

zhlineskip 宏包

张瑞熹*

2019/05/15 v1.0e

目录

第 1 节 简介	1
1.1 宏包依赖	1
1.2 中西有别	2
第 2 节 功能介绍	3
2.1 载入宏包时的键值选项	4
2.2 载入宏包后的用户命令	5
2.2.1 调整数学公式的行距	5
2.2.2 调整西文文本的行距	5
2.3 使用范例	6
2.4 Microsoft Word 中的“单倍行距”	7

第 1 节 简介

zhlineskip 宏包允许用户指定正文行距相比于正文字号的倍数（通常建议设置在 1.5 至 1.67 之间），以及脚注行距相比于脚注字号的倍数。另一方面，由于数学公式主要是由西文字符构成的，zhlineskip 还能将数学公式的行距“恢复”成西文较为紧凑的行距（通常为西文字号的 1.2 倍），使得全文的视觉密度较为均匀。最后，本宏包还支持按照 Microsoft Word 进行“多倍行距”排版。

1.1 宏包依赖

本宏包是针对中日韩文的横排文档设计出来的，它依赖于下面这些宏包：

- kvoptions 宏包：为用户提供载入本宏包的键值选项。
- xintexpr 宏包：实现精确的浮点运算，属于 xint 宏集的一个部分。
- etoolbox 宏包：处理脚注行距与数学行距时需要打补丁。

*ruixizhang42@gmail.com。

- `mathtools` 宏包：只有在恢复数学行距为西文行距时，才会载入这个宏包。

请确保你的 $\text{T}_\text{E}_\text{X}$ 发行版里已经安装好了以上这些宏包的最新版本。

1.2 中西有别

在西文排版里，相邻两行**基线** (baseline) 之间的距离称为**行距** (leading, 发音为 led-ding)。这个词的词根是 lead，即**铅**。早在铅字时代，每当工匠填满一行铅字之后要开始填下一行，都会在两行之间插入铅条，从而适当地扩大行距。因为西文的每个字母四周与其**字框** (em-box, 见图 1) 之间有较大的空隙，所以不需要插入很高的铅条。一般来说，西文的行距为**字号** (font size) 的 1.2 至 1.45 倍¹。



图 1：西文字体。绿色方框即为 em-box，它在纸上的实际边长就是西文字号。

中文排版虽然没有基线的概念，但有非常相似的概念：**底线** (ideographic baseline, 见图 2)。中文里相邻两行底线之间的距离，与西文里行距的概念是一致的。另一概念是上一行底线和下一行**顶线**之间的距离，即**行间距** (line gap)，这与西文里插入铅条的高度是一致的。由于汉字四周与其字框间的空隙较小，所以需要使用时比西文更大的行间距。根据场合不同，行间距从字号的 1/4 至 1 倍不等：以中文书刊为例，行间距一般为字号的 1/2 至 2/3 倍²，即行距约为字号的 1.5 至 1.67 倍。

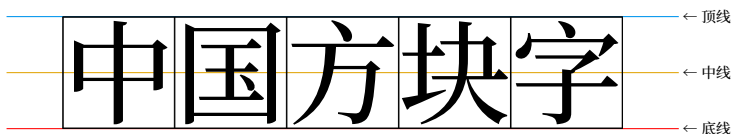


图 2：中文字体。汉字字面几乎占满整个字框，字框的边长即为中文字号。

在一般情况下， $\text{C}_\text{T}_\text{E}_\text{X}$ 会默认用 `linespread=1.3` 这个文档类选项将中文的行距设置为字号的 1.56 倍（基础行距是字号的 1.2 倍，而 $1.2 \times 1.3 = 1.56$ ）。通过这种方法扩大全文的行距，自然会影响到文章里数学公式的行距。而数学公式主要是由西文字符构成的，把它们按照中文的行距进行排版，就会显得有些松散。图 3 左边是 $\text{C}_\text{T}_\text{E}_\text{X}$ 默认排版效果，文本、数学看似一紧、一松；右边是配合用 `zhlineskip` 的效果，视觉密度比较均匀。`zhlineskip` 宏包还允许用户调整数学行距的大小。

综上所述，在进行中西文混排时，最好能够区分中文与西文的行距。在使用 `zhlineskip` 时，就可以分开处理中文文本与数学公式的行距。用户甚至还能分别指定正文行距与脚注行距，实现灵活的排版。同时，`zhlineskip` 宏包能恢复各种“多行”

