenrows}快速设定奇偶行颜色。L<sup>A</sup>T<sub>E</sub>X 毕竟是做科技排版的，太花哨的彩色表格没有多大意义，用 L<sup>A</sup>T<sub>E</sub>X 做起来也难受，不要深究。表 3.4 这种表头设定深色，奇偶行颜色交替的表格在科技排版中应用较多。

3.4.3. 斜线表头

虽然斜线表头是不符合国标的，但在非正式场合用得还挺多的。制作斜线表头需要 diagbox 宏包，刘海洋写的，中文说明。



表 3.4 彩色表格演示

姓名	学号	性别
张三	2016121	Male
李四	2016122	Male
王五	2016123	Male
赵六	2016124	Male

```

\centering
\begin{tabular}{|l|ccc|}
\hline
\diagbox{Time}{Room}{Day}
&Mon&Tue&Wed\\
\hline
Morning&used&used&\\
Afternoon& &used&used\\
\hline
\end{tabular}

```

Room \diagdown Day	Mon	Tue	Wed
	Time		
Morning	used	used	
Afternoon		used	used

#### 3.4.4. 表格标题

表格标题命令默认只能在浮动体内使用，在导言中添加如下命令，便可以在浮动体外使用`\figcaption` 和`\tabcaption` 命令来为图标添加标题。为了防止标题和图表不在一页，我们也可以用`minipage` 环境把它们包起来。

```

\makeatletter
\newcommand\figcaption{\def\@capttype{figure}\caption}
\newcommand\tabcaption{\def\@capttype{table}\caption}
\makeatother

```

表 3.5 一张课表

时间	星期	
	一	二
8:30	化学	物理
9:30	韩语	数学

## 第 4 节 插图

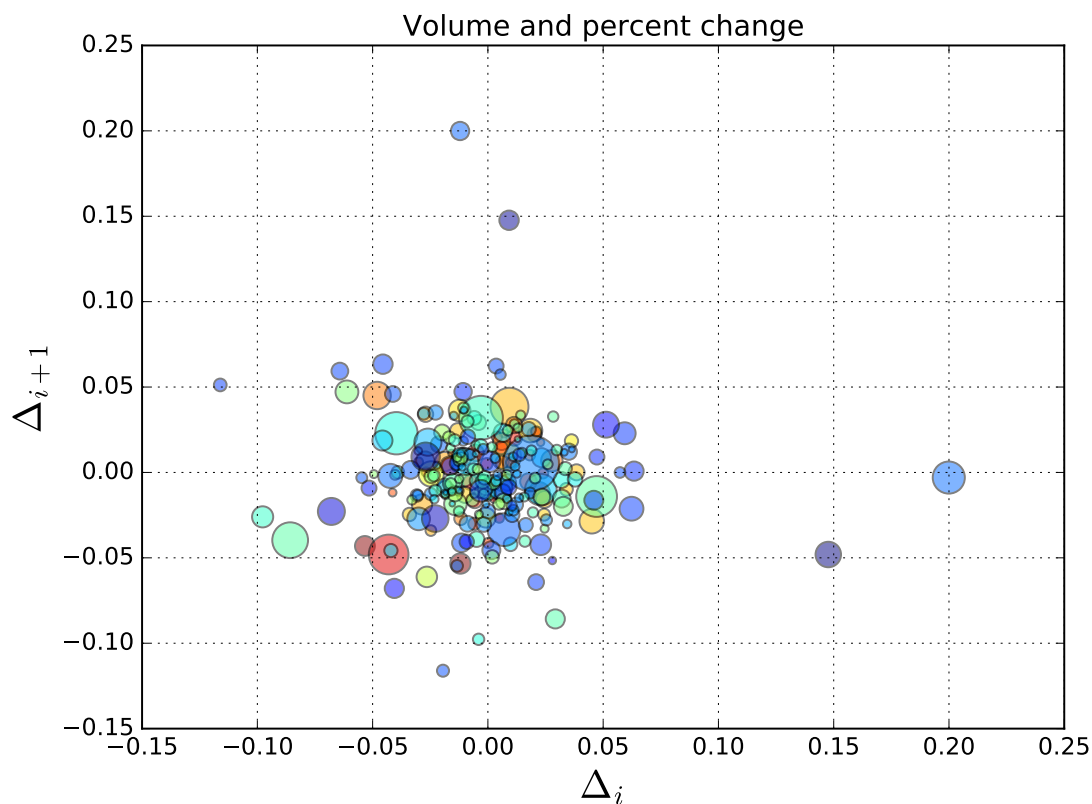


图 3.1 一副图像

## 第 5 节 正文工具

### 5.1. 目录

### 5.2. 脚注

脚注是对正文中词语的补充说明。系统提供的脚注命令如下，序号用于自行设定脚注序号，通常不需要给出。

```
|\footnote[number]{text}
```

例如，为本文作者<sup>①</sup>添加脚注。

如果要在脚注中输入带反斜杠的字符串，可使用等宽字体命令加字符串命令输入<sup>②</sup>。代码如下。如果需要更多的设置，可以调用脚注宏包 *footmisc*，对脚注命令 `\footnote` 进行扩展功能。

```
|\footnote{\texttt{\string\footnote}}
```

---

<sup>①</sup>邹思宇，男， $\text{\LaTeX}$  爱好者

<sup>②</sup>脚注命令 `\footnote`

### 5.3. 边注

LaTeX 本身提供边注命令：

```
\marginpar[左边注]{右边注}
```

边注测试。

这是边注啊

调用 `marginnote` 宏包，新定义一个边注。使用 `\bz` 调用，将会在与段落平齐的地方生成一个边注。例如：

```
% 边注和索引，来自重庆大学LaTeX团队
\renewcommand*{\marginfont}{\color{Note}\sffamily\heiti}
\DeclareDocumentCommand{\bz}{s o m}{%
  \IfBooleanTF {#1}
  {%ture
    \IfNoValueTF{#2}{\marginnote[#3]{#3}}{\marginnote[#2]{#3}}
  }{%false
    \IfNoValueTF{#2}{\marginnote[#3]{#3}}{\marginnote[#2]{#3}}
  }
  \index{#3}
}%
}
```

从这一行开始是用于重新定义边注的代码

### 5.4. 参考文献

中文著作肯定要符合《GB7714-2015 信息与文献参考文献著录规则》的要求，我习惯使用 `bibtex` 来生成参考文献。在导言区或者自定义的类文件中添加如下 1-5 行的代码，调用 `bibtex` 宏包并指定 `bib` 数据库路径<sup>①</sup>和名称。在正文中使用<sup>②</sup>第 7 行代码打印参考文献。

本书主要参考了刘海洋<sup>[1]</sup>和胡伟<sup>[2]</sup>编写的教程。使用的参考文献样式是胡振震编写的，源码托管在 [Github hushidong/bibtex-gb7714-2015](https://github.com/hushidong/bibtex-gb7714-2015) 上。

```
\usepackage[
  backend=biber,%处理方式
  style=gb7714-2015%样式
]{bibtex}
\addbibresource{Zousiyu.bib}

\printbibliography%打印参考文献
```

`bib` 参考文献数据格式如下所示，为分字段显示。各字段可以顾名思义，第一行的“刘海洋”是参考文献标识，你在文中引用参考文献时需要使用此标识。

```
@book{刘海洋,
  title={LATEX入门},
  author={刘海洋},
  publisher={电子工业出版社},
  year={2013},
}
```

参考文献使用范例，单独列出<sup>[1][2]</sup>，一起列出<sup>[1,2]</sup>

范例中使用参考文献标识引用参考文献，具体实现如下。

<sup>①</sup>文中采用的是相对路径，即数据库为我编译的 `tex` 文件的同一目录下的 `Zousiyu.bib` 文件

<sup>②</sup>一般写在 `\end{document}` 之前

```
单独列出\cite{刘海洋}\cite{胡伟}
一起列出\cite{刘海洋,胡伟}
```

