

# TP d'initiation à L<sup>A</sup>T<sub>E</sub>X \*

Ludovic GOUDENÈGE  
Univ. Paris-Est

Mathieu LEROY-LERETRE  
Univ. Toulouse

Grégory VIAL  
Univ. Lyon

9 janvier 2012

## Résumé

Ce document est une courte initiation à L<sup>A</sup>T<sub>E</sub>X rédigée dans le but de remplir, de manière synthétique, deux objectifs :

- fournir un minimum vital pour se lancer dans L<sup>A</sup>T<sub>E</sub>X ;
- présenter des fonctionnalités facilitant la rédaction d'un rapport de projet.

Les vingt premières pages suffisent pour découvrir L<sup>A</sup>T<sub>E</sub>X et avoir les bases permettant de s'en servir ; les vingt suivantes couvrent une bonne partie des situations classiques rencontrées lors de l'écriture d'un rapport.

## Mode d'emploi

Le fonctionnement de L<sup>A</sup>T<sub>E</sub>X est très différent de celui des logiciels de traitement de texte ; de plus, il existe sur Internet une foultitude d'informations : cela peut freiner son apprentissage. L<sup>A</sup>T<sub>E</sub>X nécessite donc une initiation, que le présent document se propose de fournir sous la forme d'un TP-cours.

**Qu'apporte ou non ce polycopié d'initiation ?** Il propose des *bases* pour débuter en L<sup>A</sup>T<sub>E</sub>X et un *survol* de beaucoup de possibilités : c'est un *point de départ* avant de consulter des références plus complètes. En conséquence, ce document n'est volontairement *pas exhaustif* (sinon, il serait bien plus long!).

**À qui s'adresse-t-il ?** Il est prévu pour une formation de 4 à 6 heures sur ordinateur, destinée à des étudiants en mathématiques ou en mécatronique, de niveau L3 ou M1, n'ayant jamais (ou peu) utilisé L<sup>A</sup>T<sub>E</sub>X et devant réaliser un document de moyenne taille (10 à 50 pages).

**Comment est-il organisé ?** Le début sert de support aux séances de TP-cours : les sections 1 et 2 permettent de savoir ce qu'est L<sup>A</sup>T<sub>E</sub>X, avant d'aborder la section 3 qui constitue le cœur de cette initiation. La section 4 présente un aperçu de fonctionnalités utiles (mise en page, inclusion d'images, flottants, graphiques, présentations vidéoprojetées). Enfin, des règles typographiques puis des explications volontairement éclipsées au cours des séances (installation, packages, encodage, commandes personnelles, ...) figurent en annexe.

---

\*Polycopié initiallement rédigé lorsque les trois auteurs ont effectué cet enseignement au département mathématique de l'ENS Cachan Bretagne. Contact : [mathieu.leroy-leretre@at@ens-cachan.org](mailto:mathieu.leroy-leretre@at@ens-cachan.org).

# Plan du polycopié de TP-cours

<b>1 Présentation de L<sup>A</sup>T<sub>E</sub>X</b>	<b>3</b>
1.1 Intérêt et utilité . . . . .	3
1.2 Prise en main . . . . .	3
<b>2 Structure d'un fichier .tex</b>	<b>4</b>
<b>3 Découverte de L<sup>A</sup>T<sub>E</sub>X</b>	<b>5</b>
3.1 Démarrage avec T <sub>E</sub> Xstudio (ex-TexMakerX) . . . . .	5
3.2 Composition simple . . . . .	6
3.3 Mathématiques 1 . . . . .	7
3.4 Environnements . . . . .	11
3.5 Sectionnement et références croisées . . . . .	13
3.6 Mathématiques 2 . . . . .	14
3.6.1 Agencement de formules, tableaux et matrices . . . . .	14
3.6.2 Théorèmes et énoncés similaires . . . . .	17
3.6.3 Pour aller plus loin : vers l'infini et au-delà ! . . . . .	20
<b>4 Fonctionnalités plus avancées</b>	<b>21</b>
4.1 Mise en page et structure du document . . . . .	21
4.2 Inclusion d'images . . . . .	23
4.3 Flottants . . . . .	24
4.3.1 Présentation . . . . .	24
4.3.2 Réalisation . . . . .	25
4.4 Schémas et graphiques . . . . .	26
4.4.1 Crées directement dans L <sup>A</sup> T <sub>E</sub> X . . . . .	26
4.4.2 Crées à l'extérieur de L <sup>A</sup> T <sub>E</sub> X . . . . .	27
4.5 Transparents et présentations . . . . .	28
<b>A Typographie française</b>	<b>32</b>
A.1 Avec L <sup>A</sup> T <sub>E</sub> X . . . . .	32
A.2 Quelques règles et usages . . . . .	32
<b>B Explications complémentaires au TP-cours</b>	<b>35</b>
B.1 Compréhension et utilisation de L <sup>A</sup> T <sub>E</sub> X . . . . .	35
B.2 Le \documentclass . . . . .	37
B.3 Le préambule et les packages . . . . .	38
B.4 Le préambule et les macros . . . . .	41
B.5 Gros exemple mathématique . . . . .	42
B.6 Où trouver de l'aide ? . . . . .	45
<b>Références</b>	<b>46</b>

# 1 Présentation de L<sup>A</sup>T<sub>E</sub>X

Voyons dans cette section quelques mots introductifs pour décrire L<sup>A</sup>T<sub>E</sub>X et voir ce que signifie « faire du L<sup>A</sup>T<sub>E</sub>X ».

## 1.1 Intérêt et utilité

L<sup>A</sup>T<sub>E</sub>X, logiciel libre et gratuit, disponible sous Linux, Mac OS et Windows, est un outil de composition performant conçu pour la rédaction de documents scientifiques de qualité : il conviendra en particulier à toute personne ayant besoin de dactylographier un texte contenant des équations, des symboles, etc. Toute publication de *recherche* en mathématiques est faite avec L<sup>A</sup>T<sub>E</sub>X et son utilisation s'étend à d'autres disciplines : physique, informatique ou chimie par exemple. Il s'agit aussi d'un outil efficace dans l'*enseignement* pour rédiger des supports de cours. Son usage est donc loin d'être réservé à des spécialistes !

Remarquons dès maintenant que L<sup>A</sup>T<sub>E</sub>X ne se limite pas à l'écriture d'équations : en pouvant gérer tout seul la mise en page, la création d'une table des matières et plein d'autres choses, il permet à l'utilisateur de se concentrer sur le fond (*i.e.* le contenu du texte qu'il écrit) sans se préoccuper de la forme.

## 1.2 Prise en main

À la différence des traitements de texte usuels (tel MS Word), il ne s'agit pas d'un logiciel où le résultat de la mise en forme (*i.e.* italique, indentation, etc.) s'observe au fur et à mesure que l'on tape : avec L<sup>A</sup>T<sub>E</sub>X, la phase d'édition (*i.e.* saisie du texte) et la phase de relecture (*i.e.* visualisation du texte mis en forme) sont séparées. Autrement dit, L<sup>A</sup>T<sub>E</sub>X ressemble plus à un langage qu'il faut compiler afin de pouvoir voir le résultat attendu ; c'est pour cela que sa prise en main est un peu plus longue.

Rédiger un document en L<sup>A</sup>T<sub>E</sub>X se fait donc en plusieurs étapes, que nous décrivons brièvement :

1. Saisie du code source : on crée un fichier avec l'extension `.tex` (que nous appelons ici `TP_init.tex`) et l'on tape son code dans un *éditeur de texte*.
2. Compilation du fichier source : c'est ici que l'on fait appel à L<sup>A</sup>T<sub>E</sub>X, qui est un programme autonome ; cela crée plusieurs fichiers, dont `TP_init.dvi`.
3. Prévisualisation du fichier DVI : on observe à l'écran `TP_init.dvi` pour voir s'il affiche bien ce qu'on voulait ; sinon, on corrige le fichier source.
4. Impression ou diffusion : on convertit son fichier au format PDF, pour l'imprimer ou le transmettre à quelqu'un.

Ces quatre étapes sont très simples, mais les outils pour les mettre en place peuvent porter des noms différents selon que l'on est sous un environnement Linux, Mac OS ou Windows. Des informations à ce sujet, inutiles pour le TP-cours, sont résumées à l'annexe B.1.2.

## 2 Structure d'un fichier .tex

Le fichier .tex est créé dans un éditeur de texte. Il possède en général le squelette suivant :

1. La commande \documentclass{} qui indique quelle *classe* (ou quel *type*) de document l'on souhaite rédiger.
2. Le *préambule*, où l'on indique des options, des préférences et des choix pour la compilation ; il contient habituellement deux catégories d'informations :
  - les **packages**, qui regroupent des commandes destinées à faciliter l'utilisation de certaines fonctionnalités ou à en créer de nouvelles. Il en existe un très grand nombre et quelques-uns sont cités en annexe (cf. § B.3.3) ;
  - les **macros ou commandes personnelles (de l'utilisateur)**, une richesse de L<sup>A</sup>T<sub>E</sub>X que nous ne ferons qu'effleurer en séance (cf. § B.4).
3. Le *corps de document*, délimité par \begin{document} et \end{document}, où réside le texte lui-même.

L'objectif de ce TP-cours étant une prise en main de L<sup>A</sup>T<sub>E</sub>X, nous allons nous concentrer sur le corps de document. Nous utiliserons donc la classe **article** et nous chargerons les packages de base<sup>1</sup> devenus incontournables pour un document rédigé en français : l'exemple de la figure 1 constitue un « kit de démarrage » de tout fichier .tex.

```
\documentclass[12pt,french]{article}

\usepackage[<option à choisir>]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[a4paper]{geometry}
\usepackage{amsmath, amssymb}
\usepackage{babel} %à charger en dernier

\begin{document}
    Bienvenue en séance de TP !
\end{document}
```

The diagram shows the structure of a LaTeX source code file. The code is enclosed in a box. It is divided into two main sections: 'Préambule' (Preamble) and 'Corps de document' (Document body). The Preamble section starts with \documentclass and ends with \begin{document}. It contains several \usepackage commands and ends with \usepackage{babel}. The Document body section starts with \begin{document} and ends with \end{document}. It contains the text 'Bienvenue en séance de TP !'.

L'option de inputenc doit coïncider avec l'encodage de l'éditeur (cf. table 1 page 40).

FIGURE 1 – Exemple de fichier source « minimal »

---

1. Certains de ces packages concernent des problèmes techniques, très éloignés de nos préoccupations d'utilisateur : le bon comportement est de les adopter comme postulat de départ pour bien rédiger en L<sup>A</sup>T<sub>E</sub>X (l'inverse, consistant à ne pas les utiliser sous prétexte que l'on n'en comprend pas toute l'utilité, serait une mauvaise idée). Ils sont brièvement expliqués en annexe (cf. § B.3.1).

### 3 Découverte de L<sup>A</sup>T<sub>E</sub>X

L'objet de ce paragraphe est d'utiliser les outils de base de L<sup>A</sup>T<sub>E</sub>X à travers des exemples, des remarques, des conseils et des exercices. L'accent est mis sur les outils mathématiques.

#### 3.1 Démarrage avec T<sub>E</sub>Xstudio (ex-TexMakerX)

L'éditeur de texte que nous utiliserons est T<sub>E</sub>Xstudio, disponible gratuitement et librement sous Linux, Mac OS et Windows ; c'est un éditeur « orienté L<sup>A</sup>T<sub>E</sub>X ».

0. Réglez la compilation rapide sur TEX → DVI → PS → PDF : dans le menu Options > Configurer TeXstudio > Compilation rapide, sélectionner le bouton radio LaTeX + DviPs + Ps2Pdf + Afficheur PDF.
1. Recopiez l'exemple de fichier source proposé à la figure 1, en remplaçant<sup>2</sup> *<option à choisir>* par utf8 ; T<sub>E</sub>Xstudio peut vous simplifier la tâche : allez dans le menu Fichier > Nouveau à partir d'un modèle... et choisissez Article(French). Ces packages sont à utiliser systématiquement, sauf contre-ordre explicite.

Remarquons tout de suite qu'en général, une commande s'applique à un argument parfois accompagné d'une option :

- la commande commence par le caractère \ (antislash) ;
- l'argument(s) obligatoire(s) est mis entre {} ;
- l'option(s) est mise entre [] .

Ainsi, \usepackage[utf8]{inputenc} signifie que l'on charge un package : il s'agit de inputenc, utilisé avec l'option utf8.

2. Compilez (en tapant sur F2 ; ou en cliquant sur l'icône LATEX ; ou via LaTeX dans le menu Outils). Lors de la première compilation, il vous demande d'enregistrer le fichier : donnez lui le nom TP\_init.tex.
3. Prévisualisez (en tapant sur F3 ; ou en cliquant sur l'icône du lion jaune ; ou via Afficher DVI dans le menu Outils) : cette étape ouvre une fenêtre affichant le fichier DVI. Vous devriez alors observer à l'écran ce qui est écrit dans le corps de document. Laissez cette fenêtre ouverte : elle se mettra à jour dès que vous compilerez de nouveau (*i.e.* il sera inutile de recliquer sur F3).

Si le caractère accentué n'apparaît pas correctement, c'est qu'il y a un problème d'encodage : reportez-vous à la note de bas de page n° 2.

4. Obtenez un PDF ; il y a deux manières de faire<sup>3</sup> : soit « par PostScript » (en cliquant sur F1<sup>4</sup>), soit « PDF direct » (en cliquant sur F6, puis sur F7 pour afficher).

---

2. Cela occulte un détail technique : la notion d'encodage (cf. § B.3.2). Retenir que si plusieurs personnes travaillent sur un même fichier .tex, il faut qu'elles règlent leur éditeur sur le *même* encodage lorsqu'elles modifient ce fichier (cf. § B.1.1 pour T<sub>E</sub>Xstudio). Retenir aussi que l'option du package inputenc et l'encodage sur lequel est réglé l'éditeur doivent coïncider (cf. table 1 page 40) !

3. Ce choix importe peu pour l'instant ; néanmoins, ce ne sera pas toujours le cas (cf. § B.1.3).

4. À condition d'avoir paramétré la compilation rapide comme suggéré ci-dessus ; sinon, cliquez successivement sur F2, F4 et F8, puis sur F7 pour afficher.

### 3.2 Composition simple

Présentons ici quelques outils de mise en forme dont on a l'habitude dans les logiciels de traitement de texte.

Commençons par deux exemples :

Voici un exemple simple pour voir comment s'utilisent, avec `\LaTeX{}`, la mise en `\textbf{gras}`, les formes `\textit{italique}` ou `\textsc{petites capitales}` et les familles `\texttt{type machine}` ou `\textsf{sans empattement}`.

On peut aussi changer la taille des fontes en `{\small petit}`, `{\tiny minuscule}`, `{\large grand}`, `{\huge énorme}`, etc.

Voici un exemple simple pour voir comment s'utilisent, avec LATEX, la mise en **gras**, les formes *italique* ou PETITES CAPITALES et les familles type **machine** ou **sans empattement**.

On peut aussi changer la taille des fontes en petit, minuscule, grand, énorme, etc.

Quelques remarques s'imposent :

- on n'a pas utilisé de la même manière les commandes dans les deux exemples ci-dessus : dans le premier, les commandes s'appliquent à un argument ; dans le second, elles appliquent le changement à tout ce qui suit : il faut alors les isoler pour cibler leur action ;
- les accolades `{}` permettent d'isoler un *groupe* afin que certaines commandes ne s'appliquent pas à tout le reste du document (`\tiny` par exemple) ;
- les caractères `\` et `{}` ne s'affichent pas à l'écran mais sont *interprétés* par LATEX<sup>5</sup> ;
- les accents, s'ils ne sont pas sur le clavier, sont codés à l'aide de `\'` ou `\`` ou `\^`, etc. (`\'e` donnera un « é ») ; la cédille s'obtient avec `\c{c}` (`\c{C}` donnera un « Ç ») ;
- pour passer à la ligne (c'est-à-dire débuter un nouveau paragraphe), on utilisera *systématiquement* la touche « entrée » du clavier deux fois<sup>6</sup> ;
- on utilisera `\emph{}` pour mettre en valeur du texte, et non pas `\textit{}`.

