

Guide de l'utilisateur de `simples-matrices`*

Yvon Henel[†]

19 juin 2022

Résumé

Une extension pour écrire des matrices en donnant les coefficients par ligne sous la forme d'une liste de valeurs séparées par des virgules.

Une macro permet de définir des matrices nommées et une autre permet d'écrire les matrices nommées. L'extension fournit également quelques raccourcis pour les matrices identité et les matrices nulles.

Abstract

A package to write matrices which are defined with a list of comma separated coefficients read by row.

A macro enables the definition of a named matrix, another enables the writing of a named matrix. This package provides also some shortcuts for identity matrices and null matrices.

The name of this package and of its macros are French based for there are already too many macros using the word "matrix" itself. The French "simples matrices" means "simple matrices". *Just a letter apart!*

The English documentation for the final user of the package `simples-matrices` is available in the file `simples-matrices-eng`.

Table des matières

1	Les macros	2
1.1	Macros principales	2
1.2	Valuer les clés	3
1.3	Raccourcis	3
2	Les options de l'extension	4

*Ce fichier décrit la version 1.0, dernière révision 2022/06/19.

En souvenir de ma grand-mère maternelle, Adrienne BINAUT (1908-03-23 – 1997-06-08).

[†]E-mail : le.texnicien.de.surface@yvon-henel.fr

3	Exemples	4
3.1	Sans extension particulière	4
3.1.1	Déclaration et utilisation	4
3.1.2	L'argument « type »	4
3.1.3	Utilisation de la clé « typeord »	5
3.1.4	Utilisation des macros <code>\(La)MatriceInterieur</code>	6
3.2	Changer l'aspect	6
3.2.1	Avec <code>mathtools</code>	6
3.2.2	Avec <code>delarray</code>	6
3.2.3	Avec <code>nicematrix</code>	7

L'extension `simples-matrices` utilise `xparse` et `l3keys2e` pour définir ses macros et gérer les clés d'option. Elle charge également `amsmath` pour obtenir une présentation convenable des matrices.

Attention : vous devez fournir l'environnement mathématique adéquat pour utiliser les macros qui écrivent les matrices. Seule la version non-étoilée de `\declarermatrice` peut être utilisée en dehors du mode mathématique.

1 Les macros

L'extension `simples-matrices` offre six macros de document.

1.1 Macros principales

Les cinq macros principales de document sont les suivantes :

```
\matrice <{prefixe}> <{vliste de paires de clé-valeur}> [(type)]{vliste des coefficients}
```

où `<{prefixe}>` a la même fonction que la clé `prefix` ; `<{vliste de paires de clé-valeur}>`, optionnel et vide par défaut, peut être utilisé pour redéfinir l'environnement de `matrix` ; `<{type}>` est une chaîne de caractère dont l'utilisation sera expliquée ci-dessous — voir [3.1.2](#) — et `<{vliste des coefficients}>` est la vliste des coefficients de la matrice donnée ligne par ligne.

Avec `\(\matrice{1, 2, 3, 4}\)` on obtient $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$.

Avec `\(\matrice(b)[3]{1, 2, 3, 4, 5, 6}\)` on obtient $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$.

```
\declarermatrice <{prefixe}> <{vliste de paires de clé-valeur}> {nom}[(type)]{vliste des coefficients}
```

où `<{nom}>` est le nom de la matrice pour utilisation future. Les autres arguments, optionnels ou obligatoires, ont la même signification que ci-dessus. La version étoilée définit et écrit la matrice. La version non-étoilée ne fait que la définir.

La définition est globale mais on peut redéfinir une matrice nommée existante avec la même fonction. **Aucun contrôle** n'est exercé pour s'assurer que l'on n'est pas en train de redéfinir une matrice nommée précédemment définie.

`\lamatrice` `\lamatrice(<prefix> <vliste de paires de clé-valeur> <{matrix name}>)`
écrit la matrice nommée de nom `<nom>`. Les deux arguments optionnels ont la même signification que ci-dessus.

