The `parskip` package*

Frank Mittelbach `virhuiai@qq.com` 译

March 3, 2023

Abstract

The parskip package helps in implementing paragraph layouts where the paragraphs are separated by a vertical space instead of (or in addition to) indenting them.

parskip 宏包有助于实现段落布局，其中段落是通过垂直间距而不是（或除了）缩进来分隔的。

The package can be used with any document class at any size. By default it produces the following paragraph layout: Zero `\parindent` and non-zero `\parskip`. The stretchable glue in `\parskip` helps LaTeX in finding the best place for page breaks.

该宏包可与任何文档类一起使用，任何大小。默认情况下，它产生以下段落布局：零 `\parindent` 和非零 `\parskip`。在 `\parskip` 中的可伸缩粘合剂有助于 LaTeX 找到最佳的分页位置。

1 Introduction 介绍

Many LaTeX constructs are internally built by using the paragraph mechanism even if technically there aren't text paragraphs. In most such cases the LaTeX code handles indentation and suppressed it if necessary. But unfortunately this is normally not done for `\parskip` (as that is zero in the default layouts) and thus changing it will result in vertical spaces in unexpected places.

许多 LaTeX 构造在内部使用段落机制，即使从技术上讲没有文本段落。在大多数这样的情况下，LaTeX 代码处理缩进并在必要时抑制它。但是不幸的是，这通常不适用于 `\parskip`（因为在默认布局中它为零），因此更改它将导致在意想不到的位置出现垂直空间。

This package attempts to fix the spacing in table of contents structures, list environments, and around display headings that would get screwed up by a positive `\parskip` value.

*This is a reimplementation of a package originally written by Hubert Partl in 1989 and later maintained by Robin Fairbairns.
这是 Hubert Partl 于 1989 年编写的宏包的重新实现版本，后来由 Robin Fairbairns 维护。

该宏包尝试修复目录结构、列表环境和周围显示标题中可能会受到正值 \parskip 的影响的间距。

It is, however, is no more than quick fix; the 'proper' way to achieve effects as far-reaching as this is to create a new class.

但是，它只是一个快速修复，达到如此广泛的效果的“正确”方法是创建一个新类。

1.1 History
历史

This file was originally developed by Hubert Partl in 1989 (i.e., for LaTeX 2.09) to provide a somewhat crude solution to an existing problem in case no proper document class (back then called document style) support was available.

该文件最初是由 Hubert Partl 在 1989 年（即为 LaTeX,2.09）开发的，以提供一个有些粗糙的解决方案，以应对在没有适当的文档类（当时称为文档样式）支持的情况下存在的问题。

About ten years later Robin Fairbairns picked up the orphaned package and his version was then the one available for LaTeX 2_ε during the next 15^+ years.

大约十年后，Robin Fairbairns 接手了这个被遗弃的宏包，他的版本随后在接下来的 15^+ 年中可用于 LaTeX 2_ε。

Finally, while working on the next edition of the LaTeX Companion the current author did a reimplementation, that added support for TOC data and heading structures. Also a few additional key/value options were added to make the package more useful. It still is and will remain an inferior choice compared to a properly designed document class. But it offers a starting point if nothing is around.

最后，在编写下一版 LaTeX 伴侣时，当前作者重新实现了宏包，增加了对 TOC 数据和标题结构的支持。还添加了一些额外的键/值选项，使该宏包更加有用。但它仍然是和将来会是一个次优选择，与一个适当设计的文档类相比。但如果没有其他可用的选择，它提供了一个起点。

2 The user interface
用户界面

The parskip package doesn't offer any document user commands and just needs loading with \usepackage.

parskip 宏包没有提供任何文档用户命令，只需要用 \usepackage 命令载入即可。

2.1 Options to customize the package
定制宏包的选项

All of the package options are implemented as key/value options.