¹参见 <https://practicaltypography.com/line-spacing.html>。

²参见张胜涛、王忆波著《方正飞腾 4.0 实用培训教程》，第 6.1.1 节。

<p>设 $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. 又设 $A = (a_{ij})_{m \times n}$ 为一个 m 行 n 列的实值矩阵, 即</p> $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix},$ <p>其中 $a_{ij} \in \mathbb{R}, i = 1, \dots, m, j = 1, \dots, n$. 又因为</p> $\sum_{\substack{i=1 \\ i \neq j}}^m a_{ij} = \begin{cases} 0, & j = 1, \\ 1, & j > 1, \end{cases}$ <p>我们得到……</p>	<p>设 $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. 又设 $A = (a_{ij})_{m \times n}$ 为一个 m 行 n 列的实值矩阵, 即</p> $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix},$ <p>其中 $a_{ij} \in \mathbb{R}, i = 1, \dots, m, j = 1, \dots, n$. 又因为</p> $\sum_{\substack{i=1 \\ i \neq j}}^m a_{ij} = \begin{cases} 0, & j = 1, \\ 1, & j > 1, \end{cases}$ <p>我们得到……</p>
--	--

图3：数学行距对比。在左图中，大矩阵 pmatrix 与分类 cases 两个环境受到影响，行距都被扩大了；但第一行文本里的小矩阵与末尾公式里求和号的下角标却没有受到影响，行距仍然较为紧凑。在右图中，数学公式的行距都是西文的行距，密度比较均匀，行间公式里的大括弧、大括号也不会特别突兀。

数学环境（包括矩阵、分类、多行公式推导等等）的行距，使数学公式的行距符合西文行距的规范。

最后，zhlineskip 宏包还支持用户在一定范围内按 Microsoft Word 的“多倍行距”进行排版³。用户可以指定“多倍行距”的“倍数”，但是这只保证用 T_EX 排出来的文本行距与用 Microsoft Word 排的行距相同。硬要用 T_EX 模仿 Microsoft Word 是没有太大意义的。

第2节 功能介绍

首先，请避免使用“多倍行距”这个概念：Microsoft Word 中“单倍行距”的值严重依赖于字体（参见第 2.4 节）。在严格排版的时候，一般都会给出具体的字号与行距，例如字号 12 磅、行距 22 磅。对于一般的用户，指定目标行距相比字号的倍数即可——zhlineskip 宏包可以自动提取基础行距（即 T_EX 中的单倍行距）相比字号的倍数（详见表 1），再通过用户指定的倍数来计算所需的行伸展因子。因此，不论是中日韩文还是西文的横排文档，都是可以使用本宏包的。本宏包的缺省设置更适合中日韩文文档。

³本宏包默认假定“被要求”用的字体是中易系列字体，这包括 Microsoft Word 里的“宋体”、“黑体”、“楷体”与“仿宋”。若改用其他字体，可能需要调整 MSWordSingleSpaceRatio 的值。参见第 2.1 节与第 2.4 节。

表 1：ctexart 与 article 各个文档类选项设置的基础行距倍数。

文档类选项	正文基础行距	脚注基础行距
zihao=5	1.2	1.2
zihao=-4	1.2	1.2
10pt	12/10	9.5/8
11pt	13.6/10.95	11/9
12pt	14.5/12	12/10

2.1 载入宏包时的键值选项

载入 zhlineskip 宏包时可以设定六个基本的键值选项，它们分别是：

bodytextleadingratio *<real>* 指定正文目标行距相比于正文字号的倍数。以书刊为例，建议设置在 1.5 至 1.67 之间。缺省值是 1.5，即 1/2 的行间距。

footnoteleadingratio *<real>* 指定脚注目标行距相比于脚注字号的倍数，它可以比正文的倍数稍小一些，建议设置在正文倍数的 98% 至 100% 之间。缺省值是 1.48，即为正文倍数的 98.67% 左右。

restoremathleading *<bool>* 指定是否要将数学公式的行距恢复成西文基础行距。缺省值是 true，即恢复数学行距。该选项为真时，会自动载入 mathtools 宏包，此时还能利用 `\SetMathEnvironmentSinglespace{<real>}` 命令微调数学公式的基础行距。

