Monografías Matemáticas *García de Galdeano* **vv**, 1–15 (yyyy)

PROCEEDINGS OF THE XV CONFERENCE ZARAGOZA-PAU ON MATHEMATICS AND ITS APPLICATIONS: INSTRUCTIONS FOR AUTHORS

Juan José Torrens

Abstract. In September 2018, the *Fifteenth International Conference Zaragoza-Pau on Mathematics and its Applications* took place in Jaca (Huesca, Spain). This document contains instructions for helping authors to prepare their contributions to the proceedings.

Keywords: Jaca, Applied Mathematics, Statistics, proceedings, LaTeX.

AMS classification: 00–01, 00A69, 60–06, 62–06.

§1. Introduction

Since 2003, LaTeX has been the typesetting system officially chosen to compose and print the proceedings of each *International Conference Zaragoza-Pau on Mathematics and its Applications* (formerly on Applied Mathematics and Statistics). Thus, only papers written in LaTeX can be submitted for publication.

In order to guarantee a uniform layout throughout the proceedings, you are kindly requested to use the `jaca` document class, defined in the file `jaca.cls`, which is a customized version of the `article` class. The features of `jaca` will be discussed in the next sections. Together with `jaca.cls`, you get in the same bundle the following files:

- `guide.pdf`: the document you are reading now;
- `acmurl.bst`: the bibliographic style required to write contributions to the proceedings;
- `model.tex`: a template `tex` file which exemplifies the usage of `jaca`; it can be a good starting point for writing your own paper;
- `bibmodel.bib`: the companion `bib` file of `model.tex`.

§2. Dos and don'ts

While preparing your LaTeX source file, take in mind the following remarks:

- The length of every article is limited to ten pages for communications and fourteen pages for conferences. Articles must be written in English.

- The proceedings will be quite probably typeset in a machine running an operating system different from yours. Keep your files as platform independent as possible. This means, for example, that only ASCII characters should appear in your `tex` and `bib` files (no accented characters, please!). Do not use the inputenx package or a similar one.
- Avoid any other source of possible conflicts, as special packages or fonts. As a rule of thumb, use only the packages mentioned in this guide, most of them already loaded by the document class.
- Your file will be surely edited during the submission or the publication processes. Improve its readability:
  - Include brief comments to quickly identify parts of the source code.
  - Hard wrap lines in the final version of your paper, so that each line in the source file has about 80 characters, at most.
  - Format the code and indent lines in environments, as done, for example, in the following code:

```
Let us consider the function $f:\Rset^2\to\Rset$ given by
\begin{equation*}
   f(x)=
   \begin{cases}
      \dfrac{x^2-y^3}{x^2+y^2}, & (x,y)\ne(0,0), \\
      0, & (x,y)=(0,0).
   \end{cases}
\end{equation*}
```

  See also the example given in Appendix A.
- If really required, define additional commands or environments with `\newcommand` and `\newenvironment`. Never use `\def` or similar Plain TEX commands. Likewise, do not introduce more definitions than strictly necessary. Avoid mere abbreviations to save keystrokes, such as `\be` for `\begin{equation}`. Keep all your private definitions in the preamble of the document, which should be as simple as possible. And, of course, do not define any macro whose functionality is already provided by a command in the `jaca` document class.
- Floats should be placed at the top of text pages or on separate pages. In other words, the optional argument of the `figure` and `table` environments should be `tp`.
- Use the commands provided by LATEX for cross-referencing: `\label`, `\ref`, `\eqref` (for referencing equations) and `\cite` (for bibliographic cites).
- Be sure that your file can be typeset without errors and warnings. They always indicate that something in your file does not work properly. For example, an `underfull \hbox` warning may be caused by a bad use of the `\newline` (or `\\`) command. An `overfull \hbox` warning usually means that some text expands on the right margin of the page, yielding it an unpleasant appearance. Pay attention to these details. If you get a warning that you are unable to correct, ask for advice.
- Let LATEX alone to format paragraphs. So do not remove or add paragraph indentation with

`\noindent` or `\indent`, do not put the `\\` command at the end of a paragraph[1], do not add extra space with `\smallskip` or a similar command. The use of the mentioned commands should be exceptional in ordinary text.

- Clean your file. Delete unused macro definitions, superfluous comments or the loading of unnecessary packages.
- Read carefully this guide. It shows how to use the `jaca` document class, but it also provides helpful examples and tips about mathematical typesetting (take a close look to the appendices!). You should also become familiar with the amsmath package, essential for writing mathematics with LaTeX. See its user's guide [1].
- Before submitting your paper, check carefully that you have closely followed the guidelines given in this document. If so, you will save the editors time and efforts. They will be very grateful.