## 5.5. 链接

这部分内容主要用 *hyperref* 宏包来实现。

## 5.6. 引用功能

在论文写作中，章节、插图、表格、公式和文本经常要前后调整或增添删减，这些引用的位置难以一次确定，所以不能进行直接编号。 $\text{\LaTeX}$  提供很智能的方法来解决这个问题，你不用担心引用的编号问题，只管引用就好了， $\text{\LaTeX}$  系统会帮你编号。

在你的导言区添加如下代码，重新定义自动引用的名字。

```
\AtBeginDocument{%
  \def\figureautorefname{图}
  \def\tableautorefname{表}
  \def\partautorefname{卷}
  \def\appendixautorefname{附录}
  \def\equationautorefname{式}
  \def\Itemautorefname{列表}
  \def\chapterautorefname{章}
  \def\sectionautorefname{节}
  \def\subsectionautorefname{小节}
  \def\subsubsectionautorefname{条目}
  \def\paragraphautorefname{自然段}
  \def\Hfootnoteautorefname{脚注}
  \def\AMSautorefname{式}
  \def\theoremautorefname{定理}
  \def\pageautorefname{页}
}
```

我们可以使用命令引用一个表格、公式、图片等。如使用如下命令分别引用一张表和一个带编号的公式。引用结果：如页 17，节 5.6 中式 5.6.1，表 5.1，图 5.1 所示。

```
\ref{tools-equation}
\ref{tools-tabular}
```

$$\int \operatorname{arccsc} x \, dx = x \operatorname{arccsc} x + \ln(x + \sqrt{x^2 - 1}) + C \quad (5.6.1)$$

表 5.1  $\text{\TeX}$  家族标识符

$\text{\TeX}$ 家族标识符	
$\text{\LaTeX}$	$\text{\LaTeX} 2_{\epsilon}$
$\text{\TeX}$	$\text{\XeLaTeX}$

## 5.7. 列表

### 5.7.1. 常规列表

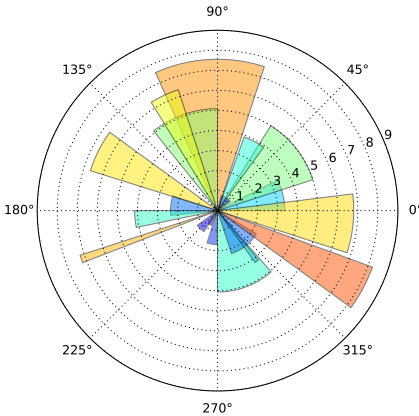


图 5.1 Demo of bar plot on a polar axis

5.7.2. 排序列表

$\text{\LaTeX}$  自带的列表环境可调整的样式很有限，调整起来也很麻烦。所以最好直接用别人写好的宏包来调整列表环境。记住一句话，要随心所欲定制  $\text{\LaTeX}$  输出的样式，就要用自由度最高的宏包，不要嫌麻烦，否则达不到想要的定制效果。`enumitem` 宏包在定制列表环境方面做得很不错，可调样式很多，能满足大部分需求。借用 `wklchris`<sup>①</sup>绘制的 `enumitem` 列表长度参数图。

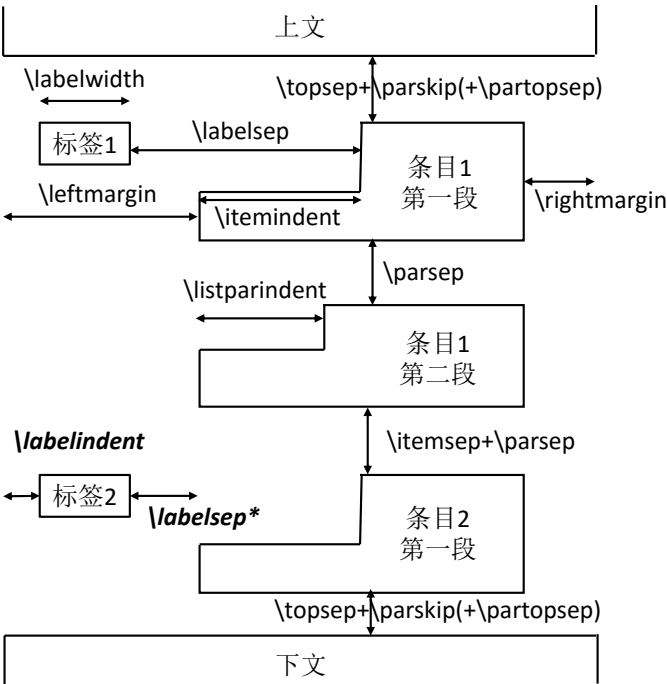


图 5.2 列表长度参数图

`enumitem` 提供的参数很多，想每一项都弄明白得花点时间，我不解释每一项参数，而是从例子开始入手。

<sup>①</sup><https://github.com/wklchris/Note-by-LaTeX>

中文文章的要求一般是，序号前缩进两个字符，列表项目之间无额外行距，列表换行后无缩进。对 `itemsep`、`topsep` 赋值，分别消除列表项目之间的间距、列表与上下文（正文）之间的间距；对 `leftmargin` 赋值，消除列表项目换行后的缩进，对 `labelindent` 赋值，控制序号前缩进两个字符，同时对 `listparindent` 赋值，控制条目换段后缩进两个字符；因为标签长度不太可控，`itemindent` 的值不好计算，所以设置其值为自动计算比较稳妥。

```
\begin{enumerate}[label=(\arabic*),itemsep=0pt,parsep=0pt,topsep=0pt,
leftmargin=0pt,labelindent=\parindent,listparindent=\parindent,itemindent=*]
\item 列表条目
\end{enumerate}
```

(1)《采桑子·辘轳金井梧桐晚》辘轳金井梧桐晚，几树惊秋。昼雨新愁，百尺虾须在玉钩。琼窗春断双蛾皱，回首边头。欲寄鳞游，九曲寒波不溯流。

《采桑子·亭前春逐红英尽》亭前春逐红英尽，舞态徘徊。细雨霏微，不放双眉时暂开。绿窗冷静芳音断，香印成灰。可奈情怀，欲睡朦胧入梦来。

(2)《长相思·一重山》一重山，两重山。山远天高烟水寒，相思枫叶丹。菊花开，菊花残。塞雁高飞人未还，一帘风月闲。

(3)《相见欢·无言独上西楼》无言独上西楼，月如钩。寂寞梧桐深院，锁清秋。剪不断，理还乱，是离愁。别是一般滋味，在心头。

### 5.7.3. 解说列表

该类型列表用于对专业术语进行解释。

### 5.7.4. 带圈数字列表

在许多文章中，特别是中文文章中，我们会见到带有圆圈的数字。它们有点是单独出现的，有点作为列表的计数出现。这里给出一个利用 `TikZ` 绘制的方法，既能在正文中调用，也能在列表中调用。基本的思路是定义一个新命令，接受一个数字参数，用 `TikZ` 在它周围画圈。同时要考虑基线和对齐的问题。代码实现如下<sup>①</sup>：

```
\usepackage{tikz}
\usepackage{etoolbox}
\newcommand{\circled}[2][1]{\tikz[baseline=(char.base)]
  {\node[shape = circle, draw, inner sep = 1pt]
    (char) {\phantom{\ifblank{#1}{#2}{#1}}};%
  \node at (char.center) {\makebox[0pt][c]{#2}};}}
\robustify{\circled}
```

这个新定义的命令可以按照 `\textcircled` 方法在正文中使用。

Numbers aligned with the text:  
`\circled{1}` `\circled{2}` `\circled{3}` end.

Numbers aligned with the text: ① ② ③  
 end.

