Tirage aléatoire uniforme de cartes planaires de degré 4 enracinées

Dominique Rossin

dominique.rossin@m4x.org

1 Présentation

De nombreuses plateformes d'expérimentation nécessitent le tirage aléatoire d'objets de manière uniforme. Nous nous intéressons ici au tirage aléatoire de cartes planaires c'est à dire de graphes planaires dessinés sur une feuille de papier, en réalité sur une sphère. Rappelons qu'un graphe planaire est un graphe que l'on peut dessiner sur une feuille de manière à ce que les arêtes ne se coupent entre elles qu'aux sommets du graphe. Une carte planaire est une représentation planaire d'un graphe planaire. Ainsi il y a plusieurs cartes planaires correspondants au même graphe planaire. Par exemple considérons les deux cartes suivantes:

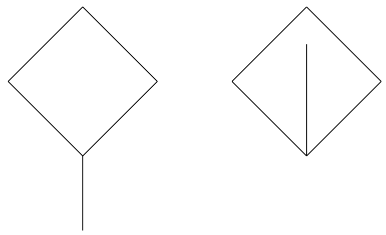

Figure 1: Deux cartes planaires correspondant au même graphe

Le tirage aléatoire uniforme de graphes est un problème dur. La restriction aux cartes planaires permet de mettre en œuvre des algorithmes simples de manipulation d'arbres et l'obtention d'une méthode optimale (temps linéaire) pour la génération aléatoire.

Le travail demandé est de réaliser un programme permettant de générer aléatoirement et de manière uniforme des cartes planaires où tous les sommets sont de degré 4. Une deuxième partie est l'étude de certaines données statistiques sur les graphes comme le diamètre moyen du graphe.

2 Présentation de l'algorithme

L'algorithme se décompose en plusieurs phases:

- Génération aléatoire uniforme d'arbres binaires complets plantés. Un arbre planté est un arbre possédant une racine

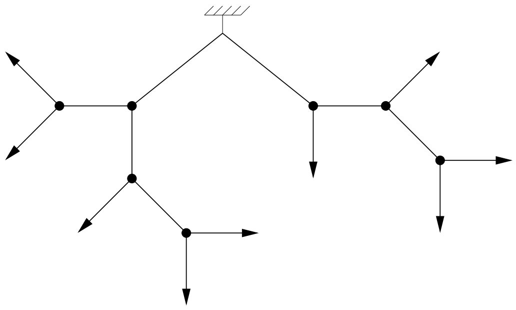

Figure 2: Arbre binaire complet planté à 8 nœuds, 10 feuilles

On dira que les feuilles de l'arbre sont noires.

- Ajout de bourgeons. On ajoute sur chaque nœud un bourgeon (ie une feuille blanche). Remarquons que pour chaque nœud, il y a 3 manières possibles d'ajouter le bourgeon.

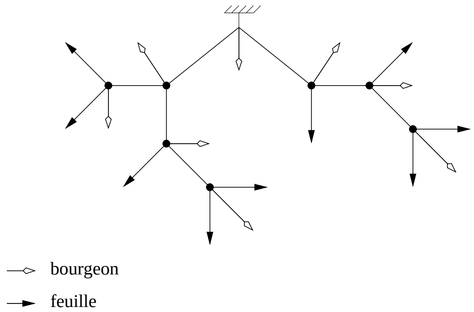

Figure 3: Arbre binaire complet planté bourgeonnant à 8 nœuds, 10 feuilles et 8 bourgeons

- Clôture partielle de l'arbre bourgeonnant. On parcours l'arbre dans un ordre préfixe gauche (dans l'ordre inverse des aiguilles d'une montre) en partant de la racine. On ne retient dans ce parcours que les bourgeons et les feuilles. Ainsi si l'on note x les bourgeons et $\bar{x}$ les feuilles (et la racine) on obtient le parcours suivant: $x\bar{x}\bar{x}x\bar{x}\bar{x}x\bar{x}xx\bar{x}\bar{x}x\bar{x}x\bar{x}x\bar{x}$. Si l'on rencontre un bourgeon (x) suivi d'une feuille ($\bar{x}$) on les apparie ensemble en créant une arête comme le montre la figure ci-dessous pour le premier appariement trouvé.

