Pavages

Etienne Moutot - `etienne.moutot@math.cnrs.fr`

Pavages par polygones

Un pavage est le recouvrement du plan (infini) par des copies de formes tel qu'il n'y ait ni chevauchement ni trou entre les formes, que l'on appelle des *tuiles*. Ces tuiles peuvent être tournées (*rotations*), mais pas retournées (pas de *symétries*)

Polygones réguliers

Quels sont les polygônes régulier qui pavent le plan (en autorisant un seul type de tuile à la fois)?

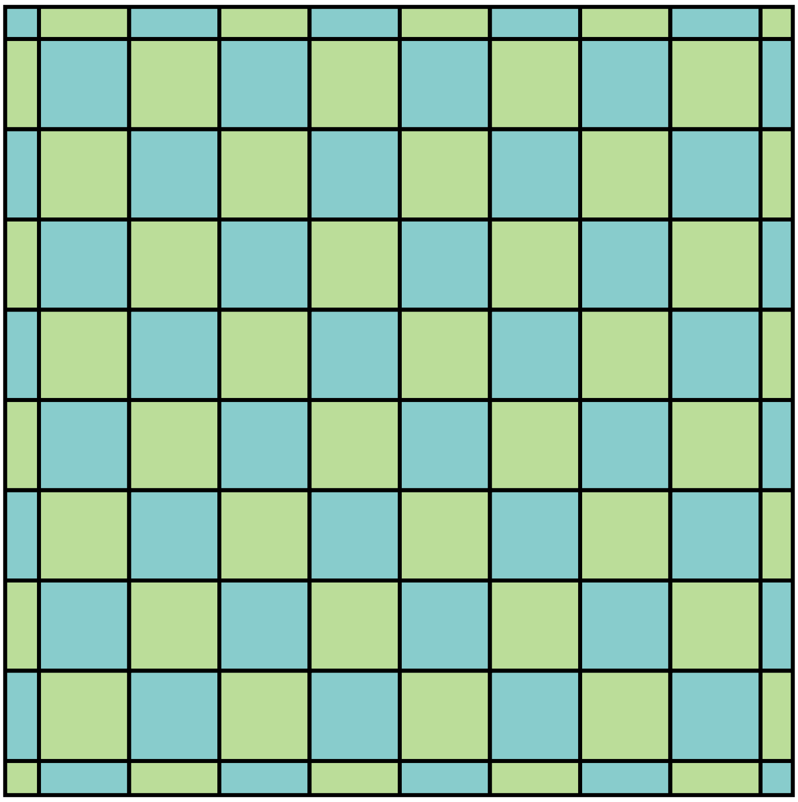

Figure 1: Pavage par un carré

Que se passe-t-il si on autorise plusieurs types de polygones réguliers ? En mélangeant les tailles, peut-on en utiliser autant qu'on veut ?

Le but n'est pas forcément de lister tous les cas possibles (mais si vous y arrivez, allez-y !), mais surtout de **démontrer** ce que vous trouvez.