如果需要用在列表中，则因为「脆弱命令」的问题，需要处理一下。这里我们选择使用 `etoolbox` 宏包提供的 `\robustify` 来处理一下，同时结合 `enumitem` 宏包，给出示例用法如下：

<sup>①</sup> 此法来源于 [tikz pgf - Good way to make \textcircled numbers? - TeX - LaTeX Stack Exchange](https://tex.stackexchange.com/questions/111111/good-way-to-make-textcircled-numbers)

```
\begin{enumerate}[label=\
dcircled{\arabic*}, noitemsep]
  \item 力微任重久神疲，再竭
    衰庸定不支
  \item 苟利国家生死以，岂因
    祸福避趋之
  \item 谪居正是君恩厚，养拙
    刚于戍卒宜
  \item 戏与山妻谈故事，试吟
    断送老头皮
\end{enumerate}
```

- ① 力微任重久神疲，再竭衰庸定不支
- ② 苟利国家生死以，岂因祸福避趋之
- ③ 谪居正是君恩厚，养拙刚于戍卒宜
- ④ 戏与山妻谈故事，试吟断送老头皮

## 5.8. 附录

## 5.9. 代码环境

首先载入 *listings* 宏包，定义基础代码环境，我取名为 *CodeBase*，这个基础代码环境定义的样式能被后续的代码环境调用，免去重复设置。也正是因为基础代码环境的通用性，所以这里只适合定义在所有代码环境中都适用的样式，如字体、各种边距、换行和标识等。

```
\lstdefinestyle{CodeBase}
{
  basicstyle=\small\ttfamily,
  frame=l,
  aboveskip=0pt,%上边距
  belowskip=0pt,%下边距
  lineskip=0pt,
  tabsize=4,%设置tab空格数
  showtabs=false,%Tab
  showspace=false,%空格标识
  showstringspaces=false,
  numbers=left,
  numbersep=5pt,%行号与代码距离
  numberstyle=\small\ttfamily,
  rulecolor=\color{cyan},
  boxpos=c,
  xleftmargin=1em,%左边距
  xrightmargin=0pt,
  breaklines=true,%自动换行
  breakindent=0pt,%换行后缩进为0
  extendedchars=false,%解决代码跨页时，章节标题，页眉等汉字不显示的问题
  framesep=3pt,
  rulesep=2pt,
  framerule=1pt,
  %代码颜色设置
  backgroundcolor=\color{gray!5},
  stringstyle=\color{green!40!black!100},
  keywordstyle=\bfseries\color{RGB}{0,0,255},
```

```
commentstyle=\slshape\color{black!60},
}
```

接下来，我们就可以用这个基本样式来定义一个专用于  $\text{\LaTeX}$  代码书写的样式和相应的环境。

```
%LaTeX代码环境用
\lstdefinestyle{LaTeX}
{
    style=CodeBase,
    language=[LaTeX]TeX,
    classoffset=0,
    morekeywords={addplot, begin, end},
}

%定义latex代码专用环境
\lstnewenvironment{latex}[1]{\lstset{style=LaTeX}}{}
```

最后，直接在正文中使用新定义的环境 *latex* 框住所需要展示的代码即可。

上面定义了一个  $\text{\LaTeX}$  专用的代码环境，实际使用肯定不只  $\text{\LaTeX}$  代码需要展示，还有诸如 *Python*, *MATLAB* 等大量其他代码需要展示。这里我们在定义一个用于展示 *MATLAB* 代码的环境，同样也是从基础样式 *CodeBase* 进行衍生，只需要几条简单的命令即可。

```
%matlab代码展示
\lstdefinestyle{Matlab}{
    style=CodeBase,
    language=Matlab
}

%定义Matlab代码专用环境
\lstnewenvironment{Matlab}[1]{\lstset{style=Matlab}}{}
```

*MATLAB* 代码高亮测试。

```
t=0:pi/10:2*pi;
[X,Y,Z]=cylinder(4*cos(t));
subplot(1,2,1);mesh(X);title('X');
subplot(1,2,2);mesh(Y);title('Y');
```

从 *CodeBase* 定义的新样式 *X*，其设置可以覆盖 *CodeBase* 中的设置，如下面这段 *Python* 代码高亮测试中，我们在代码中定义了一句 `keywordstyle=\slshape\color[RGB]{0,0,255},,` 让 *Python* 代码中的关键词变为斜体，其他代码环境不受影响。

```
\lstdefinestyle{python}{
    style=CodeBase,
    keywordstyle=\slshape\color[RGB]{0,0,255},%就是这句
    language=Python,
    morekeywords={def},
}
\lstnewenvironment{python}[1]{\lstset{style=python}}{}
```

*Python* 代码展示。



```
def ffmpeg_concat_av(files, output, ext):
    print('Merging video parts... ', end="", flush=True)
    params = [FFMPEG] + LOGLEVEL
    for file in files:
        if os.path.isfile(file): params.extend(['-i', file])
    params.extend(['-c:v', 'copy'])
    if ext == 'mp4':
        params.extend(['-c:a', 'aac'])
    elif ext == 'webm':
        params.extend(['-c:a', 'vorbis'])
    params.extend(['-strict', 'experimental'])
    params.append(output)
    return subprocess.call(params)
```

`listings` 宏包识别的代码关键词肯定是有限的，但好在它提供一个参数可以扩充关键词。比如我们为 `c++` 语言添加更多的关键词，只需要在设置里面写下如下代码。关键词想要多少都行，依据实际情况补充。

```
\lstset{
    morekeywords={alignas,continute,friend,register,true,alignof,decltype,
goto,reinterpret_cast,try,asm,default,if,return,typedef,auto,delete,inline
,short,typeid,bool,do,int,unsigned,typename,break,double,long,sizeof,union,
case,dynamic_cast,mutable,static,unsigned,catch,else,namespace,static_
assert,using,char,enum,new,static_cast,virtual,char16_t,char32_t,explicit,
noexcept,struct,void,export,nullptr,switch,volatile,class,extern,operator
,template,wchar_t,const,false,private,this,while,constexpr,float,
protected,thread_local,const_cast,for,public,throw,std}
},
```

当 `listings` 展示环境显示行号时，复制代码时会将行号也复制进去，可以使用如下代码解决。编译的 PDF 必须使用 Acrobat 等功能足够完善的 PDF 阅读器来查看，在 SumatraPDF 中复制代码仍然会复制到行号。

```
%复制listings生成的代码时不复制行号
\usepackage{accsupp}
\newcommand{\emptyaccsupp}[1]{\BeginAccSupp{ActualText={}}#1\EndAccSupp{}}
\lstset{%
    numberstyle=\small\ttfamily\emptyaccsupp,}
```

## 第 6 节 数学排版

终于到了  $\text{\LaTeX}$  最擅长的部分，数学排版。

### 6.1. 数学模式

分行内公式和行间公式。

行内公式，即： $\sum_{i=1}^n a_i$ ，得到： $\sum_{i=1}^n a_i$ 。

行间公式，即： $\sum_{i=1}^n a_i$ ，得到：

$$\sum_{i=1}^n a_i$$

## 6.2. 数学宏包

## 6.3. 数学符号

### 6.3.1. 上标与下标

上下标一般写在数学符号的右上、右下方，如果需要将它们写在正下、正上方，可以使用`\limits`。

```
$A_{ij}=2^{i+j}$\par
$\sum_{i=1}^n$\par
$\sum\limits_{i=1}^n$\par
```

$$A_{ij} = 2^{i+j}$$

$$\sum_{i=1}^n$$

$$\sum_{i=1}^n$$

如果是行间公式，上下标默认就在正下、正上方。另外，使用`\substack`命令可以加入多行的上下标，举个例子。

```
\[\sum_{\min}^{\max}\]
\[\sum_{\substack{i=1\\j=1}}^{\substack{i=n\\j=n}}\]
```

$$\sum_{\min}^{\max}$$

$$\sum_{i=1}^n$$

$$\sum_{j=1}^n$$

### 6.3.2. 画线补充

想划线，就拿`\overline`和`\underline`命令就可以了，划线的部分最好以花括号括起来。想画箭头则将`line`替换为`arrow`。想打双向箭头或其他，那么把`left/right`改成`leftright`<sup>①</sup>。举个例子。

```
$ \overleftarrow{abc} $\par
$ \underline{xy} $\par
$ a \leftrightarrows b $\par
$ \overleftrightarrow{abc} $
```