`\MatriceInterieur` `\MatriceInterieur`
donne l'intérieur de la dernière matrice écrite — via `\matrice` — ou définie — via `\declarermatrice`. On peut l'utiliser dans un environnement du genre `array`.

`\LaMatriceInterieur` `\LaMatriceInterieur{<nom>}`
donne l'intérieur de la matrice de nom `<nom>`. On peut l'utiliser dans un environnement du genre `array`.
Voir les exemples en page 4.

1.2 Valuer les clés

On peut changer les valeurs des clés de simples-matrices avec

`\simplesmatricessetup` `\simplesmatricessetup{<vliste de paires de clé-valeur>}`
où `<vliste de paires de clé-valeur>` est l'habituelle `vliste` de paires de clé-valeur valuant une ou plusieurs des trois clés de cette extension telles que présentées ci-dessous en page 4.
Pour suivre une convention désormais bien établie le nom de cette macro consiste en le nom de l'extension — réduite à ces lettres T_EXiennes — suivi de l'anglais *setup*. Je demande pardon pour cette étrange mixture linguistique.

1.3 Raccourcis

L'extension propose deux macros *raccourcis* :

`\matid` `\matid(<prefix> <vliste de paires de clé-valeur> <[coefficient]>{<nombre de colonnes>})`
qui imprime la matrice identité — par défaut — avec `<nombre de colonnes>` colonnes. Si `<coefficient>` est fourni, on obtient une matrice diagonale dont tous les coefficients non-nuls sont égaux à `<coefficient>`.

Les deux premiers arguments optionnels ont la même fonction que dans les macros précédentes.

`\matnulle` `\matnulle(<prefix> <vliste de paires de clé-valeur> <[coefficient]>{<nombre de colonnes>})`
imprime la matrice nulle `<nombre de colonnes>` colonnes — par défaut — ou une matrice de `<nombre de colonnes>` colonnes dont tous les coefficients sont égaux à `<coefficient>`.
Les deux premiers arguments optionnels ont la même fonction que dans les macros précédentes.

<code>\(\matid{3}\)\qquad \(\matid[9]{3}\)</code> <code>\par \medskip</code> <code>\(\matnulle{3}\)\qquad \(\matnulle[2]{3}\)</code>	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 9 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 9 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{pmatrix}$
--	---	---

2 Les options de l'extension

L'extension utilise le système de clefs-valeurs pour ses options. Elle possède quatre clés : `envir`, `prefix`, `typeord` et `argopt`.

`envir` (*chaîne de caractères*) la partie finale du nom de l'environnement utilisé pour imprimer la matrice. Sa valeur initiale est `matrix`.

`prefix` (*chaîne de caractères*) , caractères placés devant `envir` pour fournir le nom complet de l'environnement utilisé pour imprimer la matrice. Sa valeur initiale est `p`. Cela fait que l'environnement par défaut est `pmatrix` fourni par `amsmath`.

`typeord` (*chaîne de caractères*) définit la valeur ordinaire — c.-à-d. par défaut — de l'argument optionnel *⟨type⟩*. La valeur initiale de la clé est `C`.

`argopt` (*liste de lexèmes, anglais : token list*) . Cette clé est vide par défaut. Voir page 6 pour son utilisation.

De plus, une clé supplémentaire existe qui est une métaclé : `out-of-box`.

Avec `\simplesmatricessetup{out-of-box}` on obtient le même effet qu'avec `\simplesmatricessetup{prefix=p, envir=matrix, argopt=, typeord=C}`.

3 Exemples

3.1 Sans extension particulière

C'est-à-dire en chargeant uniquement `amsmath` comme le fait cette extension.

3.1.1 Déclaration et utilisation

<pre>\declarermatrice{A}{1, 2, 300, 400} \(\declarermatrice*{B}{a, b, c, d}\) \qqquad \(\lamatrice{A}\) \par \bigskip \(\lamatrice(V){A}\) \qqquad \(\lamatrice(b){B}\)</pre>	$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \begin{pmatrix} 1 & 2 \\ 300 & 400 \end{pmatrix}$ $\left\ \begin{matrix} 1 & 2 \\ 300 & 400 \end{matrix} \right\ \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix}$
---	--

3.1.2 L'argument « type »

L'avant dernier argument — optionnel — des macros `\matrice` et `\declarermatrice` vaut par défaut `0`. Dans ce cas, c'est la valeur de la clé `typeord` qui est utilisée pour déterminer le *type* des données de la matrice, voir 5.