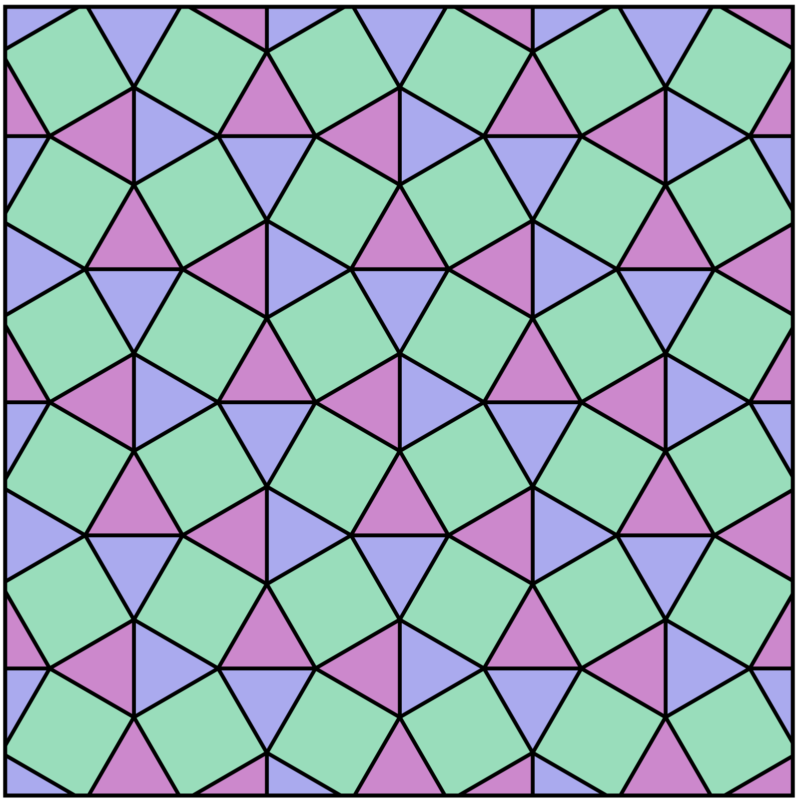

Figure 2: Exemple de pavage par un carré et un triangle

Polygone non réguliers

Un polygone est dit *convexe* si chacun des points de sa surface peut être reliée par un segment entièrement contenu dans le polygone.

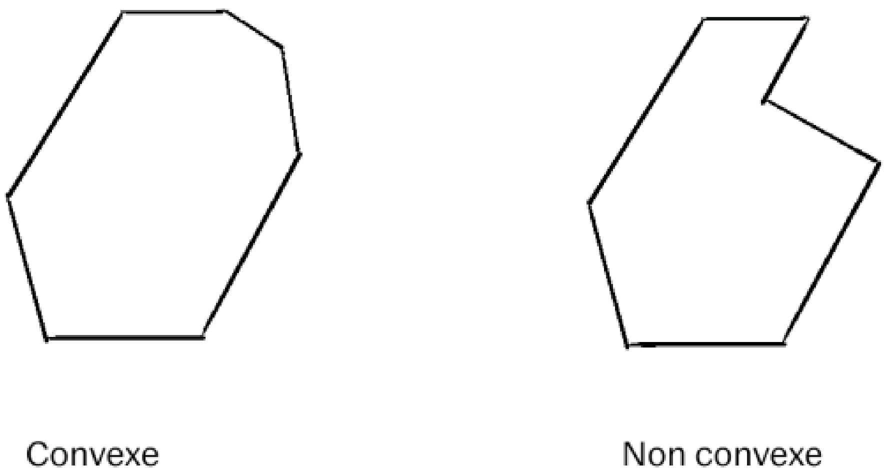

Figure 3: Exemples de polygones convexe et non convexe

On va s'intéresser aux pavages du plan par une seule tuile **convexe**.

- Quels sont les triangles convexes qui pavent le plan ?
- Les quadrilatères convexes ?
- Les pentagones convexes ?
- ... etc
- Existe-t-il un nombre maximal de côté pour qu'un polygone convexe puisse paver le plan ?

Pavages de tatamis

On cherche à paver des formes avec des tatamis: des rectangles de taille 2x1 que l'on peut placer horizontalement ou verticalement.

Rectangles

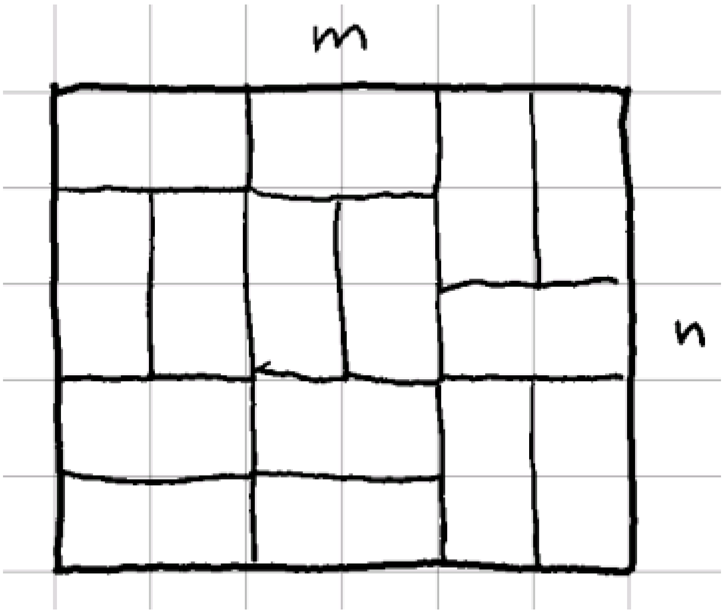

Figure 4: Pavage d'un rectangle par tatamis

Dans un premier temps, on cherche à paver un rectangle à l'aide de ces tuiles rectangulaires. Peut-on toujours paver le rectangle, quelque soit ses dimensions m et n ? Si non, pour quels m et n peut-on paver le rectangle ?