Deux autres exemples :

Pourquoi recommander la commande dite d'`\emph{emphase}` à celle d'`\textit{italique}`? Parce que dans une phrase en italique, l'`\emph{emphase}` ressort mais pas l'`\textit{italique}` !

Pourquoi recommander la commande dite d'*emphase* à celle d'*italique*? Parce que dans une phrase en *italique*, l'*emphase* ressort mais pas l'*italique* !

5. Il existe dix de ces caractères, dits spéciaux ou réservés ; cf. [mpg, § 2.1.1] pour en voir la liste et savoir les afficher à l'écran quand même.

6. L'utilisation de la commande `\\"` pour cet usage est à proscrire, sauf dans l'énoncé d'un titre.

**Conseil.** Pour mettre en valeur, proscrire l'usage du souligné (`\underline{}`) et éviter celui du gras : utiliser plutôt la commande `\emph{}`.

Ce mot est \tiny minuscule mais tout ce qui suit aussi car il n'est pas dans un groupe.

Ce mot est minuscule mais tout ce qui suit aussi car il n'est pas dans un groupe.

### 3.3 Mathématiques 1

L<sup>A</sup>T<sub>E</sub>X est prévu pour rédiger des formules mathématiques ; il existe deux modes mathématiques :

- le **mode en ligne**, utilisé pour insérer des symboles ou des formules (souvent courtes) dans une ligne de texte, au cours d'une phrase. Il s'obtient via la syntaxe `$...$` ;
- le **mode hors-texte**, permettant d'écrire une formule sur une ligne à part, centrée, au moyen de la syntaxe `\[ ... \]`<sup>7</sup>.

Considérons tout de suite des exemples :

Soit la fonction `$f$`, définie par `$f(x)=2x$`.

Soit la fonction  $f$ , définie par  $f(x) = 2x$ .

Soit la fonction `$f$`, définie par `\[ f(x)=2x^2. \]`

Soit la fonction  $f$ , définie par  $f(x) = 2x^2$ .

Soit la fonction `\[ f_n :`  
`x\mapsto\alpha^2\sqrt{\pi}+\frac{1}{2}\sum_{i=1}^n\ln x.` `\]`

Soit la fonction  

$$f_n : x \mapsto \alpha^2 \sqrt{\pi} + \frac{1}{2} \sum_{i=1}^n \ln x.$$

La fonction `\zeta` de Riemann satisfait `\[\zeta(s)=\sum_{n=1}^{+\infty}\frac{1}{n^s}=\prod_{p\in\mathcal{P}}\frac{1}{1-p^{-s}}.\]`

La fonction  $\zeta$  de Riemann satisfait  

$$\zeta(s) = \sum_{n=1}^{+\infty} \frac{1}{n^s} = \prod_{p \in \mathcal{P}} \frac{1}{1 - p^{-s}}.$$

Une différence fondamentale entre ces deux modes est que le mode hors-texte prend la place — si nécessaire — d'afficher les grands symboles, ce que ne fait pas le mode en ligne ; voici un exemple : « Soit  $f$  définie par  $f(x) = \sum_{i=1}^n \int_0^i f_i \left( \frac{\ln x + \alpha}{\sin x + 1} \right)$ , avec  $\alpha \in \mathbb{R}$ . » à comparer avec « Soit  $f$  définie par

$$f(x) = \sum_{i=1}^n \int_0^i f_i \left( \frac{\ln x + \alpha}{\sin x + 1} \right),$$

avec  $\alpha \in \mathbb{R}$ . ».

---

7. La commande `$$...$$` est à proscrire en L<sup>A</sup>T<sub>E</sub>X, même si beaucoup de gens l'utilisent.

*Certaines habitudes ou conventions sont à prendre lorsque l'on rédige des mathématiques ; les exercices suivants ont pour but de vous y entraîner. Ils vous permettront aussi de découvrir quelques spécificités du mode mathématique par rapport au mode texte.*

À l'aide des exemples ci-dessus, des commandes proposées par  $\text{\TeX}^{\text{studio}}$  et éventuellement d'une liste de symboles<sup>8</sup>, trouvez comment obtenir les résultats suivants ; vous ferez en priorité les exercices n°s 1, 2, 3, 5, 8, 10, 13, 14, 15 et 19. (On ne demande pas de reproduire l'encadrement ni les éventuels indices donnés entre parenthèses.)

*Exercice 1.* Pour  $n$  entier naturel non nul, on pose  $u_0 = 0$  et  $u_n = u_{n-1} + n$ .

Alors

$$\forall n \geq 0, u_n = \frac{n(n+1)}{2}.$$

*Exercice 2.* La formule de Stirling exprime, pour  $n$  grand, que

$$n! \sim C n^n \sqrt{n} \exp(-n),$$

où  $C = \sqrt{2\pi}$ . Cette constante peut se calculer en utilisant la formule de Wallis, que l'on trouve grâce aux intégrales éponymes :

$$\forall n \in \mathbb{N}, I_n = \int_0^{\frac{\pi}{2}} (\sin x)^n dx.$$

*Exercice 3.* La fonction  $\Gamma: \mathbb{R}_+^* \rightarrow \mathbb{R}$ , définie par

$$\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-t} dt$$

et appelée “fonction Gamma (d’Euler)”, généralise la factorielle.

En effet,  $\forall n \in \mathbb{N}^*, \Gamma(n+1) = n!$ . On peut aussi montrer que

$$\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi},$$

en se ramenant à l’intégrale de Gauss  $I = \int_0^{+\infty} e^{-t^2} dt$  (par changement de variables), cette dernière valant  $\frac{\sqrt{\pi}}{2}$  (par exemple en considérant le carré de  $I$  et un passage en coordonnées polaires).

Nous venons d’illustrer quelques principes de base du mode mathématique :  $\text{\LaTeX}$  adapte la taille de certains caractères selon le mode mathématique choisi ; il est aussi capable de modifier leur apparence selon notre demande : italique mathématique pour les variables, romain pour les opérateurs, etc. Il connaît déjà certains opérateurs ( $\cos$ ,  $\ker$ ,  $\sup$ ,  $\exp$ , etc.) mais pas tous ; pour le « e » de la fonction exponentielle, on code  $\mathit{e}$ .

---

8.  $\text{\LaTeX}$  possède un très grand nombre de symboles (cf. § 3.6.3) ; pour le TP-cours, la liste distribuée en séance (issue de [4]) suffira largement.

**Conseil.** Pour écrire un opérateur, toujours utiliser la commande prédéfinie ; si elle n'existe pas, la créer de manière *adéquate*<sup>9</sup>.

Il y a \$a\$ tel que \$a=\max x\$ pour  
\$x\$ entier relatif et \$x+2<0\$.

Il y a a tel que \$a=\max x\$ pour  
x entier relatif et \$x+2<0\$.

bien → Il y a a tel que  $a = \max x$   
pour  $x$  entier relatif et  $x + 2 < 0$ .

mal → Il y a a tel que  $a = \max x$   
pour  $x$  entier relatif et  $x + 2 < 0$ .

Une autre particularité importante est que le mode mathématique est un *monde à part* du mode texte. Ainsi,

- les mises en forme habituelles (gras, italique, etc.) se font au moyen d'autres commandes : `\mathrm{}` pour passer en romain, `\mathbf{}` pour mettre en gras<sup>10</sup>, etc.  
De plus, les lettres calligraphiques, gothiques et ajourées sont disponibles :

Soit  $f \in \mathcal{C}$  et  
 $\sigma \in \mathfrak{S}_n$  pour  $n \in \mathbb{N}$ .  
 $\Sigma \in \mathfrak{S}_n$  pour  
 $N \in \mathbb{N}$ .

Soit  $f \in \mathcal{C}$  et  $\sigma \in \mathfrak{S}_n$  pour  $n \in \mathbb{N}$ .

- les blancs sont entièrement gérés par LATEX ; néanmoins, il a parfois du mal et il faut alors forcer les espaces : on a vu `\,` (espace fine) mais il y a aussi `\quad` (espace inter-mot classique), `\quad` (1 cadratin), `\!` (espace fine négative), etc.
- les accents sont obtenus autrement ; par exemple

$\hat{x}, \ddot{x}, \vec{n}, \acute{m}, \mathring{A}, \bar{f}, \tilde{f}$ , etc.  
 $\acute{m}, \mathring{A},$   
 $\bar{f}, \tilde{f}$ , etc.

Par contre, ces accents n'adaptent pas leur taille à la largeur des caractères accentués ; on dispose néanmoins de `\widehat{}`, `\widetilde{}` et de diverses flèches :

L'angle  $\hat{ABC}$  est moins  
joli que  $\widehat{ABC}$ .

L'angle  $A\hat{B}C$  est moins joli que  $\widehat{ABC}$ .

*Exercice 4* (avec `\mathcal{}`). Pour  $M \in \mathcal{M}_n(\mathbb{Z})$ ,

$$M \in \mathrm{GL}_n(\mathbb{Z}) \iff \det M = \pm 1.$$

Il ne faut pas confondre le mode texte et le mode mathématique :

*Exercice 5.* Considérons  $\phi, \Sigma, \hbar, \varepsilon$  et  $\ell$  des réels et  $(O, \vec{i}, \vec{j})$  un repère orthonormé.

9. Voir l'annexe B.4.

10. Cette commande ne peut pas mettre en gras tout type de symbole (cf. [5, § 8.5] pour y remédier).

Cependant, il arrive que l'on ait besoin d'écrire du texte à l'intérieur du mode mathématique ; il faut alors utiliser la commande `\text{}`, qui permet de faire comme si on était en mode texte<sup>11</sup>.

```
Pour tout $x>0$, \[ \Gamma(x)
\text{ est bien définie.}
\]
```

Pour tout $x > 0$ ,	Pour tout $x > 0$ , $\Gamma(x)$ est bien définie.
---------------------	--

*Exercice 6* (avec `\overrightarrow{}`). Écrivons le moment magnétique

$$\vec{M} = \frac{1}{2} \iiint_{\mathcal{V}} \vec{OP} \wedge \vec{j}(P) d\tau \quad (\mathcal{V} \text{ étant un volume}).$$

*Exercice 7* (avec `\overset{}`). L'exercice 3 peut aider au calcul de l'intégrale de Fresnel

$$\varphi \stackrel{\text{def}}{=} \int_0^{+\infty} \exp(ix^2) dx = \frac{\sqrt{\pi}}{2} \exp\left(\frac{i\pi}{4}\right)$$

en montrant, pour  $\alpha$  dans  $]0, 1[$ , que

$$J: \alpha \mapsto \int_{]0, +\infty[} t^{\alpha-1} e^{it} dt$$

vérifie

$$J(\alpha) = \Gamma(\alpha) e^{i\alpha \frac{\pi}{2}}.$$

*Exercice 8* (avec `\xrightarrow{}`). Si  $f \in L^1(\mathbb{R})$  alors sa transformée de Fourier, notée  $\hat{f}$ , est continue et vérifie (pour une définition bien choisie)

$$\hat{f}(x) \xrightarrow{x \rightarrow \pm\infty} 0 \quad \text{et} \quad \left\| \frac{\hat{f}}{2\pi} \right\|_\infty \leq \|f\|_1.$$

*Exercice 9* (avec `\ldots`, `\displaystyle`). Soit  $a_1, \dots, a_k \in \mathbb{N}^*$ . Supposons les  $a_i$  premiers entre eux dans leur ensemble (pour  $i \in \{1, \dots, k\}$ ) et notons, pour  $n \geq 1$ ,  $u_n$  le nombre de  $k$ -uplets  $(x_1, \dots, x_k) \in \mathbb{N}^k$  tels que  $\sum_{i=1}^k a_i x_i = n$ .

Alors

$$u_n \underset{+\infty}{\sim} \frac{1}{a_1 a_2 \cdots a_k} \frac{n^{k-1}}{(k-1)!}.$$

**Conseil.** Ne pas abuser du `\displaystyle` (car il modifie la dimension de l'interligne : cela peut donc vite devenir moche!). Prendre garde à la syntaxe : `\displaystyle blabla` est correct (alors que `\displaystyle{blabla}` n'a aucun sens).

---

11. Certains ouvrages proposent l'utilisation de `\mbox{}` ou de `\mathrm{}` pour cet usage : cela ne produit pas toujours l'effet désiré, donc c'est à proscrire.

### 3.4 Environnements

Voyons maintenant un outil utile pour réaliser des objets plus évolués, comme des tableaux, des équations alignées, des énumérations, etc. : ce sont les environnements. Nous aborderons plus loin (cf. § 3.6) ceux spécifiques au mode mathématique.

Leur utilisation est simple :

```
\begin{<nom de l'environnement>}
:
\end{<nom de l'environnement>}
```

Citons les plus courants :

- `bfseries`, `sffamily`, etc. qui permettent de modifier la mise en forme des fontes, *i.e.* mise en gras, en sans serif, etc. (cf. § 3.2) ;
- `small`, `Large`, etc. qui permettent de modifier la taille des fontes (cf. § 3.2) ;
- `center`, qui permet de centrer une partie de texte horizontalement ;
- `itemize`, `enumerate` et `description`, qui servent à mettre en forme des listes ;
- `tabular`, qui s'utilise pour créer des tableaux au sein du mode texte (en mode mathématique, c'est son homologue `array` qui doit être utilisé).

Expliquons à l'aide d'exemples le fonctionnement de quelques environnements :

<pre>\begin{itemize} \item premièrement ; \item deuxièmement. \end{itemize}</pre>	<ul style="list-style-type: none"> <li>– premièrement ;</li> <li>– deuxièmement.</li> </ul>									
<pre>\begin{enumerate} \item Primo. \item Secundo. \end{enumerate}</pre>	<ol style="list-style-type: none"> <li>1. Primo.</li> <li>2. Secundo.</li> </ol>									
<pre>\begin{tabular}{ rl c } \hline right &amp; left &amp; center \\ Xavier &amp; Martine &amp; François\\ \hline Résistance &amp; Inductance &amp; Capacité\\ \hline \end{tabular}</pre>	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th style="padding: 2px;">right</th> <th style="padding: 2px;">left</th> <th style="padding: 2px;">center</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">Xavier</td> <td style="padding: 2px;">Martine</td> <td style="padding: 2px;">François</td> </tr> <tr> <td style="padding: 2px;">Résistance</td> <td style="padding: 2px;">Inductance</td> <td style="padding: 2px;">Capacité</td> </tr> </tbody> </table>	right	left	center	Xavier	Martine	François	Résistance	Inductance	Capacité
right	left	center								
Xavier	Martine	François								
Résistance	Inductance	Capacité								

Explications sur la syntaxe de `tabular` : le `\\"` indique une fin de ligne, le `&` sépare les colonnes, le `\hline` trace un trait horizontal entre deux lignes. Quant à l'argument de l'environnement (ici `|rl|c|`), il indique deux choses : l'alignement horizontal dans chaque colonne (via la lettre `r`, `l` ou `c`) et l'éventuel ajout d'un filet vertical entre deux colonnes (en insérant le caractère `|`).

*Exercice 10* (on ne demande pas de reproduire l'espacement plus aéré des itemize de cet exercice). Pour avoir la valeur d'une intégrale, deux moyens existent :

1. Calculer sa valeur exacte. Différents outils peuvent être utilisés, en particulier :
  - la règle des invariants de Bioche :
    - si  $-x \leftarrow x$  est un invariant, on utilise  $u = \cos x$ ,
    - si c'est  $\pi - x \leftarrow x$ , on utilise  $u = \sin x$ ,
    - si c'est  $\pi + x \leftarrow x$ , on utilise  $u = \tan x$ ;
  - le théorème des résidus ;
  - l'égalité de Plancherel-Parseval.
2. Calculer une valeur approchée. On distingue deux types de méthodes :
  - (a) des méthodes déterministes, contenant :
    - i. les méthodes de Newton-Cotes,
    - ii. les méthodes de Gauss ;
  - (b) une méthode probabiliste : la méthode de Monte-Carlo.