Sinon, il peut prendre une valeur numérique qui est le nombre de colonnes de la matrice ou encore une chaîne de caractères — à ce jour, une seule lettre — parmi : `C`, `D`, `I`, `J`, `S`, `T` et `x`.

`C` signifie matrice **carrée**.

<pre>\[\matrice{1, 2, 3, 4, 5, 6, 7, 8, 9} \qqquad \matrice[C]{1, 2, 3, 4, 5, 6, 7, 8, 9} \]</pre>	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$
---	---

D signifie matrice **diagonale**.

$\begin{array}{l} \backslash[\\ \quad \backslashmatrice{1, 2, 3, 4}\qqquad \\ \quad \backslashmatrice[D]{1, 2, 3, 4} \\ \backslash] \end{array}$	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$
---	--	--

I signifie matrice triangulaire **inférieure** ; J signifie matrice triangulaire inférieure avec des zéros sur la diagonale.

$\begin{array}{l} \backslash[\\ \quad \backslashmatrice[I]{1, 2, 3}\qqquad \\ \quad \backslashmatrice[J]{1, 2, 3} \\ \backslash] \end{array}$	$\begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 2 & 3 & 0 \end{pmatrix}$
--	--	---

S signifie matrice triangulaire **supérieure** ; T signifie matrice triangulaire supérieure avec des zéros sur la diagonale.

$\begin{array}{l} \backslash[\\ \quad \backslashmatrice[S]{1, 2, 3}\qqquad \\ \quad \backslashmatrice[T]{1, 2, 3} \\ \backslash] \end{array}$	$\begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 2 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \end{pmatrix}$
--	--	---

x signifie **xcas**. Avec cette valeur, on peut copier-coller de **xcas** dans le document \LaTeX .

$\begin{array}{l} \backslash[\\ \quad \backslashmatrice[x]{\{[[7,1,3],[1,0,3],[5,1,2]]\}} \\ \backslash] \end{array}$	$\begin{pmatrix} 7 & 1 & 3 \\ 1 & 0 & 3 \\ 5 & 1 & 2 \end{pmatrix}$
--	---

Un nombre donne le nombre de colonnes de la matrice.

$\begin{array}{l} \backslash[\\ \quad \backslashmatrice[2]{1, 2, 3, 4, 5, 6} \\ \quad \backslashquad \\ \quad \backslashmatrice[3]{1, 2, 3, 4, 5, 6} \\ \backslash] \end{array}$	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$
---	---	--

3.1.3 Utilisation de la clé « typeord »

Dans ce document, la clé `typeord` a comme valeur initiale `C`.

$\begin{array}{l} \backslashsimplismatricessetup{typeord=3} \\ \backslash(\backmatrice{1, 2, 3, 4, 5, 6}\backslash) \\ \backslashsimplismatricessetup{typeord=2} \\ \backslash(\backmatrice{1, 2, 3, 4, 5, 6}\backslash) \\ \backslashsimplismatricessetup{typeord=C} \\ \backslash(\backmatrice{1, 2, 3, 4}\backslash) \end{array}$	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$
--	--	---	--

Le code précédent n'est, bien entendu, donné qu'à titre d'illustration. Pour une déclaration de type ne concernant qu'une seule matrice, on aura intérêt à utiliser l'argument optionnel. Cependant si on veut rédiger un cours sur les matrices triangulaires supérieures...

