



Initiation à \LaTeX

Mathématiques, mise en page complexe

Pierre Senellart



Semaine *Informatique pratique*, septembre 2023



Plan

Mathématiques

Environnements mathématiques

Caractères

Formules

Théorèmes

Tableaux

Images et flottants



Mathématiques en ligne

On écrit des formules mathématiques **en ligne** en les encadrant de `$ $` ou, de manière équivalente, de `\(\)`.

Pour tout `$x\in \mathbb{N}$`,
`$x+1=1+x$`.

`\bigskip`

On peut aussi démontrer que

`\(`

`\sum_{n=1}^{+\infty}`

`\frac{1}{n^2}=\frac{\pi^2}{6}`

`\)` est vrai.

Pour tout $x \in \mathbb{N}$, $x+1 = 1+x$.

On peut aussi démontrer que

$\sum_{n=1}^{+\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$ est vrai.

Pour avoir le même affichage qu'en mode mise en évidence (cf. ci-après), il faut ajouter `\displaymath` à l'intérieur.



Mathématiques mises en évidence

On écrit des formules mathématiques **mises en évidence** (*displayed equations*) en les encadrant de `\[\]` ou éventuellement de

`\begin{equation*}` `\end{equation*}`.



Ne pas utiliser `$$$` qui est la manière de faire en Plain T_EX, et pose des problèmes d'espacement avec L^AT_EX.

La constante d'Euler `\gamma` est telle que:

```
\[
\lim_{n\rightarrow+\infty}
\left(\sum_{k=1}^n
\frac{1}{k}\right)
-\ln(n)=\gamma
\]
```

La constante d'Euler γ est telle que :

$$\lim_{n \rightarrow +\infty} \left(\sum_{k=1}^n \frac{1}{k} \right) - \ln(n) = \gamma$$

Équations numérotées

```
\begin{equation}
\label{equ:masse-energie}
E=m\cdot\mathsf{c}^2
\end{equation}
```

On peut utiliser l'équation%
~\eqref{equ:masse-energie}
pour prouver le théorème
suivant.

$$E = m \cdot c^2 \quad (1)$$

On peut utiliser l'équation (1)
pour prouver le théorème sui-
vant.

`\eqref` est un ajout du package amsmath. On peut aussi
simplement utiliser `\ref` (mais il faudra mettre les parenthèses à
la main).

Environnements du package amsmath

amsmath propose des environnements mathématiques permettant de regrouper plusieurs équations, ou de séparer une équation en plusieurs lignes.

<code>multline</code>	équation sur plusieurs lignes, sans alignement
<code>split</code>	équation sur plusieurs lignes, avec alignement
<code>gather</code>	groupe d'équation sans alignement
<code>align</code>	groupe d'équations avec alignement
<code>aligned</code>	alignement à l'intérieur d'une équation complexe



Ne pas utiliser `eqnarray` / `eqnarray*` qui donne de mauvais résultats.



Environnement multiline

```
\begin{multiline*}
x=1+2+3\\
+\dots+\\
+(n-1)+n
\end{multiline*}
```

$$\begin{aligned}
 x &= 1 + 2 + 3 \\
 &\quad + \cdots + \\
 &\quad + (n - 1) + n
 \end{aligned}$$

`multiline*` : non numéroté

Environnement split

```
\begin{equation}
\begin{split}
\phi = {} & (x \wedge y \wedge z) \vee \\
& (\neg x \wedge y \wedge \neg z)
\end{split}
\end{equation}
```

$$\phi = (x \wedge y \wedge z) \vee (\neg x \wedge y \wedge \neg z) \quad (2)$$

Le `{}` après `\lor` permet d'avoir un espacement correct : on dit à \LaTeX que \vee est bien utilisé comme opérateur binaire.

Environnement aligned

```
\begin{equation*}
\left.\begin{aligned}
\mathbf{B}' &= -\partial \times \mathbf{E}, \\
\mathbf{E}' &= \partial \times \mathbf{B} - 4\pi \mathbf{J}
\end{aligned}
\right\}
\quad \text{Maxwell's equations}
\end{equation*}
```

$$\left. \begin{aligned} \mathbf{B}' &= -\partial \times \mathbf{E}, \\ \mathbf{E}' &= \partial \times \mathbf{B} - 4\pi \mathbf{J} \end{aligned} \right\} \quad \text{Maxwell's equations}$$



Plan

Mathématiques

Environnements mathématiques

Caractères

Formules

Théorèmes

Tableaux

Images et flottants



Alphabet latin

Par défaut, les noms de variables, etc., sont affichés en italique. On a accès aux **polices mathématiques** suivantes, pour les lettres de l'alphabet latin :

<code>\mathrm{ABCDZ}</code>	Empattements	ABCDZ
<code>\mathsf{ABCDZ}</code>	Sans empattements	ABCDZ
<code>\mathtt{ABCDZ}</code>	Chasse fixe	ABCDZ
<code>\mathit{ABCDZ}</code>	Italique	<i>ABCDZ</i>
<code>\mathbf{ABCDZ}</code>	Gras	ABCDZ
<code>\mathcal{ABCDZ}</code>	Cursif	<i>ABCDZ</i>
<code>\mathfrak{ABCDZ}</code>	Fraktur	<i>ABCDZ</i>
<code>\mathbb{ABCDZ}</code>	Gras « façon tableau »	ABCDZ

Certaines de ces polices nécessitent un `\usepackage{amsfonts}` .

