

# The spacingtricks package\*

Antoine Missier  
antoine.missier@ac-toulouse.fr

February 27, 2023

## 1 Introduction

This package offers some macros to deal with spacing issues. Thus:

- `\centered` yields good horizontal centering without vertical spacing;
- `\footnote` has been redefined to avoid unsuitable spacing;
- `\vstrut` produces a strut with variable height or depth;
- `\indent` has been redefined to indent a line at the beginning of a particular paragraph even if `\parindent` has been set to 0;
- the `indentblock` environment produces indentation of all its content;
- the `compactlist` environment yields a compact list, without vertical spacing between the items, like here; several `aliases` are provided to type some list symbols shorter: `\bul`, `\dash`, `\ddash`, `\aster`, `\hand`, `\checksymb`, `\arrowsymb`;
- common abbreviations *i.e.* and *e.g.* are typeset by the macros `\ie` and `\eg` with correct spacing;
- the `\dualboxes` command attends to place two boxes (figures, tables, text) side by side by adjusting the vertical positioning.

Three other “spacing” packages are loaded by spacingtricks: `setspace` [1] (natively in  $\text{\LaTeX 2}\mathcal{E}$ ), for setting line spacing in a piece of text, `xspace` [2], which adds an inter-word space unless the macro is followed by a punctuation character, and `centeredline` [3] which gives an interesting alternative to our `\centered` macro (see further). We also took over the fine `juxtapose` environment from the `mafr` [4] distribution, as another way to place boxes side by side.

Otherwise, we have develop the package `arraycols` [5], that allows a good management of spacing in tabular or array environments, and `mismath` [6] that provides several macros to improve spacing in mathematical formulas.

## 2 Usage

`\centered` The `\centered{<text>}` command yields a centered line without vertical spacing. It

---

\*This document corresponds to spacingtricks v1.5, dated 2023/02/27.

acts like `\centerline` except in lists or tables where its behavior is much better (see the following examples). Moreover, the line break before (but not after) the macro is automatic.

Here is a comparative example of the centering commands inside a list:

1. Here a centered line with `\centered`:  

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Here another centered line with `\\ \centerline`:  

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Here another centered line with `\par\centerline`:  

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
4. Here a centered line with the `center` environment:  

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

After `\centered{...}`, the end of line must be explicit, otherwise `\centered` will not work. You can use `\\`, which is equivalent to `\linebreak`, or a blank line to begin a new paragraph (equivalent to `\par`). In the former case, no vertical space is added after the centered line, in the latter case, the vertical space following the centered line is a bit larger. In the example above, the centered line comes to the end, without line breaking, because of the next `\item` command, and the vertical space is set by the `enumerate` environment.

In tables, `\centered` allows to center a particular line in a cell independently of the column alignment<sup>1</sup>, here with `\begin{tabular}{|l|r|}`:

left aligned column	right aligned column
the second cell fixes the width	centered cell
centered cell	the last cell fixes the width

`\centeredline` A limitation of our `\centered` macro is that it doesn't allow to use `\verb` commands inside its argument. For this purpose, you get the `\centeredline` command, from the package `centeredline` by Jean-François Burnol [3]. This small package is now loaded by `spacingtricks`. Another advantage of `\centeredline` is that it does not require an explicit end, the text can follow the command on the same line, and the line break will be automatic. Nevertheless, it doesn't work in tables (like in the example above), except with `p` column declaration.

`onehalfspace`  
`doublespace`  
`spacing` The `setspace` package [1], loaded by `spacingtricks`, provides environments for changing the spacing between lines. The present paragraph is nested in a `onehalfspace` environment as you can see. The `spacing` environment can be used to get other interline spacing, e.g. `\begin{spacing}{2.5}`.

`\footnote` The `\footnote` command doesn't have a good management of spacing issues, in par-

---

<sup>1</sup>In tables, we can also use the powerful `\makecell` command of the `makecell` package [7]; on the other hand, neither `\centerline` nor `\centering` commands do work for a single line in a cell.

ticular when the `hyperref` package has been loaded. In English tradition, there is no space before numbers (or symbols) of note calls, and likewise at the beginning of footnotes, text begins immediately after the note number<sup>2</sup>. To avoid undesirable spaces, we don't have to put some space before or after writing `\footnote{`, for instance:

this is a note\footnote{good spacing} which works fine,

but sometimes, it is convenient to place the `\footnote` command on a new line. To achieve this, `\footnote` has been redefined to completely eliminate unwanted spaces<sup>3</sup>.

