

# L'extension frenchmath\*

Antoine Missier  
antoine.missier@ac-toulouse.fr

25 octobre 2022

## 1 Introduction

Cette extension, inspirée de `mafr` de Christian Obrecht [10], permet le respect des règles typographiques mathématiques françaises, en particulier la possibilité d'obtenir automatiquement les majuscules en romain (lettres droites) plutôt qu'en italique (voir [1] et [2]) ainsi que des espacements corrects pour les virgules et point-virgules<sup>1</sup>. Depuis la version 2.0, une option permet également de composer les minuscules grecques du mode mathématique en forme droite plutôt qu'en italique.

D'autres solutions pour composer les majuscules mathématiques en romain sont proposées dans les extensions `fourier` de Michel Bovani [12] (avec la famille des fontes Adobe Utopia) ou encore `mathdesign` de Paul Pichaureau [13] (avec les fontes Adobe Utopia, URW Garamond ou Bitstream Charter). Mais `frenchmath` fournit une solution générique s'adaptant à n'importe quelle fonte<sup>2</sup>. \$

D'autres préconisations, telles que composer en lettre droite et non en italique le symbole différentiel, les constantes mathématiques *i* et *e* [2], sont des règles internationales [6] [7] [9]. Elles ne sont donc pas implémentées dans `frenchmath`<sup>3</sup>.

L'extension fournit en outre diverses macros francisées. Quelques différences sont à signaler avec `mafr` :

- nous avons choisi de ne pas substituer les symboles français aux symboles anglo-saxons avec le même nom de commande ;
- les macros présentées dans la section 2 correspondant à des macros de `mafr` sont signalées par un astérisque en fin d'item, les autres sont nouvelles ;
- enfin quelques commandes de `mafr` ne sont pas spécifiques aux mathématiques françaises et ne sont donc pas abordées ici : c'est le cas de `\vect`<sup>4</sup>, des ensembles de nombres  $\mathbb{R}$ ,  $\mathbb{N}$ , ... (pour  $\mathbf{R}$ ,  $\mathbf{N}$ , ...) ainsi que celles relatives à la réalisation de feuilles d'exercices.

---

\*Ce document correspond à `frenchmath` v2.0, dernière modification le 25/10/2022.

1. Merci à Jean-François Burnol pour différentes améliorations proposées au code.

2. Si l'on souhaite néanmoins utiliser `mathdesign` avec `frenchmath`, il faut charger `frenchmath` après `mathdesign`.

3. Nous proposons pour cela l'extension `mismath` [23] qui fournit diverses macros pour les mathématiques internationales.

4. Pour de jolis vecteurs on dispose de l'extension `esvect` d'Eddie Sautrais [22].

Mentionnons aussi l'extension `tdsfrmath` [11] de Yvon Henel qui fournit beaucoup de commandes francisées.

## 2 Utilisation

### 2.1 Majuscules mathématiques

En France, les lettres majuscules du mode mathématique doivent toujours être composées en romain ( $A, B, C, \dots$ ) et non en italique ([1] p.107, voir aussi [2]). Il faut dire que cette convention n'est pas commode à mettre en œuvre, ni avec  $\text{\LaTeX}$ , ni avec les éditeurs de formule des traitements de textes usuels, et peu d'auteurs la respectent. La mise en œuvre automatique de cette recommandation, quelle que soit la fonte utilisée, est le principal bénéfice de `frenchmath`. Par défaut  $\text{\[ P(X)=\sum_{i=0}^n a_i X^i \]}$  donne avec `frenchmath`

$$P(X) = \sum_{i=0}^n a_i X^i.$$

`capsit` L'extension `frenchmath` possède l'option `capsit`, qui permet de conserver la composition des majuscules du mode mathématique en italique (dans la famille de fonte par défaut). Que l'option soit activée ou pas, il est toujours possible de changer l'aspect d'une lettre particulière, avec les macros  $\text{\LaTeX}$  `\mathrm` et `\mathit`.