§3. The `jaca` document class

3.1. Description and options

Essentially, the `jaca` class is the `article` class with options `10pt` and `twoside`. Valid options for `jaca` are:

`draft` Lines which extend on the right margin are marked with black boxes. Likewise, the graphic files are not actually included, just the filename is written in a box of the correct size, speeding up the compilation process.

`final` The opposite of `draft` (no lines marked, graphics included). It is the default option.

`a4paper` The paper size is set to 210×297 mm. The default size is 170×240 mm.

`Adobe | CM` By default, text is written with the TX fonts collection[2], which is essentially a group of Times, Helvetica and monospaced fonts, together with a complete set of math fonts. If you experience any problem with them, the `Adobe` and `CM` options allow to switch to the corresponding Adobe fonts or to the Computer Modern ones.

`IXConf | XConf | XIConf | Last` These options activate specific settings for the proceedings of the corresponding Jaca conference. The default option is `Last`.

`evennpages` This option produces a document with an even number of pages. To be used only by the proceeding editors.

3.2. Title, authors, abstract, keywords, AMS codes, acknowledgements and addresses

The `\title` and `\author` commands have been redefined. They are processed as soon as LaTeX reads them, so there is no need of the `\maketitle` command. Place them in the body

[1] The `\\` command is intended to insert a line break *inside* a paragraph. If it is placed at the end of a paragraph, LaTeX usually issues an `underfull \hbox` warning. There are better and "legal" methods to add space between paragraphs if really needed.

[2] The `jaca` class loads the txfonts package with the `varg` option.

of the document (i.e. after `\begin{document}`). Both commands have now a mandatory and an optional arguments. By default, the mandatory argument appears in the text and in the running heads. The optional argument, if present, replaces the mandatory one in the headings. Use it to shorten long titles or authors' lists. For example, in this document, these commands are used as follows:

```
\title[Instructions for authors]{Proceedings of the \\
    XIV Conference Zaragoza-Pau\\ on Mathematics \\
    and its Applications: \\ Instructions for authors}
\author{Juan Jos\'e Torrens}
```

Let us observe that, by means of the `\\` command, you can control how to break the title or the authors' list into several lines. Please, do not write any footnotes in the arguments of `\title` and `\author`. Acknowledgements and authors' affiliations should be written at the end of your paper. For the former, just type `\ack` and then the corresponding text. For the latter, see below the `address` environment.

The definition of the `abstract` environment has been also modified. However, you can use it as in standard LaTeX.

The keywords and the AMS classification codes are specified, respectively, in the first and the second argument of the `\KeysAndCodes` command. Separate by commas two consecutive keywords and codes. For example, if you write

```
\KeysAndCodes{Spline, smoothing, interpolation}{65D05, 65D07, 65D10}
```

you get the output

Keywords: Spline, smoothing, interpolation.

AMS classification: 65D05, 65D07, 65D10.

The authors' affiliations should appear as the last text of the article. Type your name, address and e-mail into an `address` environment, introducing `\\` commands to start new lines. This environment accepts an optional argument to fix the number of addresses per text line. Valid values are 1 (the default) and 2. If there are two or more authors, add as many `address` environments as required, separated by blank lines. Of course, you can include several authors in the same `address` if they share their affiliation. To put two consecutive addresses side by side in the same text line, do not leave any blank line between the corresponding `address` environments and set to 2 the optional argument of the first environment. The address at the end of this document has been obtained with the following code:

```
\begin{address}
   Juan Jos\'e Torrens \\
   Departamento de Estad\'istica, Inform\'atica y Matem\'aticas \\
   Universidad P\'ublica de Navarra \\
   Campus de Arrosad\'ia \\
   31006 Pamplona, Spain \\
   \texttt{jjtorrens@unavarra.es}
\end{address}
```

See also the patterns provided in `model.tex`.

§4. Lists

You will notice that lists are, by default, more compact than those provided by the `article` class, since the corresponding vertical skips have been halved. Use the standard LaTeX environments (`itemize`, `enumerate`, `description` and, eventually, `list`). Please, do not load any of the existing packages for list handling (enumerate, enumitem, paralist...) to prevent incompatibilities. In most cases, lists can be customized with simple redefinitions.