*Exercice 11.* À savoir sur les méthodes de quadrature :

Méthode	Ordre
Rectangles à gauche	0
Rectangles à droite	0
Point milieu	1
Trapèzes	1
Simpson	3

*Exercice 12* (sans package additionnel mais en utilisant `\multicolumn{2}{c}{}`). Voici un parallèle entre des méthodes de calcul approché d'intégrales et des schémas de résolution approchée d'équations différentielles ordinaires :

Méthode de quadrature		Schéma EDO
Nom	Ordre	Nom
Rectangles à gauche	0	Euler explicite
Rectangles à droite	0	Euler implicite
Point milieu	1	Euler modifié
Trapèzes	1	Crank-Nicolson
Simpson	3	Runge-Kutta d'ordre 4 (RK4)

Il existe beaucoup d'autres possibilités d'agencer les tableaux : avoir une ligne horizontale ne s'étendant que sur quelques colonnes, modifier la hauteur des cellules, fusionner des cellules, modifier le symbole affiché entre chaque colonne, etc. Enfin, des packages étendent encore ces possibilités... on renvoie à [mpg, § 7] et à [B&C, § 5].

### 3.5 Sectionnement et références croisées

*LaTEX est capable de gérer tout seul la mise en forme des titres de paragraphe et leur numérotation ; on utilise pour cela les commandes de sectionnement. Il est alors possible de faire des renvois à ces paragraphes : on parle de références croisées.*

Pour le sectionnement, les commandes `\chapter{}`, `\section{}`, `\paragraph{}`, etc. (cf. [mpg, § 3.4]) sont à disposition ; quant à leur version étoilée (`\section*{}`, etc.), elle supprime la numérotation. L'exemple suivant illustre leur fonctionnement (sous la classe `article`, `\chapter{}` n'est pas disponible).

```
\section{Première section}
\subsection{Sous-section}
\section{Deuxième section}
\section*{Conclusion}
```

<b>1 Première section</b>		2 Deuxième section
<b>1.1 Sous-section</b>		
<b>Conclusion</b>		

*Exercice 13.* Recopier l'exemple de code source ci-dessus puis compiler. Ajouter ensuite une section avant la première section et compiler de nouveau. Qu'observez-vous au niveau de la numérotation ?

Les références croisées permettent de référencer automatiquement les commandes de sectionnement : on place une étiquette juste après le titre (au moyen de `\label{}`) et on s'y réfère *n'importe où* dans le document (au moyen de `\ref{}` pour avoir le numéro du sectionnement et de `\pageref{}` pour avoir la page où il débute). Pour écrire ce polycopié, nous avons tapé

```
\section{Découverte de \LaTeX}
\label{sec-Decouv}
```

pour coder le titre de section en page 5, d'où l'exemple suivant :

La section~ <code>\ref{sec-Decouv}</code> débute page~ <code>\pageref{sec-Decouv}</code> .		La section 3 débute page 5.

**Conseil.** Prendre l'habitude de « classifier » ses références à l'aide d'un mot-clé (suivi d'un *tiret*<sup>12</sup>) : `\label{<mot-clé>-truc}` en remplaçant, par exemple, `<mot-clé>` par `sec` pour une section, `subsec` pour une sous-section, `eq` pour une équation numérotée<sup>13</sup>, etc.

12. Éviter l'usage du deux-points (en particulier si `babel/frenchb` est chargé), même si certains ouvrages le proposent.

13. Comme nous le verrons par la suite, plein d'objets (numérotés) peuvent être référencés : notes de bas de page, environnements `table` et `figure`, théorèmes, ...

**Conseil.** L'espace insécable (codé avec le caractère `\~`) permet d'empêcher un retour à la ligne à l'endroit utilisé ; prendre l'habitude d'en faire précéder chaque `\ref{}`.

*Exercice 14.* Reprendre l'exercice 13 et ajouter une étiquette à la première section, en y faisant référence dans la deuxième ; ensuite, compiler une (seule) fois : qu'observez-vous ? Compiler une seconde fois et trouver une explication.

L'intérêt de la gestion automatique des références croisées par L<sup>A</sup>T<sub>E</sub>X est majeur : en effet, si au cours de la rédaction un numéro de sectionnement ou un numéro de page doit changer (suite à une modification de texte avant), alors les commandes `\ref{}` et `\pageref{}` se mettent à jour toutes seules<sup>14</sup>.

## 3.6 Mathématiques 2

*Voyons maintenant comment enrichir le mode mathématique en utilisant la structure d'environnement et les références croisées. Ce paragraphe répondra aux besoins suivants : faire des tableaux en mode mathématique, créer des matrices, numérotter des formules, afficher des successions d'équations ; écrire proprement des énoncés de théorèmes numérotés ; savoir où trouver des informations supplémentaires.*

### 3.6.1 Agencement de formules, tableaux et matrices

1. L'environnement `equation` permet d'écrire *une* ligne en mode hors-texte (comme la syntaxe `\[ \dots \]`) mais, en plus, il la numérote :

Soit la fonction `$f$`, définie  
par `\begin{equation}`  
`f(x)=2x.`  
`\end{equation}`

Soit la fonction  $f$ , définie par  
$$f(x) = 2x. \quad (1)$$

2. L'environnement `array` permet de réaliser des tableaux au sein du mode mathématique, mais il faut au préalable être entré dans ce mode. Son utilisation est très similaire à l'environnement `tabular` ; comme ce dernier, il ne fait pas débuter une nouvelle ligne : par défaut il est centré verticalement sur la ligne en cours, mais on peut changer ce placement au moyen des options `t` ou `b`.

Soit `$\begin{array}{|c|c|}`  
`\hline`  
`f(x,t)& g(x,y)\hline`  
`t\sqrt[3]{x} &`  
`\Re\text{e}(x+iy)\hline`  
`\end{array}$ un beau tableau.`  
Soit `$\begin{array}{[b]|c|c|}`  
`...$ un autre tableau.`

Soit 

$f(x, t)$	$g(x, y)$
$t\sqrt[3]{x}$	$\Re e(x + iy)$

 un beau ta-bleau.  
Soit 

$f(x, t)$	$g(x, y)$
$t\sqrt[3]{x}$	$\Re e(x + iy)$

 un autre ta-bleau.

---

14. Après deux compilations tout de même.

3. Pour produire une matrice, on remplacera avantageusement `array` par un environnement mieux adapté : `matrix`, `pmatrix`, `bmatrix`, `vmatrix`, etc.

```
Soit $L=\begin{pmatrix} 2 & -1 & \\ -1 & 2 & -1 \\ & -1 & 2 \end{pmatrix}$ une matrice.
```

$$\text{Soit } L = \begin{pmatrix} 2 & -1 & \\ -1 & 2 & -1 \\ & -1 & 2 \end{pmatrix} \text{ une matrice.}$$

4. Pour écrire des équations sur plusieurs lignes, là encore on remplacera avantageusement `array` par un environnement mieux adapté ; citons en particulier

- `gather`, pour empiler des équations (elles seront numérotées) :

```
On a \begin{gather}
\Delta V = -\nabla \cdot (\nabla E), \\
\Delta V + \frac{\rho}{\varepsilon_0} = 0.
\end{gather}
```

$$\begin{aligned} \text{On a} \\ \Delta V &= -\vec{\nabla} \cdot (\nabla \vec{E}), & (2) \\ \Delta V + \frac{\rho}{\varepsilon_0} &= 0. & (3) \end{aligned}$$

- `align` pour empiler des équations en les alignant verticalement<sup>15</sup> (elles seront numérotées) :

```
On a aussi \begin{align}
(a+b)^2 &= (a+b)(a+b) \\
&= a^2 + 2ab + b^2.
\end{align}
```

$$\begin{aligned} \text{On a aussi} \\ (a+b)^2 &= (a+b)(a+b) & (4) \\ &= a^2 + 2ab + b^2. & (5) \end{aligned}$$

On peut faire plusieurs colonnes d'équations (en rajoutant des `&`, généralement placées avant le symbole de relation ; cf. § B.5) ;

- `multline` pour des formules très longues qui tiennent sur plusieurs lignes ; la première est alignée à gauche, la dernière à droite et celles du milieu sont centrées :

```
On a \begin{multline}
55 = 1+2+3 \\
+4+5+6+7 \\
+8+9+10.
\end{multline}
```

$$\begin{aligned} \text{On a} \\ 55 &= 1 + 2 + 3 \\
&+ 4 + 5 + 6 + 7 \\
&+ 8 + 9 + 10. & (6)
\end{aligned}$$

Ces trois environnements (ainsi que `equation`) admettent une forme étoilée qui supprime la numérotation :

---

15. Proscrire l'usage de `eqnarray`.

```
On a enfin \begin{align*}
(a+b)^2 &= (a+b)(a+b) \\
&= a^2+ab+ba+b^2 \\
&= a^2+2ab+b^2 \text{ \& \text{car}\dots} \\
\end{align*}
```

$$\begin{aligned} (a+b)^2 &= (a+b)(a+b) \\ &= a^2 + ab + ba + b^2 \\ &= a^2 + 2ab + b^2 \quad \text{car...} \end{aligned}$$

5. L'environnement `split`, un peu à part : il permet, alors que l'on est *déjà* en mode mathématique hors-texte, de scinder une équation (qui est trop longue pour tenir sur une seule ligne) ; il nécessite donc d'être placé dans un environnement mathématique et il n'apporte pas de numérotation. Donnons deux exemples d'utilisation.

(a) On souhaite un alignement vertical (ce que ne procure pas `multline`) :

```
On a encore \begin{equation}
\begin{split}
55 &= 1+2+3+4+5 \\
&\quad +6+7+8+9+10.
\end{split}
\end{equation}
```

$$\begin{aligned} 55 &= 1 + 2 + 3 + 4 + 5 \\ &\quad + 6 + 7 + 8 + 9 + 10. \end{aligned} \tag{7}$$

(b) On est déjà dans un environnement mathématique hors-texte (ce qui empêche l'usage de `multline`) :

```
Les équations d'Euler sont
\begin{gather}
\frac{\partial n}{\partial t} + \nabla_x \cdot (nu) = 0, \\
\begin{split}
\frac{\partial (nu)}{\partial t} + \nabla_x \cdot (nu \otimes u) \\
&\quad + \nabla p = 0,
\end{split}
\\
\frac{\partial E}{\partial t} + \nabla_x \cdot ((E+p)u) = 0.
\end{gather}
```

$$\begin{aligned} \frac{\partial n}{\partial t} + \nabla_x \cdot (nu) &= 0, \tag{8} \\ \frac{\partial (nu)}{\partial t} + \nabla_x \cdot (nu \otimes u) &+ \nabla p = 0, \tag{9} \\ \frac{\partial E}{\partial t} + \nabla_x \cdot ((E+p)u) &= 0. \tag{10} \end{aligned}$$

Pleins d'autres outils existent, par exemple :

- l'environnement `cases` permet des « constructions par cas » (cf. exercice 18) ;
- la commande `\intertext{}` permet d'insérer une ligne de texte entre deux lignes de `align` sans briser l'alignement vertical (cf. § B.5) ;
- la commande `\notag` permet de supprimer la numérotation sur une ligne particulière (cf. § B.5).

*Exercice 15* (avec `\eqref{}`). On a l'identité remarquable, numérotée (11) :

$$\forall a, b \in \mathbb{N}, \quad (a + b)^2 = a^2 + 2ab + b^2. \quad (11)$$

*Exercice 16* (avec `\vdots`). Pour tout  $(a_1, \dots, a_n) \in \mathbb{K}^n$ , le déterminant de Vandermonde est

$$V(a_1, \dots, a_n) = \begin{vmatrix} 1 & a_1 & \cdots & a_1^{n-1} \\ 1 & a_2 & \cdots & a_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & \cdots & a_n^{n-1} \end{vmatrix} = \prod_{1 \leq i < j \leq n} (a_j - a_i).$$

*Exercice 17* (avec `\underbrace{...}`). Soit  $(u_n)_{n \in \mathbb{N}}$  définie par  $u_0 \in ]0, \frac{\pi}{2}]$  et  $\forall n \in \mathbb{N}, u_{n+1} = \sin(u_n)$ . Alors on peut montrer successivement que :

$$\begin{aligned} \lim_{n \rightarrow +\infty} u_n &= 0, \\ u_n &\underset{+\infty}{\sim} \sqrt{\frac{3}{n}}, \\ u_n &\underset{+\infty}{=} \sqrt{\frac{3}{n}} - \underbrace{\frac{3\sqrt{3}}{10} \frac{\ln n}{n\sqrt{n}}}_{= \mathcal{O}\left(\frac{\ln n}{n\sqrt{n}}\right)} + o\left(\frac{\ln n}{n\sqrt{n}}\right). \end{aligned}$$

*Exercice 18* (avec `cases`). Soit  $f: \mathbb{R} \setminus \{0\} \longrightarrow \mathbb{R}$ . On peut prolonger  $f$  par continuité en  $\sin_c$  définie par  $\sin_c(x) = \begin{cases} f(x) & \text{si } x \in ]-\infty, 0[ \cup ]0, +\infty[ \\ 1 & \text{sinon} \end{cases}$ .

### 3.6.2 Théorèmes et énoncés similaires

LATEX permet de mettre en forme des théorèmes, propositions, remarques, etc., de les numérotter automatiquement et d'y faire référence très simplement, tout cela grâce aux environnements de type theorem.

Avant toute chose, chargeons le package `amsthm`<sup>16</sup>; ensuite, cela se fait en deux temps :

1. Définition et configuration d'un type d'énoncé dans le préambule (c'est ici que l'on choisit divers paramètres : quel nom portera-t-il ? sera-t-il numéroté à la suite ou en parallèle d'un autre type d'énoncé ? sera-t-il numéroté par section ?).
2. Utilisation au cours du document.

16. Les capacités de LATEX sont très limitées sans ce package ; à l'inverse, pour davantage de possibilités, utiliser en plus le package `thmtools` (cf. § 3.6.3).

Ainsi, en ajoutant ces deux lignes dans le préambule,

```
\newtheorem{prop}{Proposition}
\newtheorem{cor}[prop]{Corollaire}
```

on obtient les deux exemples suivants :

```
\begin{prop}
Pour tout  $\alpha \in \mathbb{R}$ ,
la série  $\sum_{n \geq 1} \frac{1}{n^\alpha}$  converge
si et seulement si  $\alpha > 1$ .
\end{prop}
```

```
\begin{cor}[Séries de Bertrand]
\label{cor-Bertrand}
Pour tout  $(\alpha, \beta) \in \mathbb{R}^2$ ,
 $\sum_{n \geq 2} \frac{n^\alpha (\ln n)^\beta}{n^\alpha}$  converge
si et seulement si
 $\alpha > 1$  ou  $\alpha = 1$  et  $\beta > 1$ .
\end{cor}
```

**Proposition 1.** Pour tout  $\alpha \in \mathbb{R}$ , la série

$$\sum_{n \geq 1} \frac{1}{n^\alpha}$$

converge si et seulement si  $\alpha > 1$ .

**Corollaire 2** (Séries de Bertrand). Pour tout  $(\alpha, \beta) \in \mathbb{R}^2$ ,

$$\sum_{n \geq 2} \frac{1}{n^\alpha (\ln n)^\beta}$$

converge si et seulement si  $(\alpha > 1)$  ou  $(\alpha = 1 \text{ et } \beta > 1)$ .

(Dans ce dernier exemple un titre a été donné à l'énoncé, via des crochets ; un `\label{}`, lui aussi facultatif, a été donné pour pouvoir faire référence au corollaire plus tard.)

*Exercice 19.* Retirer le `[prop]` dans la définition de `cor` et observer que la numérotation (des corollaires) a changé : ils sont dorénavant numérotés indépendamment des propositions.