所有的宏包选项都是实现为键值选项的形式。

skip With the package option `skip` it is possible to explicitly specify the vertical space between paragraphs. If the option is not given (or given without a value) then `.5\baselineskip` plus `2pt` of stretch is assumed.
使用选项 `skip` 可以明确指定段落之间的垂直间距。如果未指定该选项（或未给出值），则默认为 `.5\baselineskip` 加上 `2pt` 的伸展量。

tocskip By default the `\parskip` is zero within `\tableofcontents` and similar lists, regardless of its value elsewhere. With the option `tocskip` it can be given a different value. If used without an explicit value you get the same `\parskip` as elsewhere within these lists.
默认情况下，在 `\tableofcontents` 和类似列表中，`\parskip` 为零，而不管其它地方的值如何。使用选项 `tocskip` 可以给它赋予不同的值。如果在未明确给出值的情况下使用，则在这些列表中的其它地方使用相同的 `\parskip`。

indent With the package option `indent` it is possible to explicitly set the paragraph indentation. Using this option without a value keeps the document class indentation unchanged, if it is specified with a value then that value is used. If the package is loaded without this option the indentation is set to zero.
使用选项 `indent` 可以明确设置段落缩进。如果不指定值使用该选项，则文档类的缩进不变，如果指定了值，则使用该值。如果在不带该选项的情况下载入宏包，则缩进为零。

parfill With package option `parfill`, the package also adjusts `\parfillskip` to impose a minimum space at the end of the last line of a paragraph. If specified without a value then `30pt` are assumed, if a value is given that forms the minimum.
使用选项 `parfill`，该宏包还会调整 `\parfillskip`，以强制在段落的最后一行添加最小间距。如果未指定值，则假定为 `30pt`，如果指定了值，则该值成为最小值。

3 Differences to the original package
与原始宏包的差异

If the package is used without any options or just with the option `parfill` it behaves like the earlier version, except that now the spacing around headings is also adjusted (not adding extra `\parskip`). If this is not desirable when processing an old document it can be avoided by explicitly requesting version `v1` as follows:

如果未使用任何选项或仅使用 parfill 选项，则该宏包的行为类似于早期版本，只是现在也会调整标题周围的间距（而不是添加额外的 \parskip）。如果在处理旧文档时不希望出现这种情况，则可以通过明确要求使用 v1 版本来避免，方法如下：

```
\usepackage{parskip}[=v1]
```

Of course, the new options, etc. are then also not available.

当然，此时也不可用新选项等。

4 Sources, bugs and issues 源、错误和问题

The official production version is available from CTAN. The latest (development) sources are maintained at GitHub at:

官方生产版本可从 CTAN 获取。最新的（开发）源代码维护在 GitHub 上：

```
https://github.com/FrankMittelbach/fmitex-parskip
```

In case of problems with the package you can report them at

如果使用该软件包出现问题，您可以在以下网址报告问题：

```
https://github.com/FrankMittelbach/fmitex-parskip/issues
```

Please provide a minimal test example that can be run and doesn't use packages not in a standard LaTeX distribution (and only those that are needed to show the issue).

请提供一个最小化的测试例子，可以运行并且不使用非标准 LaTeX 分发包（只使用需要展示问题的那些包）。

5 The Implementation 实现

everything is inside a module
一切都在一个模块中

```
1 ⟨*package⟩
```

5.1 The main implementation part 主要实现部分

```
2 \NeedsTeXFormat{LaTeX2e}[2018-04-01]
3
4 \DeclareRelease       {v1}{2001-04-09}{parskip-2001-04-09.sty}
5 \DeclareCurrentRelease{v2}{2018-08-24}
```

```
\ProvidesPackage{parskip}[2021-03-14 v2.0h non-zero parskip adjustments]
```

5.1.1 Option handling
选项处理

Here we define all option keys for use as package options:

这里定义了所有用作软件包选项的选项键：

```
\RequirePackage{kvoptions}
\SetupKeyvalOptions{family=parskip,prefix=parskip@}
```

The key `indent` defines the amount of indentation for each paragraph. If not given the indentation will be zero (default) and if given without a value then the outer value from the document class will get used, otherwise the given value is used.