In addition, you may use the `myitemize` environment, defined in `jaca`, to write bulleted lists. The default label can be changed with the optional argument of this environment. The left margin is adjusted to fit the label width. For example, the list in Section 1 is obtained with the following code:

```
\begin{myitemize}[$\circ$]
   \item ...
   \item ...
   .........
\end{myitemize}
```

§5. Graphics

The proceedings will be typeset with pdfLaTeX. This program is able to deal with several graphics formats, including `pdf`, `png` and `jpeg`, but excluding `ps` and `eps`. To simplify things, we shall only accept `pdf`, `png` and `jpeg` files. If required, convert your figures to a valid format before submitting your paper. The file size may also be an issue, so keep it into reasonable bounds.

To include graphic files, load the graphicx package and use the `\includegraphics` command, which has the following syntax:

`\includegraphics[`*key*`]{`*filename*`}`

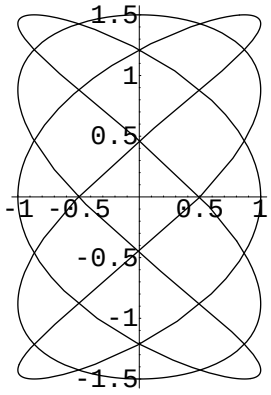
Figure 1 illustrates different keys that can be used in the optional argument.

§6. Typing Mathematics

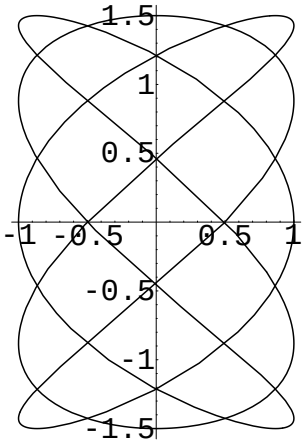
The amsmath package. It is directly loaded by the `jaca` class. We encourage you to take advantage of the many features provided by this package: environments for multiline formulas (`split`, `gather`, `multline`, `align`, `alignat`...), matrix environments (`matrix`, `pmatrix`, `bmatrix`...), references to equations (`\eqref` command), common numbering of a group of equations (`subequations` environment), etc. See the user's guide [1] and Appendix B.

Fonts and mathematical symbols. The TX math fonts provide all the symbols included in the Computer Modern and AMS fonts, as well as new symbols like ≔, ⟺ or ⨌ (see [5] for a complete list[3]). Even if you use the class options `Adobe` or `CM`, there is no need of loading the amssymb package, since the `jaca` class takes care of that.

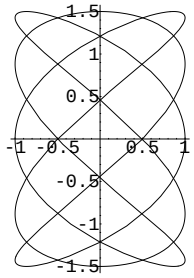
[3] The symbols ≔, ⟺ and ⨌, in particular, are written with the commands `\coloneqq`, `\DiamonddotRight` and `\varoiiintctrclockwise`, respectively.



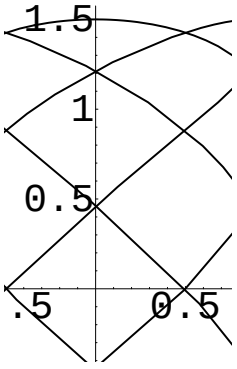
No optional argument



`height=6cm`



`scale=0.7`



`trim=1cm 2cm 0.5cm 0cm,clip,width=3cm`

Figure 1: Graphics inserted with the `\includegraphics` command. The value of the optional argument is shown below every picture.

The bold versions of symbols and letters are obtained through the `\boldsymbol` and `\mathbf` commands. They are commonly used to denote vectors. To this end, the `jaca` class redefines the `\vec` command[4] and provides its starred form `\vec*`: they are equivalent, respectively, to `\mathbf` and `\boldsymbol`. For example, to get

$$(\boldsymbol{\alpha}\times\boldsymbol{\beta})\cdot(\mathbf{a}\times\mathbf{b}) = \begin{vmatrix} \boldsymbol{\alpha}\cdot\mathbf{a} & \boldsymbol{\beta}\cdot\mathbf{a} \\ \boldsymbol{\alpha}\cdot\mathbf{b} & \boldsymbol{\beta}\cdot\mathbf{b} \end{vmatrix}$$

just type

```
\[
  (\vec*{\alpha}\times\vec*{\beta}) \cdot (\vec{a}\times\vec{b}) =
  \begin{vmatrix}
     \vec*{\alpha}\cdot\vec{a} & \vec*{\beta}\cdot\vec{a} \\
     \vec*{\alpha}\cdot\vec{b} & \vec*{\beta}\cdot\vec{b}
  \end{vmatrix}
\]
```

The starred form of `\vec` is mainly intended for symbols. However, it can be also applied to latin letters, yielding italic bold characters (compare $\mathbf{f}$ with $\boldsymbol{f}$, that is, `\vec{f}` with `\vec*{f}`).