This is a note  
`\footnote{`  
 The old command.}  
 with bad spacing.

This a note<sup>a</sup> with bad spacing.

---

<sup>a</sup>The old command.

This is a note  
`\footnote{`  
 The new command.}  
 with good spacing.

This a note<sup>a</sup> with good spacing.

---

<sup>a</sup>The new command.

`\footnotespace`  
`\footnoteindent`

Like the old one, the new `\footnote` command can take an optional argument to force the number of the note. Likewise, we have always the customization macros `\footnotesize`, `\footnotesep`, `\footnoterule`, but two new macros have been added to manage spacing: `\footnotespace` produces the space before the note call symbol and `\footnoteindent` produces the space at the beginning of the footnote text. For instance, with `\renewcommand{\footnotespace}{\,}` and `\renewcommand{\footnoteindent}{\enskip}`<sup>4</sup> we get:

This is a note<sup>a</sup> with particular space settings.

---

<sup>a</sup> en dash spacing at the beginning of the note.

`\vstrut`

`\vstrut[⟨depth⟩]{⟨height⟩}` produces a strut with variable height or depth, in order to increase the line's height (above the base line) or depth (below the base line, optional); this command can be used in a text line, a table, a list, a formula, etc. If the values of `⟨height⟩` and `⟨depth⟩` are inferior to the height and depth of the current line, the command has no effect. Here are some examples.

$$\left[\frac{\sqrt{0.5p}}{10}\right] = \frac{\sqrt{0.5p}}{10} = \frac{\sqrt{0.5p}}{10}$$

`\fbox{\vstrut{2ex}$\sigma(X)=1$}` gives  $\sigma(X)=1$  better than  $\sigma(X)=1$ .

The height adjustment is done by trial and error. We could also have used a vertical phantom box; for example in the previous square root, we get a good result with `\vphantom{\bar{t}}`, but it's not obvious to know what to put in the phantom box, moreover, `\vstrut` allows a finer tuning.

---

<sup>2</sup>The typesetting of footnotes and note calls depends on national typographic rules which are, in principle, managed by `babel`. For instance, by activating the `french` option of `babel`, a thin space is added before the note calls, and the new `\footnote` macro does not alter this behavior.

<sup>3</sup>Probably it would have been enough to recommend the use of the `%` symbol at the end of line; its effect is to cancel the space produced by a line break, but we do not always think of using it.

<sup>4</sup>`\enskip` is equivalent to `\hspace{0.5em}`.

In a table, `\renewcommand{\arraystretch}{\stretch}` allows to increase the height of the rows but this command has a global effect, whereas `\vstrut` allows to adjust properly the height of each row, as in the following table:

bad	good	
$\lim_{\substack{x \rightarrow 1 \\ x > 1}} \ln \left( \frac{x^2}{x-1} \right)$	$\lim_{\substack{x \rightarrow 1 \\ x > 1}} \ln \left( \frac{x^2}{x-1} \right)$	obtained with <code>\vstrut{3.8ex}</code>
$\frac{a}{\frac{b}{b}}$	$\frac{a}{b}$	<code>\vstrut[2ex]{3ex}</code>
$\int_1^x \frac{1}{t} dt$	$\int_1^x \frac{1}{t} dt$	<code>\vstrut[2.5ex]{4.2ex}</code>

However, for tables, we have the `arraycols` package [5], based on `cellspace` [8], which allows to adjust row heights automatically. Nevertheless, `\vstrut` can be useful for fine adjustments.

In a text line, `\vstrut` can be used in place of `\vspace`.

`\indent` The command `\setlength{\parindent}{0cm}` allow to eliminate any indentation of lines at the beginning of every paragraph. But in this case, the `\indent` command does not work anymore if we want exceptional indentation of a particular paragraph. So, the initial length of `\parindent` has been saved in `\parindentlength` and the command `\indent` has been redefined to still allow indentation of length `\parindentlength`.

`indentblock` The `indentblock` environment allows indentation of a whole block of lines. It has an optional argument which is the length of indentation (set by default to `\parindentlength`). The following lyrics have been indented (and typeset in italic shape) with `\begin{indentblock}\itshape` and stanzas 2 and 4 have been affected by an additional indentation with `\begin{indentblock}[3em]`.