Polyominos

Un polyomino est un union de carrés par leurs côtés, ou de manière équivalente, un rectangle duquel on a supprimé certains carrés.

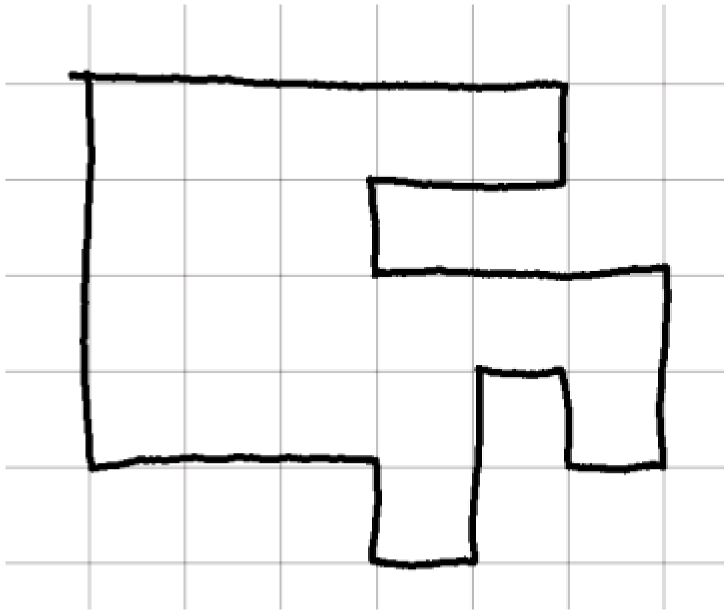

Figure 5: Exemple de polyomino

Même question que précedemment: existe-t-il un critère permettant de savoir si un polyomino admet un pavage par tatmis ou non ?

Des formes plus générales

On peut généraliser la question précédente à des formes encore plus abstraites. Pour cela il faut définir ce qu'est un graphe. Un *graphe* est un ensemble de *sommets*, reliés deux à deux par des *arêtes* (qui peuvent éventuellement se croiser).

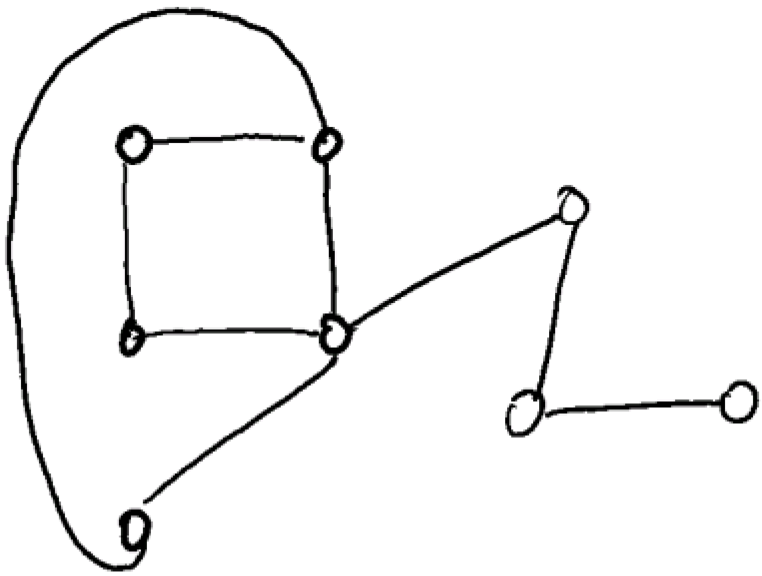

Figure 6: Exemple de graphe

On appelle un *couplage* d'un graphe le fait de selcetionner des arêtes de celui-ci, de telle manière à ce qu'aucune arête ne partage un sommet. Un *couplage parfait* est un couplage dont les arêtes couvrent tous les sommets du graphe.

