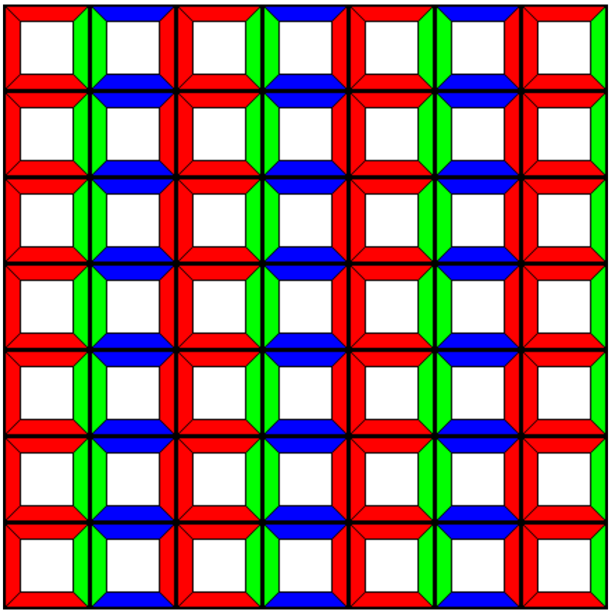
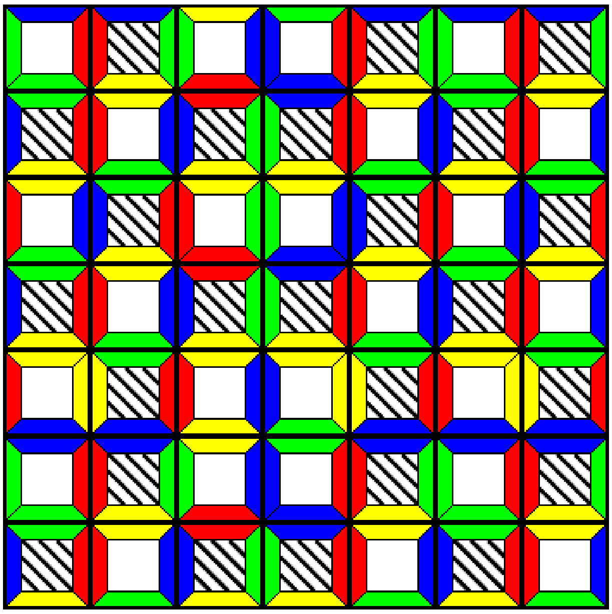
Pavages Périodiques et Quasipériodiques

(par Jean-Christophe Lavocat)



Pavage périodique



Pavage quasipériodique

Mots clés:

- Pavages de Wang
- Systèmes à contraintes locales
- Périodicité et Quasipériodicité
- Décidabilité
- Relation d'ordre
- Extraction diagonale
- Fonction d'énumération