*Overhead the albatross hangs motionless upon the air  
And deep beneath the rolling waves in labyrinths of coral caves  
The echo of a distant time comes willowing across the sand  
And everything is green and submarine*

*And no one showed us to the land  
And no one knows the where's or why's  
But something stirs and something tries  
Starts to climb towards the light*

*Strangers passing in the street  
By chance two separate glances meet  
And I am you and what I see is me  
And do I take you by the hand  
And lead you through the land  
And help me understand the best I can?*

*And no one calls us to move on  
And no one forces down our eyes  
No one speaks and no one tries  
No one flies around the sun*

`compactlist` As its name tells it, the `compactlist` environment allows to create a “compact” list, i.e. without vertical space neither above nor between items. As for lists in  $\text{\LaTeX}$ , items are generated by the `\item` command. The environment has an optional argument: `\begin{compactlist}[\langle symbol \rangle]`.

`\bul` The default item symbol is `\textbullet`, but can be changed. We provide  
`\dash` aliases for several symbols commonly used in lists: `\bul` • (alias for `\textbullet`),  
`\ddash` `\dash` – (alias for `\textendash`), `\ddash` — (alias for `\textemdash`), `\aster` \*  
`\aster` (alias for `\textasteriskcentered`).

`[pifont]` Some other common symbols can be found in the `pifont` package [9]. Therefore  
`\hand` `spacingtricks` has a `pifont` option, that loads this package and creates the appropriate  
`\check symb` aliases: `\hand` ☞ for `\ding{43}`, `\check symb` ✓ for `\ding{51}` and `\arrowsymb` ➤  
`\arrowsymb` for `\ding{226}`.

The following example is produced by `\begin{compactlist}[\check symb]`:

- ✓ First item.
- ✓ Second item.
- ✓ Third item.

☞ These aliases can also be used directly in text mode, of course. For `\hand`,  
`\check symb` and `\arrowsymb`, the symbol is followed by a space if there is no punctu-  
 ation character just after it (thanks to the macro `xspace` from the `xspace` package [2]).

`\compactlistindent` Indentation length is set by `\compactlistindent` (fixed at 0.5em by default) and  
 can be modified with `\setlength`. Notice that there are several other ways to construct  
 a compact list in particular with the `noitemsep` key of the `enumitem` package [10].

`dinglist` The `pifont` package has also a list environment `\begin{dinglist}\{\langle number \rangle\}` to  
 typeset a list with normal spacing, using the `\ding{\langle number \rangle}` symbol for each `\item`  
 (instead of using `\item[\ding{\langle number \rangle}]`). So `\begin{dinglist}\{226\}` yields:

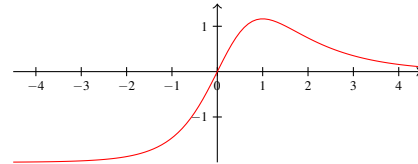
- ✓ The first item in the list.
- ✓ The second item in the list.
- ✓ The third item in the list.

`\ie` In English, at the end of a sentence, the point is followed by an em space which  
`\eg` is larger than an inter-word space. We provide the `\ie` (*id est*) and `\eg` (*exempli gra-*  
*tia*) macros, suggested in The  $\text{\LaTeX}$  Companion [17], to get correct spacing after these  
 abbreviations e.g. here. In American typography, a comma is often placed after these ab-  
 breviations, what we can get with `\ie`, on the other hand, some authors prefer to typeset  
*i.e.* in italic shape, which is always possible with `\textit{\ie}`.

`\dualboxes` Several packages intend to typeset text around a figure or a table, but some of them  
 need to give the width of the box containing the figure or the table. We provide here  
 another approach with the macro `\dualboxes[\langle pos \rangle]\{\langle left \rangle\}\{\langle right \rangle\}`, which places  
 two boxes, `\langle left \rangle` and `\langle right \rangle`, side by side. These boxes can contain figures, tables, one  
 small line text, parbox text, minipage environments (for several paragraphs and lists),  
 etc. The optional `\langle pos \rangle` parameter sets the vertical level on which the boxes are aligned:  
 a number between 0 (bottom) and 1 (top, default value).

Here is a first example in which we have adjusted the position of the boxes vertically, with `\dualboxes[0.6]`.

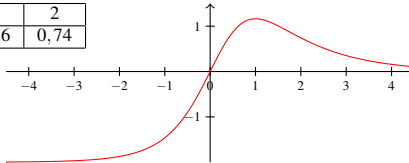
$x$	-2	-1	0	1	2
$f(x)$	-1,87	-1,46	0	1,16	0,74



The horizontal space is equally shared between left margin, inter-box space and right margin.

In the following example boxes are aligned by top and the right box has been shifted back (left) from 3 cm to make an overlapping with the left one, what `picins` cannot do.

