Visualisation de graphes en format postscript

Description générale

Le langage ou format postcript permettant de décrire une page composé de dessins et de textes. Il repose sur des formules vectorielles pour la plupart de ses éléments, mais peut aussi comporter des description matricielle de certains composants (images etc). Ce format est un standard, la plupart des imprimantes lasers haut de gamme peuvent traiter directement le format PostScript. Le développement du PostScript est arrêté par Adobe depuis 2007, afin que le PDF puisse prendre la relève. Néanmoins il s'agit d'un format très léger et simple qui permet de dessiner rapidement de très grands dessins composés d'une grande quantité d'éléments simples. C'est pourquoi il est parfaitement adapté pour représenter des graphes.

On peut visualiser un dessin codé par un langage postscript en utilisant le logiciel ghostview (gv). On peut aussi noter que la plupart des outils apte à lire les formats pdf peuvent la plupart du temps interpréter le format postscritp, mais il le font plus lentement que ghostview qui est dédié à cet effet. Il est également possible de transformer un dessin en langage postscript en un fichier pdf par exemple (ou autre format). Sous linux, le logiciel ps2pdf permet de réaliser cette transformation. Le dessin résultat peut alors être visualisé, imprimé ou transformé dans un format facile à diffuser (pdf, jpg),...

Langage poscript

Le langage postscript est constitué d'un fichier texte qui peut être interprété directement par une imprimante.

Afin de ne pas dépendre des formats de page (A4, american letter,...), une page postscript est défini par une unité de mesure appelée points. Le format A4 correspond à 595 points sur et 842 points. Afin de dessiner un graphe dont les coordonnées des points ont un écart entre $dimX$ et $dimy$, il suffit alors de faire une homothétie pour se ramener à ces coordonnées A4.

Visualiser des graphes

Dans ce document, nous nous intéressons uniquement à la représentation d'un graphe en langage postscript dans le cas où le graphe est décrit:
- par un ensemble de sommets repérés par des coordonnées planaires
- par une liste d'arêtes données par des couples de points.

Il s'agit donc ici d'un graphe non-orienté (vous pourrez néanmoins déduire facilement de ce document des façons de représenter les arcs d'un graphe orienté).

Noter que dans le cas où les coordonnées des sommets ne sont pas connus, il est préférable d'utiliser des outils de représentations de graphe capable d'attribuer automatiquement des coordonnées bien choisies aux sommets de manière à représenter le graphe le plus agréablement possible. Le logiciel Graphviz est par exemple très adapté pour cela (en revanche, il ne permet pas de représenter des graphes de dimensions importantes).

Commandes utiles pour visualiser des graphes

On va donc ici représenter les sommets d'un graphe par des cercles et les arêtes par des lignes droites:
- Pour dessiner un cercle de 2,5 points de rayon et de centre de coordonnées x, y, il suffit d'utiliser les trois lignes suivantes

```
x y 2.5 0 360 arc
fill
stroke
```

- Pour dessiner une droite entre les points $x1, y1$ et $x2, y2$, il suffit d'utiliser les trois lignes suivantes

```
x1 y1 moveto
x2 y2 lineto
stroke
```

- Par défaut, le trait de dessin est en noir, si l'on désire dessiner d'une autre couleur, il suffit de faire précéder la commande précédente par

```
r g b setrgbcolor
```

où *rgb* est le codage en pourcentage de la quantité de rouge, vert et bleu de la couleur demandée: par exemple `1 0 0` correspond au rouge et `0 1 0` correspond au vert.

Exemple de fichier

Il est à noter que pour un tel dessin, il n'est même pas nécessaire de mettre une entête à un fichier texte correspondant à un fichier postscript.

Afin d'automatiser le dessin d'un graphe à partir d'un programme possédant une structure de données contenant ce graphe, il suffit donc de créer un fichier texte contenant les commandes précédentes les unes à la suite des autres.

Voici un exemple de fichier postscript et le graphe correspondant:

```
100 200 moveto
200 300 lineto
stroke
200 300 moveto
200 400 lineto
stroke
200 400 moveto
300 400 lineto
stroke
1 0 0 setrgbcolor
300 400 moveto
300 300 lineto
stroke
0 0 0 setrgbcolor
300 300 moveto
200 300 lineto
stroke
100 200 2.5 0 360 arc
fill
0 1 0 setrgbcolor
200 300 2.5 0 360 arc
fill
0 0 0 setrgbcolor
200 400 2.5 0 360 arc
fill
300 400 2.5 0 360 arc
fill
300 300 2.5 0 360 arc
fill
```