### 2.2 Lettres grecques

La norme concernant l'usage des lettres grecques en italique ou en forme droite pour les mathématiques françaises ne semble pas très claire et les auteurs divergent sur ce point. Plusieurs recommandent l'usage des lettres grecques minuscules en forme droite [12] [13], mais d'autres préconisent l'italique, comme pour toutes les variables mathématiques [3]. Le lexique des règles typographiques en usage à l'Imprimerie Nationale [1] les compose en forme droite et relativement grasses (p.108) sans préciser s'il s'agit vraiment d'une règle s'appliquant aux variables, au même titre que celles énoncées pour l'alphabet latin.

Pour les physiciens (et chimistes) l'affaire est plus claire puisque les quantités doivent toujours être écrites en italique et les unités ou les constantes en romain (forme droite), conformément à la norme ISO [5] [6] [7]. Ainsi la constante  $\pi \approx 3,14$  ne s'écrit pas de la même manière qu'une variable  $\pi$ . Les extensions `mathdesign` [13] ou `fourier` [12] disposent d'une option permettant l'écriture automatique des lettres grecques minuscules en forme droite.

`upgreek` Avec la même philosophie, `frenchmath` fournit l'option `upgreek` basée sur l'extension `upgreek` de Walter Schmidt [14] qui donne accès aux lettres grecques minuscules en forme droite : `\upalpha`, `\upbeta`, ... L'extension `upgreek` sera chargée avec l'option par défaut, `Euler`. Si l'on veut, par contre, utiliser l'extension `upgreek` avec l'une de deux autres options disponibles, `Symbol` ou `Symbolsmallscale` (utilisant la fonte Adobe Symbol), il faut charger l'extension `upgreek` avec l'option

souhaitée *avant frenchmath*<sup>5</sup>. L’option `upgreek` de `frenchmath` redéfinit les commandes `\alpha`, `\beta`, ... pour produire automatiquement les lettres en forme droite  $\alpha$ ,  $\beta$ , ..., les formes italiques,  $\alpha$ ,  $\beta$ , ... restant cependant disponibles avec les commandes `\italpha`, `\itbeta`, ...

Mentionnons ce commentaire de Walter Schmidt [14] que le mu utilisé pour le préfixe des unités physiques,  $\mu$ , doit se composer avec `\textmu`, disponible en mode texte dans la plupart des fontes (ou avec `textcomp`) ; il diffère du  $\mu$  de `\upmu`.

**Upgreek**

Avec L<sup>A</sup>T<sub>E</sub>X, les lettres grecques majuscules sont automatiquement composées en forme droite, néanmoins il peut être utile de disposer de toutes les lettres grecques dans le même style ; ainsi l’extension `upgreek` propose aussi `\Upgamma`, ..., `Upomega` :  $\Gamma$ , ...,  $\Omega$ . L’option `Upgreek` de `frenchmath` redéfinit les majuscules `\Gamma`, ..., `\Omega` pour correspondre à ces variantes. Par contre l’on n’a alors plus accès aux caractères d’origine :  $\Gamma$ , ...,  $\Omega$ .

L’option `Upgreek` couvre aussi toutes les fonctionnalités des minuscules grecques de l’option `upgreek`, qu’il est donc inutile d’invoquer simultanément.

Les caractères grecs de l’extension `upgreek` ne se marient pas toujours de manière harmonieuse avec n’importe quelle fonte de texte. On peut vouloir charger une autre fonte de lettres grecques compatible avec la fonte de texte utilisée, si tant est qu’elle soit disponible. Il suffit alors simplement de charger l’extension voulue sans appeler l’option `upgreek` dans `frenchmath`. Citons `pxgreek` [15], `txgreek` [16] ou `libgreek` [17] de Jean-François Burnol, qui donnent de beaux résultats pour une utilisation avec respectivement les fontes Palatino, Times et Libertine. Mais attention ces extensions doivent être chargées *après* `amsmath` (et donc aussi `mismath`)<sup>6</sup>. Ces paquets fournissent également des options pour redéfinir les commandes de lettres grecques, minuscules comme majuscules.