$$\overleftarrow{abc}$$

$$\underline{xy}$$

$$a \leftrightarrow b$$

$$\overleftrightarrow{abc}$$

如果想在数学环境里面写中文<sup>②</sup>，那么记住两件事，一是在开头引用`CTEX`宏包，二是在引用中文的之前使用`\text`命令，举一个例子。

```
$\overbrace{(a_0,a_1,\dots,a_n)}$
^{\text{共 $n+1$ 项}}$
%\dots命令能在基线上产生三个点
```

$$\overbrace{(a_0, a_1, \dots, a_n)}^{\text{共 } n+1 \text{ 项}}$$

### 6.3.3. 分式

使用命令`\frac`写出正常的分式而不是`a/b`这种的，命令之后有两个参数，如果分子分母均只有一个字符，则可以不加花括号。举例如下。

<sup>①</sup>连写，先`left`后`right`

<sup>②</sup>并不推荐这样做，只是为了符合国情才有教材在数学公式里面排版中文

%行内公式形式

```
$\frac{1}{2} \quad \frac{2}{n}
```

%行间公式形式

```
\[ \frac{1}{2} \quad \frac{2}{n}
```

```
\quad \frac{2}{2+n} \]
```

$$\frac{1}{2} \quad \frac{2}{n} \quad \frac{2}{2+n}$$

$$\frac{1}{2} \quad \frac{2}{n} \quad \frac{2}{2+n}$$

如果你想玩点花样，随意使用行内公式和行间公式，那么这里的`\frac`可以分支为`\dfrac`和`\tfrac`，t即text（行内，文本模式），d即display（行间，显示模式）。我们可以用这两个命令调节嵌套分式的大小，举个例子。

```
\[ \frac{1}{\tfrac{1}{a}+\tfrac{1}{b}+c} \]
```

```
\[ \frac{1}{\dfrac{1}{a}+\dfrac{1}{b}+c} \]
```

$$\frac{1}{\frac{1}{a} + \frac{1}{b} + c}$$

$$\frac{1}{\frac{1}{a} + \frac{1}{b} + c}$$

### 6.3.4. 斜线分式和斜线除号

对于一些需要用到斜除号的地方，如果斜除号两边的字符比较高，用常规的`/`会导致式子很不协调，这个时候可以使用`\middle/`来使得斜除号的高度与两侧字符高度相匹配。如下所示：

```
$x=a^{\frac{1}{2}}/b$\\
```

```
$x=\left.a^{\frac{1}{2}}\middle/
```

```
b\right.$
```

$$x = a^{\frac{1}{2}}/b$$

$$x = a^{\frac{1}{2}}/b$$

此外，有时候还需要用到行内斜线分式。通常，我们输入斜线分数都是键入`X/Y`，但是这个真心有点难看。我们可以用专业的`xfrac`宏包来处理这些斜线分式，它提供一个命令：

```
\sfrac{1}{2}
```

注意：这个命令可以在数学环境外使用，即在文本模式中直接使用

```
\sfrac{1}{2} \\
```

```
$ \sfrac{7}{20} $ \\
```

```
$\sfrac{1}{4}$ cups of sugar
```

$$\frac{1}{2}$$

$$\frac{7}{20}$$

$$\frac{1}{4} \text{ cups of sugar}$$

### 6.3.5. 根式

开方的次数<sup>①</sup>用方括号`[]`括起来。注意，根式的开方次数如果过大，写在左边就很影响美观，这个时候一般都改为指数形式。

```
\[ \sqrt{x^2+1} \quad \sqrt[3]{x^4+1} \]
```

$$\sqrt{x^2+1} \quad \sqrt[3]{x^4+1}$$

### 6.3.6. 嵌套

所有的公式都可以做到嵌套，这样子就可以形成相对比较复杂的公式。

<sup>①</sup>非数学专业，不知道用次数表达是否合理，欢迎指正

```
\[
\frac{-b\pm \sqrt{b^2-4ac}}{2a}\\
\lim\limits_{x\to 0}\frac{
{x\cdot \frac{\cos x-1}{\cos x}}}{x^3}
\]
```

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \lim_{x \rightarrow 0} \frac{x \cdot \frac{\cos x - 1}{\cos x}}{x^3}$$

除了在分式中会经常用到嵌套以外，矩阵里这种情况也很常见，比如分块矩阵，举个例子。当然，我们也可以把零弄大一点，我们只需要将 0 修改为`\text{\large 0}` 就好

```
\[
A=\begin{pmatrix}
\begin{matrix}
1 & 0 \\
0 & 1
\end{matrix} & 0 \\
\text{\large 0} & \begin{matrix}
1 & 0 \\
0 & 1
\end{matrix}
\end{pmatrix}
\]
```

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

### 6.3.7. 定界符

嵌套多了式子会变得非常复杂，也就会变得越来越大！可是这个时候如果你使用括号你会发现，它的大小并没有什么变化，这就显得非常的 low，影响美观，因此我们会在括号外加一个 `left` 或者是 `right` 进行大小的控制。举例如下。

```
\[
\lim\limits_{x\to 0}\left(\frac{
{a^x+b^x+c^x}}{3}\right)^{\tfrac{1}{x}}
\]
```

$$\lim_{x \rightarrow 0} \left( \frac{a^x + b^x + c^x}{3} \right)^{\frac{1}{x}}$$

```
\[ \Bigg< \bigg\{ \Big[ \big(
xyz \big) \Big] \bigg\} \Bigg>
\]
```

$$\left\langle \left\{ \left[ (xyz) \right] \right\} \right\rangle$$

学了定界符之后，就可以完全实现矩阵的部分形态了，比方说排版一个增广矩阵。

```
\[
\left(
\begin{tabular}{ccc|c}
1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1
\end{tabular}
\right)
\]
```

$$\left( \begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array} \right)$$

定界符必须成对出现, 对公式组而言, 定界符需要用在公式组环境 (`align`、`alignat`、`gather`、`aligned`、`alignedat`、`gathered`) 外面。另外, 定界符必须成对出现, 没有定界符的一侧使用 `\left.` 或者 `\right.` 来代替。了解定界符之后, 我们就可以利用定界符和公式组环境做出如下排版, 和高数书上的公式效果差不多!

$$\begin{cases} \frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i \\ N_i = -D_i \nabla c_i + u c_i \end{cases} \quad (6.3.1)$$

```
\begin{equation}
\left\{
\begin{gathered}
\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i \\
N_i = -D_i \nabla c_i + u c_i
\end{gathered}
\right.
\end{equation}
```

### 6.3.8. 数学字体

标准的  $\text{\LaTeX}$  提供的数学字体有以下几种。简单的文档中, 这些字体已经够用了, 如果要使用更高级的字体, 可查阅  $\text{\CTEX}$  宏包说明。

```
\[
\mathit{ABCDE}\]
\[\mathrm{ABCDE}\]
\[\mathbf{ABCDE}\]
\[\mathsf{ABCDE}\]
\[\mathtt{ABCDE}\]
```

*ABCDE*

ABCDE

**ABCDE**

ABCDE

ABCDE

### 6.3.9. 希腊字母

有时间排个表在这, 不着急。

### 6.3.10. 符号

规范的函数符号输入是使用命令来输入，比如指数、对数以及简单的三角函数符号等。

```
$ \sin~\cos~\exp~\log $\\  
\\[\mathcal{L}(X_i|\lambda) = \  
sum\limits_{j=1}^{n_i}\log p(  
x_{ij}|\lambda) \]
```

sin cos exp log

$$\mathcal{L}(X_i|\lambda) = \sum_{j=1}^{n_i} \log p(x_{ij}|\lambda)$$

调用 amsmath 宏包后，大多数函数符号能够使用反斜杠加名称直接打出，例如：

```
$ \sin \quad \quad \ln \quad \quad \arccos  
$
```

sin ln arccos

像 arcsec arccot arccsc 这三个函数，amsmath 宏包就没有定义，这就需要我们自己定义这样一个新的函数命令。

```
%定义一些amsmath没有定义的函数  
\DeclareMathOperator{\arcsec}{arcsec}  
\DeclareMathOperator{\arccot}{arccot}  
\DeclareMathOperator{\arccsc}{arccsc}
```