键 `indent` 定义了每个段落的缩进量。如果没有给出，则缩进量为零（默认值）。如果没有给出值，则使用文档类的外部值，否则使用给定值。

```
\DeclareStringOption[0pt]{indent}[\parindent]
```

The key `parfill` defines a minimum amount of white space that should be left in the last line. By default the last line can get completely fill up. If given without a value the default (as before) is to require a minimum of `30pt`, otherwise the given value is used.
键 `parfill` 定义了应该在最后一行留下的最小白色空间量。默认情况下，最后一行可以完全填充。如果没有给出值，则默认值（与以前相同）为要求最小 `30pt`。否则，使用给定值。

```
\DeclareStringOption[0pt]{parfill}[30pt]
```

The key `skip` defines the vertical separation between paragraphs. If not given the default (as before) is to use half a `\baselineskip` plus a stretch of `2pt` to add some flexibility. If given, one need to provide an explicit value which is then used as a separation (and it needs to contain any extra stretch if that is wanted, i.e., there is no extra stretch added in this case).
键 `skip` 定义了段落之间的垂直间隔。如果没有给出，则默认值（与以前相同）为使用半个`\baselineskip` 加上 `2pt` 的拉伸，以增加一些灵活性。如果给定，需要提供一个明确的值，然后将其用作分离（如果需要任何额外的拉伸，则需要包含它，即在这种情况下不会添加额外的拉伸）。

```
\DeclareStringOption{skip}[]
```

The key `tocskip` defines the vertical separation inside the lists `\tableofcontents`, `\listoffigures` and `\listoftables`. By default there is no extra separatation (i.e., `0pt`). If specified without a value the standard `\parskip` is used, otherwise the given value.

关键字 tocskip 定义了在列表 \tableofcontents、\listoffigures 和 \listoftables 中的垂直间距。默认情况下没有额外的间隔（即 0pt）。如果未指定值，则使用标准的 \parskip，否则使用给定的值。

```
\DeclareStringOption[0pt]{tocskip}[\parskip]
```

Execute any package options:

执行任何包选项：

```
\ProcessKeyvalOptions*
```

So now we can evaluate the given options and adjust the parameter settings:

现在我们可以评估给定的选项并调整参数设置：

```
\ifx\parskip@skip\@empty
```

If no skip was given (or it was empty) set \parskip to half of .5\baselineskip plus 2pt stretch. Stretch or shrink inside \baselineskip is ignored in this case.

如果没有给出 skip（或为空），则将\parskip 设置为.5\baselineskip 的一半加上 2pt 的伸展。在此情况下，\baselineskip 内的伸展或收缩被忽略。

```
  \parskip=.5\baselineskip plus 2pt\relax
\else
```

Otherwise set it to the specified value:

否则将其设置为指定的值：

```
  \setlength\parskip\parskip@skip
\fi
```

Setting \parfillskip was suggested by Donald Arseneau at some point on comp.text.tex:

Donald Arseneau 在某些时候在 comp.text.tex 上建议设置\parfillskip：

```
\setlength\parfillskip\parskip@parfill
\advance\parfillskip 0pt plus 1fil\relax
```

\parindent gets whatever was specified. If the key was given without an option this will essentially reassign the now "current" value.

如果没有指定选项，则\parindent 将获得任何指定的值。这实际上将重新分配当前值。

```
\setlength\parindent\parskip@indent
```

5.2 Handling document elements
处理文档元素

Setting up a non-zero \parskip has some side-effects in document elements such as lists or headings etc. Here we try to keep these side-effects somewhat under control.

设置非零的\parskip 会对列表或标题等文档元素产生一些副作用。在这里，我们尽量控制这些副作用。

We make use of the etoolbox package to do patching.