## 2.3 Virgules et point-virgule

Dans le mode mathématique de L<sup>A</sup>T<sub>E</sub>X, la virgule est toujours, par défaut, un symbole de ponctuation et sera donc suivi d’une espace. Ceci est légitime dans un intervalle : `$(a,b)$` donne  $[a, b]$ , mais pas pour un nombre en français : `$12,5$` donne 12,5 au lieu de 12,5. L’extension `babel`, avec l’option `french` [18], fournit deux bascules : `\DecimalMathComma` et `\StandardMathComma`, qui permettent d’adapter le comportement de la virgule du mode mathématique. Deux autres extensions bien commodes permettent néanmoins de se passer de ces bascules<sup>7</sup>. En mode mathématique :

- avec `icomma` (intelligent comma) de Walter Schmidt [19], la virgule se comporte comme un caractère de ponctuation si elle est suivie d’une espace, sinon c’est un caractère ordinaire,
- avec `nccomma` de Alexander I. Rozhenko [20], la virgule se comporte comme un caractère ordinaire si elle est suivie d’un chiffre (sans espace), sinon c’est un caractère de ponctuation.

5. L’option `Symbol` de `upgreek` se marie mieux avec une fonte comme Times par exemple.

6. En effet, elles redéfinissent les commandes `\iint`, `\iiint`, `\iiint`, `\idotsint`.

7. Dans ce cas il ne faut pas utiliser les bascules, au risque de rendre ces extensions inopérantes.

Cette deuxième approche paraît meilleure, néanmoins `nccomma` ne fonctionne pas avec l'option `autolanguage`<sup>8</sup> de l'extension `numprint`. Par contre `icomma` fonctionne et était utilisé jusqu'à la version 1.5 de `frenchmath`. Un code proposé par Jean-François Burnol permet, depuis la version 1.6, d'utiliser `nccomma` qui sera donc chargée par `frenchmath` à la place de `icomma`.

Mentionnons enfin l'article *Intelligent commas* de Claudio Beccari [21] qui propose une solution simplifiée par rapport à `nccomma` mais dont l'avantage est discutable.

Lorsque l'on utilise l'extension `pstricks-add` de `PSTricks` pour tracer des axes de coordonnées, l'appel `\psset{comma=true}` permet d'avoir les graduations avec une virgule au lieu du point décimal. Ce réglage est effectué par défaut ici.

Le symbole « ; » a été redéfini pour le mode mathématique car l'espace précédant le point-virgule est incorrecte en français `$x \in [0,25 ; 3,75]$` donne  $x \in [0,25; 3,75]$  sans `frenchmath` et  $x \in [0,25 ; 3,75]$  avec `frenchmath`; le comportement de « ; » devient identique à celui de « : »<sup>9</sup>.

## 2.4 Quelques macros et alias utiles

<code>\curs</code>	Les lettres cursives ( $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \dots$ ), provenant de l'extension <code>mathrsfs</code> chargée par <code>frenchmath</code> , sont composées avec <code>\curs</code> et sont différentes de celles obtenues avec <code>\mathcal</code> <sup>10</sup> ( $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \dots$ ). La commande <code>\curs</code> permet aussi de composer ces lettres en mode texte, sans les délimiteurs du mode mathématique.* <sup>11</sup>
<code>\infeg</code> <code>\supeg</code>	Les relations $\leq$ et $\geq$ s'obtiennent avec les commandes <code>\infeg</code> et <code>\supeg</code> et diffèrent des versions anglaises de <code>\leq</code> ( $\leq$ ) et <code>\geq</code> ( $\geq$ ). Ce sont des alias de <code>\leqslant</code> et <code>\geqslant</code> de l'extension <code>amssymb</code> , chargée par <code>frenchmath</code> .*
<code>\vide</code>	Le symbole de l'ensemble vide $\emptyset$ s'obtient avec <code>\vide</code> (alias de la commande <code>\varnothing</code> de l'extension <code>amssymb</code> ); il diffère de celui obtenu avec <code>\emptyset</code> (particulièrement laid dans la fonte par défaut Computer Modern : $\emptyset$ ).*
<code>\paral</code>	La commande <code>\paral</code> fournit la <i>relation</i> <sup>12</sup> du parallélisme : $\parallel$ , plutôt que sa version anglaise <code>\parallel</code> : $\parallel$ .*
<code>\ssi</code>	La commande <code>\ssi</code> produit « si, et seulement si, ».
<code>\cmod</code>	Le modulo se compose normalement entre parenthèses, avec <code>\pmod</code> , mais on rencontre aussi, en français, le modulo entre crochets, ce que permet la commande <code>\cmod</code> en respectant le bon espacement propre au modulo : $5 \equiv 53 \pmod{12}$ .