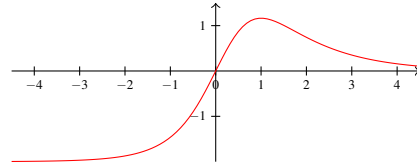
$x$	-2	-1	0	1	2
$f(x)$	-1,87	-1,46	0	1,16	0,74



```
\dualboxes{\scriptsize
  $\begin{array}{|*{6}{c}|} \hline
    x      & -2      & -1      & 0 & 1      & 2      \\ \hline
    f(x)    & -1,87   & -1,46   & 0  & 1,16   & 0,74   \\ \hline
  \end{array}$
}{\mbox{} \hspace{-3cm}
  \begin{tikzpicture}[scale=0.6]\tiny
    \draw[->] (-4.5,0) -- (4.5,0);
    \draw[->] (0,-2) -- (0,1.5);
    \foreach \x in {-4,...,4} {
      \draw (\x,0.1cm) -- (\x,-0.1cm) node[below] {$\x$};
    }
    \foreach \y in {-1,1} {
      \draw (0.1cm,\y) -- (-0.1cm,\y) node[left] {$\y$};
    }
    \draw[domain=-4.5:4.5,samples=100,color=red]
      plot ({\x},{2*\x/(exp(\x)-\x)});
  \end{tikzpicture}
}
```

`\dualboxes*` This command has a starred version that eliminates space before the first box and after the second one, but not between them: `\dualboxes*[\langle pos \rangle]{\langle left \rangle}{\langle right \rangle}`.

In practice, `\dualboxes` is suitable for boxes in which you put figures or tables or small one line text. For longer text, this one must be nested in a paragraph box, e.g. here we used `\parbox{6.5cm}{...}`. If you want several paragraphs, a list or a mathematical displayed formula, then you have to put them in a `minipage` environment.



Let us indicate that it is unfortunately not possible to use `verbatim` environments (nor the in-line `\verb` command) within `\dualboxes` arguments, as well as it is not possible for footnotes or margin notes.

For putting two figures side by side, or a table and a figure, `\dualboxes` is perfect. But for text around a figure, some other package may be more efficient. Let us mention first `wrapfig` [11], one of the easiest, but text must contain in a single paragraph (like above), without list or you have to place it in a `minipage` environment.

More sophisticated is the `picins` package [12], cited in The  $\LaTeX$  Companion [17]. Unlike most others, it can be used with lists, nevertheless vertical positioning can be tricky and it doesn't belong to common distributions  $\TeX$  Live or  $\text{MiK}\TeX$ . A last one we mention is `picinpar` [13] in which text can begin after the first line and can flow to the two side of the figure.

`juxtapose` As a very interesting alternative, we present the powerful `juxtapose` environment, written by Christian Obrecht in the `fiche.cls` class file, which is part of the `mafr` distribution [4]. We made small improvements, among others to work correctly with lists and get customization of some lengths: top and bottom lengths are set by default to `\smallskipamount` and separation space is set to 1.5em. For spacing above or below the environment, if the default length doesn't fit well, you can add a `\bigskip` or a `\vspace{<dimen>}` for example, or at worst redefine the length `\juxtopskip` and `\juxbottomskip`.

In this example the environment begins here. `juxtapose` can deal with figures, tables, but also:

- verbatim text: `\mytestmacro`,
- text with several paragraphs,
- lists, like here (a compact list),
- displayed mathematics as shown on the right side,

without the need of `multipage` environments.

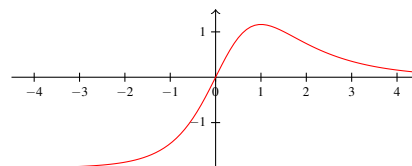
The two boxes are aligned by their center, but unlike `\dualboxes`, you cannot adjust vertical positioning.

Here begins the following text. The syntax of `juxtapose` is simple:

```
\begin{juxtapose} ... \otherside ... \end{juxtapose}
```

It has an optional argument `\begin{juxtapose}[<lefratio>]` where `<lefratio>` is the width ratio of the left side relatively to `\textwidth` (the default value is 0.5 which means  $0.5\text{\textwidth}$ ).

Paragraphs are indented inside each side, except the first one, because for a figure, there's no need to shift it to the right (from indent length) and otherwise the user may forgot to put `\noindent` before the figure. If you still want to indent the first paragraph use `\indent`. For horizontal alignment inside one side, you can use `\centering` or `\flushright` (useful for a figure). If you want a global indentation of the first (or the second) box, you can put it in an `indentblock` environment, it works fine!



$$f(x) = \frac{2x}{e^x - x}$$

This is the last line of the environment.