Pour rédiger une preuve, l'environnement `proof` peut être utilisé :

```
\begin{proof}
Étude de cas et comparaison
série/intégrale pour  $\alpha > 1$ .
\end{proof}
```

*Démonstration.* Étude de cas et comparaison série/intégrale pour  $\alpha > 1$ .  $\square$

```
\begin{proof}[Preuve du
corollaire-\ref{cor-Bertrand}]
\end{proof}
```

*Preuve du corollaire 2.*  $\square$

Pour varier l'affichage des énoncés (*i.e.* avoir par exemple une mise en forme pour les définitions qui soit différente de celle des théorèmes, propositions, lemmes, ...), on dispose de trois styles de composition : `plain`, `definition` et `remark` ; il est aussi possible de ne pas numérotter un type d'énoncé. Voir l'exercice 20 ci-après pour la syntaxe.

*Exercice 20.* Charger `amsthm` (au moyen de `\usepackage{amsthm}` placée avant `\usepackage{babel}`) et écrire les quelques lignes suivantes dans le préambule.

```
\theoremstyle{plain}
\newtheorem{thm}{Théorème}

\theoremstyle{definition}
\newtheorem{defi}[thm]{Définition}

\theoremstyle{remark}
\newtheorem{exe}{Exemple}
\newtheorem*{rmq}{Remarque}
```

Ensuite, obtenir le résultat présenté dans la figure 2 (ci-dessous), largement emprunté de [1]. Que se passe-t-il si la 2<sup>e</sup> ligne de code ci-dessus est remplacée par `\newtheorem{thm}{Théorème}[section]` ?

**Définition 1** (Base hilbertienne). Soit  $(H, \langle \cdot, \cdot \rangle)$  un espace de Hilbert. On dit qu'une famille  $(e_i)_{i \in I}$  est une *base hilbertienne* de  $H$  si elle est orthogonale, normée et totale.

**Définition 2.** Un espace métrique  $(E, d)$  est dit *séparable* si tout ouvert non vide de  $E$  contient au moins un point d'une partie dénombrable de  $E$ .

*Exemple 1.*  $\mathbb{R}^n$  est séparable.

**Théorème 3** (Caractérisation). Soit  $(H, \langle \cdot, \cdot \rangle)$  un espace de Hilbert séparable et  $(e_n)_{n \in \mathbb{N}}$  une famille orthonormée de  $H$ . Les propriétés suivantes sont équivalentes :

1. La famille orthonormée  $(e_n)_n$  est une base hilbertienne.
2. Pour tout  $x \in H$ ,

$$x = \sum_{n=0}^{+\infty} \langle x, e_n \rangle e_n \quad (\text{égalité de Parseval}).$$

3. On a

$$(e_n, \quad n \in \mathbb{N})^\perp = \{0\}.$$

*Remarque.* Le théorème 3 s'étend au cas des espaces de Hilbert non séparables.

FIGURE 2 – Texte à reproduire (sans l'encadrement) dans l'exercice 20

**Conseil.** Pour faire référence à un énoncé numéroté, *toujours* utiliser `\label{}` et `\ref{}`; ne jamais faire une numérotation « en dure »<sup>17</sup>.

---

<sup>17</sup>. Autrement dit (cf. figure 2), *ne jamais écrire « Le théorème~3 » dans son fichier source mais « Le théorème~\ref{thm-caractBaseHilb} » (avec un `\label{thm-caractBaseHilb}` bien placé).*

### 3.6.3 Pour aller plus loin : vers l'infini et au-delà !

*Donnons ici quelques références commentées apportant des informations pour aller plus loin en mathématiques avec L<sup>A</sup>T<sub>E</sub>X.*

#### Documents généraux :

- Consulter [mpg, § 4] pour trouver plus ou moins les mêmes outils que dans ce polycopié, mais rédigés sous forme de cours ; on y trouvera à coup sûr les informations nécessaires pour traiter les exercices de ce TP-cours.
- Consulter [B&C, § 4] pour trouver une présentation riche et pédagogique des possibilités de base en mathématiques ; ce livre, d'une grande qualité, amène progressivement le débutant vers un très bon niveau d'utilisateur.
- Consulter [2], qui est une tentative de rassembler toutes les possibilités (commandes, options, packages, etc.) pertinentes pour faire des mathématiques. Il s'agit d'un document très riche mais de haut niveau ; en particulier, savoir que les parties I (traitant de L<sup>A</sup>T<sub>E</sub>X sans `amsmath`) et III (traitant de T<sub>E</sub>X) dépassent largement le cadre de ce TP-cours d'initiation.

#### Packages généraux :

- Consulter [3] (documentation de `amsmath`) pour exploiter à fond ses possibilités. En particulier : § 3 (dont la table 3.1) pour d'autres environnements sur l'agencement des formules, § 4 et 5 pour diverses notations mathématiques (coefficients binomiaux, fractions continuées, parenthésages, modulo, ...), § 8 pour des diagrammes commutatifs *simples* (*i.e.* sans diagonale<sup>18</sup>).
- Se servir de `mathtools`. Ce package peut avantageusement remplacer `amsmath` ; en fait, il le charge<sup>19</sup>, corrige certains de ses bugs et le complète ( $e_n \xrightarrow{n \rightarrow \infty} 0$ ).

#### Packages ciblés :

- Se servir de `thmtools`. Il s'utilise *conjointement* avec `amsthm`<sup>20</sup>, en proposant une interface plus facile à utiliser et en étendant certaines de ses possibilités (passage à la ligne après l'affichage du titre numéroté, affichage d'une liste des théorèmes, ...). Des exemples figurent dans [B&C, § 4.8].

#### Symboles

- Avoir la liste distribuée en cours (issue de [4]) ; elle regroupe les plus courants.
- Avoir [5], qui est une liste exhaustive (§ 3 pour les mathématiques).
- Utiliser le site Internet Detexify [6] qui permet, en dessinant un symbole à la souris, de trouver la commande qui lui correspond !

---

18. Pour des diagrammes plus compliqués, mais au détriment d'une manipulation moins facile, voir les outils graphiques de L<sup>A</sup>T<sub>E</sub>X (cf. § 4.4).

19. Toutes les fonctionnalités d'`amsmath` sont donc disponibles.

20. Ou avec `ntheorem`, package qui peut s'utiliser à la place de `amsthm` : il propose plus de fonctionnalités (`ntheorem` est à charger après `babel`).

## 4 Fonctionnalités plus avancées

### 4.1 Mise en page et structure du document

*Présentons ici des commandes et fonctionnalités permettant d'améliorer la présentation d'un document (se reporter à [mpg] et à [B&C] pour davantage d'informations).*

**Titre d'un document et page de garde :** il est possible de formaliser l'affichage du titre d'un document ; cela se fait en deux temps :

1. Donner les informations à L<sup>A</sup>T<sub>E</sub>X (intitulé, auteur(s), date), de préférence dans le préambule.
2. Demander à L<sup>A</sup>T<sub>E</sub>X d'afficher le titre : il suffit pour cela d'insérer la commande `\maketitle` à l'endroit voulu (généralement au début du document!).

Par exemple, en ayant écrit dans le préambule

```
%Intitule  
\title{Initiation \LaTeX{}\thanks{ENS Cachan Bretagne}}  
%Auteurs  
\author{\bsc{Nom1}\\\ Univ.\~1 \and  
       \bsc{Nom2}\thanks{\texttt{prenom.nom@ens-cachan.org}}\\\ Univ.\~2}  
%Date  
\date{9 janvier 2012}
```

la commande `\maketitle` produit un titre proche de celui de ce polycopié. Observez deux autres moyens de gérer la date en supprimant la dernière ligne ci-dessus puis en la remplaçant par `\date{}`.

Il existe l'environnement `titlepage` pour personnaliser entièrement la page de titre.

**Table des matières :** la commande `\tableofcontents` permet d'afficher la table des matières à l'endroit où elle est appelée.

*Exercice 21.* En reprenant l'exercice 13, insérer une table des matières au tout début. Compiler, observer ; compiler et observer de nouveau. Ensuite, faire chacune des étapes suivantes, en observant à chaque fois le résultat de la (des) compilation(s) :

1. Modifier un titre de section et ajouter une section.
2. Ajouter `\section[Titre court]{Titre long, vraiment long}`.
3. Placer `\addcontentsline{toc}{section}{\protect Conclusion}` juste après `\section*{Conclusion}`.<sup>21</sup>

**Notes de bas de page :** elles s'utilisent avec `\footnote{}` comme ceci<sup>22</sup>. Observez que dans un tableau cette commande ne fonctionne pas et trouvez comment faire<sup>23</sup>.

21. Explication : cette nouvelle commande permet d'insérer une ligne dans la `toc` (*table of contents*), contenant un titre de niveau `section` qui s'appelle `Conclusion`. `\protect` permet de ... protéger  $\odot$ .

22. Je suis une note de bas de page ; je commence par une majuscule et me termine par un point.

23. Solution : grâce à `\footnotemark` et `\footnotetext{...}`...

**Bibliographie** : nous décrivons ici une manière « manuelle » de faire une bibliographie qui convient pour quelques références (par contre, dans l'optique d'une utilisation intensive de L<sup>A</sup>T<sub>E</sub>X, il convient de faire appel à des outils plus puissants<sup>24)</sup> ; il s'agit de l'environnement `thebibliography` qui s'utilise comme suit :

1. À l'endroit où doit apparaître la bibliographie, on place cet environnement ; il contient la liste des données bibliographiques, chacune étant introduite par `\bibitem{<étiquette>}`.
2. Lorsque l'on veut renvoyer à un ouvrage de la bibliographie au cours du document, on utilise `\cite[<endroit>]{<étiquette>}`.

Par exemple, en écrivant à la fin d'un rapport de stage

```
\clearpage
\begin{thebibliography}{99}
\bibitem{degond} Pierre \textsc{Degond}. \og Macroscopic
limits of the Boltzmann equation\string: a review\fg.
Chapitre issu de \emph{...}, Springer, \oldstylenums{2004}.
\end{thebibliography}
```

cela affiche la bibliographie sur une nouvelle page et permet d'écrire à tout endroit du fichier source : `voir~\cite[$~$]{degond}` ou `voir~\cite{degond}`.

**Entêtes et pieds de page** : là encore, plusieurs possibilités. On peut utiliser les outils déjà existants de L<sup>A</sup>T<sub>E</sub>X (mais assez limités), faire appel à un package (`fancyhdr` en particulier) ou créer ses propres entêtes et pieds de page. Pour la première solution, il suffit d'utiliser les commandes `\pagestyle{<style>}` (dans le préambule) ou `\thispagestyle{<style>}` (à l'endroit désiré dans le document), en remplaçant `<style>` par `plain` (valeur par défaut sous `article` et `report`), `empty` ou `headings`.

**Annexes** : il suffit d'écrire `\appendix` à l'endroit où débute les annexes puis d'utiliser les commandes de sectionnement déjà vues.

**Gestion des espaces** : les blancs tapés dans le fichier `.tex` ne sont pas toujours interprétés par L<sup>A</sup>T<sub>E</sub>X, qu'ils soient horizontaux ou verticaux (cela peut permettre de rendre plus lisible le fichier source) ; par exemple :

Un espace dans le code source.	Un espace dans le code source.
Deux espaces.	Deux espaces.
Plein d'espaces.	Plein d'espaces.

L<sup>A</sup>T<sub>E</sub>X possède une précision de l'ordre du nanomètre et est capable de comprendre un certain nombre d'unités de longueur ; citons-en trois : le centimètre (`cm`) et

---

24. Tels que le programme `bibtex` conjointement au package `biblatex`. Plein d'autres packages ont fleuri autour de l'outil BIBT<sub>E</sub>X ; `biblatex` a pour vocation d'en simplifier, unifier et dépasser les capacités. Quant au programme `bibtex`, il pourra poser des problèmes si le fichier `.bib` contient des caractères accentués (en particulier s'il n'est pas en UTF-8) : les écrire alors sous la forme `{\'e}` au lieu de `é`. À l'heure actuelle (janvier 2012), autant se lancer dans l'apprentissage de `biber`, le « successeur désigné » de `bibtex` qui ne souffre pas de ses limitations ; cf. [B&C, § 12].

le point ( $1 \text{ pt} \sim 10^{-2} \text{ cm}$ ), qui sont des unités fixes, ainsi que le « em », qui est une unité relative ( $1 \text{ em} \simeq \text{largeur de la lettre M majuscule dans la police courante}$ ). On comprend très vite l'immense intérêt de cette unité relative : si l'on est amené à changer la taille des fontes d'un document, ces espaces relatifs s'adapteront automatiquement !

Pour faire des espaces horizontales, on dispose en particulier de : `\,` (espace fine), `\_` (espace inter-mot classique), `\quad` (cadratin, qui vaut 1 em) et `\quad\quad` (double cadratin), qui sont des espaces de longueurs prédéfinies ; citons enfin `\hspace{}`, pour avoir une espace de taille quelconque, qui prend en argument une valeur positive ou négative, avec une unité de longueur. En mode mathématique, on dispose aussi de la commande<sup>25</sup> `\!` qui permet de réaliser une espace fine négative.

Pour faire des espacements verticaux, on peut utiliser `\vspace{}` ; mais si l'on souhaite espacer ses paragraphes, on privilégiera `\smallskip`, `\medskip` et `\bigskip`, qui sont des blancs relatifs. Cependant, pour utiliser correctement ces trois dernières possibilités, il faut les appeler en dehors d'un paragraphe, c'est-à-dire avoir une ligne vide avant et après dans le fichier source :

<code>\dots fin de paragraphe.</code>	<code>\dots fin de paragraphe.</code>
<code>\medskip</code>	Début de paragraphe...
<code>Début de paragraphe\dots</code>	

Enfin, pour changer de page, on utilisera `\clearpage` (qui a l'avantage de placer tous les flottants<sup>26</sup> en attente, s'il y en a).

## 4.2 Inclusion d'images

*Dans de nombreux documents, on est amené à insérer des images au fil du texte : graphiques, schémas, résultats numériques, etc. Insistons sur le fait qu'on traite ici de l'insertion d'images existantes, et non de leur production<sup>27</sup>.*

Le package utilisé pour insérer des images est `graphicx`<sup>28</sup> : il fournit la commande

```
\includegraphics[<options>]{fichier}
```

qui permet d'insérer une image existante et d'ajuster l'échelle, l'orientation, etc. Les exemples suivant présentent quelques-unes des options ; on renvoie à [mpg, § 6.1.3] pour une description plus détaillée. Remarquons que dans ces exemples, l'extension du fichier `img` n'est pas précisée : c'est tout à fait licite, et même préférable (cf. conseil ci-dessous).

---

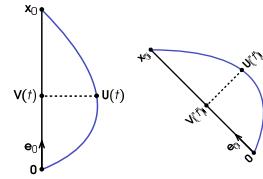
25. En fait, grâce au package `amsmath`, cette commande fonctionne en mode texte.

26. Voir le paragraphe 4.3.

27. Si les graphiques ou autres courbes proviennent naturellement de logiciels scientifiques (`scilab`, `matlab`, `maple`, etc.), la création de dessins figuratifs est plus délicate et fera l'objet du paragraphe 4.4.

28. Ne pas charger `graphics`, sur lequel il repose mais à la syntaxe moins agréable.

```
\begin{center}
\includegraphics[scale=.4]{img}
\includegraphics[width=2cm,
angle=45]{img}
\end{center}
```



Attention : les formats d'image autorisés sont totalement dépendants du mode de compilation utilisé (cf. § B.1.3) :

- compilation **TEX** → **DVI** → **PS** → **PDF** : seules les images au format PostScript sont autorisées (extensions **.ps**, **.eps** ou **.epsf**) ;
- compilation **TEX** → **PDF** : seuls les fichiers **.jpg**, **.png** ou **.pdf** sont autorisés.

Insistons lourdement : une fois qu'une image est insérée, il y a incompatibilité entre ces deux modes de compilation, *i.e.* si une image est insérée au format EPS, alors une compilation PDF**L****A****T****E****X** ne fonctionnera plus.

**Conseil.** Pour contourner cette limitation, nous vous suggérons d'avoir systématiquement chacune de vos images en deux formats (p. ex. EPS et PDF)<sup>29</sup> et de ne pas indiquer d'extension dans le `\includegraphics[]{}{}` : ainsi, selon le mode de compilation effectué, **L****A****T****E****X** choisira tout seul l'extension qu'il lui faut pour ne pas produire d'erreurs!<sup>30</sup>

Les images insérées dans **L****A****T****E****X** peuvent être numérotées et légendées, voir le paragraphe suivant sur les *flottants*.