8. L'option `autolanguage` de `numprint` utilisée conjointement avec l'option `french` de `babel` garantit un espacement correct entre les groupes de trois chiffres dans les grands nombres, qui doit être une espace insécable et non dilatable [1], légèrement plus grande que l'espace que l'on obtient sans cette option.

9. Un autre problème d'espacement, non spécifique au français, se pose avec les délimiteurs [ et ], par exemple  $x \in ]0, \pi[$ . L'extension `mismath` corrige ce problème.

10. L'extension `calrsfs` fournit les mêmes cursives mais en redéfinissant la commande `\mathcal`.

11. Comme précisé dans l'introduction, l'astérisque en fin d'item signale une fonctionnalité similaire dans `mafr`.

12. Pour noter que deux objets sont perpendiculaires, on utilise `\perp`, défini comme une *relation* mathématique plutôt que `\bot` défini comme un *symbole* (les espacements diffèrent).

## 2.5 Identifiants de « fonctions » classiques

- `\pgcd`    En arithmétique, nous avons les classiques `\pgcd` et `\ppcm`, qui diffèrent de `\ppcm` leur version anglo-saxonne `\gcd` et `\lcm`<sup>13</sup>.
- `\card`    Pour le cardinal d'un ensemble, nous proposons `\card`, cité dans [1] et [3], ou `\Card` `\Card`, qui est aussi d'usage courant (cf. Wikipedia).
- `\Ker`     $\text{\LaTeX}$  fournit les macros `\ker` et `\hom`, alors que l'usage français est souvent `\Hom` de commencer ces noms par une majuscule pour obtenir `Ker`<sup>14</sup> et `Hom`.
- `\rg`    Le rang d'une application linéaire ou d'une matrice (`rg`) s'obtient avec la commande `\rg` et l'espace vectoriel engendré par une famille de vecteurs avec `\Vect`.
- `\ch`    En principe, les fonctions hyperboliques s'écrivent en français avec les macros `\sh`  $\text{\LaTeX}$  standard `\cosh`, `\sinh`, `\tanh`. Néanmoins les écritures `ch x`, `sh x` et `th x`, qui sont la norme avec les langues d'Europe de l'Est [8], sont aussi utilisées en France [1]. On les obtient avec les commandes `\ch`, `\sh` et `\th`<sup>15</sup>.

## 2.6 Bases et repères

- `\Oij`    Les repères classiques du plan ou de l'espace seront composés avec des hauteurs de flèches homogénéisées : `\Oij` compose  $(O, \vec{i}, \vec{j})$ , `\Oijk` compose  $(O, \vec{i}, \vec{j}, \vec{k})$  et `\Ouv` `\Ouv` compose  $(O, \vec{u}, \vec{v})$  (utilisé dans le plan complexe). On peut écrire ces commandes en mode texte, sans les délimiteurs du mode mathématique.
- `\Oij*`    Les versions étoilées utilisent le point-virgule et non la virgule comme séparateur après le point `O`, comme mentionné dans [1]. On obtient  $(O ; \vec{i}, \vec{j})$ ,  $(O ; \vec{i}, \vec{j}, \vec{k})$ , `\Oijk*`  $(O ; \vec{u}, \vec{v})$ .
- `\ij`    Enfin les macros `\ij` et `\ijk` composent les bases du plan et de l'espace,  $(\vec{i}, \vec{j})$  `\ijk` et  $(\vec{i}, \vec{j}, \vec{k})$ , en homogénéisant la hauteur des flèches. Notons que la macro `\ij` existait déjà (ligature entre `i` et `j` pour le hollandais) et a été redéfinie.

