

Chronique 3

Insertion d'image

3.1 Mode de compilation

Pour visualiser un document tapé en \LaTeX , il y a plusieurs façons. Moi je visualise en PostScript (au moyen de `Gsview` et `Ghostscript`). Ce mode de compilation et de visualisation permet aussi d'utiliser `PsTricks` sans problème.

On peut également visualiser en pdf ; mais là, il y a deux modes de compilation : soit par `PdfLatex`, soit par `ps2pdf` en passant par `dvips`.

Ce que je vais expliquer dans cette rubrique ne fonctionne qu'en PostScript ou en pdf compilé par `ps2pdf` (donc en passant également par PostScript).

L'éditeur que j'utilise, `Texmaker`, doit donc être configuré correctement ; on choisit dans le menu un de ces deux modes :

- Options
- Configurer Texmaker
- Compil rapide
- LaTeX + dvips + View PS
- Options
- Configurer Texmaker
- Compil rapide
- LaTeX + dvips + ps2pdf + View pdf

Ensuite l'appui sur `F1` entraîne la compilation et affiche le résultat attendu.

3.2 Flottant ou pas

Il existe, en \LaTeX , deux modes de gestion des images : le mode flottant et le mode non flottant.

En fait la notion de « flottant » est plus générale, et on doit plutôt parler d'« objet flottant », l'objet en question pouvant être une image, une photo, une table...

On peut même rendre flottant n'importe quel objet.

Un objet flottant sera placé « au mieux » par \LaTeX , c'est-à-dire qu'il ne sera pas forcément à l'endroit où cet objet est inséré dans le texte source ; ainsi il peut être plus bas dans la page ou encore la page suivante, notamment s'il y a déjà un objet flottant dans la page courante.

Ce n'est donc pas de ce mode dont je vais parler dans cette chronique car je préfère que les figures soient à l'endroit précis où je le souhaite.

3.3 Format EPS

Le format de sortie originel de \TeX (puis celui de \LaTeX) était le `POSTSCRIPT (PS)` ; le format d'images associé est l'`ENCAPSULÉ POSTSCRIPT (EPS)`.

Il est donc tout à fait naturel que l'insertion d'image `EPS` dans un document écrit en \LaTeX se fasse simplement ; une seule instruction suffira.

3.3.1 Création d'une image EPS avec GeoGebra

Le très bon logiciel de géométrie GeoGebra permet d'exporter des graphiques et d'en faire des images EPS que l'on pourra directement insérer dans un texte en \LaTeX .

On crée une figure quelconque (fonction carré, un cercle et un trapèze rectangle), avec des couleurs, pas seulement pour faire plus joli !

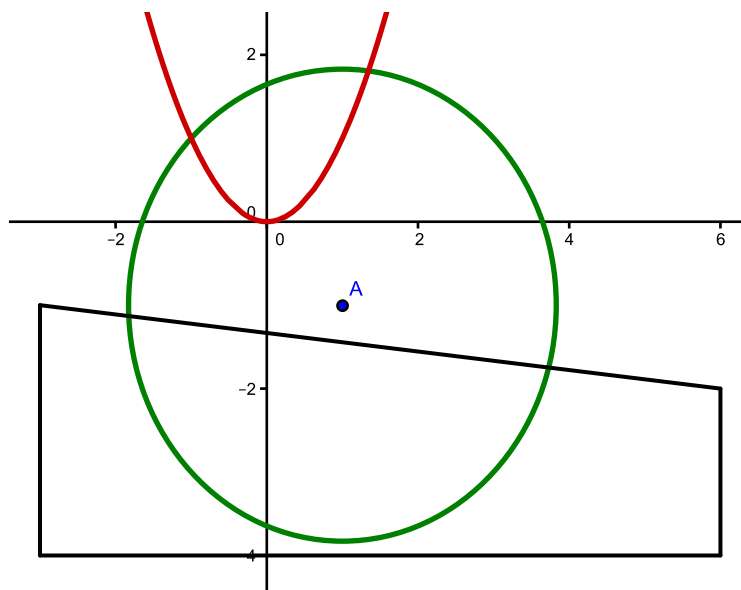
À la souris, on sélectionne la partie du graphique à découper puis on choisit :