One may ask why not using multi column environments instead. In the classic multi-col environment you cannot choose unbalanced columns and you cannot choose where to pass to the other column.

Some nice packages improve multicols, e.g. `vwcol` [14] in which you can fix on the columns width, but unfortunately it doesn't accept verbatim text, and doesn't manage correctly lists and figures like above.

Let us mention `paracol` [15] and `reledpar` [16], which are intended to typeset two column of text in parallel, useful for translation or critical edition of text. They work fine with lists, figures, verbatim text, and also footnotes and margin notes, but it is not immediate to obtain just a chosen vertical positioning of two boxes. They have synchronization mechanisms, convenient for text, but not when you place two figures side by side. Besides they are a bit oversized for our need. They don't have the same purpose. So we can hope `juxtapose` or `\dualboxes` are useful.

### 3 Implementation

```

1 \newif\ifspacingtricks@pifont
2 \DeclareOption{pifont}{\spacingtricks@pifonttrue}
3 \ProcessOptions \relax
4
5 \RequirePackage{ifthen}
6 \RequirePackage{calc}
7 \RequirePackage{setspace}
8 \RequirePackage{xspace}
9 \RequirePackage{centeredline}
10 \ifspacingtricks@pifont \RequirePackage{pifont} \fi
11
12 \newcommand*{\centered}[1]{\setlength{\parskip}{0pt}\par\noindent\hfill
13     #1\hfill\mbox{}}

```

The double braces are necessary here to ensure that the `\parskip` modification applies locally within the command and not globally to the rest of the document.

```

14 \newcommand{\footnotespace}{}
15 \newcommand{\footnoteindent}{}
16 \let\footnt\footnote
17 \renewcommand{\footnote}[2][\unskip\footnotespace%
18     \ifthenelse{\equal{#1}{}}{
19         \unskip\footnt{\footnoteindent\ignorespaces #2}
20     }{
21         \unskip\footnt[#1]{\footnoteindent\ignorespaces #2}
22     }\unskip
23 }

```

`\unskip` eliminates undesirable spaces before and `\ignorespaces` after.

```

24 \newlength{\strutheight}
25 \newcommand*{\vstrut}[2][0pt]{%
26     \setlength{\strutheight}{#2}%
27     \addtolength{\strutheight}{#1}%

```



```

28 \unskip
29 \ensuremath{\rule[-#1]{0pt}{\strutheight}}%
30 \ignorespaces%
31 }
32
33 \newlength{\parindentlength}
34 \setlength{\parindentlength}{\parindent}
35 \renewcommand{\indent}{\hspace{\parindentlength}}
36
37 \newenvironment*{indentblock}[1][\parindentlength]{
38   \begin{list}{}{
39     \setlength{\leftmargin}{#1}
40     \setlength{\itemsep}{0pt}
41     \setlength{\topsep}{0pt} % previously 1ex
42     \setlength{\partopsep}{0pt}
43   }
44   \item[]
45 }{\end{list}}
46
47 \newlength{\compactlistindent}
48 \setlength{\compactlistindent}{0.5em}
49 \newenvironment*{compactlist}[1][\textbullet]{
50   \par % sometimes necessary
51   \begin{list}{#1\unskip}{% \unskip gobbles space created by \xspace
52     \setlength{\itemsep}{0pt}
53     \setlength{\parsep}{0pt}
54     \setlength{\topsep}{0ex}
55     \setlength{\partopsep}{0pt}
56     \setlength{\labelwidth}{1em}
57     \setlength{\leftmargin}{\labelwidth}
58     \addtolength{\leftmargin}{\labelsep}
59     \addtolength{\leftmargin}{\compactlistindent}
60   }
61 }{\end{list}}
62
63 \providecommand{\bul}{\textbullet}
64 \providecommand{\dash}{\textendash}
65 \providecommand{\ddash}{\textemdash}
66 \providecommand{\aster}{\textasteriskcentered}

```

The command `\asterisk` already exists in the `mathabx` package.  
The following macros need the `pifont` package.

```

67 \ifspacingtricks@pifont
68   \providecommand{\hand}{\ding{43}\xspace}
69   \providecommand{\checksymb}{\ding{51}\xspace}
70   \providecommand{\arrowsymb}{\ding{226}\xspace}
71 \fi
72
73 \providecommand{\ie}{i. e. \@ \xspace}
74 \providecommand{\eg}{e. g. \@ \xspace}