## 3 Le code

```

1 \RequirePackage{ifthen}
2 \newboolean{capsit}
3 \DeclareOption{capsit}{\setboolean{capsit}{true}}
4 \newboolean{upgreek}
5 \newboolean{Upgreek}
6 \DeclareOption{upgreek}{\setboolean{upgreek}{true}}
7 \DeclareOption{Upgreek}{\setboolean{upgreek}{true}}
8   \setboolean{Upgreek}{true}}
9 \ProcessOptions \relax
10
```

13. Cette dernière n'est pas implémentée en standard dans  $\text{\LaTeX}$  (mais dans `mismath`).

14. La commande `\Im` existe déjà pour la partie imaginaire des nombres complexes et produit  $\Im$ ; elle est redéfinie en `Im` par l'extension `mismath` et peut aussi être utilisée pour l'image.

15. La commande `\th` existe déjà (pour le mode texte uniquement); elle a été redéfinie, uniquement pour le mode mathématique, à la place de l'ancienne commande `\tgh`, désormais obsolète.

```

11 \@ifpackageloaded{mathdesign}{
12   \PackageWarningNoLine{frenchmath}{Package mathdesign is loaded,
13     I don't load mathrsfs and amssymb packages}
14 }{
15   \RequirePackage{mathrsfs} % fournit les majuscules cursives
16   \RequirePackage{amssymb} % fournit \leqslant, \geqslant, \varnothing
17 }
18 \RequirePackage{amsopn} % fournit \DeclareMathOperator
19 \RequirePackage{xspace} % utile pour les commandes \curs, \ssi, \Oij
20 % \RequirePackage{icomma}
21 \RequirePackage{nccomma} % virgule intelligente
22 \AtBeginDocument{\mathcode'\,="8000\relax
23   \@ifpackageloaded{babel}{
24     \addto\extrasfrench{\mathcode'\,="8000\relax}
25   }{}
26 }

```

Merci à J.F. Burnol pour ce patch permettant d'utiliser nccomma à la place de icomma, ainsi que pour l'amélioration du code ci-après, redéfinissant les majuscules mathématiques.

Sauf si l'option capsit est activée, on redéfinit toutes les lettres majuscules du mode mathématique ; \AtBeginDocument est nécessaire pour que ces définitions soient prises en compte avec la classe beamer par exemple.

```

27 \ifthenelse{\boolean{capsit}}{
28   \AtBeginDocument{
29     \DeclareMathSymbol{A}\mathalpha{operators}{'A}
30     \DeclareMathSymbol{B}\mathalpha{operators}{'B}
31     \DeclareMathSymbol{C}\mathalpha{operators}{'C}
32     \DeclareMathSymbol{D}\mathalpha{operators}{'D}
33     \DeclareMathSymbol{E}\mathalpha{operators}{'E}
34     \DeclareMathSymbol{F}\mathalpha{operators}{'F}
35     \DeclareMathSymbol{G}\mathalpha{operators}{'G}
36     \DeclareMathSymbol{H}\mathalpha{operators}{'H}
37     \DeclareMathSymbol{I}\mathalpha{operators}{'I}
38     \DeclareMathSymbol{J}\mathalpha{operators}{'J}
39     \DeclareMathSymbol{K}\mathalpha{operators}{'K}
40     \DeclareMathSymbol{L}\mathalpha{operators}{'L}
41     \DeclareMathSymbol{M}\mathalpha{operators}{'M}
42     \DeclareMathSymbol{N}\mathalpha{operators}{'N}
43     \DeclareMathSymbol{O}\mathalpha{operators}{'O}
44     \DeclareMathSymbol{P}\mathalpha{operators}{'P}
45     \DeclareMathSymbol{Q}\mathalpha{operators}{'Q}
46     \DeclareMathSymbol{R}\mathalpha{operators}{'R}
47     \DeclareMathSymbol{S}\mathalpha{operators}{'S}
48     \DeclareMathSymbol{T}\mathalpha{operators}{'T}
49     \DeclareMathSymbol{U}\mathalpha{operators}{'U}
50     \DeclareMathSymbol{V}\mathalpha{operators}{'V}
51     \DeclareMathSymbol{W}\mathalpha{operators}{'W}
52     \DeclareMathSymbol{X}\mathalpha{operators}{'X}
53     \DeclareMathSymbol{Y}\mathalpha{operators}{'Y}