- Fichier
- Exporter ▶
- Graphique en tant qu'image (png, eps)
- Format Postscript encapsulé (eps)
- on laisse 1 cm pour échelle
- Sauvegarder
- sous le nom de FigGeo

On a donc créé une figure qui porte le nom `FigGeo.eps` que l'on va copier dans le même répertoire que le document \LaTeX sur lequel on travaille ; si cette figure n'est pas dans le même répertoire, il faudra indiquer le chemin d'accès de cette figure.

3.3.2 Insertion de l'image

\LaTeX sait lire les caractéristiques de toute image au format EPS ; il suffit d'une instruction pour inclure l'image `FigGeo` dans le texte : `\includegraphics{FigGeo}`



La figure a la taille de la figure découpée dans GeoGebra.

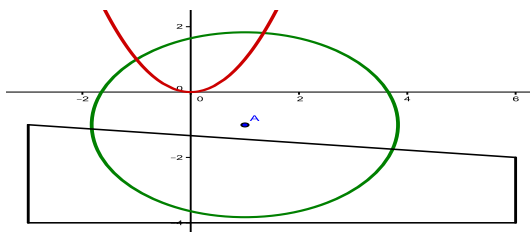
3.3.3 Options

Tout d'abord, pour utiliser des options avec l'instruction `\includegraphics`, il faut avoir chargé l'extension `graphicx` par un `\usepackage{graphicx}` dans le préambule du document.

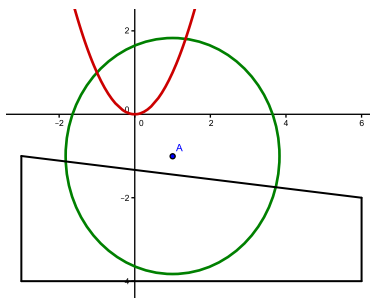
La syntaxe complète est alors : `\includegraphics[options]{nom_image}`.

On peut très facilement alors fixer la largeur (`width`) ou la hauteur (`height`) de la figure à afficher ; il suffit de rajouter des options entre crochets :

- `\includegraphics[width=7cm]{FigGeo}` impose une largeur de 7 cm ;
- `\includegraphics[height=3cm]{FigGeo}` impose une hauteur de 3 cm ;
- `\includegraphics[width=7cm, height=3cm]{FigGeo}` impose à la fois une largeur de 7 cm et une hauteur de 3 cm ; la figure est donc déformée :



Une autre façon simple de modifier les dimensions d'une figure est d'utiliser le facteur d'échelle `scale` : `\includegraphics[scale = 0.5]{FigGeo}` multiplie chaque dimension par 0,5.



On peut tourner les figures en entrant en option la valeur, en degrés, de l'angle de la rotation que l'on veut appliquer :

```
\includegraphics[scale = 0.5, angle=45]{FigGeo}
```

Et on peut même imposer le centre de la rotation en rajoutant l'option `origin = label` où `label` est une combinaison de lettres parmi : `l` (left), `r` (right), `c` (center), `t` (top), `b` (bottom) ou `B` (baseline). L'origine par défaut est `bl`, en bas à gauche.

L'inclusion d'image est naturellement compatible avec le multicolonnage, ainsi qu'avec la notion de minipage (voir chroniques de la première saison).

3.3.4 Légende

On peut ajouter une légende très simplement à une figure.

Pour cela il faut d'abord charger l'extension `caption` par `\usepackage{caption}`.