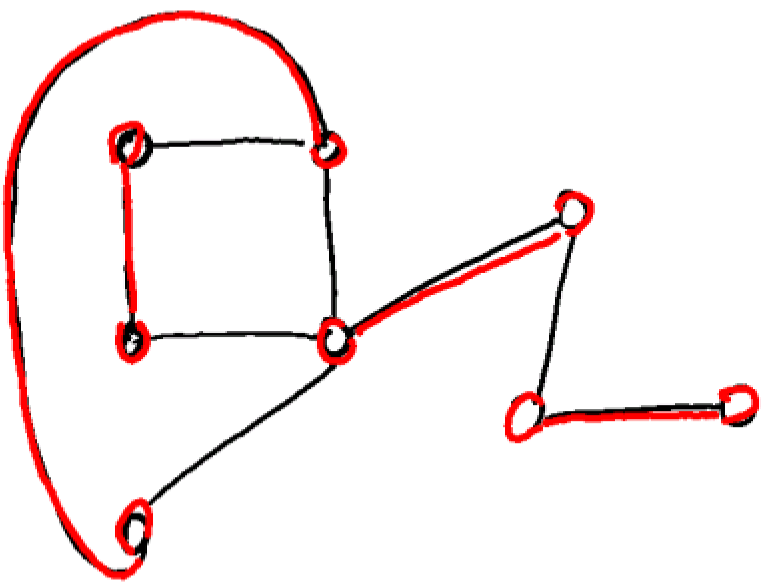

Figure 7: Un *couplage parfait* du graphe précédent (les arêtes séléctionnées sont en rouge)

Et quel est le rapport avec les pavages par tatamis ? Chaque arête choisie dans un couplage peut être vue comme le fait de “placer un tatami” sur cette arête. Trouver un couplage parfait revient donc à recouvrir le graphe de tatamis, en effet: le couplage assure que deux tatamis ne se chevauchent pas (les arêtes séléctionnées n'ont pas de sommet en commun) et le fait qu'il soit parfait assure que le pavage recouvre bien tout le graphe.

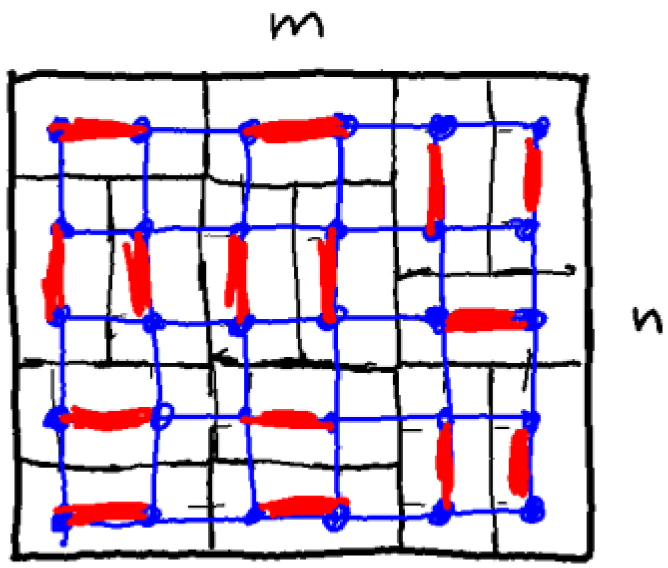

Figure 8: Le pavage du rectangle plus haut vu comme un couplage parfait (en rouge) du graphe (en bleu)

Et encore une fois, la même question peut être posée: existe-t-il un critère sur le graphe qui permette de dire s'il existe ou non un couplage parfait ?

Si vous avez envie programmer...

Ce sujet se prête bien à de la programmation, si vous en avez envie.

- On peut imaginer un programme cherchant à paver des formes en *force brute*: il teste toutes les combinaisons de tatamis possible jusqu'à en trouver une qui pave la forme. (Évidemment, ça risque d'être lent, mais sur des petites formes, ça peut être utile quand même)
- Quitte à essayer toutes les combinaisons, on peut écrire un programme qui *compte* les pavages des formes (ou couplages parfaits)
- Vous pouvez programmer des fonctions qui testent vos critères trouvés “à la main” pour savoir si une forme (ou un graphe) peuvent être pavés ou non.