### 4.3 Flottants

*Les flottants sont un outil efficace pour placer (dans le document final) des objets déjà créés<sup>31</sup> ; cependant, l'utilisateur se retrouve souvent confronté à des problèmes. Présentons ici brièvement ce que sont les flottants et à quel moment des difficultés peuvent intervenir ; ensuite, donnons la syntaxe pour créer de tels objets et pour aborder la résolution des problèmes qui s'y rapportent.*

#### 4.3.1 Présentation

**L****A****T****E****X** permet de créer des tableaux et d'insérer des images : ce sont parfois des objets occupant beaucoup de place verticalement et qui peuvent donc poser des problèmes de mise en page. Imaginons par exemple qu'il reste moins de 5 cm en bas d'une page au moment d'insérer une figure : que va-t-il se passer si cette dernière fait plus de 5 cm ? Eh bien nous devons la mettre sur la page suivante, où elle aura de la place ; mais alors surgit un problème : cela va laisser 5 cm de vide sur la page précédente<sup>32</sup>.

---

29. Des outils gratuits (disponibles sous Linux, Mac OS et Windows) font ces conversions : **Inkscape**, en ouvrant une image puis en la sauvegardant dans un autre format (**TheGimp** procède de même mais sans la gestion des formats vectoriels, *i.e.* EPS et PDF) ; **ImageMagick**, en ligne de commande.

30. Cela prend bien sûr plus de place sur le disque dur mais évite bien des déboires... .

31. Tels que des images (cf. § 4.2) ou des tableaux.

32. Et ce sera moche !

$\text{\LaTeX}$  peut remédier à ce problème car il offre la possibilité de faire de cette figure un flottant, c'est-à-dire que sa position dans le document pourra être différente de celle dans le fichier source ; en quelque sorte, on laisse une certaine liberté de placement : on laisse sa figure « flotter ». Dans notre exemple ci-dessus,  $\text{\LaTeX}$  va débuter une nouvelle page pour insérer la figure mais continuera d'écrire du texte sur la page d'avant, tant qu'il y aura de la place.

Arrive alors le problème fréquent suivant : l'utilisateur n'est pas content des choix de placement faits par  $\text{\LaTeX}$  ; en effet, un flottant est souvent affiché en début (ou fin) de page, ce qui ne semble pas toujours cohérent. Deux réponses au moins à cela :

- on peut préciser à  $\text{\LaTeX}$  ses préférences, afin d'influencer ces choix ;
- il faut garder à l'esprit<sup>33</sup> que  $\text{\LaTeX}$  sait très bien gérer la mise en page, afin que le document soit agréable à lire : cette gestion ne satisfait pas toujours l'auteur du premier coup, mais les décisions de  $\text{\LaTeX}$  sont généralement très performantes<sup>34</sup>.

### 4.3.2 Réalisation

Deux environnements existent pour créer des flottants : **table** et **figure** ; ils font tous les deux sensiblement la même chose, mais le premier est habituellement utilisé pour des tableaux et le second pour des figures, schémas, images, etc. On peut alors

1. Émettre des préférences de placement, parmi les quatre suivantes : ici, en haut de page, en bas de page, sur une page flottante réservée à des flottants ; ces choix se font respectivement via les quatre lettres : h, t, b et p. Plusieurs choix peuvent être donnés, par exemple `\begin{table}[htbp]` :  $\text{\LaTeX}$  essayera de placer le flottant « ici » puis, s'il ne le peut pas, passera aux autres options.
2. Légender un flottant, à l'aide de `\caption{}`.
3. Référencer (cf. § 3.5) un flottant, à l'aide de `\label{}` (prendre garde à bien mettre cette étiquette *après* la légende).

Voici un exemple (avec fichiers `sphero.eps` et `sphero.pdf` dans dossier `image`) :

```
\begin{figure}[ht]
\centering %pour centrer
\includegraphics{image/sphero}
Bla bla (commentaires et explications sur la figure)
\caption{Les trois zones d'un sphéroïde}
\label{fig-troisZones}
\end{figure}
```

Attention : il faut bien comprendre que **figure** ne sert pas du tout à *insérer* une image, mais à la faire *flotter* (c'est `\includegraphics[]{}` qui l'insère).

---

33. Ou se convaincre...

34. Les règles de placement ainsi que les paramètres pris en compte pour la gestion des flottants sont nombreux et techniques ; cela est donc à double tranchant : le résultat est satisfaisant dans de nombreux cas mais en contrepartie, lorsqu'il ne l'est pas, il devient compliqué de savoir d'où vient le désaccord avec les choix de  $\text{\LaTeX}$ .

Plein de contraintes rentrent en compte pour le placement des flottants (nombre maximal de flottants par page, etc.) ; cela fait que les problèmes apparaissent essentiellement dans deux cas : si les flottants sont très gros ou s'il y en a beaucoup<sup>35</sup>. Le premier réflexe peut être de se demander s'il est pertinent de mettre plein de flottants dans son document ; ensuite, on peut agir sur les options de placement<sup>36</sup>. Enfin, il est aussi possible (mais déconseillé) de relâcher certaines de ces contraintes, via un « ! » : on écrira `[!h]` si l'on souhaite *vraiment* que le flottant soit ici, au détriment des règles esthétiques de  $\text{\LaTeX}$ . Voir [B&C, § 7.1.1] pour mieux comprendre le fonctionnement des flottants.

Enfin, comme pour la note page 19, ne jamais écrire « `la figure ci-dessous` » mais « `la figure~\ref{fig-truc}` » (en ayant préalablement référencé cette figure).

## 4.4 Schémas et graphiques

*Présentons un aperçu partiel de ce qui existe pour créer un dessin destiné à être intégré dans un fichier .tex. Il peut être obtenu de multiples façons : soit dans un logiciel de dessin classique : on le sauvegarde alors dans un fichier image, que l'on insère avec un \includegraphics (cf. § 4.2) ; soit à l'aide d'outils externes de dessin proposant un export vers \LaTeX ; soit directement dans le .tex, à l'aide de commandes.*

### 4.4.1 Crées directement dans $\text{\LaTeX}$

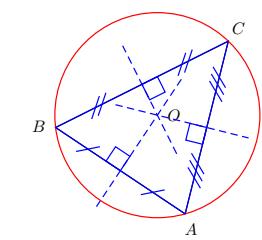
Il existe deux groupes de packages pour créer des dessins (à vocation plutôt scientifique) directement dans le fichier source : `PSTricks` et `PGF/TikZ`. Le premier est puissant et très complet<sup>37</sup> mais il ne fonctionne pas avec une compilation PDF $\text{\LaTeX}$  (cf. § B.1.3) ; le second ne souffre pas de cette incompatibilité mais il est moins riche<sup>38</sup>.

Pour une introduction à `PGF/TikZ`, voir par exemple [pg] ou [7, § 1 et 2].

Pour utiliser `PSTricks`, il faut charger le package `pstricks` (cf. [B&C, § 9]), parfois avec des packages auxiliaires, et s'interdire une compilation  $\text{\TeX} \xrightarrow{\text{pdflatex}} \text{PDF}$ . Voici un aperçu de quelques possibilités de `PSTricks` (il en existe bien d'autres : arbres, formules et représentations moléculaires, morceaux de musiques, figures de diffraction, ...).

Des figures de la géométrie euclidienne sont prédéfinies dans l'extension `pst-eucl` :

```
\begin{pspicture}(6,6)
\pstTriangle[PointSymbol=none](4,1){A}(1,3){B}(5,5){C}
\pstCircleABC[CodeFig=true,CodeFigColor=blue,
  linecolor=red,PointSymbol=none]{A}{B}{C}{O}
\end{pspicture}
```



35. En effet, tant qu'un flottant n'est pas placé, tous les suivants (même s'ils ne gênent pas  $\text{\LaTeX}$ ) sont mis en attente (pour respecter l'ordre d'apparition).

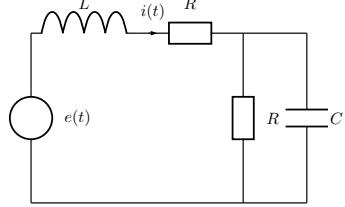
36.  $\text{\LaTeX}$  est assez chatouilleux sur ces options. S'il y a beaucoup de flottants, leur attribuer une page entière (option `p`) peut être utile. Pour un contrôle fin du placement des flottants, voir [B&C, § 7.4].

37. Peut-être trop... ? Certains de ses packages sont incompatibles entre eux.

38. Tout débutant y trouvera néanmoins largement son bonheur ; et il s'enrichit d'année en année.

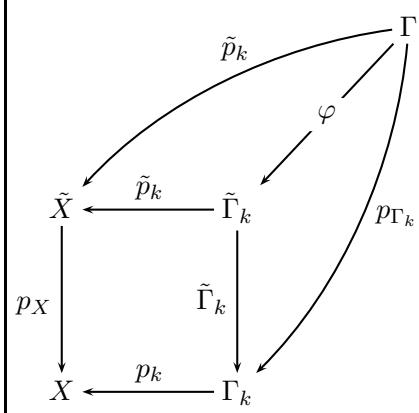
L'extension `pst-circ` fournit des macros pour dessiner des circuits électriques :

```
\begin{pspicture}(7,4.5)
\pnode(0,0){A}\pnode(5,0){B}\pnode(0,4){C}
\pnode(5,4){D}\Ucc[labeloffset=-1.1](A)(C){$e(t)$}
\wire(A)(B)\multidipole(C)(D)\coil{$L$}
\resistor[intensitylabel=$i(t)$]{$R$}.
\resistor(D)(B)\capacitor[parallel](D)(B){$C$}
\end{pspicture}
```



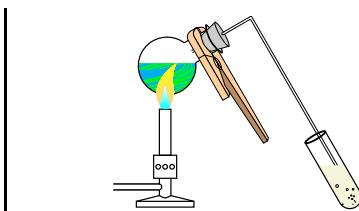
Avec l'extension `pst-node`, on peut créer des graphes ou diagrammes :

```
\begin{psmatrix} [colsep=2cm, rowsep=2cm]
& \Gamma \\
\tilde{X} & \{\tilde{\Gamma}_k\} & \\
X & {\displaystyle \Gamma_k} & \\
\psset{arrows=>, labelsep=3pt, nodesep=3pt}
\ncarc[arcangle=-20]{1,3}{2,1}^{\{\tilde{p}_k\}}
\ncline{1,3}{2,2}\ncput*{\varphi}
\ncarc[arcangle=20]{1,3}{3,2}>{\p_{\Gamma_k}}
\ncline{2,2}{2,1}^{\{\tilde{p}_k\}}
\ncline{2,1}{3,1}<{p_X}
\ncline{2,2}{3,2}<{\tilde{\Gamma}_k}
\ncline{3,2}{3,1}^{\{p_k\}}
\end{psmatrix}
```



Citons enfin `pst-labo` pour la chimie :

```
\psset{prince=true, glassType=ballon}
\pstChauffageTube[becBunsen, barbotage,
    substance=\pstFilaments[10]{green}]
```



#### 4.4.2 Crées à l'extérieur de L<sup>A</sup>T<sub>E</sub>X

Citons `Inkscape`, éditeur d'image vectorielle. Avec l'export<sup>39</sup> PDF+L<sup>A</sup>T<sub>E</sub>X (respectivement EPS+L<sup>A</sup>T<sub>E</sub>X), le dessin (créé dans `Inkscape` ou même ailleurs) est sauvegardé sous forme d'image PDF (resp. EPS) et les zones de texte sous forme de fichier `.tex` : elles peuvent donc contenir du code L<sup>A</sup>T<sub>E</sub>X, qui sera compilé ultérieurement<sup>40</sup>. Avec l'export L<sup>A</sup>T<sub>E</sub>X, le dessin est converti en fichier `.tex` sous forme de commandes PSTricks.

Citons `lpe`, éditeur d'image vectorielle créant des graphiques (scientifiques) aux formats EPS et PDF. Les zones de texte sont compilées *dans*<sup>41</sup> `lpe` (`File > Run LaTeX`).

39. Faire « Enregistrer sous » en `.pdf` (respectivement `.eps`) puis cocher la case PDF+LaTeX. Cela crée deux fichiers ; le `.pdf_tex` (ouvrable dans `TeXstudio`) contient des instructions d'utilisation.

40. Au même moment que le reste du fichier source où sera incluse l'image : cela permet ainsi d'y écrire toute commande interne au `.tex`, donc même des `\ref` !

41. Cela laisse entendre que l'on ne peut utiliser de commandes personnelles ou issues de packages :

## 4.5 Transparents et présentations

LATEX fournit la classe *slides* pour mettre en page des transparents. Toutefois, il existe un meilleur moyen de créer des présentations ou diaporamas au format PDF grâce à LATEX : il s'agit d'une nouvelle classe LATEX, nommée *beamer*. Elle génère des « transparents » (*frames*) qui peuvent être affichés en plusieurs étapes par une succession de couches (*slides*). Pour obtenir une sortie *beamer*, le début du document est un peu différent de celui habituellement utilisé (cf. page 4) : la figure 3 en présente un exemple.

```
\documentclass[french]{beamer}

\usepackage[<option à choisir>]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage{amsmath, amssymb}
\usepackage{babel} %à charger en dernier

\usetheme{Warsaw}

\begin{document}
\begin{frame}
Bienvenue devant ma présentation !
\end{frame}
\end{document}
```

FIGURE 3 – Exemple de fichier source *beamer*

On retrouve les packages liés aux mathématiques ou à la langue française et il existe une ligne supplémentaire qui définit le thème *beamer*. Ici c'est le thème *Warsaw* qui a été choisi (il existe de nombreux autres thèmes dont le détail ici serait fastidieux). Un petit tour sur Internet permet de trouver rapidement le thème voulu. On citera les plus classiques (certains peuvent utiliser des options) :

———— Quelques thèmes ———

```
\usetheme{Warsaw}
\usetheme{AnnArbor}
\usetheme[secheader]{Boadilla}
\usetheme[secheader]{Madrid}
\usetheme[compress]{Ilmenau}
```

Après le préambule, on peut définir le titre, le sous-titre, l'auteur, la date, etc. ; viendront ensuite les transparents (frames) : un frame se compose d'un titre, d'un sous-titre et de texte. Autour de cet ensemble, on trouvera (suivant le thème choisi) des

---

que nenni ! En effet, il est possible de définir à la main un « préambule de travail » dans *lpe*, voire même de se servir directement du préambule du fichier *.tex* où sera incluse l'image (cf. manuel d'*lpe*).

bandes de couleurs, des symboles de navigation, le nom de l'auteur, le titre de la section, un logo, etc. Une image étant toujours plus parlante que des mots, un exemple de frame est donné sur la figure 4.



FIGURE 4 – Exemple de frame beamer

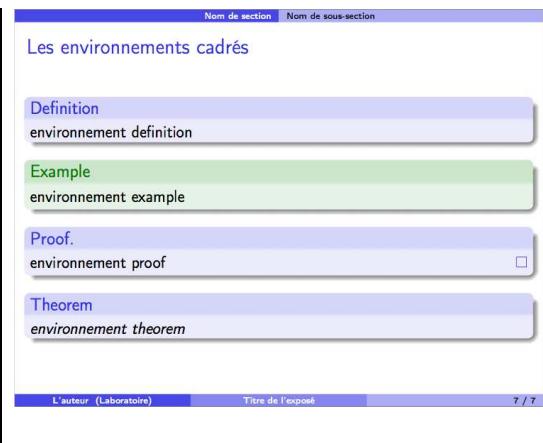
Dans le corps du transparent, on peut intégrer des tableaux, des listes, des formules, des définitions, des théorèmes, des remarques et à peu près tout ce qu'on peut faire classiquement en LATEX. Attention, il existe des exceptions à cette règle, par exemple l'insertion d'un environnement `verbatim` nécessite une option `fragile`. Sans cette option, la compilation échoue.

```
\begin{frame}[fragile]
Ce texte est normal.
\verb_Ici on est en mode verbatim._
\end{frame}
```

Ce texte est normal.  
Ici on est en mode verbatim.

