The cellspace package

Josselin Noirel

2017/08/02—v1.7

Abstract

This package is intended to allow automatic spacing out of the lines of an array. People often complain about text touching the `\hline` of a tabular when it is too high or too deep. For example (with the amsmath package) with `\dfrac{1}{2}` surrounded by `\hline`s. This package provides a modifier `S` to usual column types (`l`, `c`, `r`, `p`, `m`, and `b`) that ensures a minimal spacing between rules and cells of an array.

Introduction

The mechanism used by LaTeX to build tables—using struts—has an important consequence: the cells of a table that extend too much tend to touch horizontal rules. First example:

```
\begin{tabular}{cc}
  \hline
  \itshape Bond &
  \itshape Distance ($\mathrm{\mathring{A}})$ \\
  \hline
  C--C          & $1.53$ \\
  C--H          & $1.10$ \\
  \hline
\end{tabular}
```

Bond	*Distance (Å)*
C–C	1.53
C–H	1.10

Second example:

```
\begin{tabular}{c}
  \hline
  \itshape Formula \\
  \hline
  $\displaystyle
   e = 1 + \frac{1}{2} + \frac{1}{6}  + \cdots
                       + \frac{1}{k!} + \cdots $ \\
  $\displaystyle
   \pi = 4 \left( 1 - \frac{1}{3} + \frac{1}{5} + \cdots
                   + \frac{(-1)^k}{2 k + 1}
                   + \cdots \right)$ \\
  \hline
\end{tabular}
```

Formula
$e = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \frac{1}{k!} + \cdots$
$\pi = 4 \left(1 - \frac{1}{3} + \frac{1}{5} + \cdots + \frac{(-1)^k}{2k+1} + \cdots \right)$

The classical workarounds `\\[`⟨*dimen*⟩`]` and `\noalign{\vspace{`⟨*dimen*⟩`}}` are not very powerful and need fine tuning.

How to solve this?

The cellspace loads several packages to carry out its job: array, ifthen, and calc (along `amsmath` when the option `math` is passed to `cellspace`). It redefines several internals, still not too many (`\@startpbox` and `\@endpbox`). By default, the tables will behave as usual. To improve the spacing of your tables, you must change the table preamble and prepend `S` to the column types `l`, `c`, `r`. The same holds for the paragraph columns `p`, `m`, and `b`, except that they must be surrounded by an extra pair of braces. For instance, the default behaviour of a table beginning with

```
\begin{tabular}{l l l p{3cm}}
```

should be changed into

```
\begin{tabular}{Sl Sl Sl S{p{3cm}}}
```

The cellspace has two parameters governing the spacing of the cells. The dimension `\cellspacetoplimit` is the minimal spacing required between the actual text the cell is made up of and the top of the cell (where `\hline`s may appear), if the spacing is less than this threshold, a space of `\cellspacetoplimit` is added, otherwise nothing is done. Conversely, at the bottom the dimension `\cellspacebottomlimit` is the minimal space required between the bottom of the cell and the text itself. This parameters can be changed in the document preamble using `\setlength`. Together with booktabs, the results look rather good. cellspace may work with other column types (defined through `\newcolumntype` for instance); it will assume by default that an unknown column is a LR-mode column. Otherwise you have to tell cellspace something like

```
\addparagraphcolumntypes{X}
```

(The previous example is needed to make tabularx work with cellspace.)

The following examples use a setting of 2 pt for both top and bottom space limits.

Bond	*Distance* (Å)
C–C	1.53
C–H	1.10

Bond	*Distance* (Å)
C–C	1.53
C–H	1.10

Some space is added at the top the first row because it doesn't fulfil the condition imposed by the package. Similarly some space is added at the bottom of the first row. It has to be noted, however, that no space is added in other rows because they are far enough from the cell separations.

Formula
$e = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \frac{1}{k!} + \cdots$
$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} + \cdots + \frac{(-1)^k}{2k+1} + \cdots\right)$

Formula
$e = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \frac{1}{k!} + \cdots$
$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} + \cdots + \frac{(-1)^k}{2k+1} + \cdots\right)$

Here it can be immediately remarked that the 'Formula' row has the same height in both examples. Only the formula lines are affected. The best layout is obtained with booktabs:

Formula
$e = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \frac{1}{k!} + \cdots$
$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} + \cdots + \frac{(-1)^k}{2k+1} + \cdots\right)$

Formula
$e = 1 + \frac{1}{2} + \frac{1}{6} + \cdots + \frac{1}{k!} + \cdots$
$\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} + \cdots + \frac{(-1)^k}{2k+1} + \cdots\right)$

The following example shows that how it works with paragraph columns.

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

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

Matrices

The package can be loaded with the options `nomath` (default) and `math`. The latter does two things: it first loads the `amsmath` package and then redefines the command that is invoked when one typesets matrices (`\env@matrix` — this improvement was suggested by Bastien Roucaries). The following examples illustrate what happens with (left) or without (right) cellspace's correction in a `pmatrix` environment.

`nomath` option $\begin{pmatrix} +\frac{1}{2} & +\frac{1}{3} \\ -\frac{1}{3} & +\frac{1}{2} \end{pmatrix}$ `math` option $\begin{pmatrix} +\frac{1}{2} & +\frac{1}{3} \\ -\frac{1}{3} & +\frac{1}{2} \end{pmatrix}$

As a consequence, the `amsmath` package can be loaded beforehand with other packages (such as `empheq` or `mathtools`), were an incompatibility to arise from one's loading it later.

Using a different column name

An other column name than `S` can be used if `S` conflicts with the user's own definitions or a package's. A keyval option can be passed to the package to that effect;

```
\usepackage[column=O]{cellspace}
```

can be used to provide cellspace's functionality through the `O` column modifier instead of the `S` column modifier.

Bugs and limitations

This package hasn't been heavily tested, so there may be plenty of bugs. As usual, bugs will certainly arise in complicated situations because tables can become very complicated (for instance, this package hasn't been designed with nested tables into mind). In simple cases however, it should work nicely. The package loads array ensuring a peaceful collaboration with each other. Other packages of special interest haven't been tested: tabularx[1] and longtable in particular.

A	B B B B B B B B
A A	B B B B B B B B B B B B B B B B
A A A A	$\frac{1}{2}$B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B $\frac{1}{2}$

[1] A one minute test seems to indicate that it works provided tabularx is loaded after cellspace. The example on this page illustrates this. A thirty second test indicates that cellspace works correctly with longtable as long as only LR-mode columns are concerned.

tabls[2] is a package that does a similar job as cellspace. The differences are: tabls affects a whole table while cellspace affects only the columns on which applies the modifier `S`. Each cell affected by cellspace is affected in the same way: there must be enough space above (> `\cellspacetoplimit`) and below the text (> `\cellspacebottomlimit`), even if no `\hline` is present. In the other hand, tabls ensures a minimal distance using one threshold parameter `\tablinesep` applying between rows not separated by a rule and using a systematic spacer `\arraylinesep` when there is a rule (whereas cellspace can leave a cell as is if the distance to the rule is big enough). As a last point, tabls, as far as I can see, cannot work with array.

[2]Thanks to Jim Hefferon for pointing this out.