我们使用 etoolbox 宏包进行修补。

```
\RequirePackage{etoolbox}
```

5.2.1 Lists
列表

To accompany this, the vertical spacing in the list environments is changed to use the same as \parskip in all relevant places (for \normalsize only), i.e.

为了配合这一点,列表环境中的垂直间距更改为在所有相关位置（仅针对\normalsize）使用与\parskip 相同的间距，即

```
\parsep = \parskip
\itemsep = \z@ % add nothing to \parskip between items
\topsep = \z@ % add nothing to \parskip before first item
```

However, if the user explicitly asked for a zero parskip (via the skip option) we shouldn't do this but rather keep the default list settings, so we better check for this.

然而，如果用户通过 skip 选项明确要求零段落间距，则我们不应这样做，而应保留默认的列表设置，因此我们最好检查一下。

```
\ifdim \parskip > 0pt

  \def\@listI{\leftmargin\leftmargini
     \topsep\z@ \parsep\parskip \itemsep\z@}
  \let\@listi\@listI
  \@listi

  \def\@listii{\leftmargin\leftmarginii
     \labelwidth\leftmarginii\advance\labelwidth-\labelsep
     \topsep\z@ \parsep\parskip \itemsep\z@}

  \def\@listiii{\leftmargin\leftmarginiii
      \labelwidth\leftmarginiii\advance\labelwidth-\labelsep
      \topsep\z@ \parsep\parskip \itemsep\z@}
%
% and finally ...
```

```
%   \partopsep = \z@ % don't even add anything before first item (beyond
%                    % \parskip) even if the list is preceded by a blank line
  \partopsep=\z@
\fi
```

5.2.2 TOCs and similar lists
目录和类似列表

Within a table of contents or a list of figures we don't want any additional vertical spacing just because the individual lines in such a list are implemented as one-line paragraphs. So we locally set the `\parskip` to zero by default. Should be really something that is done already in LaTeX.

在目录或图表清单中，我们不想因为这样一个列表中的每行都是一行段落而有任何额外的垂直间距。因此，我们默认将 `\parskip` 局部设置为零。这应该是在 LaTeX 中已经实现的东西。

```
\patchcmd\@starttoc
    {\begingroup \makeatletter}
    {\begingroup \makeatletter
```

Just setting `\parskip` to zero as it was done in the original version of the package, does not always work. If the list starts out with an ordinary paragraph (and not with `\addvspace` as it usually does) we will get a zero `\parskip` but the heading above assumes we get the normal `\parskip` and has therefore removed that amount from its own vertical skip. As long as the parskip value is not too large people didn't notice that heading and list moved closer to each other but if you use, say, `[skip=20pt]` you will even see an overlap.

但是，就像在包的原始版本中所做的一样，只是将 `\parskip` 设置为零并不总是有效的。如果列表以普通段落开头（而不是通常使用的 `\addvspace`），我们将获得零的 `\parskip`，但上面的标题假定我们获得正常的 `\parskip`，因此已经从自己的垂直跳过中删除了该量。只要段落距离不太大，人们就不会注意到标题和列表彼此移动，但是如果您使用，例如，`[skip=20pt]`，您甚至会看到重叠。

We therefore do the following: we look at the last skip, undo it and then issue a skip that is equal to `\parskip` + `\lastskip`. This way the skip seen by any following code has the right value which is important for `\addvspace` calulations. Only then we locally set `\parskip` to zero or rather to `\parskip@tocskip`, the parameter that the user can set through an option.