Une utilisation classique de beamer est l'emploi des *blocks*. Par défaut, les théorèmes, exemples, définitions et preuves sont dans des blocks.

```
\begin{frame}
\frametitle{Les environnements cadrés}
\begin{definition}
  environnement definition
\end{definition}
\begin{example}
  environnement example
\end{example}
\begin{proof}
  environnement proof
\end{proof}
\begin{theorem}
  environnement theorem
\end{theorem}
\end{frame}
```



Par ailleurs, il existe d'autres types de blocks, dont les détails peuvent être redéfinis manuellement.

```
\begin{frame}
\begin{block}{Un bloc normal}
Texte du block \texttt{block}
\end{block}
\begin{alertblock}{Un bloc alerte}
Texte du block \texttt{alertblock}
\end{alertblock}
\begin{exampleblock}{Un bloc exemple}
Texte du block \texttt{exampleblock}
\end{exampleblock}
\end{frame}
```

Nom de section Nom de sous-section

Un bloc normal  
Texte du block block

Un bloc alerte  
Texte du block alertblock

Un bloc exemplaire  
Texte du block exampleblock

L'auteur (Laboratoire) Titre de l'exposé 6 / 7

Enfin, il existe une commande très pratique : la commande `\pause`. Elle permet d'empêcher une partie du transparent d'apparaître : elle gèle en quelque sorte l'affichage complet du transparent.

```
\begin{frame}
Ceci s'affiche en premier.\\
\pause
Cette partie ne s'affiche que
lorsque je suis pr\^et.\\
\pause
Cette derni\`ere partie
appara\^it encore apr\`es.
\end{frame}
```

Nom de section Nom de sous-section

Ceci s'affiche en premier.

L'auteur (Laboratoire) Titre de l'exposé 8 / 8

Nom de section Nom de sous-section

Ceci s'affiche en premier.  
Cette partie ne s'affiche que lorsque je suis prêt.

L'auteur (Laboratoire) Titre de l'exposé 8 / 8

Nom de section Nom de sous-section

Ceci s'affiche en premier.  
Cette partie ne s'affiche que lorsque je suis prêt.  
Cette dernière partie apparaît encore après.

L'auteur (Laboratoire) Titre de l'exposé 8 / 8

On peut se demander à quoi cela sert, car il suffirait de faire 3 transparents différents

pour obtenir le même résultat. Passant le fait que cela est beaucoup plus contraignant, on note en particulier que le compteur de transparents ne s'incrémente pas. De plus, de nombreuses options permettent de changer l'ordre d'affichage, revenir en arrière, modifier la forme du texte de manière temporaire, etc. C'est ici qu'apparaît la notion de couches, notion quelque peu complexe qu'on n'abordera pas ici.

Pour plus de détails, on consultera l'excellente page [8] ou le livre [B&C, § 10]; l'exemple de **beamer** proposé en dernière séance pourra aussi servir !

## A Typographie française

*L'objet de cette annexe est de présenter quelques-unes des règles les plus élémentaires de la typographie française et leur mise en forme (simplifiée) grâce à L<sup>A</sup>T<sub>E</sub>X.*

### A.1 Avec L<sup>A</sup>T<sub>E</sub>X

Il suffit d'utiliser le package `babel` avec l'option `french`. Nous listons quelques-unes des abréviations proposées par `babel/frenchb`<sup>42</sup> en page 33 du TP-cours et nous renvoyons à l'annexe B.3.1 pour plus d'informations.

### A.2 Quelques règles et usages

Un point capital est de respecter une cohérence au cours de son document.

#### A.2.1 Espacement et ponctuation

Grâce à `babel/frenchb`, il suffit d'utiliser la « barre d'espacement » du clavier chaque fois qu'une espace est nécessaire ; en effet, L<sup>A</sup>T<sub>E</sub>X saura alors la transformer automatiquement en le type d'espace adéquat :

- virgule et point sont suivis d'un blanc ;
- point d'interrogation, point d'exclamation, point-virgule et deux-points sont suivis d'un blanc et précédés d'une « espace fine insécable » (sauf le deux-points, en général précédé d'une « espace insécable ») ;
- les guillemets ouvrants ou fermants sont respectivement, précédés ou suivis d'un blanc, suivis ou précédés d'une espace insécable ;
- les parenthèses ou crochets ouvrants sont précédés d'un blanc ;
- les parenthèses ou crochets fermants sont suivis d'un blanc ;
- les apostrophes et traits d'union ne sont ni précédés ni suivis de blanc ;
- le tiret long (codé `---`) utilisé pour ouvrir ou fermer une incise est respectivement, précédé ou suivi d'un blanc, suivi ou précédé d'une espace fine insécable ;
- le point de suspension (codé `\dots`) est suivi d'un blanc (pour que ce blanc apparaisse, on rajoute des accolades après le nom de la commande : `\dots{}`).

#### A.2.2 Abréviations

- Une abréviation qui ne se compose que des premières lettres du mot se termine par un point (référence = réf.).
- Une abréviation qui se termine par la dernière lettre du mot ne comporte pas de point final (boulevard = bd).
- L'abréviation de groupe de mots n'a pas de point final (s'il vous plaît = svp).

---

42. C'est ainsi que nous désignons ce package muni de cette option dans ce polycopié.

Voici quelques exceptions souvent utilisées :

c'est-à-dire	c.-à-d. ou c-à-d
<i>confer</i>	cf.
environ	env.
<i>et cetera</i>	etc. (pas de point de suspension ni de répétition)
exemple	ex.
figure	fig.
<i>idem</i>	<i>id.</i>
<i>id est</i>	<i>i.e.</i>
page	p. (ne se répète pas : p. 127 et 128)
paragraphe	§ (ne se répète pas : § 4 et 5)
numéro, numéros	n°, n <sup>os</sup> (et N°, N <sup>os</sup> )
premier, premiers	1 <sup>er</sup> , 1 <sup>ers</sup>
première, premières	1 <sup>re</sup> , 1 <sup>res</sup>
deuxième, deuxièmes	2 <sup>e</sup> , 2 <sup>es</sup>
primo, secundo, tertio	1 <sup>o</sup> , 2 <sup>o</sup> , 3 <sup>o</sup>
degré (Celcius)	39 °C (espace insécable)
degré (alcoolique)	40° (pas d'espace en français)
monsieur, messieurs	M., MM. (pas de « Mr », abréviation de l'anglais <i>mister</i> )
madame, mademoiselle	M <sup>me</sup> , M <sup>lle</sup>

L'abréviation « § » est fournie par L<sup>A</sup>T<sub>E</sub>X : \S ; les autres se codent facilement grâce à babel/frenchb : \no, 1\iers{}, 2\ieme{}, \primo{}, 39~\degres{} C, M\up{me}, ...

### A.2.3 Ponctuation

Un mauvais emploi de la ponctuation peut rendre difficile la lecture, ou même changer le sens d'une phrase. Quelques rappels :

- les phrases se terminent par un point final, un point d'exclamation, un point d'interrogation ou un point de suspension ;
- la virgule sépare les parties d'une proposition ou d'une phrase à condition qu'elles ne soient pas déjà réunies par les conjonctions et, ou, ni. Elle sépare les énumérations. Elle isole les mots mis en apostrophe. On ne met pas de virgule avant une parenthèse, un tiret ou un crochet ;
- le point virgule sépare différentes propositions de même nature ;
- le deux-points introduit une explication, une citation, une énumération ;
- le point de suspension est toujours composé de trois points (utiliser \dots{}) ;
- les parenthèses intercalent une précision dans la phrase ;
- les crochets indiquent une précision à l'intérieur d'une parenthèse ou une coupure dans une citation ;
- employé seul, le tiret signale chaque terme d'une énumération ou le changement d'interlocuteur dans un dialogue. Employés par paires, ils s'utilisent comme les parenthèses ; en fin de phrase, le tiret fermant est supprimé avant le point final.

#### A.2.4 Énumérations

On peut distinguer deux types d'énumérations :

- celles qui arrivent en cours de phrase :
  - elles sont introduites par un deux-points et chaque terme commence par une minuscule,
  - celles de premier rang sont introduites par un tiret et se terminent par un point-virgule (même s'il y a plusieurs phrases dans l'énumération), sauf la dernière par un point final,
  - celles de second rang sont introduites par un tiret décalé et se terminent par une virgule, sauf la dernière par un point virgule ;
- celles qui débutent une phrase : chaque terme commence alors par une majuscule et se termine par un point (voir un exemple au paragraphe A.2.5).

En général, lorsque chaque élément de l'énumération est introduit par un numéro (ou autre symbole) suivi d'un point, on applique toujours le 2<sup>e</sup> type ci-dessus.

#### A.2.5 Divers

- On rappelle que l'accent a pleine valeur orthographique ; en conséquence **on met les accents sur les majuscules**.
- On écrit « on résout », « quel que soit », « quelles que soient », « intéresser » et « on rappelle qu'il est rappelé ».
- Une erreur très fréquente est l'abus de majuscules.

Enfin, on n'oubliera pas d'être informé de la « nouvelle orthographe », réforme orthographique de l'Académie Française datant de 1990 ; l'emploi de ces rectifications n'est pas imposé, mais il est recommandé (en particulier, aucune des deux graphies ne peut être tenue pour fautive). Citons deux des dix nouvelles règles :

- « On emploie l'accent grave (plutôt que l'accent aigu) dans un certain nombre de mots (pour régulariser leur orthographe), au futur et au conditionnel des verbes qui se conjuguent sur le modèle de céder, et dans les formes du type puissè-je ». Ex : évènement, je cèderai ;
- « L'accent circonflexe disparait sur *i* et *u*. On le maintient néanmoins dans les terminaisons verbales du passé simple, du subjonctif et dans cinq cas d'ambigüité » (motivation : « sur *i* et *u*, l'accent circonflexe ne joue aucun rôle phonétique ; il est l'une des principales causes d'erreurs et son emploi, aléatoire, ne peut être justifié par l'étymologie. »). Ex. : cout, il paraît ; dû, sûr, mûr, jeûne.

Ces quelques lignes sont issues de [9], court document qui résume cette réforme, l'explique, en décrit les avantages et donne des remarques ainsi que des exemples.

## B Explications complémentaires au TP-cours

Dans cette annexe, expliquons certains points non détaillés pendant les séances, afin de mieux les comprendre et de savoir les utiliser efficacement. Pour des explications plus exhaustives, se reporter aux ouvrages sur  $\text{\LaTeX}$  ([mpg] et [B&C] sont des valeurs sûres).

### B.1 Compréhension et utilisation de $\text{\LaTeX}$

#### B.1.1 Installation et paramétrage

**Sous Windows** Différents outils sont à installer. Nous recommandons de se servir de ProTeXt (`protext.exe` sur <http://www.tug.org/protext>), qui s'occupe de tout :

$$\text{ProTeXt 3.0} \simeq \text{MiK}\text{\TeX} + \text{T}\text{\EX}{}_{\text{studio}}^{43} + (\text{Ghostscript} + \text{GSview}).$$

Lancer le `.exe`; choisir un dossier où seront extraits les fichiers d'installation<sup>44</sup>. Ensuite :

1. Lancer le `Setup.exe` de ce dossier et cliquer sur `Open` pour ouvrir le manuel d'installation : il suffit de suivre du § 1.2 au § 1.6, à une exception : faire une installation `Complete MiK\TeX` au lieu du `Basic MiK\TeX` illustré figure 1.3 du manuel. Cela installe `MiK\TeX`.
2. Pour installer `Ghostscript` et `GSview` (assurant la gestion du format PostScript), aller dans le sous-dossier `gsv` et lancer `gs902w32.exe` puis `gsv49w32.exe`.
3. Pour installer `T\EX{}_{\text{studio}}`, revenir au manuel d'installation et suivre le § 1.7.

**Sous Mac OS** Nous recommandons MacTeX (<http://www.tug.org/mactex>) ainsi que `T\EX{}_{\text{studio}}`<sup>45</sup> (`T\EX{}_{\text{Shop}}` est une alternative à `T\EX{}_{\text{studio}}`) :

$$\text{MacTeX} \simeq \text{T}\text{\EX}{}_{\text{Live}} + \text{T}\text{\EX}{}_{\text{Shop}}.$$

**Installation sous Linux** Nous recommandons la distribution `T\EX{}_{\text{Live}}` (<http://www.tug.org/texlive>), ainsi que `T\EX{}_{\text{studio}}`<sup>45</sup> (l'éditeur Kile est une alternative à `T\EX{}_{\text{studio}}`). Il est préférable de ne pas utiliser la version de `T\EX{}_{\text{Live}}` proposée sous forme de paquets par le système d'exploitation (car elle n'est pas à jour); procédure ici ([B&C, § 1.1]) ou là (<http://www.tug.org/texlive/doc/texlive-fr/texlive-fr.html>)<sup>46</sup>.

**Paramétrage de `T\EX{}_{\text{studio}}`** Pour régler l'encodage sur UTF-8 : aller dans le menu Options > Configurer TeXstudio > Éditeur, sélectionner UTF-8 en face du champ Codage d'entrée des caractères; s'assurer que la case Auto-détection est cochée. Après cela, UTF-8 doit s'afficher en bas à droite de l'écran. Pour un changement occasionnel d'encodage, aller dans le menu Éditer > Sélectionner le codage d'entrée...

---

43. Précédemment dénommé TexMakerX.

44. Ce dossier, ainsi que le `.exe` téléchargé, pourront être supprimés après l'installation.

45. Pour installer `T\EX{}_{\text{studio}}`, aller sur <http://texstudio.sourceforge.net/>.

46. En particulier, pour éviter d'avoir des paramétrages à faire après l'installation, mettre le champ Créer les liens symboliques dans les répertoires systèmes à Oui.

### B.1.2 Fonctionnement : les quatre étapes décrites à la section 1

L'étape 1 pourrait se faire via un éditeur « quelconque », comme **Nedit** (sous Linux), **TextEdit** (sous Mac OS) ou **WordPad** (sous Windows) ; mentionnons aussi **Emacs**, éditeur de texte non graphique mais polyvalent et disponible sur tous les systèmes d'exploitation : ses adeptes sont nombreux (la saisie d'un fichier `.tex` est facilitée grâce à l'extension **AucTEX**). Les autres étapes se feraient alors via une fenêtre de commande (où l'on taperait « à la main » `latex TP_init.tex` pour compiler). Néanmoins, il existe des éditeurs de texte orientés **L<sup>A</sup>T<sub>E</sub>X** qui permettent de faire ces étapes beaucoup plus agréablement, en quelques clics : c'est ce que nous avons utilisé en séance et nous conservons ce point de vue dans la suite, où nous donnons quelques remarques propres à chaque étape.

Pour saisir son document (étape 1), plusieurs éditeurs de texte existent, dont

- sous Linux : **Kile**, **Texmaker**, **T<sub>E</sub>Xstudio** (ex-**TexMakerX**) ;
- sous Mac OS : **T<sub>E</sub>XShop**, **Texmaker**, **T<sub>E</sub>Xstudio** (ex-**TexMakerX**) ;
- sous Windows : **T<sub>E</sub>XnicCenter**, **Texmaker**, **T<sub>E</sub>Xstudio** (ex-**TexMakerX**), **WinEdt**<sup>47</sup>.

Ces éditeurs de texte (ou interfaces graphiques) sont l'endroit où l'on tape son fichier source ; ils sont à distinguer de « ce qui fait fonctionner L<sup>A</sup>T<sub>E</sub>X » à proprement parler, c'est-à-dire de ce qui fait fonctionner la compilation (étape 2). On parle alors de *distributions L<sup>A</sup>T<sub>E</sub>X* ; ces dernières ne sont pas non plus les mêmes selon les systèmes d'exploitation. Citons sous Linux et Mac OS : **T<sub>E</sub>XLive** ; sous Windows : **MiK<sub>T</sub>E<sub>X</sub>** ou **T<sub>E</sub>XLive**.