Puis, après avoir inséré la figure, on en donne le nom au moyen de l'instruction `\captionof` comme dans l'exemple :

```
\begin{center}
\includegraphics[height=4cm]{FigGeo}
\captionof{figure}{GeoGebra}
\end{center}
```

ce qui donne :

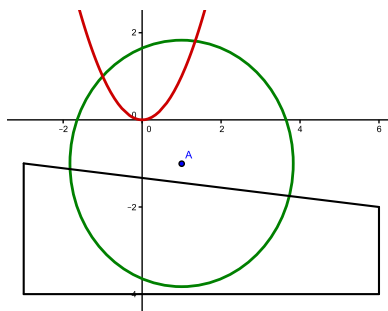


FIGURE 3.1 – GeoGebra

Le mot `figure` qui est le premier paramètre de la commande `\captionof` signifie que le nom « GeoGebra » est le nom d'une figure et apparaîtra ainsi dans une éventuelle table des figures.

Vous remarquerez que devant le mot `GeoGebra` se trouve le mot `FIGURE` (en petites capitales) suivi de deux nombres séparés par un point : à gauche du point se trouve le numéro du chapitre en cours, et à droite du point le numéro de la figure dans le chapitre.

À condition toutefois d'être dans une classe (`report` ou `book`) dans laquelle le niveau de structure `\chapter` est reconnu ; sinon, seul le nombre à droite du point est affiché.

3.4 Format JPG

Disons-le tout net : la sortie en `POSTSCRIPT` n'est pas faite pour insérer des images au format `JPG`. Il faut pour cela compiler avec `pdflatex`, et tout ce qui a été dit précédemment fonctionne ; on entrera donc une commande du style : `\includegraphics[options]{nom_image.jpg}`.

Mais si l'on est très têtu et que l'on veut à tout prix insérer une image au format `JPG` dans un fichier visualisé en `POSTSCRIPT` (par exemple si on travaille avec `PsTricks` qui n'aime pas une compilation en `pdflatex`) ?

3.4.1 Insertion de l'image

On place une photo intitulée `Hoss01.jpg` dans le répertoire de ce document source et on entre :

```
\includegraphics{Hoss01.jpg}
```

On obtient un magnifique message d'erreur :

```
Cannot determine size of graphic in Hoss01.jpg (no BoundingBox)
```

En effet, `LATEX` sait lire les caractéristiques d'une image en `EPS`, mais pas la taille d'une image en `JPG`. Il faut donc passer en paramètre les dimensions de la photo, et en même temps une largeur pour que la photo rentre dans la page !

Cette photo est de 490 pixels sur 368 pixels, donc on entrera :

```
\begin{center}
\includegraphics[bb=0 0 490 368, width=8cm]{Hoss01.jpg}
\captionof{figure}{Café de Paris}
\end{center}
```

Vous aurez compris que le `bb` signifie `BoundingBox`, et que l'on indique ainsi à `LATEX` les dimensions de la photo à insérer.

Et que se passe-t-il alors ?



FIGURE 3.2 – Café de Paris

Catastrophe! Les couleurs de la photo ont disparu et je ne connais pas de moyen d'empêcher ça! On oubliera donc l'insertion d'image JPG dans un texte compilé en POSTSCRIPT. Mais peut-être qu'on peut convertir le format JPG en format EPS?

3.4.2 Conversion de format

J'ai trouvé sur Internet un logiciel de dessin vectorisé qui permet (entre autres) de passer du format JPG au format EPS; il s'agit de `inkscape` que l'on peut télécharger (gratuitement) à l'adresse <http://inkscape.org/?lang=fr>

Il y a une grosse documentation pour ce logiciel sur Internet, la plupart du temps en anglais. Moi je ne l'ai utilisé que pour convertir les formats : j'ouvre un fichier JPG et je l'exporte en EPS en utilisant `Enregistrer sous`.

Seul défaut, le fichier EPS obtenu est dix fois plus volumineux que le fichier JPG d'origine.

Après transformation du format si on entre :

```
\includegraphics[width=8cm]{Hoss01}
\captionof{figure}{Café de Paris}
```

on obtient :



FIGURE 3.3 – Café de Paris

C'est quand même mieux en couleur!