\ldots 是列举中用的省略号，而 \cdots 是运算（连加、连乘）中用的省略号，二者主要区别在于位置一高一低，切勿混用。

关于数学环境中的省略号

```
\begin{gather*}  
\{ (X_1,y_1), \ldots, (X_i,y_i)  
, \ldots, (X_{N_B},y_{N_B}) \}  
\}\\  
N_I=n_1 +n_2 + \cdots +n_T  
\end{gather*}
```

$$\{(X_1, y_1), \dots, (X_i, y_i), \dots, (X_{N_B}, y_{N_B})\}$$

$$N_I = n_1 + n_2 + \cdots + n_T$$

### 6.3.11. 转置符号

转置符号并没有严格的规定，好几种都在普遍被使用。但是有一点是明确的，转置符号不能是斜体。常见的转置符号大概有四种。

```
$\mathbf{A}^{\mathrm{T}}$\  
$\mathbf{A}^{\top}$\  
$\mathbf{A}^{\mathsf{T}}$\  
$\mathbf{A}^{\intercal}$
```

$\mathbf{A}^T$

$\mathbf{A}^\top$

$\mathbf{A}^\mathsf{T}$

$\mathbf{A}^\intercal$

推荐使用第三或者第四个，其中 \intercal 符号需要使用  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{T}\mathcal{E}\mathcal{X}$  宏包。

## 6.4. 公式环境

### 6.4.1. 单行公式 equation

无论公式多长，都被排版成一行，并给出一个序号。其间，换行命令无效，换段非法并会报错。

```
\begin{equation}
\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i
\end{equation}
```

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i \quad (6.4.1)$$

### 6.4.2. 公式组 align 和 alignat

该环境可以使公式组或者多行公式关于某个字符对齐，\\ 换行，& 用于分列，奇数列会右对齐，偶数列会左对齐，公式组的每一行都会有一个编号。

```
\begin{align}
&\lim_{x \rightarrow 1} \left( \frac{1}{1-x} - \frac{3}{1-x^3} \right) \\
&= \lim_{x \rightarrow 1} \left( \frac{x^2+x-2}{1-x^3} \right) \\
&= \lim_{x \rightarrow 1} \frac{(x+2)(x-1)}{(1-x)(x^2+x+1)} \\
&= \lim_{x \rightarrow 1} \frac{-(x+2)}{x^2+x+1} \\
&= -1
\end{align}
```

$$\lim_{x \rightarrow 1} \left( \frac{1}{1-x} - \frac{3}{1-x^3} \right) \quad (6.4.2)$$

$$= \lim_{x \rightarrow 1} \left( \frac{x^2+x-2}{1-x^3} \right) \quad (6.4.3)$$

$$= \lim_{x \rightarrow 1} \frac{(x+2)(x-1)}{(1-x)(x^2+x+1)} \quad (6.4.4)$$

$$= \lim_{x \rightarrow 1} \frac{-(x+2)}{x^2+x+1} \quad (6.4.5)$$

$$= -1 \quad (6.4.6)$$

```
\begin{align}
A_1 & \& = B_1 B_2 & A_2 & \& = B_2 \\
A_3 & \& = B_3 & A_4 & \& = B_4
\end{align}
```

$$A_1 = B_1 B_2 \qquad A_2 = B_2 \quad (6.4.7)$$

$$A_3 = B_3 \qquad A_4 = B_4 \quad (6.4.8)$$

注意：列对之间的空白与列对两侧的空白相等，像这种公式比较短的情况下就很丑。这时可以使用 alignat 环境手动控制公式间的空白，该环境必须在参数中指定列队个数。

```
\begin{alignat}{2}
A_1 & \& = B_1 B_2 & \quad \& A_2 & \& = B_2 \\
A_3 & \& = B_3 & \quad \& A_4 & \& = B_4
\end{alignat}
```

$$A_1 = B_1 B_2 \quad A_2 = B_2 \quad (6.4.9)$$

$$A_3 = B_3 \quad A_3 A_4 = B_4 \quad (6.4.10)$$

### 6.4.3. 公式组 gather

用于编写中心对称的公式组，以\\换行以区分每个公式，每个公式都会被编号。

```
\begin{gather}
\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i \\
N_i = -D_i \nabla c_i + u c_i
\end{gather}
```

$$\frac{\partial c_i}{\partial t} + \nabla \cdot (-D_i \nabla c_i) + u \cdot \nabla c_i = R_i \quad (6.4.11)$$

$$N_i = -D_i \nabla c_i + u c_i \quad (6.4.12)$$

### 6.4.4. 多行公式 multiline

适用于长公式在公式中间直接换行的情况，长公式换行并无规矩，通常在关系符（如=）和二元符之后换行。

```
\begin{multiline}
\frac{\rho}{\epsilon_p} \left( \frac{\partial u}{\partial t} + (u \cdot \nabla) \frac{u}{\epsilon_p} \right) = \\
\nabla \cdot \left[ -pl + \frac{\mu}{\epsilon_p} (\nabla u + (\nabla u)^T) - \frac{2\mu}{3\epsilon_p} (\nabla \cdot u) l \right] - \left( \mu \kappa^{-1} + \beta_F u + \frac{Q_{br}}{\epsilon_p^2} \right) u + F
\end{multiline}
```

$$\frac{\rho}{\epsilon_p} \left( \frac{\partial u}{\partial t} + (u \cdot \nabla) \frac{u}{\epsilon_p} \right) = \nabla \cdot \left[ -pl + \frac{\mu}{\epsilon_p} (\nabla u + (\nabla u)^T) - \frac{2\mu}{3\epsilon_p} (\nabla \cdot u) l \right] - \left( \mu \kappa^{-1} + \beta_F u + \frac{Q_{br}}{\epsilon_p^2} \right) u + F \quad (6.4.13)$$

### 6.4.5. 多行公式 split

适用于关于某个符号对齐的长公式，例如，我们将式(6.4.2)用 split 环境排版，用 equation 环境赋予其编号，整个公式只会得到一个编号，更符合排版规范。该环境以 & 分列，至多两列，以\\换行。

注意：split 环境不能产生编号，需要外在的公式环境提供；split 环境不能与 multiline 嵌套；autoref 可以生成“式 1.1”，eqref 可以生成“(1.1)”，视情况使用，展示无法做到直接引用成“式 (1.1)”

```
\begin{equation}
\begin{split}
&\lim_{x \rightarrow 1} \left( \frac{1}{1-x} - \frac{3}{1-x^3} \right) \\
= &\lim_{x \rightarrow 1} \left( \frac{x^2+x-2}{1-x^3} \right)
\end{split}
\end{equation}
```



```
= & \lim\limits_{x\to 1}\frac{(x+2)(x-1)}{(1-x)(x^2+x+1)}\\
= & \lim\limits_{x\to 1}\frac{-(x+2)}{x^2+x+1}\\
= & -1
\end{split}
\end{equation}
```

$$\begin{aligned}
 & \lim_{x \rightarrow 1} \left( \frac{1}{1-x} - \frac{3}{1-x^3} \right) \\
 &= \lim_{x \rightarrow 1} \left( \frac{x^2 + x - 2}{1 - x^3} \right) \\
 &= \lim_{x \rightarrow 1} \frac{(x+2)(x-1)}{(1-x)(x^2+x+1)} \\
 &= \lim_{x \rightarrow 1} \frac{-(x+2)}{x^2+x+1} \\
 &= -1
 \end{aligned} \tag{6.4.14}$$

#### 6.4.6. breqn 宏包

冲突太多，有待测试。

#### 6.4.7. 公式块

`align(alignat)`、`gather` 产生的公式组只能出现在行间，无法做为一个块出现在行内，而很多情况下我们需要一个公式组做为块出现在行间。这时我们可以使用 `aligned(alignedat)`、`gathered` 公式块环境来完成，每一行可以放置多个公式块，但块环境不提供编号。

```
\begin{equation}
\begin{aligned}
f(x,y) &= 0 \\
z &= c
\end{aligned}
\quad \text{以及} \quad
\begin{gathered}
x = t \cos t \\
z = at
\end{gathered}
\end{equation}
```

$$\begin{aligned}
 f(x,y) = 0 \\
 z = c
 \end{aligned}
 \quad \text{以及} \quad
 \begin{aligned}
 x = t \cos t \\
 z = at
 \end{aligned}
 \tag{6.4.15}$$