```

```

75
76 \newcommand{\@@dualboxes}[3][1]{
77   \par\noindent
78   \raisebox{\depth-#1\totalheight}{#2} \hfill % needs calc
79   \raisebox{\depth-#1\totalheight}{#3} \smallskip
80 }
81 \newcommand{\@dualboxes}[3][1]{
82   \par\noindent \hfill
83   \raisebox{\depth-#1\totalheight}{#2} \hfill
84   \raisebox{\depth-#1\totalheight}{#3} \hfill\mbox{}\smallskip
85 }
86 \newcommand{\dualboxes}{\@ifstar{\@@dualboxes}{\@dualboxes}}
87

```

The following macro has been found in the `fiche.cls` file, distributed with the `mafr` package by Christian Obrecht [4]. It had some tiny bugs in spacing. I just made small changes: adding some lengths parameters and the `\linewidth` instruction (without whom, lists in the first side send the second side out of the page).

```

88 \newlength\juxtopskip
89 \setlength\juxtopskip{\smallskipamount}
90 \newlength\juxbottomskip
91 \setlength\juxbottomskip{\smallskipamount}
92 \newlength\juxsepspace
93 \setlength\juxsepspace{1.5em}
94 \newlength\side@width
95
96 \newenvironment{juxtapose}[1][0.5]{% left side ratio
97   \def\otherside{\egroup % internal function remembers \side@width
98     \@tempdima=\side@width
99     \side@width=\textwidth
100     \advance\side@width by -\@tempdima
101     \advance\side@width by -\juxsepspace
102     \hskip\juxsepspace
103     \linewidth=\side@width % for lists environments
104     \vcenter\bgroup\hsize=\side@width\noindent % \ignorespaces useless
105   }
106   \side@width=#1\textwidth
107   \advance\side@width by -0.5\juxsepspace
108   \linewidth=\side@width % for lists environments
109   \par\vskip\juxtopskip%
110   \hbox\bgroup\begin{math}
111   \vcenter\bgroup\hsize=\side@width\noindent\ignorespaces
112   \}{\egroup\end{math}\egroup\vskip\juxbottomskip}
113

```

## References

- [1] *The setspace Package*, Robin Fairbairns, Geoffrey Tobin, CTAN, v9.7b 2022/12/04.

- [2] *The xspace package*, David Carlisle, Morten Høgholm, CTAN v1.13 2014/10/28.
- [3] *centeredline – A macro for centering lines*, Jean-François Burnol, CTAN, v1.2 2022/10/10.
- [4] *La distribution mafr*, Christian Obrecht, CTAN, v1.0 17/09/2006.
- [5] *The arraycols package*, Antoine Missier, CTAN, v1.2 2021/09/20.
- [6] *mismath – Miscellaneous mathematical macros*, Antoine Missier, CTAN, v2.5 2023/02/24.
- [7] *The makecell package*, Olga Lapko, CTAN, v0.1e 2009/08/03.
- [8] *The cellspace package*, Josselin Noirel, CTAN, v1.9.0 2022/01/04.
- [9] *pifont – Using common PostScript fonts with L<sup>A</sup>T<sub>E</sub>X*, Walter Schmidt, CTAN, v9.3 2020/03/25.
- [10] *Customizing lists with the enumitem package*, Javier Bezos, CTAN, v3.9 2019/06/20.
- [11] *The wrapfig package*, Donald Arseneau, CTAN, v3.6 2003/01/31.
- [12] *Bilder in L<sup>A</sup>T<sub>E</sub>X-Dokumenten – PicIns-Benutzerhandbuch*, Joachim Bleser, Edmund Lang, CTAN, v3.0 sept. 1992.
- [13] *The picinpar package*, Friedhelm Sowa, Stefan Blochwitz, CTAN, v1.3 2022/11/25.
- [14] *The vwcol package*, Will Robertson, CTAN, v0.2 2015/02/10.
- [15] *Package paracol: Yet Another Multi-Column Package to Typeset Columns in Parallel*, Hiroshi Nakashima, CTAN, v1.35 2018/12/31.
- [16] *Parallel typesetting for critical editions: the reledpar package*, Maïeul Rouquette, Peter Wilson, Herries Press, CTAN, 2.25.3 2022/04/22.
- [17] *The L<sup>A</sup>T<sub>E</sub>X Companion*. Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, 2nd edition, Pearson Education, 2004.