UseMSWordMultipleLineSpacing *<bool>* 在排版论文时，如果被要求按照 Microsoft Word 来设置“多倍行距”，那么用户可以将该选项设置为 true，并通过设置 `MSWordLineSpacingMultiple` 指定“倍数”，这会忽略用户之前指定的正文行距与脚注行距倍数，但是与数学行距的设置独立。该选项的缺省值是 false。

MSWordLineSpacingMultiple *<real>* 设置 Microsoft Word “多倍行距”的“倍数”，仅在 `UseMSWordMultipleLineSpacing` 为真时生效。缺省值是 1.15，在不修改 `MSWordSinglespaceRatio` 时，相当于设置了目标行距为字号的 1.49140625 倍，适用于中易字体（参见第 2.4 节）。

MSWordSinglespaceRatio *<real>* 设置 Microsoft Word 的“单倍行距”相比字号的倍数，仅在 `UseMSWordMultipleLineSpacing` 为真时生效。缺省值是 1.296875，适用于中易字体（参见第 2.4 节）。若改用其他字体，则需调整该选项的值。

2.2 载入宏包后的用户命令

2.2.1 调整数学公式的行距

当键值选项 `restoremathleading` 为 `true` 时，数学公式的行距被恢复成字号的 1.2 倍。对于某些字面较大的数学字体（例如类似 Palatino 的字体），这个基础行距会显得过小。此时，用户可以通过如下命令微调数学行距：

`\SetMathEnvironmentSinglespace{<real>}` 如果数学字体来自 `newpxmath` 或是 TeX Gyre Pagella Math，那么数学行距在字号 1.2 倍的基础上再扩大 1.05 倍更加合适。此时，只需指定 `\SetMathEnvironmentSinglespace{1.05}` 即可。

本宏包恢复的多行数学环境包括：

L^AT_EX 环境 `array`;

amsmath 宏包各环境 `matrix`, `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix`, `Vmatrix`, `cases`, `aligned`, `alignedat`, `gathered`, `gather`, `gather*`, `align`, `align*`, `flalign`, `flalign*`, `alignat`, `alignat*`, `xalignat`, `xalignat*`, `xxalignat`, `multline`, `multline*`, `split`;

mathtools 宏包各环境 `matrix*`, `pmatrix*`, `bmatrix*`, `Bmatrix*`, `vmatrix*`, `Vmatrix*`, `cases*`, `dcases`, `dcases*`, `rcases`, `rcases*`, `drcases`, `drcases*`, `multlined`, `lgathered`, `rgathered`。

超出上述列表范围、用户自定义的**数学环境**，可用如下命令恢复其行距：

`\RestoreMathEnvironmentLeading{<env name>}` 例如本宏包恢复 L^AT_EX 数学环境 `array` 的行距，是通过 `\RestoreMathEnvironmentLeading{array}` 实现的。`<env name>` 可以是由若干环境名构成的逗号列表。

注意，在 `restoremathleading` 为 `false` 时，`\SetMathEnvironmentSinglespace` 与 `\RestoreMathEnvironmentLeading` 无效。

2.2.2 调整西文文本的行距

与数学行距命令对应，本宏包还提供两个调整**西文文本**行距的命令，用法类似。

`\SetTextEnvironmentSinglespace{<real>}` 如果西文字体来自 `newpxtext` 或是 TeX Gyre Pagella，那么可以指定 `\SetTextEnvironmentSinglespace{1.05}`。

`\RestoreTextEnvironmentLeading{<env name>}` 使用范例：假设文中的表格仅含西文、数字，此时如果想要文本环境 `tabular` 的行距与西文行距一致，可通过 `\RestoreTextEnvironmentLeading{tabular}` 实现。亦可参见例 4。`<env name>` 可以是由若干环境名构成的逗号列表。

如果作者没有顾及到某些**基本环境**（数学或文本），鼓励用户向 `zhlineskip` 的 [GitHub 维护页](#) 提供相关信息。

2.3 使用范例

下面以CT_EX提供的ctexart文档类为例，展示zhlineskip的使用方法。

例1：直接载入

```
\documentclass{ctexart}
\usepackage{zhlineskip}
\begin{document}
正文测试。
\end{document}
```