```

```

54     \DeclareMathSymbol{Z}\mathalpha{operators}'{Z}
55   }
56 }

```

Avec l'option `upgreek`, on charge l'extension `upgreek` (sauf si elle est déjà chargée, ce qui évite les incompatibilités d'option) et on redéfinit les commandes `\alpha`, `\beta`, ... `Upgreek` transforme en outre les majuscules grecques pour garder le même style.

```

57
58 \ifthenelse{\boolean{upgreek}}{
59   \@ifpackageloaded{upgreek}{}{\RequirePackage[Euler]{upgreek}}
60   \@ifundefined{italpha}{\let\italpha\alpha}{
61     \PackageWarningNoLine{frenchmath}{Italicized Greek \bslash italpha
62       also defined, no more warning for the other letters}
63   }
64   \@ifundefined{itbeta}{\let\itbeta\beta}{}
65   \@ifundefined{itgamma}{\let\itgamma\gamma}{}
66   \@ifundefined{itdelta}{\let\itdelta\delta}{}
67   \@ifundefined{itepsilon}{\let\itepsilon\epsilon}{}
68   \@ifundefined{itzeta}{\let\itzeta\zeta}{}
69   \@ifundefined{iteta}{\let\iteta\eta}{}
70   \@ifundefined{ittheta}{\let\ittheta\theta}{}
71   \@ifundefined{itiota}{\let\itiota\iota}{}
72   \@ifundefined{itkappa}{\let\itkappa\kappa}{}
73   \@ifundefined{itlambda}{\let\itlambda\lambda}{}
74   \@ifundefined{itmu}{\let\itmu\mu}{}
75   \@ifundefined{itnu}{\let\itnu\nu}{}
76   \@ifundefined{itxi}{\let\itxi\xi}{}
77   \@ifundefined{itpi}{\let\itpi\pi}{}
78   \@ifundefined{itrho}{\let\itrho\rho}{}
79   \@ifundefined{itsigma}{\let\itsigma\sigma}{}
80   \@ifundefined{ittau}{\let\ittau\tau}{}
81   \@ifundefined{itupsilon}{\let\itupsilon\upsilon}{}
82   \@ifundefined{itphi}{\let\itphi\phi}{}
83   \@ifundefined{itchi}{\let\itchi\chi}{}
84   \@ifundefined{itpsi}{\let\itpsi\psi}{}
85   \@ifundefined{itomega}{\let\itomega\omega}{}
86   \@ifundefined{itvarepsilon}{\let\itvarepsilon\varepsilon}{}
87   \@ifundefined{itvartheta}{\let\itvartheta\vartheta}{}
88   \@ifundefined{itvarpi}{\let\itvarpi\varpi}{}
89   \@ifundefined{itvarsigma}{\let\itvarsigma\varsigma}{}
90   \@ifundefined{itvarphi}{\let\itvarphi\varphi}{}
91
92   \renewcommand\alpha{\upalpha}
93   \renewcommand\beta{\upbeta}
94   \renewcommand\gamma{\upgamma}
95   \renewcommand\delta{\updelta}
96   \renewcommand\epsilon{\upepsilon}
97   \renewcommand\zeta{\upzeta}
98   \renewcommand\eta{\upeta}