我们可以用公式块做一些怪东西，比如让  $f(x,y) = 0$  以及  $x = t \cos t$  做为行间公式出现在一行。让人想起了高数书上的排版呢!! 需要注意的是公式块环境只能用在数学环境中，实现的代码如下。

```
\begin{aligned}f(x,y) &= 0 \\z &= c\end{aligned}$
~以及~
\begin{gathered}x = t \cos t \\z = at\end{gathered}$
```

#### 6.5. 矩阵环境

矩阵的环境和表格有点相似，所以用法也和列表几乎相同，举个最简单的矩阵例子。

```
\[
A=\begin{matrix}
a_{11} & a_{12} & a_{13}\\
0 & 0 & 0\\
0 & 0 & 0
\end{matrix}
\]
```

$$A = \begin{matrix} & a_{11} & a_{12} & a_{13} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{matrix}$$

那我要写带括号的呢？没关系，不同的矩阵环境会形成不同的括号。这里的 `matrix` 就不形成括号，`pmatrix` 形成小括号，`bmatrix` 形成中括号，`vmatrix` 形成竖线（行列式形式），`Bmatrix` 形成大括号，`Vmatrix` 形成双竖线。

```
\begin{gather*}
%居中的公式组环境，不编号
\begin{pmatrix}1 & 2\\
3 & 4\end{pmatrix}
\begin{bmatrix}1 & 2\\
3 & 4\end{bmatrix}
\begin{vmatrix}1 & 2\\
3 & 4\end{vmatrix}
\begin{Bmatrix}1 & 2\\
3 & 4\end{Bmatrix}
\begin{Vmatrix}1 & 2\\
3 & 4\end{Vmatrix}
\end{gather*}
```

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix} \begin{Bmatrix} 1 & 2 \\ 3 & 4 \end{Bmatrix} \begin{Vmatrix} 1 & 2 \\ 3 & 4 \end{Vmatrix}$$

矩阵的元素有时候会很多，需要使用省略号去忽略，而省略号在 `tex` 中有专门的命令，列举如下。

```
\[
\ldots \cdots \vdots \ddots \dotsc
\]
```

..... $\dotsc$ .....

```
\[\begin{matrix}
1 & & & 2 & & & \cdots & & 4 \\
& & & & & & & & \\
7 & & & 6 & & & \cdots & & 5 \\
& & & & & & & & \\
\vdots & & \vdots & & \ddots & & & & \vdots \\
\vdots & & & & & & & & \\
8 & & & 9 & & & \cdots & & 0 \\
& & & & & & & & \\
\end{matrix}\]
```

$$\begin{bmatrix} 1 & 2 & \cdots & 4 \\ 7 & 6 & \cdots & 5 \\ \vdots & \vdots & \ddots & \vdots \\ 8 & 9 & \cdots & 0 \end{bmatrix}$$

## 6.6. 定理环境

## 第 7 节 宏包

L<sup>A</sup>T<sub>E</sub>X 的强大之处在于有各种各样的宏包进行扩展，这些宏包能帮助 L<sup>A</sup>T<sub>E</sub>X 完成多样的排版任务。

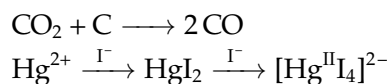
例如，在文本中排版化学式，有两个宏包比较流行，mhchem 和 chemfig。前者功能简单，用法也简单，书写无机化学式首选；后者功能众多，用法也较为繁复，主要用来书写复杂的有机化学式。

## 7.1. mhchem

使用下面的语句调用 mhchem 宏包，添加 version=4 参数是为了使用宏包的一些新特性，编写的文档如果要照顾老旧的 L<sup>A</sup>T<sub>E</sub>X 版本用户，可以酌情降低版本。

```
\usepackage[version=4]{mhchem}
```

```
\ce{CO2 + C -> 2 CO}\par
\ce{Hg^{2+} ->[I-] HgI2}
\ce{Hg^{2+} ->[I-] [Hg^{II}I4]^{2-}}
```



化学方程式

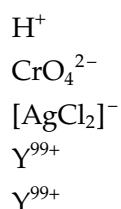
分子式可用在文本模式、数学模式和标题里面。

```
\ce{H2O}\par
\ce{Sb2O3}
```



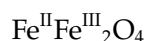
化学分子式

```
\ce{H+}\par
\ce{CrO4^{2-}}\par
\ce{[AgCl2]-}\par
\ce{Y^{99+}}\par
\ce{Y^{99+}}
```



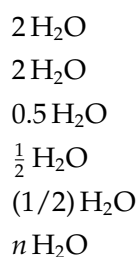
离子

```
\ce{Fe^{II}Fe^{III}2O4}
```



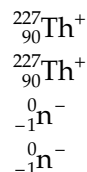
氧化价态

```
\ce{2H2O}\par
\ce{2 H2O}\par
\ce{0.5H2O}\par
\ce{1/2H2O}\par
\ce{(1/2)H2O}\par
\ce{$n$H2O}
```



化学计量数

```
\ce{^{227}_{90}Th+}\par
\ce{^{227}_{90}Th+}\par
\ce{^{0}_{-1}n^{-}}\par
\ce{^{0}_{-1}n^{-}}
```



同位素

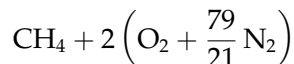
() ,[] 可以正常表示，但需要使用 \{ \} 输出花括号。大型的花括号只能在数学环境里面使用。

括弧, 方括号, 花括号

<code>\ce{(NH4)2S}\par</code>	$(\text{NH}_4)_2\text{S}$
<code>\ce{[\{(X2)3\}_2]^3+}</code>	$[\{(X_2)_3\}_2]^{3+}$

`|\[\ce{CH4 + 2 $\left( \ce{O2 + 79/21 N2} \right)$}\]`

大花括号



<code>\ce{H2(aq)}\par</code>	$\text{H}_2(\text{aq})$
<code>\ce{CO3^{2-}_{(aq)}}\par</code>	$\text{CO}_3^{2-}(\text{aq})$
<code>\ce{NaOH(aq,\$\infty\$)}</code>	$\text{NaOH}(\text{aq}, \infty)$

聚合态

<code>\ce{OCO^{.-}}\par</code>	$\text{OCO}^{\bullet-}$
<code>\ce{NO^{(2.-)}}</code>	$\text{NO}^{(2\bullet)-}$

未成对电子,  
自由基

排版约定, 变量使用斜体排版, 而其他元素 (如化学式) 则使用直立字体排版。

变量

<code>\$\ce{NO_x}\$\par</code>	$\text{NO}_x$
<code>\$\ce{Fe^{n+}}\$</code>	$\text{Fe}^{n+}$

`|\ce{x Na(NH4)HPO4 ->[\Delta] (NaPO3)_x + x NH3 ^ + x H2O}$`



<code>\ce{\mu-Cl}\par</code>	$\mu\text{-Cl}$
<code>\ce{[Pt(\eta^2-C2H4)Cl3]-}</code>	$[\text{Pt}(\eta^2\text{-C}_2\text{H}_4)\text{Cl}_3]^-$

希腊字符

<code>\ce{KCr(SO4)2*12H2O}\par</code>	$\text{KCr}(\text{SO}_4)_2 \cdot 12 \text{H}_2\text{O}$
<code>\ce{KCr(SO4)2.12H2O}\par</code>	$\text{KCr}(\text{SO}_4)_2 \cdot 12 \text{H}_2\text{O}$
<code>\ce{KCr(SO4)2 * 12 H2O}</code>	$\text{KCr}(\text{SO}_4)_2 \cdot 12 \text{H}_2\text{O}$

加成化合物

<code>\ce{C6H5-CHO}\par</code>	$\text{C}_6\text{H}_5\text{-CHO}$
<code>\ce{A-B=C#D}\par</code>	$\text{A-B=C}\equiv\text{D}$
<code>\sffamily\bfseries\ce{A-B=C#D}</code>	<b><math>\text{A-B=C}\equiv\text{D}</math></b>