On peut aussi utiliser `\text{texte}` (amsmath) pour mettre du texte à l'intérieur d'une formule.



Alphabet grec

Les lettres de l'alphabet grec sont disponibles par leur noms.

```
\[ \alpha \beta \gamma \delta \epsilon \zeta \eta \theta \iota \kappa \lambda \mu \nu \xi \pi \rho \sigma \tau \upsilon \phi \chi \psi \omega
  \iota \kappa \alpha \rho \lambda \lambda \mu \nu \xi \pi \rho \sigma \tau \upsilon \phi \chi \psi \omega
  \upsilon \phi \chi \psi \omega \]
```

```
\[ \Gamma \Delta \Theta \Lambda \Xi \Pi \Sigma \Upsilon \Phi \Psi \Omega
  \Sigma \Upsilon \Phi \Psi \Omega \]
```

$\alpha\beta\gamma\delta\epsilon\zeta\eta\theta\iota\kappa\lambda\mu\nu\xi\pi\rho\sigma\tau\upsilon\phi\chi\psi\omega$

$\Gamma\Delta\Theta\Lambda\Xi\Pi\Sigma\Upsilon\Phi\Psi\Omega$

Il y a aussi `\varepsilonpsilon` ε , `\varpi` ϖ , `\varsigma` ς ,

`\varphi` φ .



Autres caractères

Symboles `\aleph` \aleph , `\infty` ∞

« Accents » `\vec{x}` \vec{x} , `\bar{x}` \bar{x} , `\widehat{abc}` \widehat{abc}

Opérateurs binaires `\leq` \leq , `\cup` \cup , `\notin` \notin

Grands opérateurs `\sum` \sum , `\bigcup` \bigcup

Flèches `\leftarrow` \leftarrow , `\iff` \iff

Pointillés `\ldots` \ldots ou `\cdots` \cdots (`\dots` d'amsmath fait ce qu'il faut en fonction des opérateurs)

Se référer à :

- *The Comprehensive L^AT_EX Symbol List*, accessible par « `texdoc symbols-a4` »
- Trouver un symbole L^AT_EX en le dessinant : <http://detexify.kirelabs.org/classify.html>



Plan

Mathématiques

Environnements mathématiques

Caractères

Formules

Théorèmes

Tableaux

Images et flottants



Noms de fonctions

Les noms de fonctions et opérateurs ne sont traditionnellement pas en italique :

<code>\sin</code>	sin
<code>\Pr</code>	Pr
<code>\max</code>	max

...

Pour définir son propre opérateur (extension amsmath) :

```
% En-tête du document
\DeclareMathOperator*{\argmin}{argmin}
\DeclareMathOperator{\argmax}{argmax}
% Corps
\[\argmin_x f(x)=0\]
\[\argmax_x f(x)=1\]
```

$$\argmin_x f(x) = 0$$

$$\argmax_x f(x) = 1$$



Indices et exposants

```
\[
\prod_{\substack{x \in X \\ f(x)=0}}
g(x^{2^n})
\]
```

$$\prod_{\substack{x \in X \\ f(x)=0}} g(x^{2^n})$$

`\substack` est une fonctionnalité de amsmath.

On peut contrôler le placement (en-dessous ou à côté) avec les instructions `\limits` ou `\nolimits` directement après l'opérateur.



Fractions, racines, grands délimiteurs

```
\[
\left\{
\frac{1}{\sqrt{
\left(\frac{x}{y}\right)^2}}
}
\middle|
y \neq 0 \right\}
\]
```

$$\left\{ \frac{1}{\sqrt{\left(\frac{x}{y}\right)^2}} \middle| y \neq 0 \right\}$$

On peut utiliser `\left.`, `\right.` quand un seul des deux côtés a un délimiteur.



Coefficients binomiaux

Convention française:

```
\[
\mathsf{C}_n^p=
\frac{n!}{p!(n-p)!}
\]
```

Convention anglaise:

```
\[
\binom{n}{p}=
\frac{n!}{p!(n-p)!}
\]
```

Convention française :

$$C_n^p = \frac{n!}{p!(n-p)!}$$

Convention anglaise :

$$\binom{n}{p} = \frac{n!}{p!(n-p)!}$$

`\binom` est une macro du package **amsmath**.