Likewise, you can use the `\mathfrak` and `\mathbb` commands to access the Euler Fraktur and Blackboard fonts. In fact, the `jaca` class provides the commands `\Nset`, `\Zset`, `\Qset`, `\Rset` and `\Cset` to get the number sets $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ and $\mathbb{C}$, respectively.

Log-like functions. You may use the following new log-like commands: `\curl`, `\diam`, `\div`[5], `\grad`, `\meas`, `\rot`, `\sign` and `\supp`. For example, the code

```
$\div(f\,\vec{u})=(\grad f)\cdot\vec{u} + f\,\div\vec{u}$
```

yields $\operatorname{div}(f\,\mathbf{u}) = (\operatorname{grad} f)\cdot\mathbf{u} + f\,\operatorname{div}\mathbf{u}$.

Delimiters. The `\pdelim`, `\bdelim`, `\Bdelim`, `\vdelim`, `\Vdelim` and `\adelim` commands surround their mandatory argument by a pair of matching delimiters: ( ), [ ], { }, | |, ‖ ‖ and ⟨ ⟩, respectively. The `\abs` and `\norm` commands are suitable alias of `\vdelim` and `\Vdelim`. The size of the delimiters is adjusted to that of the mandatory argument. For finer control, one can use the optional argument of these commands, which should be a number between 0 and 4. Check the following example:

```
\[
  \norm{v}_{m,\Omega}=\pdelim{\sum_{\abs{\alpha}\leq m}
      \int_{\Omega} \abs{\partial^\alpha v(x)}^2\,dx}^{1/2} \quad
  \pdelim[4]{ \bdelim[3]{ \Bdelim[2]{ \vdelim[1]{
     \adelim[0]{ \vec{x},\vec{y}} } } } } \quad
  \norm[4]{\sum_{i=1}^n \lambda_i (\vec{x}_i^\ast + \vec{y}_i)}
     \leq \sum_{i=1}^n \abs{\lambda_i}
     \pdelim[1]{\norm[0]{\vec{x}_i^\ast} + \norm{\vec{y}_i}}
\]
```

[4]The old definition of `\vec` is accessible through the command `\Vec`.

[5]The standard `\div` command has been redefined. To get $a \div b$, write `$a\Div b$`.

Environment	Heading	Counter	Environment	Heading	Counter
`thm`	**Theorem**	`thm`	`cor`	**Corollary**	`thm`
`prop`	**Proposition**	`thm`	`lem`	**Lemma**	`thm`
`claim`	**Claim**	`thm`	`axiom`	**Axiom**	`thm`
`conj`	**Conjecture**	`thm`	`fact`	**Fact**	`thm`
`hyp`	**Hypothesis**	`thm`	`assum`	**Assumption**	`thm`
`crit`	**Criterion**	`thm`			
`defn`	**Definition**	`defn`	`exmp`	**Example**	`exmp`
`prob`	**Problem**	`prob`	`case`	**Case**	`case`
`alg`	**Algorithm**	`alg`			
`rem`	*Remark*	`rem`	`note`	*Note*	`rem`

Table 1: Predefined theorem-like environments.

$$\|v\|_{m,\Omega} = \left(\sum_{|\alpha|\leq m} \int_\Omega |\partial^\alpha v(x)|^2 \, dx \right)^{1/2} \qquad \left(\left[\left\{ \left| \langle \mathbf{x}, \mathbf{y} \rangle \right| \right\} \right] \right) \qquad \left\| \sum_{i=1}^{n} \lambda_i (\mathbf{x}_i^* + \mathbf{y}_i) \right\| \leq \sum_{i=1}^{n} |\lambda_i| \left(\|\mathbf{x}_i^*\| + \|\mathbf{y}_i\| \right)$$

The commands `\ooi`, `\oci` and `\coi` serve to write intervals with French notation[6]. For example, write

```
$(x,y,z)\in\ooi{-1,+\infty}\times\coi{-3,2}\times\oci{-\infty,5}$
```

to get $(x, y, z) \in \,]-1, +\infty[\times [-3, 2[\times \,]-\infty, 5]$. For completenes, there also exists the command `\cci`, equivalent to `\bdelim`. These four commands also have an optional argument to control the delimiter size.