Après la compilation, L<sup>A</sup>T<sub>E</sub>X a produit le fichier `TP_init.dvi` ; il en a aussi créé d'autres (qui lui ont été utiles), mais les éditeurs de texte savent choisir le bon lorsque l'on demande de visualiser (étape 3). Le logiciel de visualisation est en général fourni avec la distribution L<sup>A</sup>T<sub>E</sub>X (Yap pour Windows, xdvi ou kdvi pour Linux, xdvi pour Mac) ; ce format DVI nécessite cependant que L<sup>A</sup>T<sub>E</sub>X soit installé pour s'ouvrir correctement : c'est en quelque sorte un outil de travail pour le rédacteur, un peu comme un brouillon<sup>48</sup>.

Une fois que le document est sous sa version finale, on souhaite en général l'imprimer ou le diffuser (étape 4) : on utilise alors le format PDF (le format PS existe aussi). Les éditeurs de texte cités ci-dessus permettent de faire ces conversions en quelques clics : on obtient alors un fichier lisible et imprimable par tous !

### B.1.3 Compilation et obtention du PDF

Nous avons vu au paragraphe 3.1 page 5 qu'il y avait deux manières de produire un PDF ; ce choix est parfois capital. Le schéma suivant résume les étapes de réalisation dites « par PostScript » (ou L<sup>A</sup>T<sub>E</sub>X) :

$$\text{TEX} \xrightarrow{\text{ latex }} \text{DVI} \xrightarrow{\text{ dvips }} \text{PS} \xrightarrow{\text{ ps2pdf }} \text{PDF}.$$

---

47. Ce dernier est payant.

48. On remarquera que ces logiciels mettent automatiquement à jour le fichier DVI après une nouvelle compilation : il est donc inutile de les redémarrer à chaque fois.

L'autre mode de compilation dit « PDF direct » (ou PD<sub>F</sub>LT<sub>E</sub>X) compile, comme son nom l'indique, directement `TP_init.tex` en `TP_init.pdf` :

$$\text{TEX} \xrightarrow{\text{pdflatex}} \text{PDF}.$$

Attention : ce deuxième mode de compilation *n'est pas un simple raccourci* du premier. Certains fichiers sources ne se compileront pas correctement avec PD<sub>F</sub>LT<sub>E</sub>X (notamment si l'on utilise PSTricks<sup>49</sup>, qui a besoin — en général — d'un passage par le format PS)<sup>50</sup>, alors que pour d'autres ce sera l'inverse.

Il est surtout très important de faire attention au mode de compilation lors de l'inclusion d'images (cf. § 4.2) : PD<sub>F</sub>LT<sub>E</sub>X ne permet d'incorporer que des fichiers `.jpg`, `.png` et `.pdf`, alors que l'autre mode ne permet d'incorporer que des fichiers `.ps` et `.eps`. Nous suggérons d'utiliser les formats EPS et PDF (qui sont des formats vectoriels) : beaucoup de logiciels mathématiques (`maple`, `matlab`, `scilab`, etc.) proposent d'enregistrer leur graphique en `.eps` ; de plus, des logiciels gratuits (cf. note n° 29 page 24) permettent de faire des conversions vers une telle extension ou en `.pdf`.

**Que retenir de ces deux modes de compilation ?** Le plus simple nous paraît de faire un choix et de *s'y tenir dans un maximum de cas* ; ne faire alors appel à l'autre mode que lorsqu'il semble incontournable.

## B.2 Le `\documentclass`

Le `\documentclass` permet de donner la *classe* (ou le *type*) du document que l'on souhaite rédiger ; L<sup>A</sup>T<sub>E</sub>X en propose au moins cinq (`article`, `report`, `book`, `letter`, `slides`) mais il en existe d'autres (comme `beamer`). Nous ne présentons ici que les trois premières (voir paragraphe 4.5 pour `slides` et `beamer`) :

- `article` convient pour un document court (moins de 30 pages) tel qu'un article de recherche, un petit rapport (TP, projet, etc.), une documentation, etc. ;
- `report` utile pour un document de taille moyenne, tel qu'un rapport de stage, etc. ;
- `book` est à réservier pour un long document tel qu'un livre, un mémoire de thèse, un polycopié de cours, etc.

En fait, le choix de la classe n'est pas aussi rigide : chaque classe offre quelques fonctionnalités différentes (voir ci-dessous) et c'est selon ses préférences que l'on choisit la classe pour les cas limites (comme un rapport de stage d'une trentaine de pages...).

Des options peuvent être ajoutées à la déclaration de classe, en particulier :

- la taille du corps de base : `10pt` (par défaut), `11pt` ou `12pt` ;
- l'option `titlepage` qui indique qu'une nouvelle page doit être commencée après l'ajout du titre par `\maketitle`<sup>51</sup> ;

---

49. Voir paragraphe 4.4.

50. Signalons que des packages existent pour contourner certains de ces obstacles : voir `hyperref`, `urlbreak`, `pdftricks`, ... ce dernier n'étant pas efficace dans 100% des cas.

51. Voir paragraphe 4.1.

- le choix d'une composition en recto-verso (`twoside`) ou en recto (`oneside`) : un document recto-verso a des marges de droite et de gauche différentes selon que la page est paire ou impaire (pour permettre une reliure centrale après impression)<sup>52</sup>; cette option n'influe en rien sur le fait d'imprimer en recto-verso ou non !

Elles s'utilisent suivant la syntaxe

```
| \documentclass[12pt,titlepage]{article}
```

Citons maintenant deux différences entre ces trois classes : tout d'abord, `report` et `book` admettent le sectionnement `\chapter`, contrairement à `article`<sup>53</sup>. Ensuite, L<sup>A</sup>T<sub>E</sub>X choisit les options par défaut et ces choix peuvent dépendre de la classe (ainsi `article` et `report` sont par défaut sous `oneside`, alors que `book` ne l'est pas). Il est déconseillé de modifier un grand nombre d'options car, pour une classe de document donnée, les choix de L<sup>A</sup>T<sub>E</sub>X sont bien souvent judicieux du point de vue de la mise en page.

## B.3 Le préambule et les packages

### B.3.1 Les packages « de base »

Nous décrivons ici très brièvement le rôle des packages présentés à la section 2 ; des informations complètes peuvent être trouvées dans la documentation de chaque package (celle-ci étant disponible sur le Web en tapant CTAN `<nom du package>`).

- `inputenc` permet de taper directement au clavier les accents et cédilles dans le fichier source (ce package sert à spécifier le codage d'entrée). Attention : le choix de l'option (cf. table 1 page 40) dépend du paramétrage de l'éditeur de texte sous lequel est *saisi* (ou *a été saisi*) le fichier ; cela est détaillé en annexe B.3.2. Sans ce package, il faut saisir ces caractères « à l'ancienne » (`\'e` ↠ é, etc.).
- `fontenc`, avec l'option `T1`, permet une bonne césure<sup>54</sup> des mots de langues latines et un copier-coller depuis le PDF (ce package sert à spécifier le codage des fontes).
- `lmodern` permet d'adapter la police de base de L<sup>A</sup>T<sub>E</sub>X aux nouveaux formats apparus depuis sa création, le PDF notamment (ce package charge la police *Latin Modern*).
- `babel` permet de prendre en compte la langue de rédaction (éventuellement plusieurs) pour la mise en page, les règles typographiques, les césures, etc. ; `french` est l'option<sup>55</sup> actuellement à utiliser pour un document en français (cf. § A.1). On notera que `babel/frenchb` apporte des commandes supplémentaires pour la langue française (par exemple, `\ier{}` permet d'écrire l'exposant de 1<sup>er</sup>) et francise les mots-clés de L<sup>A</sup>T<sub>E</sub>X (par exemple, l'appel à `\tableofcontents` affichera « Table des matières » au lieu de « Contents »). Nous renvoyons à [B&C, § 2.6] pour une description plus précise de ses actions.

---

52. Ce n'est pas la seule différence.

53. L<sup>A</sup>T<sub>E</sub>X estime qu'un document de classe `article` n'a pas besoin de ce niveau de sectionnement : si au cours d'une rédaction sous `article` vous en ressentez le besoin, c'est qu'il faut changer de classe.

54. Coupure de fin de ligne.

55. À exceptionnellement écrire au niveau du `\documentclass` et non du `\usepackage{babel}`.

- `geometry` permet de spécifier la dimension des pages du document (l’option par défaut étant le format en vigueur aux États-Unis, il est important de préciser l’option `a4paper` pour une impression qui se fera sur du papier A4). Ce package est bien plus puissant : il facilite aussi un contrôle fin et complet des marges du document (essayez `[a4paper, body={16cm, 24cm}]`).
- `amsmath` et `amssymb` apportent des fonctionnalités supplémentaires en mode mathématique. Remarquez la syntaxe (nous aurions pu aussi les charger au moyen d’un `\usepackage` chacun) : en regroupant de la sorte, on peut rendre plus lisible le fichier source<sup>56</sup>.

### B.3.2 Encodage et package `inputenc`

Ce paragraphe est l’occasion de parler brièvement d’encodage. C’est une notion technique, déjà abordée en note n° 2 (page 5) et au paragraphe B.3.1 (avec `inputenc`) ; on a besoin de s’en soucier dans deux cas au moins :

- lorsque plusieurs personnes travaillent sur un même fichier `.tex` ;
- lorsqu’une personne travaille son fichier sur des éditeurs de texte différents (par exemple un à la maison et un à la fac).

**Qu’est-ce que l’encodage ?** Grossièrement, chaque caractère saisi au clavier est codé par un nombre lorsque le fichier `.tex` est enregistré sur l’ordinateur ; ce nombre est défini par l’encodage de l’éditeur au moment de la saisie. Exemple : supposons que le nombre 130 représente un « é » dans l’encodage  $\mathcal{A}$  et un « ç » dans l’encodage  $\mathcal{B}$  ; supposons qu’un utilisateur règle son éditeur sur le  $\mathcal{A}$  et qu’il saisisse plusieurs é ; alors, s’il ouvre son fichier sur un éditeur réglé sur l’encodage  $\mathcal{B}$ , tous ces é seront devenus des ç... D'où l'intérêt de faire attention ! L'ordinateur (et L<sup>A</sup>T<sub>E</sub>X) ayant été développés aux États-Unis (où les caractères accentués n'existent pas), l'encodage « de base » ne possède pas ces caractères.

À l’heure actuelle (janvier 2012) le choix de l’encodage UTF-8 est à privilégier dès que possible<sup>57</sup>, même si dans de (très) rares occasions il pose problème ; en effet, ses avantages sont bien supérieurs à ses inconvénients. Donnons nos motivations.

- Pourquoi choisir l’UTF-8 ? C’est un encodage commun aux trois plateformes habituelles (Linux, Mac OS, Windows) et universel<sup>58</sup>, qui a vocation à s’imposer.
- Pourquoi peut-il poser problème ? L<sup>A</sup>T<sub>E</sub>X a été créé avant l’UTF-8 : de rares packages ou fonctionnalités ne sont donc pas encore compatibles (il suffit alors, en général, de revenir à la saisie « à l’ancienne » des accents). Par exemple, on ne peut pas saisir de caractères accentués dans un code mis en forme par le package `listings`<sup>63</sup>.

---

56. Ce qui ne change rien pour L<sup>A</sup>T<sub>E</sub>X lors de la compilation, mais facilite la lecture du fichier `.tex`.

57. Il n'est pas possible de l'utiliser avec les versions  $\leq 1$  de l'éditeur de texte T<sub>E</sub>XnicCenter (Windows) qui ne supporte que l'encodage windows-1252. Nous recommandons de fuir cet éditeur, bien qu'il eut sa période de popularité.

58. Il est commun à tous les systèmes d’exploitation et vise à regrouper l’ensemble des caractères utilisés dans toutes les langues officielles de la Terre. (Whaou !)

D'autres encodages existent : ISO-8859-1 (ou latin1) sur Linux, Apple Roman propre à Mac OS, windows-1252 propre à Windows (mais très proche du latin1).

**Quel lien avec `inputenc` ?** Grossso modo, ce package autorise l'utilisation dans L<sup>A</sup>T<sub>E</sub>X d'un encodage autre que celui de base. Il permet, via son option, d'indiquer à L<sup>A</sup>T<sub>E</sub>X quel est l'encodage utilisé par l'éditeur (car L<sup>A</sup>T<sub>E</sub>X ne peut pas le deviner tout seul). Il faut bien comprendre que ce choix d'option ne modifie en rien l'encodage sur lequel est réglé l'éditeur : l'utilisateur *se doit de donner à `inputenc` l'option qui coïncide avec le paramétrage* de l'éditeur ; pour faire ce choix, voir la table 1.

encodage <sup>59</sup>	option
UTF-8	utf8
ISO-8859-1	latin1
windows-1252	cp1252
Apple Roman	applemac

TABLE 1 – Correspondance encodage/option pour le package `inputenc`

### En pratique, que faire ?

1. Savoir sur quel encodage est réglé<sup>60</sup> votre éditeur (en choisissant l'UTF-8 dès que possible) ; pour T<sub>E</sub>Xstudio, cf. § B.1.1.
2. Choisir l'option d'`inputenc` (cf. table 1) qui coïncide avec le réglage de l'éditeur.
3. Indiquer aux éventuelles personnes modifiant le fichier `.tex` l'encodage choisi ; ou se mettre d'accord *dès le début* de la rédaction sur un encodage commun (il n'est pas possible de mélanger plusieurs encodages dans un même fichier `.tex`).

### B.3.3 Quelques autres packages utiles

*Donnons ici une liste très courte de packages (non abordés dans ce TP-cours) qui peuvent se révéler utiles. Pour toute information, se reporter à la documentation du package<sup>61</sup>.*

Il existe un très grand nombre de packages ; ils pourraient être classés en trois catégories : certains sont plus ou moins incontournables (cf. § B.3.1), d'autres enrichissent des fonctionnalités déjà existantes (tels les entêtes et pieds de page) et d'autres encore sont très spécialisés (pour la composition de molécules chimiques par exemple).

Il faut savoir que les packages évoluent au gré de leur(s) concepteur(s) ; ainsi, certains deviennent obsolètes et sont remplacés par des plus performants. Un bon conseil est de vérifier la date de dernière mise à jour avant de s'en servir. Les packages ci-dessous sont d'usage fréquent ; leur fonctionnement est illustré dans [mpg], [pg] ou [B&C].

---

59. Les noms d'encodage sont ceux qui apparaissent dans T<sub>E</sub>Xstudio.

60. Souvent, il suffit de regarder dans un menu du genre Options ou Préférences.

61. Trouvable sur Internet en cherchant « CTAN <nom du package> ».

**enumitem** pour personnaliser les trois types de listes ; l'option `shortlabel` est souvent celle recherchée. (Attention à la compatibilité avec `babel/frenchb`.)

**fancyhdr** pour personnaliser les entêtes et pieds de page.

**fancyvrb** pour personnaliser les saisies en mode verbatim.

**floatrow** pour personnaliser les flottants.

**hyperref** pour faciliter la navigation hypertexte dans les versions PDF (*i.e.* transformer les `\ref{}`, `\cite{}`, etc. en liens cliquables) et pour gérer quelques outils offerts par le format PDF (cf. [mpg, § 3.5.3]). À charger après `babel`.

**listings** pour mettre en forme des lignes de code ; il connaît énormément de langages informatiques<sup>62</sup>. Attention, utiliser des caractères propres à l'UTF-8 à l'intérieur d'un environnement `lstlisting` peut être source d'incompatibilités<sup>63</sup>.

**siunitx** pour gérer l'écriture des unités ; on évitera l'usage de `\SI{units}{}`...

**subfig** pour gérer les sous-flottants.

**thmtools** pour accroître et simplifier les possibilités de `amsthm` ; à charger avec `amsthm`.

**titlesec** pour personnaliser l'apparence des titres de sectionnement.

**tocbibind** pour ajouter automatiquement à la table des matières des entrées qui ne s'y mettent pas toutes seules (commandes de sectionnement étoilées, etc.).

**varioref** pour améliorer et personnaliser l'utilisation des références (`\ref` et `\pageref`). À charger après `babel`.

**xspace** pour ne pas avoir à taper des accolades à la fin de certaines commandes si l'on veut une espace derrière (*i.e.* `1\ier exercice` au lieu de `1\ier{} exercice`, etc.).