因此，我们执行以下操作：我们查看最后一个跳过，撤消它，然后发出一个等于 `\parskip` + `\lastskip` 的跳过。这样，任何随后的代码看到的跳过都具有正确的值，这对于 `\addvspace` 计算非常重要。然后，我们将 `\parskip` 局部设置为零或者更确切地说，设置为用户可以通过选项设置的参数 `\parskip@tocskip`。

```
      \skip@\lastskip
      \advance\skip@\parskip
      \vskip-\lastskip
      \vskip\skip@
```

```
47       \parskip\parskip@tocskip}
48     {}{\typeout{Couldn't patch \string\@starttoc}}
```

5.2.3 Standard headings

For the same reason we don't want to see an additional `\parskip` being added before and after a display heading, so we subtract its value (in two places):

同样的原因，我们不想在显示标题前后添加额外的 `\parskip`，因此我们减去了它的值（在两个地方）。

```
49 \patchcmd\@startsection
50     {\addvspace\@tempskipa}
51     {\advance\@tempskipa-\parskip\addvspace\@tempskipa}
52     {}{\typeout{Couldn't patch \string\@startsection}}

53 \patchcmd\@xsect
54     {\vskip\@tempskipa}
55     {\advance\@tempskipa-\parskip\vskip\@tempskipa}
56     {}{\typeout{Couldn't patch \string\@xsect}}
```

5.2.4 titlesec headings
titlesec 标题

If titlesec is used then headings are built using different commands and we have to cancel the `\parskip` there. The principle is the same. Of course, the patching should only happen if that package really got loaded, so we defer it to the start of the document and test for it:

如果使用 titlesec 包，则标题将使用不同的命令构建，因此我们必须在那里取消 `\parskip`。原则是相同的。当然，只有在确实加载了该软件包时，我们才应该进行修补，因此我们将其推迟到文档的开头并进行测试：

```
57 \AtBeginDocument{%
58 \ifx\ttl@straight@ii\@undefined\else  % titlesec got loaded
59 \patchcmd\ttl@straight@ii
60     {\addvspace{\@tempskipa}}%
61     {\advance\@tempskipa-\parskip \addvspace\@tempskipa}%
62     {}{\typeout{Couldn't patch \string\ttl@straight@ii}}%
63 \patchcmd\ttl@straight@ii
64     {\vspace{\@tempskipb}}%
65     {\advance\@tempskipb-\parskip \vspace\@tempskipb}%
66     {}{\typeout{Couldn't patch \string\ttl@straight@ii}}%
67 \patchcmd\ttl@part@ii
68     {\vspace*{\@tempskipa}}%
69     {\advance\@tempskipa-\parskip \vspace*\@tempskipa}%
70     {}{\typeout{Couldn't patch \string\ttl@part@ii}}%
71 \patchcmd\ttl@part@ii
72     {\vspace{\@tempskipb}}%
73     {\advance\@tempskipb-\parskip \vspace\@tempskipb}%
```

```
    {}{\typeout{Couldn't patch \string\ttl@part@ii}}%
\patchcmd\ttl@page@ii
    {\vspace*{\@tempskipa}}%
    {\advance\@tempskipa-\parskip \vspace*\@tempskipa}%
    {}{\typeout{Couldn't patch \string\ttl@page@ii}}%
\patchcmd\ttl@page@ii
    {\vspace{\@tempskipb}}%
    {\advance\@tempskipb-\parskip \vspace\@tempskipb}%
    {}{\typeout{Couldn't patch \string\ttl@page@ii}}%
\fi}
```

5.2.5 amsthm theorems
amsthm 定理

The amsthm package is one of the few packages that make an explicit correction for `\parskip` which isn't any longer adequate if this parskip package is loaded. We therefore remove that setting from the package if it was loaded.

amsthm 包是为数不多的对 `\parskip` 进行明确修正的软件包之一，如果加载了此 parskip 包，则不再适用。因此，如果已加载该软件包，则从该软件包中删除该设置。

```
\AtBeginDocument{%
\ifx\deferred@thm@head\@undefined\else  % amsthm got loaded
\patchcmd\deferred@thm@head
  {\addvspace{-\parskip}}{}%
  {}{\typeout{Couldn't patch \string\deferred@thm@head!}}%
\fi}
```

5.3 Closing shop

⟨∗package⟩