化学键

<code>\ce{A\bond{-}B\bond{=}C\bond{#}D}\par</code>	
<code>\ce{A\bond{1}B\bond{2}C\bond{3}D}\par</code>	$\text{A-B=C}\equiv\text{D}$
<code>\ce{A\bond{~}B\bond{~-}C}\par</code>	$\text{A-B=C}\equiv\text{D}$
<code>\ce{A\bond{~-}B\bond{~=}C\bond{~-}D}\par</code>	$\text{A}\cdots\text{B}\equiv\text{C}$
<code>\ce{A\bond{...}B\bond{....}C}\par</code>	$\text{A}\equiv\text{B}\equiv\text{C}\equiv\text{D}$
<code>\ce{A\bond{-&gt;}B\bond{&lt;-}C}</code>	$\text{A}\cdots\text{B}\cdots\text{C}$
	$\text{A}\rightarrow\text{B}\leftarrow\text{C}$

反应箭头

```
\ce{A -> B}\par
\ce{A <- B}\par
\ce{A <-> B}\par
\ce{A <- -> B}\par
\ce{A <=>> B}\par
\ce{A <=> B}
```

$A \longrightarrow B$   
 $A \longleftarrow B$   
 $A \longleftrightarrow B$   
 $A \longleftrightarrow B$   
 $A \rightleftharpoons B$   
 $A \rightleftharpoons B$

```
\ce{A ->[H2O] B}\par
\ce{A ->[\text above][\text
below] B}\par
\ce{A ->[\$x\$][\$x_i\$] B}\par
\ce{A ->[\$x\$] B}
```

$A \xrightarrow{\text{H}_2\text{O}} B$   
 $A \xrightarrow[\text{text below}]{\text{text above}} B$   
 $A \xrightarrow[x_i]{x} B$   
 $A \xrightarrow{x} B$

带参数的反  
应箭头

```
\ce{A + B}\par
\ce{A - B}\par
\ce{A = B}\par
\ce{A \pm B}
```

$A + B$   
 $A - B$   
 $A = B$   
 $A \pm B$

化学方程式  
计算符

```
\ce{S04^2- + Ba^2+ -> BaS04 v
}\par
\ce{A v B (v) -> B ^ B (^)}
```

$\text{SO}_4^{2-} + \text{Ba}^{2+} \longrightarrow \text{BaSO}_4 \downarrow$   
 $A \downarrow B \downarrow \longrightarrow B \uparrow B \uparrow$

沉淀和气体

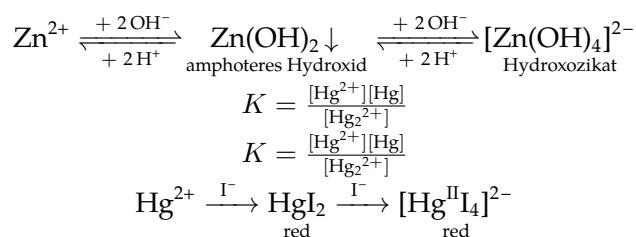
```
\ce{Zn^2+
<=>[+ 2OH-][+ 2H+]
$\underset{\text{amphoterer Hydroxid}}{\ce{Zn(OH)2 v}}$
<=>[+ 2OH-][+ 2H+]
$\underset{\text{Hydroxozikat}}{\ce{[Zn(OH)4]^2-}}$
}

$K = \frac{[\ce{Hg^2+}][\ce{Hg}]}{[\ce{Hg2^2+}]}$

$K = \frac{[\ce{Hg^2+}][\ce{Hg}]}{[\ce{Hg2^2+}]}$

\ce{Hg^2+ ->[I-]
$\underset{\text{red}}{\ce{HgI2}}$
->[I-]
$\underset{\text{red}}{\ce{[Hg^{II}I4]^2-}}$
}
```

极好的示例



## 7.2. Chemfig

## 7.3. CTeXzhnumber

以中文格式输出数字。这里的数字可以是整数、小数和分数。

```
\zhnumber{2012020120} \\
\zhnumber{2 012 020 120} \\
\zhnumber{2,012,020,120} \\
\zhnumber{2012.020120} \\
\zhnumber{2012.} \\
\zhnumber{.2012} \\
\zhnumber{20120/20120} \\
\zhnumber{/2012} \\
\zhnumber{2012/} \\
\zhnumber{201;2020/120}
```

二十亿零一千二百零二万零一百二十  
二十亿零一千二百零二万零一百二十  
二十亿零一千二百零二万零一百二十  
二千零一十二点零二零一二零  
二千零一十二点零  
零点二零一二  
二万零一百二十分之二万零一百二十  
二千零一十二分之零  
零分之二千零一十二  
二百零一又一百二十分之二千零二十

将阿拉伯数字转换为中文字符串。

```
\zhdigits{2012020120} \\
\zhdigits*{2012020120}
```

二〇一二〇二〇一二〇  
二零一二零二零一二零

将 L<sup>A</sup>T<sub>E</sub>X 计数器数值转换为中文。

```
\zhnum{section}
```

输出当天的星期。

```
\zhweekday{2012/5/20}
```

星期日

以中文格式输出日期。

```
\zhdate{2012/5/21} \\
\zhdate*{1995/01/26}
```

二〇一二年五月二十一日  
一九九五年一月二十六日星期四

以中文输出当天日期。

```
\zhtoday
```

二〇一七年一月十七日

以中文输出时间。

```
\zhtime{23:56}
```

二十三时五十六分

输出当前时间。

```
\zhcurrtime
```

二十时二十六分

输出天干计数，数字范围是 1-10。

```
\zhtiangan{1} \zhtiangan{2}
\zhtiangan{3} \zhtiangan{4}
\zhtiangan{5} \zhtiangan{10}
```

甲 乙 丙 丁 戊 癸

输出地支计数，数字范围 1-12。

带 \* 才能映  
射 0 为零

带 \* 的可以  
输出星期

```
\zhdizhi{1} \zhdizhi{2} \
zhdizhi{3}
\zhdizhi{4} \zhdizhi{5} \
zhdizhi{12}
```

子丑寅卯辰亥

输出干支计数，数字范围 1-60。

```
\zhganzhi{1} \zhganzhi{2}
\zhganzhi{3} \ \zhganzhi{4}
\zhganzhi{5} \zhganzhi{60}
```

甲子 乙丑 丙寅  
丁卯 戊辰 癸亥

输出公元纪年对应的干支纪年，公元前用负数。

```
\zhganzhinian{1898}
\zhganzhinian{-246} \ \
\zhganzhinian{-2697}
\zhganzhinian{\year}
```

戊戌 乙卯  
甲子 丁酉

*zhnumsetup* 的样式控制选项。

**Simplified** 以简体中文输出数字（对 Big5 编码无效）

**Traditional** 以繁体中文输出数字（对 Big5 编码无效）

**Normal** 以小写形式输出中文数字

**Financial** 以大写形式输出中文数字

**Ancient** 以廿输出 20，以卅输出 30，以卌输出 40，以佰输出 200

```
\zhnumsetup{
  style={Traditional,Normal}
}
\zhnumber{62012.3} \ \
\zhnumsetup{style=Ancient}
\zhnumber{21}
```

六萬二千零一十二點三  
廿一

```
\zhnumsetup{
  style={Traditional,
  Financial}}
\zhnumber{62012.3} \ \
\zhnumsetup{style=Ancient}
\zhnumber{21}
```

陸萬貳仟零壹拾貳點叁  
廿一

## 7.4. siunitx

这是一个用于书写标准国际单位制的宏包。

### 7.4.1. 数字

使用 d, D 代表  $\times 10^x$ ，使用 e, E 代表  $10^x$ ，英文的逗号, 和句号. 均可以做为小数点。

**数字输入**

```
\num{12345} \\
\num{0.123} \\
\num{0,1234} \\
\num{.12345} \\
\num{3.45d-4} \\
\num{2.2D-9} \\
\num{E9} \\
\num{-e10}
```