Theorem-like environments. The `jaca` class also defines several theorem-like environments to write your results. They are summarized in Table 1. Environments with the same internal counter share the same numbering. To add new environments of this type, use the `\newtheorem` command. Likewise, there exists the `proof` environment for obvious purposes. All these environments admit an optional argument to modify the default heading. Some of them are exemplified in Appendix A.

Displayed formulas. The `jaca` class disables the `eqnarray` environment as well as the `$$` command for entering in display math mode. There are good reasons for doing so (cf., for example, [4]). Thus, if you try something like `$$a+b=c$$`, LaTeX will stop typesetting and will issue an error message. Displayed formulas must be written enclosed by `\[` and `\]` or inside any of the suitable environments defined in amsmath. See Appendix B for examples. You will also find there tips for breaking long formulas, aligning multiline formulas, etc.

[6] In the names of these commands, the letters o and c stand for "open" and "close".

§7. Bibliography

Write the bibliography of your article with the help of BibTeX. To this end, first build a suitable `bib` file, say, `foo.bib`, and then add the following lines at the right place of your `tex` file:

```
\bibliographystyle{acmurl}
\bibliography{foo}
```

The bibliographic style `acmurl.bst` results from processing the style `acm.bst` with the Perl script urlbst (cf. [3]). This style supports the inclusion of webpages, URL's and DOI's in the bibliography. Specifically, you can add the fields `url` and `doi` to any entry. In addition, there is a new entry type called `webpage`. Its required fields are `title` and `url`, whereas the optional fields are `author`, `editor` (or maintainer of the webpage), `note`, `year`, `month` and `lastchecked`.

For your own work (but not for the proceedings), you may want to load the hyperref package. Then you will get an active hyperlink at every URL and DOI contained in the bibliography. This may be useful when reading your article on-screen.

References

[1] American Mathematical Society. *User's guide for the amsmath package (version 2.0)*, 2002. Available from: http://www.ctan.org/pkg/amsmath.

[2] Ciarlet, P. G. *The Finite Element Method for Elliptic Problems*, vol. 40 of *Classics in Applied Mathematics*. Siam, Philadelphia, 2002. Firstly published by North-Holland, Amsterdam, 1978.

[3] Gray, N. Urlbst [online]. 2010. Available from: http://purl.org/nxg/dist/urlbst [cited October 2018].

[4] Madsen, L. Avoid eqnarray! *The PracTeX Journal*, 4 (2006). Available from: http://www.tug.org/pracjourn/2006-4/madsen/madsen.pdf.

[5] Ryu, Y. *The TX fonts*, 2000. Available from: http://www.ctan.org/pkg/txfonts.

§A. Examples of theorem-like environments

The following code shows the use of the environments `defn`, `thm`, `rem` and `proof`. See below the corresponding output.

```
\begin{defn}
   Let $V$ be a Hilbert space and let
   $a(\cdot,\cdot):V\times V\to\Rset$ be a bilinear form.
   We say that $a(\cdot,\cdot)$ is \emph{$V$-elliptic} if there
   exists $\alpha>0$ such that, for all $v\in V$,
   $a(v,v)\geq \alpha\norm{v}^2$.
\end{defn}

\begin{thm}[Lax-Milgram Lemma] \label{thm1}
```

```
  Let $V$ be a Hilbert space,
  let $a(\cdot,\cdot):V\times V\to\Rset$ be a continuous
  $V$-elliptic bilinear form, and let $f:V\to\Rset$ be a
  continuous linear form. Then, the abstract variational
  problem: find $u$ such that
  \begin{equation*}\label{eq:p}
    \left\{
    \begin{aligned}
       & u\in V,\\
       & \forall v\in V,\ a(u,v)=f(v),
    \end{aligned}
    \right.
    \tag{$\mathcal{P}$}
  \end{equation*}
  has one and only one solution.
\end{thm}

\begin{rem}
  Let us observe that the bilinear form $a(\cdot,\cdot)$ is not
  assumed to be symmetric. Thus, the variational problem
  \eqref{eq:p} is not necessarily associated to a minimization
  problem.
\end{rem}

\begin{proof}[Proof of Theorem \ref{thm1}]
  See P. G. Ciarlet~\cite[Theorem 1.1.3]{Cia02}.
\end{proof}
```

Definition 1. Let V be a Hilbert space and let $a(\cdot,\cdot) : V \times V \to \mathbb{R}$ be a bilinear form. We say that $a(\cdot,\cdot)$ is *V-elliptic* if there exists $\alpha > 0$ such that, for all $v \in V$, $a(v,v) \geq \alpha \|v\|^2$.