## B.4 Le préambule et les macros

Revenons ici sur la deuxième partie d'un préambule classique, introduit à la section 2. Il s'agit d'un outil de personnalisation très puissant de  $\text{\LaTeX}$  : on peut créer ses propres commandes (voir [B&C, § C.1] disponible sur Internet ou [mpg, § 5.2]).

Cela se fait avec la commande `\newcommand*{nom}[]{}{code}` (l'étoile est à retirer si la commande doit s'appliquer à un objet plus grand qu'un paragraphe). Illustrons cette fonctionnalité à travers différents exemples (sauf mention contraire, les extraits de code qui suivent sont à taper *dans le préambule*, après le chargement des différents packages) :

1. Abréger un nom de commande : `\newcommand*{\eps}{\varepsilon}`. Il suffira alors de taper (dans le corps de document) `$\eps$` pour afficher  $\varepsilon$ .
2. Distinguer le fond et la forme : `\newcommand*{\nom}[1]{\textsf{#1}}`. Cela illustre d'ailleurs l'utilisation des commandes à paramètres (un seul ici). C'est une macro utilisée dans ce polycopié : dès que l'on a cité un nom de package,

---

62. Cela signifie qu'il est capable de reconnaître les mots-clés, les commentaires, etc. d'un langage et d'adapter la mise en forme en conséquence.

63. Une solution est de définir des caractères d'échappement (champ `escapeinside`) afin de pouvoir saisir les accents « à l'ancienne » (`\`e` → é, etc.).

d'environnement, de logiciel, etc. on a écrit, par exemple, `\nom{beamer}` au lieu de `\textsf{beamer}`. C'est à priori se compliquer la vie, mais il n'en est rien : en distinguant ainsi le fond et la forme dans le fichier source, on le rend plus lisible et plus facile à corriger ; autre avantage : si au dernier moment on doit afficher ces noms en **type machine à écrire** (par exemple pour se coordonner avec un collègue qui aurait rédigé une autre partie du document), il suffit de changer un seul endroit du fichier source et tout le reste en découlera automatiquement. D'autres exemples convaincants sont présents dans le gros exemple (cf. § B.5).

3. Créer de nouveaux opérateurs mathématiques (certains opérateurs courants ou francisés ne figurent pas dans L<sup>A</sup>T<sub>E</sub>X) : `\DeclareMathOperator{\divg}{\div}` pour l'opérateur divergence ; il permet d'écrire (dans le corps de document) :

Formule de la divergence : $\oint_{\partial\Omega} \vec{X} \cdot d\vec{S} = \int_{\Omega} \operatorname{div} \vec{X} \, dx.$	Formule de la divergence : $\oint_{\partial\Omega} \vec{X} \cdot d\vec{S} = \int_{\Omega} \operatorname{div} \vec{X} \, dx.$
--	--

Ces deux commandes ne permettent pas de renommer une macro déjà existante (par exemple, `\div` existe sous L<sup>A</sup>T<sub>E</sub>X). On remarque aussi que pour utiliser `\eps` définie ci-dessus, on doit être en mode mathématique : en rajoutant la bonne commande dans la définition de cette macro, on peut faire en sorte que L<sup>A</sup>T<sub>E</sub>X passe automatiquement en mode mathématique quand c'est nécessaire...

Mentionnons pour finir qu'il est possible de créer plein d'autres choses, dont des environnements et des compteurs (cf. [B&C, § C.1 et C.2] librement accessible sur Internet).

## B.5 Gros exemple mathématique

Enfin, terminons en donnant un gros exemple regroupant plein de choses (utiles, à faire ou à éviter). La présentation du code source est obtenue au moyen du package `listings` ; regardez le résultat de la compilation (situé page 44) et revenez ensuite consulter le source ci-dessous pour savoir ce qui s'y cache<sup>64</sup>.

Lignes à ajouter dans le préambule :

```

1 \usepackage{dsfont} %pour un alphabet different de \mathbb{ }
2 \usepackage{esvect} %pour les fleches \vv{} plus jolies que \overrightarrow{ }
3
4 \newcommand*{\abs}[1]{\left| #1 \right|} %valeur absolue
5 \newcommand*{\norm}[1]{\left\| #1 \right\|} %norme
6 \newcommand*{\N}{\mathbb{N}} %entiers naturels
7 \newcommand*{\ensemble}[1]{\left\{ #1 \right\}} %ensemble
8 \newcommand*{\tq}{\enskip\middle|\enskip} %le 'tel que' ; les espaces autour
9 %peuvent etre reduits, selon les gouts
10 \newcommand*{\sachant}{\mid} %le 'sachant que' des proba conditionnelles
11 \newcommand*{\dpart}[2]{\frac{\partial #1}{\partial #2}} %derivee partielle
12
13 \DeclareMathOperator{\grad}{\nabla} %operateur gradient
14 \DeclareMathOperator{\divg}{\operatorname{div}} %operateur divergence

```

---

64. Pour toute explication, consulter [mpg], [3], [B&C], etc. ou votre moteur de recherche favori.

```

\DeclareMathOperator{\rot}{\overrightarrow{\operatorname{curl}}}} %opérateur
gradient

Code à compiler :

16 Ne pas confondre \verb|_|_, \verb|\lvert|_, \verb|\rvert| et \verb|\mid| ; ni \verb|
|_|_, \verb|\lVert|_ et \verb|\rVert| ; ni \verb|\backslash| et \verb|\setminus|

18 \[
  \begin{array}{r@{\quad\longrightarrow\quad}l}
    \text{pas bien} & \text{bien}\\
    a|b & a\mid b \\
  \end{array}
\]
22 \mathbb{P}(A|B) & \mathbb{P}(A\backslash sachant B) \\
\{x\in A | x\not=0\} & \{x\in A | tq x\not=0\} \\
24 ||\lambda x||=|\lambda x| & \|\lambda x\|=\abs{\lambda}\norm{x} \\
\mathcal{N}^*=\mathcal{N}\backslash\{0\} & \mathcal{N}^*=\mathcal{N}\setminus\{0\}
26 \end{array}
\]
28 Regardez les deux codes sources distincts pour obtenir ces deux morceaux
identiques :
30 \left[ \frac{x}{\left| x \right|} \middle| \frac{x}{\left| x \right|} \right] \qquad \{x\in A | tq \frac{x}{\left| x \right|} \not= 0\}
et adoptez l'utilisation des commandes personnelles !
32 'Equations de transport sous formes non conservative et conservative ($a\colon\mathbf{R}^n\rightarrow\mathbf{R}^n$) :
34 \begin{subequations}
  \begin{aligned}
36   \dpart{u}{t} + a\cdot\nabla u &= 0, \\
   \dpart{u}{t} + \nabla\cdot(au) &= 0.
  \end{aligned}
38 \end{subequations}
40 Nous effectuons les calculs suivants \begin{align}
42   A &= \dpart{f}{t}(x,v,t) + v\cdot\nabla_x f(x,v,t) + \nabla_v\cdot\nabla_x f(x,v,t)\biggr) \\
   &\quad + \cdots \notag \\
44   A &= \dpart{f}{t}(x,v,t) + v\cdot\nabla_x f(x,v,t) + \nabla_v\cdot\nabla_x f(x,v,t)\biggr) + \nabla_v\cdot\nabla_x f(x,v,t)\biggr) \\
   &\quad + \cdots \notag \\
   &\quad \intertext{qui donnent, plus lisiblement}
46   A &= \dpart{f}{t}(x,v,t) + v\cdot\nabla_x f(x,v,t) + \nabla_v\cdot\nabla_x f(x,v,t)\biggr) \\
   &\quad + \cdots \notag \\
   &\quad \end{align}
48 \begin{equation*}
50 x \in \emptyset \iff x \in \varnothing \iff \left\{ \begin{aligned}
   &\begin{aligned}
52   x &\leq 0 \quad \text{et} \quad x \geq 0 \\
   &x = 0
54 \end{aligned} \right. \right.
   \end{aligned} \right. \right.
56 \begin{aligned}
58   \nabla\cdot E &= \frac{\rho}{\epsilon_0} \cdot B \quad \& \nabla\cdot B = 0 \\
   \nabla\cdot E &= -\partial_t B \quad \& \nabla\cdot B = \mu_0(\nabla\cdot E + \frac{\rho}{\epsilon_0})
   \end{aligned}
60 \end{aligned}

```

Résultat de la compilation (après ajout de traits horizontaux pour séparer les exemples) :

---

Ne pas confondre  $\mid$ ,  $\backslash\lvert$ ,  $\backslash rvert$  et  $\backslash mid$ ; ni  $\backslash\mid$ ,  $\backslash lVert$  et  $\backslash rVert$ ; ni  $\backslash backslash$  et  $\backslash setminus$ :

$$\begin{array}{ll} \text{pas bien} & \longrightarrow \text{bien} \\ a|b & \longrightarrow a \mid b \\ \mathbb{P}(A|B) & \longrightarrow \mathbb{P}(A \mid B) \\ \{x \in A | x \neq 0\} & \longrightarrow \{x \in A \mid x \neq 0\} \\ \|\lambda x\| = |\lambda| \|x\| & \longrightarrow \|\lambda x\| = |\lambda| \|x\| \\ \mathbb{N}^* = \mathbb{N} \setminus \{0\} & \longrightarrow \mathbb{N}^* = \mathbb{N} \setminus \{0\} \end{array}$$


---

Regardez les deux codes sources distincts pour obtenir ces deux morceaux identiques :

$$\left\{ x \in A \mid \frac{x}{10} \ll 1 \right\} \quad \left\{ x \in A \mid \frac{x}{10} \ll 1 \right\}$$

et adoptez l'utilisation des commandes personnelles !

---

Équations de transport sous formes non conservative et conservative ( $a: \mathbb{R}^n \rightarrow \mathbb{R}^n$ ) :

$$\frac{\partial u}{\partial t} + a \cdot \overrightarrow{\text{grad}} u = 0, \tag{12a}$$

$$\frac{\partial u}{\partial t} + \text{div}(au) = 0. \tag{12b}$$

---

Nous effectuons les calculs suivants

$$\begin{aligned} A &= \frac{\partial f}{\partial t}(x, v, t) + v \cdot \nabla_x f(x, v, t) + \nabla_v \cdot (F(x, v)f(x, v, t)) \\ &= \dots \\ A &= \frac{\partial f}{\partial t}(x, v, t) + v \cdot \nabla_x f(x, v, t) + \nabla_v \cdot (F_I(x, v)f(x, v, t)) + \nabla_v \cdot (F_F(x, v)f(x, v, t)) \end{aligned} \tag{13}$$

qui donnent, plus lisiblement

$$A = \frac{\partial f}{\partial t} + v \cdot \nabla_x f + \nabla_v \cdot (F_F f) + \nabla_v \cdot (F_I f). \tag{♣}$$


---

$$x \in \emptyset \iff x \in \varnothing \iff \begin{cases} x \not\leq 0 & \text{et} \\ x \neq 0 & \end{cases} x \not\geq 0$$


---

$$\text{div } E = \frac{\rho}{\varepsilon_0} \qquad \qquad \text{div } B = 0 \tag{14}$$

$$\overrightarrow{\text{curl}} E = -\partial_t B \qquad \qquad \overrightarrow{\text{curl}} B = \mu_0 \left[ \vec{j} + \varepsilon_0 \partial_t E \right] \tag{15}$$


---

## B.6 Où trouver de l'aide ?

Si vous avez besoin d'aide, sachez qu'il y a de fortes chances que quelqu'un ait déjà été confronté à votre problème avant vous : une réponse figure donc sans doute dans une quelconque documentation... .

- Parmi les polycopiés librement téléchargeables : polycopié de cours [**mpg**] et plus généralement tous les documents accessibles depuis la page web de l'auteur (dont les **beamer** auquel est rattaché ce polycopié), [**pg**] et plus particulièrement les “comptes-rendus des séances” qui fournissent des introductions à plein d'outils (dont tableaux, **TikZ**, **BIBTEX**, index, glossaires, personnalisation, listings informatiques, rudiments de programmation, raffinements mathématiques) et les “Documents de référence” ; **ces deux pages web destinées à des débutants suffiront amplement pour trouver plein d'informations utiles et efficaces.** Il y a aussi [4] (dont la version PDF est régulièrement actualisée sur Internet) et [12].
- Parmi les livres (niveaux débutant et intermédiaire) : le très pédagogique [**B&C**] (dont des annexes sont accessibles librement en ligne) ou [10].
- Parmi les FAQ : le PDF en cours de rédaction [11], qui est sans doute la plus complète en français.
- Pour ceux disposant d'un accès permanent à Internet, des FAQ et des forums (pour débutants) en ligne existent : la partie **L<sup>A</sup>T<sub>E</sub>X** des sites **forum.mathemateX.net** et **les-mathematiques.net/phorum/** par exemple.
- Pour les mathématiques, quelques pistes figurent au paragraphe 3.6.3.

Le dernier recours reste d'utiliser son moteur de recherche favori et de s'accrocher pour trouver la réponse la plus pertinente, en prenant garde au fait que la très populaire FAQ de l'équipe Grappa de Lille ne semble plus mise à jour depuis longtemps.

## Références

- [B&C] Denis BITOUZÉ, Jean-Côme CHARPENTIER. *L<sup>A</sup>T<sub>E</sub>X, l'essentiel*. Pearson Education France, 2010. (<http://latex-pearson.org/ressources-2010.php> pour les compléments librement accessibles en ligne : approfondissements, solutions des exercices du livre, etc.)
- [mpg] Manuel PÉGOURIÉ-GONNARD. *Apprentissage et pratique de L<sup>A</sup>T<sub>E</sub>X*, aout 2009 (polycopié de cours PDF sur <http://www.math.jussieu.fr/~mpg/lm204/>).
- [pg] Philippe GOUTET. <http://www.math.jussieu.fr/~goutet/latex/> (supports pédagogiques au module *LM204 – Initiation à L<sup>A</sup>T<sub>E</sub>X*).
- [1] Vincent BECK, Jérôme MALICK, Gabriel PEYRÉ. *Objectif Agrégation*, 2<sup>e</sup> édition. H&K, 2005.
- [2] Herbert Voß. *Math mode*, décembre 2010 (document *Mathmode.pdf* sur Internet).
- [3] American Mathematical Society. *User's Guide for the amsmath Package*, février 2002 (document *amsldoc.pdf* sur Internet).
- [4] Tobias OETIKER (pour la version anglaise), traduit en français. *Une courte (?) introduction à L<sup>A</sup>T<sub>E</sub>X 2<sub>E</sub>*, avril 2011 (document *lshort-fr.pdf* sur Internet).
- [5] Scott PAKIN. *The Comprehensive L<sup>A</sup>T<sub>E</sub>X Symbol List*, novembre 2009 (document *symbols-a4.pdf* sur Internet).
- [6] Site web Detexify (<http://detexify.kirelabs.org/>).
- [7] Gérard Tisseau, Jacques Duma. *TikZ pour l'impatient*, aout 2011 (document PDF sur <http://math.et.info.free.fr/TikZ/>).
- [8] Mc Collilieux. *Découverte de Beamer 3*, novembre 2007 (document PDF sur <http://mcclinews.free.fr/latex/introbeamer.php>).
- [9] *La nouvelle orthographe, parlons-en !*, septembre 2007 (document PDF sur <http://www.orthographe-recommandee.info/>).
- [10] Céline CHEVALIER et co-auteurs. *L<sup>A</sup>T<sub>E</sub>X pour l'impatient*. H&K, 2005.
- [11] *La FAQ de fr.comp.text.tex*, novembre 2004 (document PDF sur <http://faqfctt.fr.eu.org/>).
- [12] Vincent LOZANO. *Tout ce que vous avez toujours voulu savoir sur L<sup>A</sup>T<sub>E</sub>X sans jamais oser le demander*, octobre 2008 (document PDF sur <http://cours.enise.fr/info/latex/>).
- [13] Jacques ANDRÉ. *Petites leçons de typographie*, novembre 2008 (document PDF sur <http://jacques-andre.fr/>).