例2：设置正文行距为字号的1.6倍

```
% 导言区
\usepackage[
  bodytextleadingratio=1.6, % 设置正文行距倍数为 1.6
  footnoteleadingratio=1.57 % 设置脚注行距倍数为 1.57
]{zhlineskip} % 缺省数学行距倍数为 1.2
```

例3：按照Microsoft Word设置“1.62倍行距”

```
% 导言区
\usepackage[
  restoremathleading=false, % 可选
  UseMSWordMultipleLineSpacing, % 需设置为真
  MSWordLineSpacingMultiple=1.62
]{zhlineskip}
```

例4：中文正文里需要插入成段的西文

如果需要插入成段的西文，最好的方法是先定义一个西文专用的环境。比如说可以在导言区定义如下这个环境：

```
\newenvironment{english} % 新建“英文”段落环境
{\addvspace\medskipamount} % 段前间距，上文段落需结束
{\par\addvspace\medskipamount} % 段后间距
```

注意要先用空行结束上文段落。这时，可以通过zhlineskip声明

```
\RestoreTextEnvironmentLeading{english} % 恢复“英文”使用紧凑行距
```

将“英文”段落的行距恢复成西文较为紧凑的行距。这样一来，正文（也许是中英交替的译文）就会有比较均匀的视觉密度，而不至于“中文很挤、英文很散”。

```
\RestoreTextEnvironmentLeading{english}
人人生而自由，在尊严和权利上一律平等。他们赋有
理性和良心，并应以兄弟关系的精神相对待。
```

```
\begin{english}
All human beings are born free and equal in
dignity and rights. They are endowed with
reason and conscience and should act towards
one another in a spirit of brotherhood.
\end{english}
```

人人有资格享有本宣言所载的一切权利和自由，不分种族、肤色、性别、语言……

```
\begin{english}
Everyone is entitled to all the rights and
freedoms set forth in this Declaration,
without distinction of any kind, such as
race, colour, sex, language\dots
\end{english}
```

人人生而自由，在尊严和权利上一律平等。他们赋有理性和良心，并应以兄弟关系的精神相对待。

All human beings are born free and equal in dignity and rights. They are endowed with reason and conscience and should act towards one another in a spirit of brotherhood.

人人有资格享有本宣言所载的一切权利和自由，不分种族、肤色、性别、语言……

Everyone is entitled to all the rights and freedoms set forth in this Declaration, without distinction of any kind, such as race, colour, sex, language...

2.4 Microsoft Word 中的“单倍行距”

Microsoft Word 中“单倍行距”的设置，其行距值相比字号的倍数严重依赖于字体，甚至还可能因为操作系统的不同而迥然不同。表 2 列出几种常见字体对应的单倍行距倍数。正是因为“单倍行距”本身随字体、操作系统而变化，所以请尽量避免使用“多倍行距”的概念！

表 2：在 Microsoft Word 中设置“单倍行距”后，实际的行距依赖于字体。

字体名称	“单倍行距”除以字号的倍数
Arial	$2355/2048 = 1.14990234375$
Times New Roman	$2355/2048 = 1.14990234375$
中易系列字体	$332/256 = 1.296875$
华文中宋 (Windows)	$1479/1000 = 1.479$
微软雅黑 Light (Windows)	$3400/2048 = 1.66015625$
微软雅黑 Regular/Bold	$3513/2048 = 1.71533203125$
华文中宋 (macOS)	$1723/1000 = 1.723$
微软雅黑 Light (macOS)	$3542/2048 = 1.7294921875$
苹方 (macOS)	$1820/1000 = 1.82$
思源宋体 1.001	$1869/1000 = 1.869$
思源黑体 2.000	$1882/1000 = 1.882$
思源黑体 1.004	$1924/1000 = 1.924$

参考文献

- [1] BUTTERICK, MATTHEW. *Butterick's Practical Typography: Line spacing*. <https://practicaltypography.com/line-spacing.html>, 访问日期: 2018/10/28.
- [2] KNUTH, DONALD ERVIN. *The T_EXbook*. Addison–Wesley, 1986.
- [3] LUNDE, KEN. *CJKV Information Processing: Chinese, Japanese, Korean & Vietnamese Computing* (2 ed.). O'Reilly Media, Inc., 2008.
- [4] 张胜涛 & 王忆波. **方正飞腾 4.0 实用培训教程**. 清华大学出版社, 2005.