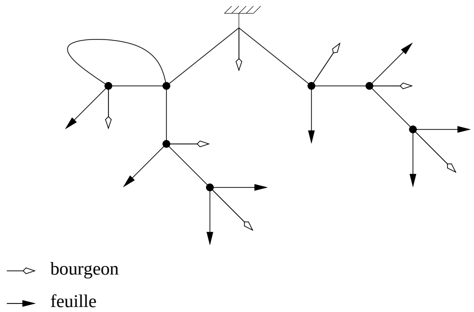

Figure 4: Premier appariement dans le processus de cloture

Les appariements suivants :

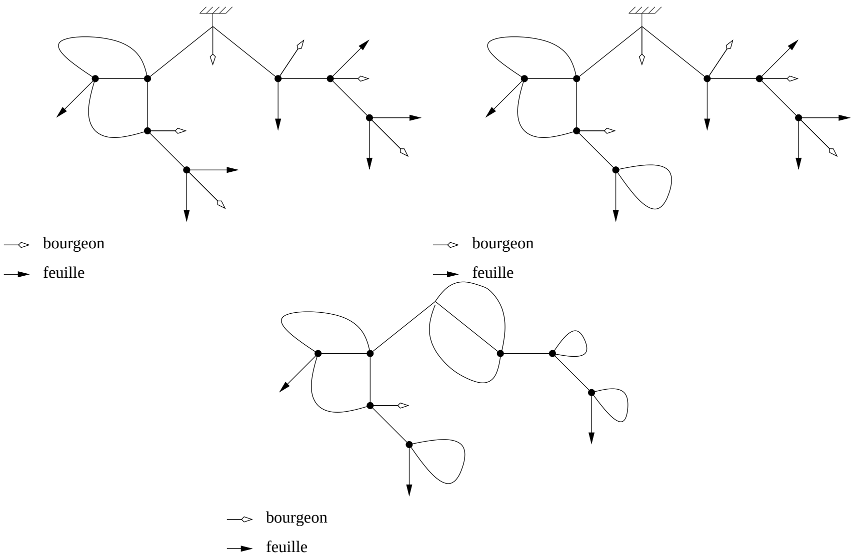

Figure 5: Appariements dans le processus de cloture

A la fin de ce stade on a fait un tour de l'arbre. On recommence maintenant le tour jusqu'à ce qu'il ne reste plus que 2 feuilles libres. (il y a deux feuilles de plus que de bourgeons). On obtient alors:

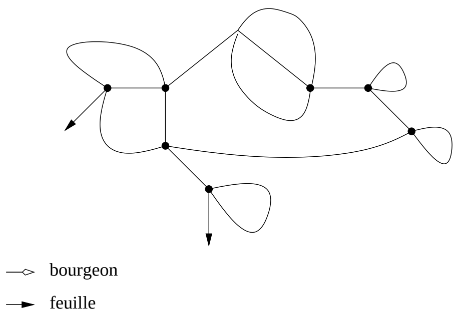

Figure 6: Appariements dans le processus de cloture

A la fin de ce processus il reste deux feuilles libres. On relie alors ces deux feuilles en orientant l'arête obtenue. On remarquera que ce processus est indépendant du choix de la racine de l'arbre. On ne demande pas de démonstration. On obtient alors une carte planaire où tous les sommets sont de degré 4.