```

```

99     \renewcommand\theta{\uptheta}
100    \renewcommand\iota{\upiota}
101    \renewcommand\kappa{\upkappa}
102    \renewcommand\lambda{\uplambda}
103    \renewcommand\mu{\upmu}
104    \renewcommand\nu{\upnu}
105    \renewcommand\xi{\upxi}
106    \renewcommand\pi{\uppi}
107    \renewcommand\rho{\uprho}
108    \renewcommand\sigma{\upsigma}
109    \renewcommand\tau{\uptau}
110    \renewcommand\upsilon{\upupsilon}
111    \renewcommand\phi{\upphi}
112    \renewcommand\chi{\upchi}
113    \renewcommand\psi{\uppsi}
114    \renewcommand\omega{\upomega}
115    \renewcommand\varepsilon{\upvarepsilon}
116    \renewcommand\vartheta{\upvartheta}
117    \renewcommand\varpi{\upvarpi}
118    \renewcommand\varrho{\upvarrho}
119    \renewcommand\varsigma{\upvarsigma}
120    \renewcommand\varphi{\upvarphi}
121  }{}
122
123  \ifthenelse{\boolean{Upgreek}}{
124    \renewcommand\Gamma{\Upgamma}
125    \renewcommand\Delta{\Updelta}
126    \renewcommand\Theta{\Uptheta}
127    \renewcommand\Lambda{\Uplambda}
128    \renewcommand\Xi{\Upxi}
129    \renewcommand\Pi{\Upipi}
130    \renewcommand\Sigma{\Upsigma}
131    \renewcommand\Upsilon{\Upupsilon}
132    \renewcommand\Phi{\Upphi}
133    \renewcommand\Psi{\Uppsi}
134    \renewcommand\Omega{\Upomega}
135  }{}
136
137  \DeclareMathSymbol{;}\mathbin{operators}'73} % \mathpunct à l'origine
138  \AtBeginDocument{\ifpackageloaded{pstricks-add}{\psset{comma=true}}{}}
139
140  \newcommand*\curs[1]{\ensuremath{\mathscr{#1}}\xspace}
141  \newcommand\infeg{\leqslant}
142  \newcommand\supieg{\geqslant}
143  \newcommand\vide{\varnothing}
144  \newcommand\paral{\mathrel{/\!/}} % \parallel existe déjà : ||
145  \newcommand\ssi{si, et seulement si,\xspace}
146  \newcommand*\cmod[1]{\quad[1]}
147
148  \DeclareMathOperator{\pgcd}{pgcd}

```

```

149 \DeclareMathOperator{\ppcm}{ppcm}
150 \DeclareMathOperator{\card}{card}
151 \DeclareMathOperator{\Card}{Card}
152 \DeclareMathOperator{\Ker}{Ker}
153 \DeclareMathOperator{\Hom}{Hom}
154 \DeclareMathOperator{\rg}{rg}
155 \DeclareMathOperator{\Vect}{\Vect}
156 \DeclareMathOperator{\ch}{ch}
157 \DeclareMathOperator{\sh}{sh}
158 \AtBeginDocument{\let\oldth\th %\th existe déjà (mode texte uniquement)
159 \renewcommand{\th}{\TextOrMath{\oldth}{\mathop{\operatorfont th}}}}
160 {\catcode'\|=z@ \catcode'\=12 \gdef\bslash{\} % la commande \bslash
161 \PackageWarningNoLine{frenchmath}{
162     Command \bslash tgh in now obsolete, use \bslash th instead}
    La commande \bslash provient du package doc.sty de Frank Mittelbach. Elle est
    utilisée à la place de \textbackslash (qui ne fonctionne pas ici).
163
164 \newcommand\@Oij{%
165     \ensuremath{\left(0, \vec{\imath}, \vec{\jmath}\,,\right)\xspace}
166 \newcommand\@OOij{%
167     \ensuremath{\left(0 ; \vec{\imath}, \vec{\jmath}\,,\right)\xspace}
168 \newcommand\Oij{\@ifstar{\@OOij}{\@Oij}}
169
170 \newcommand\@Oijk{%
171     \ensuremath{%
172         \left(0, \vec{\vphantom{t}\imath}, \vec{\vphantom{t}\jmath},
173         \vec{\vphantom{t}\smash{k}}\,,\right)%
174     \xspace}
175 \newcommand\@OOijk{%
176     \ensuremath{%
177         \left(0 ; \vec{\vphantom{t}\imath}, \vec{\vphantom{t}\jmath},
178         \vec{\vphantom{t}\smash{k}}\,,\right)%
179     \xspace}
180 \newcommand\Oijk{\@ifstar{\@OOijk}{\@Oijk}}
181
182 \newcommand\@Ouv{%
183     \ensuremath{\left(0, \vec{u}, \vec{v}\,,\right)\xspace}
184 \newcommand\@OOuv{%
185     \ensuremath{\left(0 ; \vec{u}, \vec{v}\,,\right)\xspace}
186 \newcommand\Ouv{\@ifstar{\@OOuv}{\@Ouv}}
187
188 \AtBeginDocument{
189     \renewcommand\ij{%
190         \ensuremath{\left(\vec{\imath}, \vec{\jmath}\,,\right)\xspace}}
191 \newcommand\ijk{%
192     \ensuremath{%
193         \left(\vec{\vphantom{t}\imath}, \vec{\vphantom{t}\jmath},
194         \vec{\vphantom{t}\smash{k}}\,,\right)%
195     \xspace}