Espacement mathématique et fantômes

Normalement, \LaTeX fait un bon travail de gestion de l'espacement. Des fois, il faut cependant ajuster à la main :

<code>x\,x</code>	$x\,x$	espace fine
<code>x\:x</code>	$x\:x$	espace moyenne
<code>x\;x</code>	$x\;x$	espace grande
<code>x\ x</code>	$x\ x$	espace normale
<code>x!\,x</code>	xx	espace négative
<code>x\quad x</code>	$x\quad x$	cadratin (1 em)
<code>x\qquad x</code>	$x\qquad x$	double cadratin
<code>xx</code>	$x\qquad x$	de même largeur que l'argument

Certaines de ces commandes sont aussi utilisables en mode texte.

`\vphantom`, qui occupe la hauteur de son argument et une largeur nulle, existe aussi.



Plan

Mathématiques

Environnements mathématiques

Caractères

Formules

Théorèmes

Tableaux

Images et flottants



Théorèmes, définitions, etc.

On peut définir un environnement de type théorème de la manière suivante :

```
\newtheorem{definition}{Définition}
```

On l'utilise ensuite :

```
\begin{definition}
Une formule est
\emph{satisfiable}
s'il existe une valuation
qui la rend vraie.
\end{definition}
```

Définition

Une formule est satisfiable s'il existe une valuation qui la rend vraie.

amsthm propose plusieurs styles différents de théorème.



Preuves

amsthm propose un environnement de démonstration :

```
\begin{proof}  
Ce résultat est trivial.  
\end{proof}
```

Démonstration.

Ce résultat est trivial.



Plan

Mathématiques

Tableaux

Bases

Fonctionnalités avancées

Images et flottants



L'environnement tabular

On définit un tableau en \LaTeX avec l'environnement **tabular** :

```
\begin{tabular}{specification}
l1,c1&l1,c2 ... \\
l2,c1&l2,c2 ... \\
...
\end{tabular}
```

- La spécification est une suite de n caractères, un pour chaque colonne du tableau. Chaque caractère décrit l'alignement de la colonne (m et b nécessitent le package **array**) :
 - l texte sur une seule ligne, aligné à gauche
 - r texte sur une seule ligne, aligné à droite
 - c texte sur une seule ligne, centré
 - p{1cm} paragraphe de 1 cm de large, aligné verticalement en haut
 - m{1cm} paragraphe de 1 cm de large, aligné verticalement au milieu
 - b{1cm} paragraphe de 1 cm de large, aligné verticalement en bas



L'environnement tabular, suite

- Les lignes sont indiquées l'une après l'autre, séparées par des `\\` (ou `\tabularnewline`) ; les colonnes sont séparées par des esperluettes `&`.

```
\begin{tabular}{lcc}
&Vrai&Faux\\
A&oui&\non\\
B&\non&\non
\end{tabular}
```

	Vrai	Faux
A	oui	non
B	non	non

- Les tableaux sont aussi souvent utilisés (avec p, m ou b) pour mettre en page deux « blocs » l'un à côté de l'autre ; dans ce cas on exprime souvent la taille en une fraction de la largeur de la ligne : `.3\linewidth` (voir aussi `\minipage` et `\parbox`)



L'environnement array

En mode mathématique, on utilise **array** à la place de **tabular**. Les fonctionnalités sont les mêmes.

```
\[
I_3=\left(
\begin{array}{ccc}
1&0&0\\
0&1&0\\
0&0&1
\end{array}\right)
```

$$I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

amsmath définit aussi des commandes spéciales pour les matrices.



Plan

Mathématiques

Tableaux

Bases

Fonctionnalités avancées

Images et flottants



Colonnes uniformes

Avec le package **array**, on peut indiquer du code \LaTeX arbitraire à mettre au début d'une colonne (préfixer le caractère de spécification par `>\code`), à la fin (suffixer le caractère de spécification par `<\code`) ou entre deux colonnes (ajouter `@{\}` dans la spécification).

```
\begin{tabular}
  >{\bfseries}c
  >{\color{blue}}c}
a&b\\
b&a\\
\end{tabular}
```

a	b
b	a



Colonnes fusionnées

On peut utiliser `\multicolumn{num}{specification}{contenu}` pour fusionner des colonnes sur une ligne donné. **num** est le nombre de colonnes, **specification** le caractère de spécification.

```
\begin{tabular}{cc}
a&b\\
\multicolumn{2}{c}{bouh}\\
b&a\\
\end{tabular}
```

```
a    b
bouh
b    a
```



Lignes fusionnées (multirow)

Avec le package **multirow** on peut fusionner deux lignes au sein d'une colonne donné. La syntaxe est