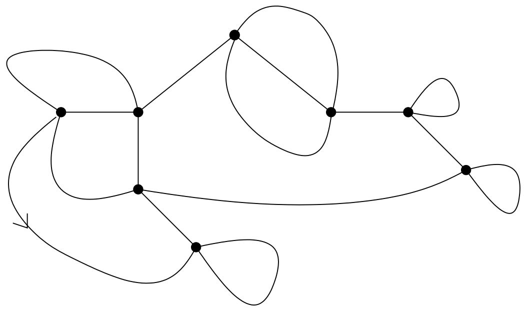

Figure 7: Carte planaire finale

La carte planaire obtenue est une carte planaire de degré 4 enracinée (une arête orientée) uniformément aléatoire.

On peut montrer que ces cartes sont en bijection avec les cartes planaires enracinées sans conditions de degré. Nous nous limiterons ici aux cartes planaires de degré 4.

3 Algorithme détaillé

Les structures de données à utiliser sont laissées au choix du lecteur. L'algorithme décrit ci-dessus ne comporte qu'une seule difficulté, la génération aléatoire d'arbres binaires complets. Mais, un arbre binaire complet peut se coder sous la forme d'un mot formé de x et de y où x représente un nœud et y une feuille. Attention à ne pas confondre cette notation avec la notation de la partie précédente. Le codage d'un arbre enraciné s'obtient en réalisant un parcours préfixe gauche de l'arbre. (ce qui s'obtient très facilement par un parcours en profondeur de l'arbre).Il y a un y de plus que les x.

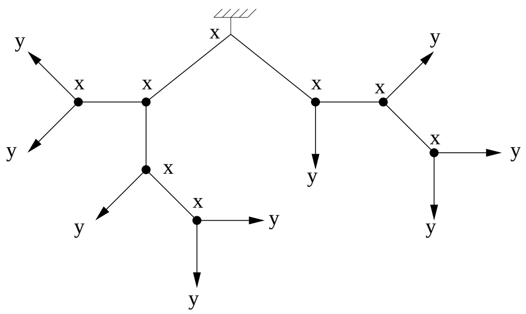

Figure 8: Arbre binaire complet

Son codage est alors $xxxyyxyxyyxyxxyyy$. A chaque lettre est aussi associée une hauteur. On part de 0 et quand on rencontre un x on fait $+1$ et -1 pour y. Ainsi les hauteurs associés au mot précédent sont

$$(1)(2)(3)(2)(1)(2)(1)(2)(1)(0)(1)(0)(1)(2)(1)(0)(-1)$$

. Si on considère ce mot comme circulaire -après la dernière lettre on recommence au début- on voit que la racine est située juste après le minimum de la fonction. Ainsi, pour trouver un arbre aléatoire on tire un mot constitué de x et de y de la manière suivante:

- Il y a n lettres x
- Il y a $n+1$ lettres y

Les lettres sont tirées aléatoirement et de manière uniforme. Par exemple il y a n chances sur $2n+1$ que la première lettre soit un x.

On obtient alors un mot de $2n+1$ lettres. En pratique, ce mot n'est pas forcement un mot d'un parcours préfixe d'un arbre binaire complet. Il faut pour celà calculer les hauteurs associées à chaque lettre et couper le mot en deux parties autour du dernier minimum strict puis remettre les deux moitiés dans l'ordre inverse. Par exemple, considérons le mot suivant $yyxyxyyxx$. Les hauteurs associés sont

$$(-1)(-2)(-1)(-2)(-1)(-2)(-3)(-2)(-1)$$

. Le minimum strict est (-3) donc on coupe le mot après la lettre correspondante et on inverse les deux parties pour obtenir le mot $xxyyxyxyy$. On admettra que cette méthode permet de tirer un arbre aléatoire de manière uniforme.

4 Travail demandé

Vous devez programmer les fonctionnalités suivantes:

1. Tirage aléatoire d'arbres.
2. Tirage aléatoire de cartes planaires de degré 4.
3. Représentation graphique des arbres et cartes (pour les cartes une représentation même non planaire est possible)
4. Calcul du diamètre de la carte obtenue (plus grande distance entre deux sommets)