3.1.4 Utilisation des macros `\(La)MatriceInterieur`

```

\declearmatrice{B}[4]{
\hline ein, zwei, drei, vier,
\hline unos, dos, tres, cuatro,
\hline unu, du, tri, kvar}
\declearmatrice{A}[3]{one, two, three,
un, deux, trois, uno, due, tre}
\begin{tabular}{*{3}{1}}
\MatriceInterieur
\end{tabular}
\par \medskip
\begin{tabular}{|*{4}{1|}}
\LaMatriceInterieur{B}\hline
\end{tabular}

```

one two three
un deux trois
uno due tre

ein	zwei	drei	vier
unos	dos	tres	cuatro
unu	du	tri	kvar

3.2 Changer l'aspect

On peut charger d'autres extensions qui permettent de présenter des matrices comme `mathtools`, `delarray` ou `nicematrix`. Dans ce cas on peut changer l'aspect de nos matrices grâce aux clés d'option `envir`, `prefix` et `argopt`.

Ce document charge les trois extensions `mathtools`, `delarray` et `nicematrix` pour présenter les exemples suivants.

3.2.1 Avec `mathtools`

Comme déjà signalé, il y a deux façons d'utiliser les clés d'option : à l'aide de `\simplesmatricessetup` ou en utilisant l'argument optionnel `< \liste de paires de clé-valeur >`. Je montre les deux.

```

\simplesmatricessetup{
  envir=matrix*, prefix=p, argopt=[r]}
\(\matrice{1, 2, 30, 40}\) \quad
\simplesmatricessetup{out-of-box}
\(\matrice{1, 2, 30, 40}\)
\par \medskip
\(\matrice(b)<envir=matrix*, argopt=[l]>
  {1, 2, 30, 40}\)

```

$\begin{pmatrix} 1 & 2 \\ 30 & 40 \end{pmatrix}$ $\begin{pmatrix} 1 & 2 \\ 30 & 40 \end{pmatrix}$
 $\begin{bmatrix} 1 & 2 \\ 30 & 40 \end{bmatrix}$

3.2.2 Avec `delarray`

```

\simplesmatricessetup{
  envir=array, prefix=,
  argopt=[c][1 c r]\}
\(\matrice{1, 2, 30, 40}\) \quad
\(\matid(<envir=array,
  argopt={c}[1 c r]\})>{3}\)

```

$\left[\begin{array}{cc} 1 & 2 \\ 30 & 40 \end{array} \right]$ $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

3.2.3 Avec nicematrix

```
\(\matrice<prefix=b, envir=NiceArray,  
argopt={*{2}{c}|cc}>{1, 2, 3, 4, 5, 6,  
7, 8, 9, 10, 11, 12, 13, 14, 15, 16}\)
```

$$\left[\begin{array}{cc|cc} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{array} \right]$$

Dans ce cas il faut doubler les accolades entourant les descripteurs de colonnes de l'environnement `array` car \LaTeX retire un niveau d'accollades en valuant la clé.

Dans l'exemple suivant, pour la deuxième matrice, il n'est pas nécessaire d'entourer la valeur de `argopt` d'une paire d'accollades — même si ça n'est pas dangereux — car la présence de `[margin]` empêche \LaTeX de supprimer une paire d'accollades.

```
\(\matrice<prefix=b, envir=NiceArray,  
argopt={{1 c r}}>{1, 2, 3, 40, 50, 60,  
700, 800, 900}\)  
\par \medskip  
\(\matrice<prefix=b, envir=NiceArray,  
argopt={1 c r}[margin]>{1, 2, 3,  
40, 50, 60, 700, 800, 900}\)  
\par \medskip  
\(\matrice<prefix=b, envir=NiceArray,  
argopt={{1 c r}[margin]}>{1, 2, 3,  
40, 50, 60, 700, 800, 900}\)
```

$$\left[\begin{array}{ccc} 1 & 2 & 3 \\ 40 & 50 & 60 \\ 700 & 800 & 900 \end{array} \right]$$

$$\left[\begin{array}{ccc} 1 & 2 & 3 \\ 40 & 50 & 60 \\ 700 & 800 & 900 \end{array} \right]$$

$$\left[\begin{array}{ccc} 1 & 2 & 3 \\ 40 & 50 & 60 \\ 700 & 800 & 900 \end{array} \right]$$

Le TeXnicien de Surface scripsit.