Theorem 1 (Lax-Milgram Lemma)**.** *Let V be a Hilbert space, let $a(\cdot,\cdot) : V \times V \to \mathbb{R}$ be a continuous V-elliptic bilinear form, and let $f : V \to \mathbb{R}$ be a continuous linear form. Then, the abstract variational problem: find u such that*

$$\begin{cases} u \in V, \\ \forall v \in V,\ a(u,v) = f(v), \end{cases} \tag{$\mathcal{P}$}$$

has one and only one solution.

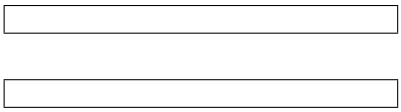
Remark 1. Let us observe that the bilinear form $a(\cdot,\cdot)$ is not assumed to be symmetric. Thus, the variational problem ($\mathcal{P}$) is not necessarily associated to a minimization problem.

Proof of Theorem 1. See P. G. Ciarlet [2, Theorem 1.1.3]. □

§B. Displayed formulas

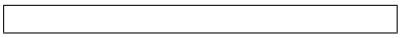
In the examples below, `\foo` is just a command which draws a rectangle whose width is controlled by an optional argument (its value is 2, by default). Replace each instance of `\foo` with a meaningful mathematical expression.

Example 1. Single unnumbered formula.



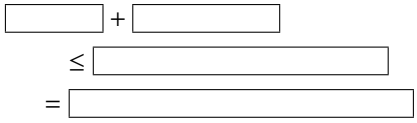
```
\[ \foo[8] \]

\begin{equation*}
   \foo[8]
\end{equation*}
```

Example 2. Single numbered formula.

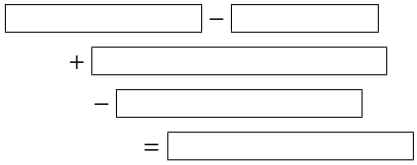


```
\begin{equation}
   \foo[8]
\end{equation}
```

Example 3. Long single formula split in several lines with no alignment.



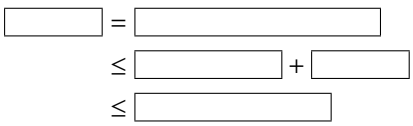
```
\begin{multline}
   \foo + \foo[3] \\
   \leq \foo[6] \\
   = \foo[7]
\end{multline}
```



```
\begin{multline}
   \foo[4] - \foo[3] \\
   + \foo[6] \\
   - \foo[5] \\
   = \foo[5]
\end{multline}
```

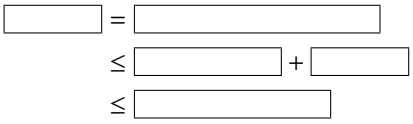
The `multline*` environment suppresses numbering.

Example 4. Long single formula split in several lines with an alignment point. Let us start with an unnumbered formula:



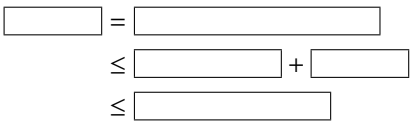
```
\begin{align*}
   \foo &= \foo[5]  \\
     &\leq \foo[3] + \foo  \\
     &\leq \foo[4]
\end{align*}
```

The alignment point is marked with `&`. To obtain the corresponding numbered formula, we replace `align*` by `split` and include everything in an `equation` environment:



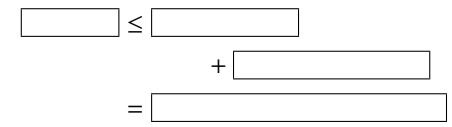
```
\begin{equation}
   \begin{split}
      \foo &= \foo[5]  \\
        &\leq \foo[3] + \foo  \\
        &\leq \foo[4]
   \end{split}
\end{equation}
```

In some cases, it suffices to replace `align*` by `align` and suppress the numbering of some lines with `\notag`:

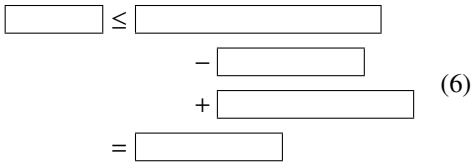


```
\begin{align}
   \foo &= \foo[5] \notag \\
     &\leq \foo[3] + \foo  \\
     &\leq \foo[4] \notag
\end{align}
```

It is often necessary to indent some lines, due to breaks at connectives of different types, a long left-most expression, etc. We mark indentations with the `\tab` command, defined by `\newcommand{\tab}{\qquad}`. Of course, you could replace `\qquad` by, say, `\quad`, `\hspace{0.5em}` or even nothing, depending on the degree of indentation you want. There are several alignment strategies. Some of them are shown in following examples:

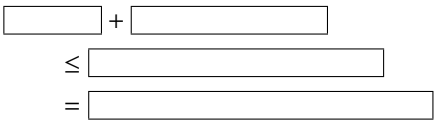


```
\begin{align*}
   \foo \leq{}& \foo[3] \\
      &\tab + \foo[4] \\
   ={}& \foo[6]
\end{align*}
```



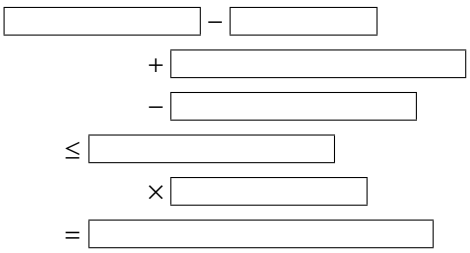
(6)

```
\begin{equation}
   \begin{split}
      \foo \leq{}& \foo[5] \\
         &\tab - \foo[3] \\
         &\tab + \foo[4] \\
      ={}& \foo[3]
   \end{split}
\end{equation}
```

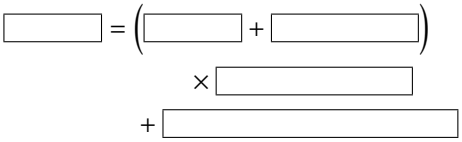


(7)

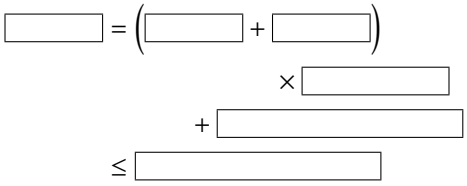
```
\begin{equation}
   \begin{split}
      &\foo + \foo[4] \\
      &\tab \leq \foo[6] \\
      &\tab = \foo[7]
   \end{split}
\end{equation}
```



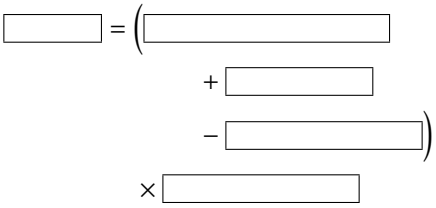
```
\newcommand{\phEq}{\phantom{{}={}}}
\begin{align*}
   &\foo[4] - \foo[3] \\
     &\tab\phEq\tab + \foo[6] \\
     &\tab\phEq\tab - \foo[5] \\
   &\tab\leq\foo[5] \\
     &\tab\phEq\tab \times\foo[4] \\
   &\tab=\foo[7]
\end{align*}
```



```
\begin{align*}
   \foo ={}
     &\pdelim[3]{\foo + \foo[3]} \\
     &\tab \times \foo[4] \\
   & + \foo[6]
\end{align*}
```

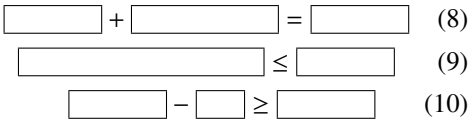


```
\newcommand{\phEq}{\phantom{{}={}}}
\begin{align*}
   \foo ={}
     &\pdelim[3]{\foo + \foo[2]} \\
     &\tab\phEq\tab \times\foo[3] \\
     &\tab + \foo[5] \\
   \leq{}& \foo[5]
\end{align*}
```



```
\begin{align*}
   \foo ={}&\!
   \begin{aligned}[t]
      \biggl( &\foo[5] \\
        &\tab + \foo[3] \\
        &\tab - \foo[4] \biggr)
   \end{aligned} \\
   &\times \foo[4]
\end{align*}
```

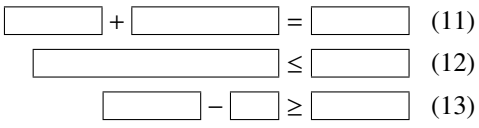
Example 5. Several formulas without alignment.



```
\begin{gather}
   \foo + \foo[3] = \foo \\
   \foo[5] \leq \foo \\
   \foo - \foo[1] \geq \foo
\end{gather}
```

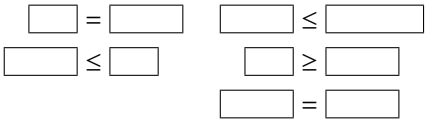
The `gather*` environment suppresses numbering. You can also remove the number of a specific line with `\notag`.

Example 6. Several formulas with an alignment point. This case is similar to that of Example 4.



