

Utiliser SageTeX pour l'intégration de Sage et L^AT_EX

Nadia Lafrenière

1^{er} juillet 2017

Note : Plusieurs exemples de ce fichier sont issus de la page *Utiliser SageTex* du Tutoriel Sage [Tea].

1 Pourquoi utiliser SageTeX

SageTeX est un paquet de Sage qui permet de faire des calculs avec Sage en plaçant automatiquement les résultats dans un document L^AT_EX. Le résultat des calculs est affiché sans que vous ayez à les copier-coller dans votre document.

Pour l'utiliser, il faut :

- Que votre distribution de L^AT_EX connaisse Sage. Pour cela, une façon est de copier un fichier `sagetex.sty` dans le répertoire dans lequel vous travaillez. Ce fichier se trouve dans le dossier
`$SAGE_ROOT/local/share/texmf/tex/latex/sagetex`
où `$SAGE_ROOT` désigne l'endroit où vous avez installé Sage sur votre ordinateur.
- Que vous ajoutiez le paquet `sagetex` à la liste des paquets utilisés dans le préambule de votre fichier L^AT_EX.
- Compiler L^AT_EX (normalement) va générer un fichier `.sage` qui contiendra toutes les lignes à exécuter par Sage.

2 Les différentes façons d'utiliser SageTeX

2.1 Comme des mathématiques, dans le texte

On peut intégrer une courte réponse de Sage dans le texte, en mode maths.

Par exemple, l'entier `150` admet `$$\sage{number_of_partitions(150)}$` partitions.

Par exemple, l'entier 150 admet 40853235313 partitions.

Le nombre chromatique du graphe de Petersen est
`\sage{graphs.PetersenGraph().chromatic_number()}` et celui
d'un graphe biparti est
`\sage{graphs.CompleteBipartiteGraph(5, 7).chromatic_number()}$.`

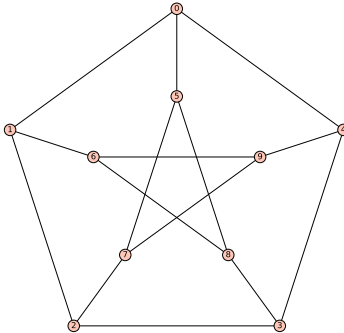
Le nombre chromatique du graphe de Petersen est 3 et celui d'un graphe biparti aléatoire est 2.

2.2 Pour faire des dessins : Sageplot

La commande `sageplot` se comporte un peu comme `includegraphics`, mais les images sont générées par Sage.

Je peux, par exemple, dessiner des graphes :

```
\begin{center}
\sageplot[height=5cm]{graphs.PetersenGraph().plot()}
\end{center}
```



2.3 Sageblock

Un bloc `sageblock` est comme un bloc `verbatim`, mais il écrit aussi dans le fichier `.sage`.

Un peu de code Sage :

```
\begin{sageblock}
f(x) = exp(x) * sin(2*x)
\end{sageblock}
```

La d'ériv'ée seconde de f est

```
\[
\frac{\mathrm{d}^2}{\mathrm{d}x^2} \sage{f(x)} =
```

```
\sage{diff(f, x, 2)(x)}.
\]
```

Cela permet d'afficher :

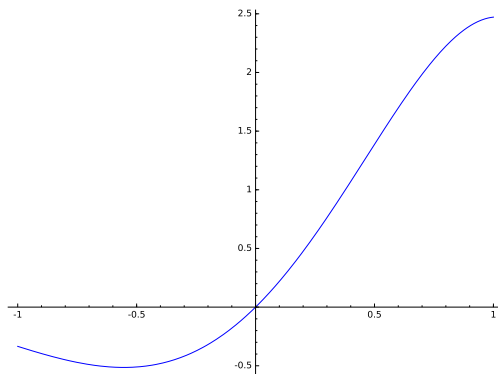
$$f(x) = \exp(x) * \sin(2*x)$$

La dérivée seconde de f est

$$\frac{d^2}{dx^2} e^x \sin(2x) = 4 \cos(2x) e^x - 3 e^x \sin(2x).$$

Voici enfin le graphe de f sur $[-1, 1]$:

```
\begin{center}
\sageplot[height=5cm]{plot(f, -1, 1)}
\end{center}
```

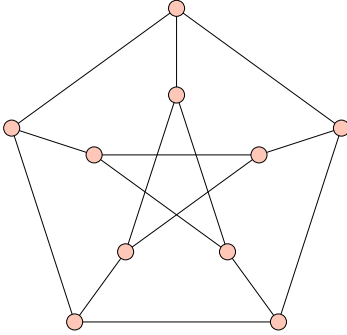


2.4 Sagesilent

On peut utiliser Sagesilent pour inscrire des lignes au fichier qui sera exécuté par Sage, mais sans les afficher. Par exemple, Voici un graphe :

```
\begin{sagesilent}
G = graphs.PetersenGraph()
c = G.coloring(hex_colors=True)
\end{sagesilent}

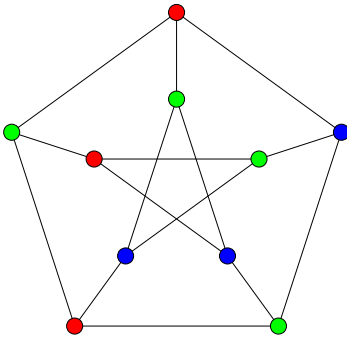
\begin{center}
\sageplot[height=5cm]{G.plot(vertex_labels=False, vertex_size=400)}
\end{center}
```



Son nombre chromatique est $\text{\sage{G.chromatic_number()}}$, comme l'indique la figure ci-dessous~:\

```
\sageplot[height=5cm]{G.plot(vertex_labels=False, vertex_colors=c,
vertex_size=400)}
```

Son nombre chromatique est 3, comme l'indique la figure ci-dessous :



3 Compilation

La compilation d'un fichier sageTeX se fait en trois étapes :

1. Compiler le fichier L^AT_EX (normalement.) Ça inscrit notamment les bonnes au fichier `.sage`.
2. Exécuter le fichier `.sage`. Dans un terminal, dans le dossier où se trouve votre fichier T_EX, exécuter la commande

```
sage [nom-de-fichier].sagetex.sage
```

3. Recompiler le fichier L^AT_EX. Votre fichier devrait apparaître avec les résultats produits par Sage.

4 Partager son code

Pour partager son code avec des gens qui n'utilisent pas Sage :

- Le fichier `sagetex.sty`
- Le fichier (génééré automatiquement) `.sout`
- Le dossier `sage-plots-for-[nom de fichier].tex` et son contenu.

Références

[Da] Dan DRAKE et AUTRES : *The SageTEX package*.

[Tea] The Sage Development TEAM : *Utiliser SageTex – Tutoriel Sage v7.6*.