```

## Références

- [1] *Lexique des règles typographiques en usage à l'Imprimerie Nationale*. Édition du 26/08/2002.
- [2] *Composition des textes scientifiques*. Inspection générale de mathématiques (IGEN-DESCO), 06/12/2001.
- [3] *Règles françaises de typographie mathématique*. Alexandre André, 02/09/2015.
- [4] *Le petit typographe rationnel*. Eddie Saudrais, 20/03/2000.
- [5] *Norme ISO 31-11*, 1992 et sa révision *ISO 80000-2*, 2009.
- [6] *Typesetting mathematics for science and technology according to ISO 31/XI*, Claudio Beccari, TUGboat Volume 18 (1997), N° 1.
- [7] *On the Use of Italic and up Fonts for Symbols in Scientific Text*, I.M. Mills and W.V. Metanomski, ICTNS (Interdivisional Committee on Terminology, Nomenclature and Symbols), dec 1999.
- [8] *L<sup>A</sup>T<sub>E</sub>X Companion*. Frank Mittelbach, Michel Goossens, 2<sup>e</sup> édition, Pearson Education France, 2005.
- [9] *The Not So Short Introduction to L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*. Manuel L<sup>A</sup>T<sub>E</sub>X de Tobias Oetiker, Hubert Partl, Irene Hyna et Elisabeth Schlegl, CTAN, v6.2 28/02/2018.
- [10] *La distribution mafr*. Christian Obrecht, CTAN, v1.0 17/09/2006.
- [11] *L'extension tdsfrmath*. Yvon Henel, CTAN, v1.3 22/06/2009.
- [12] *Fourier-GUTenberg*. Michel Bovani, CTAN, v1.3 30/01/2005.
- [13] *The mathdesign package*. Paul Pichaureau, CTAN, 29/08/2013.
- [14] *The upgreek package for L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*. Walter Schmidt, CTAN, v2.0 12/03/2003.
- [15] *The pxgreek package*. Jean-François Burnol, CTAN, v1.0 16/03/2011.
- [16] *The txgreek package*. Jean-François Burnol, CTAN, v1.0 16/03/2011.
- [17] *The libgreek package*. Jean-François Burnol, CTAN, v1.0 14/03/2011.
- [18] *A Babel language definition file for French*. Extension L<sup>A</sup>T<sub>E</sub>X babel-french de Daniel Flipo, CTAN, v3.5c 14/09/2018.
- [19] *The icomma package for L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*. Walter Schmidt, CTAN, v2.0 10/03/2002.
- [20] *The nccomma package*. Alexander I. Rozhenko, CTAN, v1.0 10/02/2005.
- [21] *Intelligent commas*. Claudio Beccari, The PracT<sub>E</sub>X Journal, 2011, No.1
- [22] *Typesetting vectors with beautiful arrow with L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*. Extension L<sup>A</sup>T<sub>E</sub>X esvect d'Eddie Saudrais, CTAN, v1.3 11/07/2013.
- [23] *mismath – Miscellaneous mathematical macros*. Antoine Missier, CTAN, v1.10 25/10/2022.