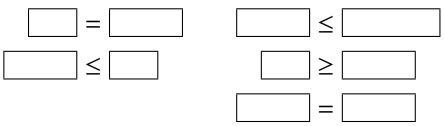
```
\begin{align}
   \foo + \foo[3] &= \foo \\
   \foo[5] &\leq \foo \\
   \foo - \foo[1] &\geq \foo
\end{align}
```

Example 7. Several formulas with at least two alignment points. Divide the group of formulas into columns and choose in each column an alignment point. Then, line by line, put the & sign at the end of every column (except the last one) and at every alignment point.



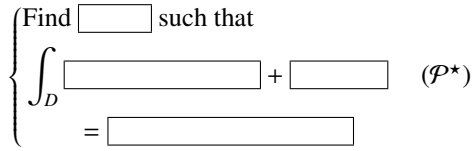
```
\begin{align*}
   \foo[1] &=\foo[1.5] &
   \foo[1.5] &\leq \foo[2] \\
   \foo[1.5] &\leq \foo[1] &
   \foo[1] &\geq \foo[1.5] \\
   & & \foo[1.5] &= \foo[1.5]
\end{align*}
```

The `align` environment computes automatically the space between columns. The `alignat` environment, instead, allows to explicitly set such a space. The number of columns should be specified in the mandatory argument of this environment:



```
\begin{alignat*}{2}
   \foo[1] &=\foo[1.5] \qquad &
   \foo[1.5] &\leq \foo[2] \\
   \foo[1.5] &\leq \foo[1] &
   \foo[1] &\geq \foo[1.5] \\
   & & \foo[1.5] &= \foo[1.5]
\end{alignat*}
```

Example 8. Formulas grouped with a brace on the left. This case frequently happens in the statement of problems, as problem ($\mathcal{P}$) in page 10. Here we have another example:

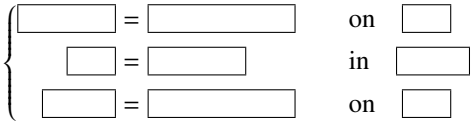


```
\begin{equation*}\label{eq:ps}
   \left\{
   \begin{aligned}
      &\text{Find \foo[1.5]
             such that} \\
      &\int_D\foo[4] + \foo \\
      &\tab = \foo[5]
   \end{aligned}
   \right.
   \tag{$\mathcal{P}^\star$}
\end{equation*}
```

The preceding example shows two additional features:

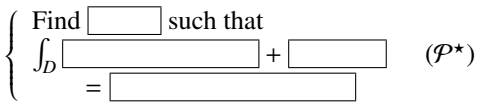
- the `\text` command allows to properly insert text in a formula;
- using the `\tag` command, you can put a custom label in a formula instead of an equation number; then you can simply refer to the formula, as usual, by combining the `\label` and `\eqref` commands: in the above example, if you write "`problem \eqref{eq:ps}`" in the source file, you get "problem ($\mathcal{P}^\star$)" (see also the example in Appendix A).

If there are two or more alignment points, the `alignedat` environment may be better suited:



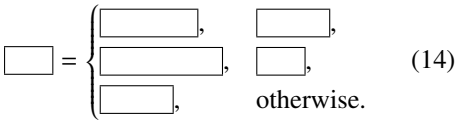
```
\[
  \left\{
  \begin{alignedat}{2}
     \foo     &= \foo[3] &\qquad
        & \text{on}\quad \foo[1] \\
     \foo[1]  &= \foo[2] &
        & \text{in}\quad \foo[1.5]\\
     \foo[1.5]&= \foo[3] &
        & \text{on}\quad \foo[1]
  \end{alignedat}
  \right.
\]
```

In any case, do not use the `array` environment, which usually gives worse results. Compare the above statement of problem ($\mathcal{P}^\star$) with the next one, yielded by a straightforward use of the `array` environment:



```
\begin{equation*}
   \left\{
   \begin{array}{l}
      \text{Find \foo[1.5]
            such that} \\
      \int_D\foo[4] + \foo \\
      \tab= \foo[5]
   \end{array}
   \right.
   \tag{$\mathcal{P}^\star$}
\end{equation*}
```

For the piecewise definition of functions, use the `cases` environment (and avoid `array`!):



(14)

```
\begin{equation}
   \foo[1]=
   \begin{cases}
      \foo, & \foo[1.5], \\
      \foo[2.5], & \foo[1], \\
      \foo[1.5], & \text{otherwise}.
   \end{cases}
\end{equation}
```

Juan José Torrens
Departamento de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Campus de Arrosadía
31006 Pamplona, Spain
jjtorrens@unavarra.es