`\multirow{num}{largeur}{contenu}` . La largeur peut être donnée de manière absolue, ou on peut juste mettre `*` qui sera interprété comme « décider suivant la largeur du contenu ».

```
\begin{tabular}{lr}
Bouh&x\\
Bah&y\\
\multirow{2}{*}{Bli}
&z\\
&t\\
\end{tabular}
```

Bouh	x
Bah	y
Bli	z
	t



Traits de séparation

On peut ajouter des traits horizontaux avec `\hline` et des traits verticaux en mettant un caractère `|` aux endroits appropriés de la spécification.

```
\begin{tabular}{|c|c|}
\hline
a&b\\
\hline
b&a\\
\hline
\end{tabular}
```

a	b
b	a



Les typographes recommandent de ne **jamais** mettre de traits verticaux dans un tel tableau, et les lignes produites par `\hline` sont peu esthétiques. On préférera celle du package **booktabs**.



Tableaux plus jolis avec booktabs

```
\begin{tabular}{cc}
\toprule
T&Y\\
\midrule
a&b\\
b&c\\
b&a\\
\bottomrule
\end{tabular}
```

T	Y
a	b
b	c
b	a

On peut aussi utiliser `\cmidrule` pour des lignes partielles, voir la documentation de [booktabs](#).



Plan

Mathématiques

Tableaux

Images et flottants

Images

Flottants



Le package graphicx

Pour charger des images, on utilise le package graphicx et la commande `\includegraphics`.

```
\includegraphics[options]{nom_fichier_sans_extension}
```

Options reconnues :

`width=.5\linewidth` pour définir la largeur en fonction de la largeur de la ligne

`height=1cm` pour définir une hauteur constante

`angle=90` pour tourner l'image



Ne pas utiliser epsfig ou graphics, ces packages sont plus anciens.

Formats graphiques supportés

Compilation par latex puis dvips seulement EPS (Encapsulated PostScript)

Compilation par pdflatex, lualatex

- PDF pour les images vectorielles (description logique de l'image)
- PNG pour les schémas, logos, etc., bitmaps (description de l'image pixel par pixel)
- JPG pour les photos et autres images bitmaps à tons continus

Conversion entre formats graphiques

D'un format bitmap vers PNG ou EPS (GIF, BMP, TIF, RLE...)

Utiliser n'importe quel logiciel de dessin bitmap (Paint, PhotoShop, Gimp...). Sous Unix, convert (ImageMagick) en ligne de commande.



Ne pas convertir un logo en JPG! JPG est un format avec compression qui n'est adapté qu'aux images à ton continu.

D'un format vectoriel vers EPS ou PDF (SVG, WPG, WMF...)

Utiliser un programme de dessin vectoriel comme Inkscape.

De EPS vers PDF Utiliser epstopdf, fourni dans les distributions de \LaTeX .

De PDF vers EPS Utiliser pdftops, avec l'option « -eps », qui est fourni avec GhostScript.



Produire des schémas de bonne qualité

- Avec n'importe quel logiciel de **dessin vectoriel** (Inkscape, OpenOffice Draw, etc.) Certains de ces logiciels (en particulier, Inkscape, xfig, IPE) proposent soit un export directement en \LaTeX , soit la possibilité d'intégrer des formules \LaTeX au schéma.
- Avec une description programmatrice du schéma, dans un **langage externe** qui s'intègre bien avec \LaTeX (MetaPost, Asymptote)
- En décrivant le schéma programmatoirement à l'intérieur de \LaTeX lui-même (xypic, pstricks, **tikz**)



Dans la mesure du possible, sauvegarder les schémas dans un **format vectoriel** !



Plan

Mathématiques

Tableaux

Images et flottants

Images

Flottants



Flottants

Deux environnements : `\begin{figure}` et `\begin{table}` permettant de définir des figures et tables **flottantes**, c'est-à-dire, ne suivant pas le flux normal du document.

```
\begin{figure}[thp]
  \includegraphics{toto}
\end{figure}
```

L'environnement a un paramètre optionnel qui est une **spécification de position**, purement indicative. **t** pour haut de page, **b** pour bas de page, **h** pour position courante, **p** pour page séparée.

Possible de définir de nouveaux environnement flottants avec le package **float**.

Certaines classes ou package gérant l'affichage en multicolonnnes (**multicol**) proposent `figure*` et `table*` qui indiquent que le flottant s'étend sur plusieurs colonnes.

Légendes et références

```
\begin{table}
\centering
\begin{tabular}{cr}
Carottes&10~\texteuro\\
Fromage&15~\texteuro\\
Lapin&100~\texteuro\\
\end{tabular}
\caption{Montant des dépenses}
\label{tab:depenses}
\end{table}
```

Ainsi qu'indiqué en Table~\ref{tab:depenses},
le lapin coûte cher.



Le `\label` doit impérativement être après le `\caption`.