12 345  
0.123  
0.1234  
0.123 45  
 $3.45 \times 10^{-4}$   
 $2.2 \times 10^{-9}$   
 $10^9$   
 $-10^{10}$

```
\numlist{10;30;50;70}
\numrange{10}{30}
```

10, 30, 50 and 70 10 to 30

数字列表和  
范围

使用英文的分号; 分割角度 (度分秒), 英文的逗号, 和句号. 均可以做为小数点。

角度

```
\ang{10} \\
\ang{12.3} \\
\ang{4,5} \\
\ang{1;2;3} \\
\ang{;;1} \\
\ang{+10;;} \\
\ang{-0;1;}
```

$10^\circ$   
 $12.3^\circ$   
 $4.5^\circ$   
 $1^\circ 2' 3''$   
 $1''$   
 $10^\circ$   
 $-0^\circ 1'$

#### 7.4.2. 单位

当仅仅输入单位时, 使用\si, 英文. 和 ~ 隔开的单位会被视为相邻单位。

```
\si{kg.m/s^2} \\
\si{g_{polymer}~mol_{cat}.s^{-1}}
```

$\text{kg m/s}^2$   
 $\text{g}_{\text{polymer}} \text{mol}_{\text{cat}} \text{s}^{-1}$

很多时候, 数字和单位是在一起的, 同时输入使用\SI命令。 \SI命令结合了\num和\si两者的作用。可以使用可选参数输入一个前置单位, 该单位将会排版在数字之前。

注意: 在 siunitx 宏包环境外使用 siunitx 宏包的命令需要小心, 因为如果其他宏包定义了相同的命令, 就会有命名冲突。

\per有多种模式, 可以是分数形式。

```
\SI[mode=text]{1.23}{J.mol^{-1}.K^{-1}} \\
\SI{.23e7}{\candela} \\
\SI[per-mode=symbol]{1.99}[\$]{\per\kilogram} \\
\SI[per-mode=fraction]{1,345}{\coulomb\per\mole}
```

$1.23 \text{ J mol}^{-1} \text{ K}^{-1}$   
 $0.23 \times 10^7 \text{ cd}$   
 $\$1.99/\text{kg}$   
 $1.345 \frac{\text{C}}{\text{mol}}$

```
\SIlist{10;30;45}{\metre}
\SIrange{10}{30}{\metre}
```

10 m, 30 m and 45 m 10 m to 30 m

带单位的数  
字列表和范  
围



### 7.4.3. 单位命令

### 7.4.4. 在表格中使用单位命令

## 7.5. pgfplots

这是一个用于 2D/3D 图像绘制的宏包。

## 7.6. mathtools

这是一个数学宏包，主要填补了  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{T}\mathcal{E}\mathcal{X}$  宏包的不足。

### 7.6.1. 单花括号环境

`mathtools` 提供了更好的单花括号环境，用法与  $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{T}\mathcal{E}\mathcal{X}$  宏包提供的别无二致，但显示的效果更完美。&用于分列，奇数列会右对齐，偶数列会左对齐。

```
\begin{dcases} & \end{dcases}
\begin{dcases*} & \end{dcases*}
\begin{rcases} & \end{rcases}
\begin{rcases*} & \end{rcases*}
\begin{drcases} & \end{drcases}
\begin{drcases*} & \end{drcases*}
\begin{cases*} & \end{cases*}
```

```
\[
\begin{dcases}
E = m c^2 & c \approx 3.00 \times 10^8 \text{m/s} \\
\int x-3 \, dx & \text{Integral is display style}
\end{dcases}
\]
```

$$\begin{cases} E = mc^2 & c \approx 3.00 \times 10^8 \text{ m/s} \\ \int x - 3 \, dx & \text{Integral is display style} \end{cases}$$

带 \* 的环境有个细微的差别，就是第二列会默认用罗马体（直立的）显示，更加方便输入纯文字。

```
\[
a= \begin{dcases*}
E = m c^2 & c \approx 3.00 \times 10^8 \text{m/s} \\
\int x-3 \, dx & \text{Integral is display style}
\end{dcases*}
\]
```

$$a = \begin{cases} E = mc^2 & c \approx 3.00 \times 10^8 \text{ m/s} \\ \int x - 3 \, dx & \text{Integral is display style} \end{cases}$$

```
\[
\begin{rcases*}
x^2 & \text{for } x > 0 \\
x^3 & \text{else}
\end{rcases*} \quad \quad \quad \rightarrow \cdots
\]
```

$$\left. \begin{array}{ll} x^2 & \text{for } x > 0 \\ x^3 & \text{else} \end{array} \right\} \Rightarrow \cdots$$

### 7.6.2. 更好的矩阵环境

$\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{T}\mathcal{E}\mathcal{X}$  宏包提供的矩阵默认是每列都居中的，`mathtools` 宏包提供了一些带星号的矩阵环境，可以手动设置列对齐的方式。

```
\begin{matrix*}[position] \end{matrix*}%无括号
\begin{pmatrix*}[position] \end{pmatrix*}%圆括号
\begin{bmatrix*}[position] \end{bmatrix*}%方括号
\begin{Bmatrix*}[position] \end{Bmatrix*}%花括号
\begin{vmatrix*}[position] \end{vmatrix*}%单竖线，行列式形式
\begin{Vmatrix*}[position] \end{Vmatrix*}%双竖线
```

对比一下两者的排版效果，`mathtools` 可按需求设置对齐方式，排版效果要更好。

```
\[\begin{pmatrix*}[r]
-1 & 3 \\
2 & -4
\end{pmatrix*}\]
\[\begin{pmatrix*}
-1 & 3 \\
2 & -4
\end{pmatrix*}\]
```

$$\begin{pmatrix} -1 & 3 \\ 2 & -4 \end{pmatrix}$$

$$\begin{pmatrix} -1 & 3 \\ 2 & -4 \end{pmatrix}$$

`mathtools` 还提供小矩阵环境，做为行内公式时更加协调，例如： $\begin{bmatrix} a & -b \\ -c & d \end{bmatrix}$ 。其用法与行间矩阵完全一致，带 \* 的可调整对齐方式。

```
\begin{smallmatrix} \end{smallmatrix}
\begin{smallmatrix*}[position] \end{smallmatrix*}
\begin{psmallmatrix} \end{psmallmatrix}
\begin{psmallmatrix*}[position] \end{psmallmatrix*}
\begin{bsmallmatrix} \end{bsmallmatrix}
\begin{bsmallmatrix*}[position] \end{bsmallmatrix*}
\begin{Bsmallmatrix} \end{Bsmallmatrix}
\begin{Bsmallmatrix*}[position] \end{Bsmallmatrix*}
\begin{vsmallmatrix} \end{vsmallmatrix}
\begin{vsmallmatrix*}[position] \end{vsmallmatrix*}
\begin{Vsmallmatrix} \end{Vsmallmatrix}
\begin{Vsmallmatrix*}[position] \end{Vsmallmatrix*}
```

## 7.6.3. 长分式

有时候遇到的分式分子特别长，长到爆的那一种。这时候就需要分子能换行书写了。`math-tools` 宏包提供两个命令用来书写分子过长的分式。

```
\splitfrac
\splitdffrac
```

```
\[
a=\frac{
  \splitfrac{xy + xy + xy + xy + xy}
  {+ xy + xy + xy + xy}
}
{z}
=\frac{
  \splitdffrac{xy + xy + xy + xy + xy}
  {+ xy + xy + xy + xy}
}
{z}
\]
```

---


$$a = \frac{\begin{array}{c} xy + xy + xy + xy + xy \\ + xy + xy + xy + xy \end{array}}{z} = \frac{\begin{array}{c} xy + xy + xy + xy + xy \\ + xy + xy + xy + xy \end{array}}{z}$$

## 参考文献

- [1] 刘海洋. LATEX 入门[M]. [S.l.]: 电子工业出版社, 2013.
- [2] 胡伟. LATEX2e 完全学习手册 (第 2 版)[M]. [S.l.]: 清华大学出版社, 2013.