Plan de l'exposé

I) Préliminaires et notions essentielles

a)Définitions élémentaires
b)Problème du pavage du plan

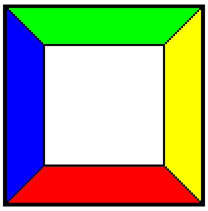
II) Quasipériodicité

a)Relation d'ordre
b)Procédé d'extraction diagonale
c)Quasipériodicité

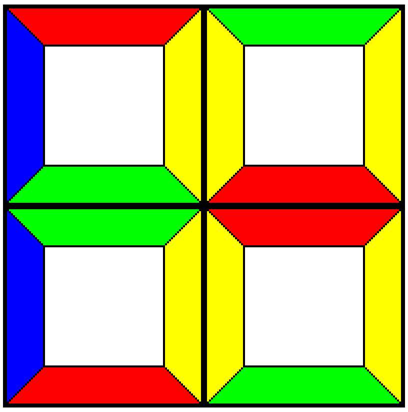
III)Fonctions sur les pavages

a)Fonction de quasipériodicité
b)Fonction de dénombrement

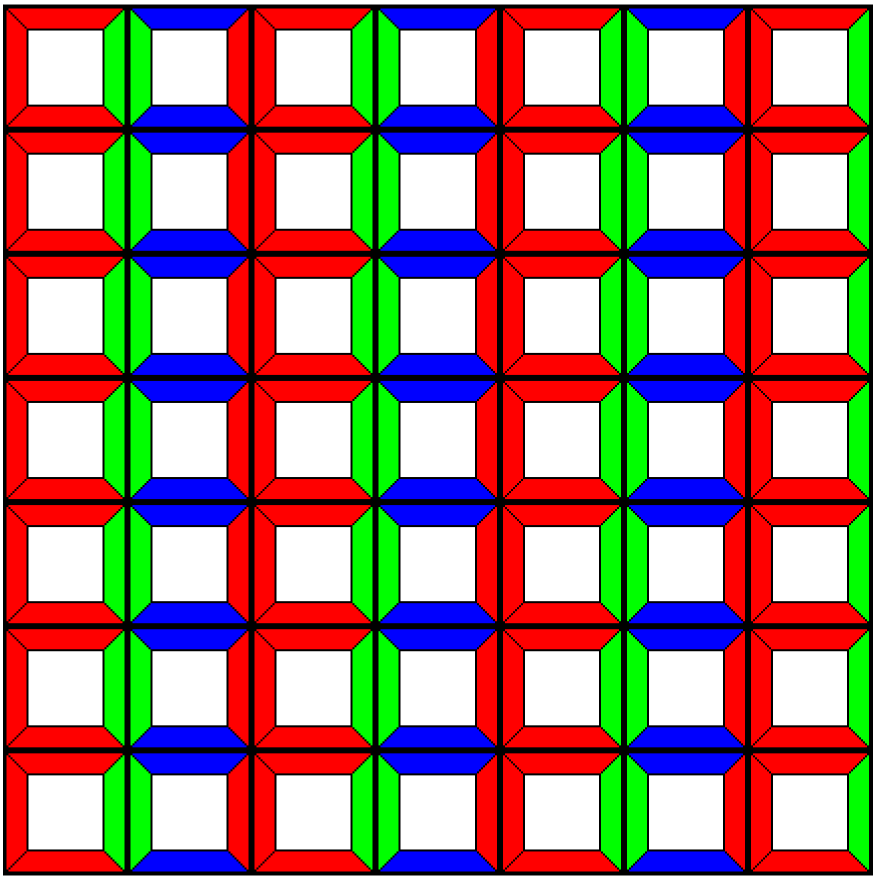
<u>Définitions</u>



Tuile de Wang



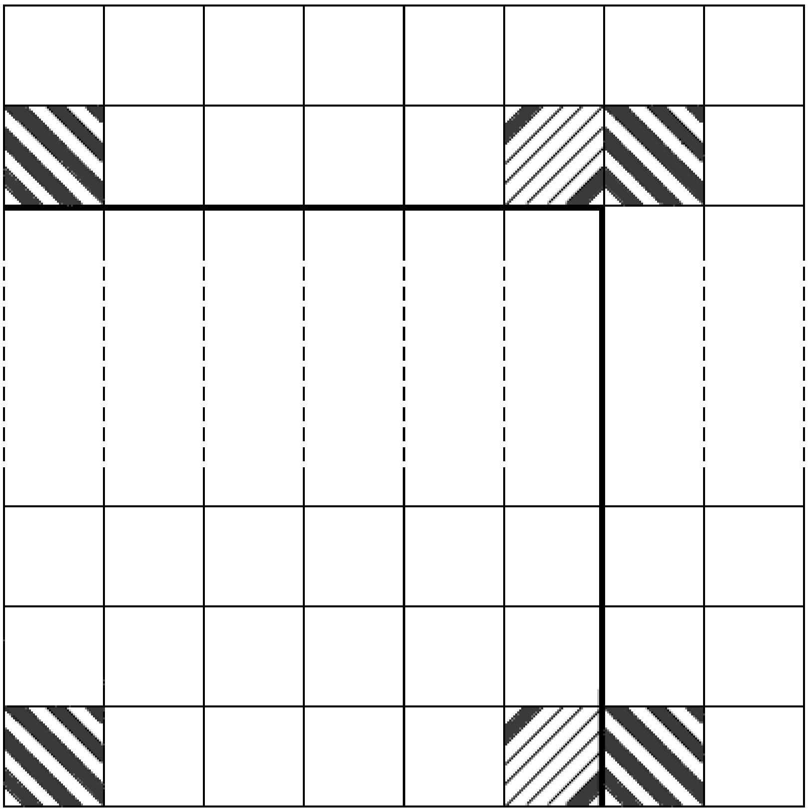
Motif valide



Configuration périodique

<u>Propriété</u>: Tout jeu de tuile acceptant un pavage périodique selon un axe accepte un pavage périodique

<u>Preuve:</u> Nombre fini de tuiles



Pavage du plan

Problème décidable: Existence d'un algorithme donnant une valeur de vérité en temps fini

Problème du pavage du plan: Un jeu de tuile donné pave-t-il le plan?

- **Conjecture de Wang**:
  Tout jeu de tuile pavant le plan permet un pavage périodique

- **Algorithme de construction récursif**:
  - On rempli l'ensemble des motifs valides de taille n
  - On cherche si il en existe de périodique
  - Si oui, on s'arrète en renvoyant le premier périodique trouvé
  - Sinon, on passe au rang n+1

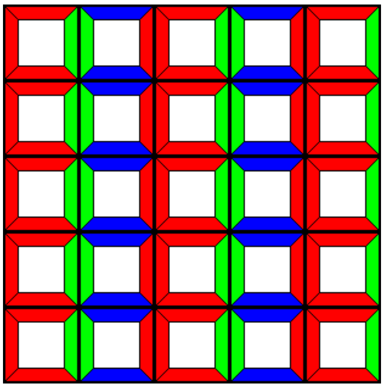
Contre-exemple de la conjecture donné par **Berger** grâce à un jeu de tuile pavant le plan, mais n'admettant aucune configuration périodique

Indécidabilité prouvée plus tard par **Gurevich et Koriakov**

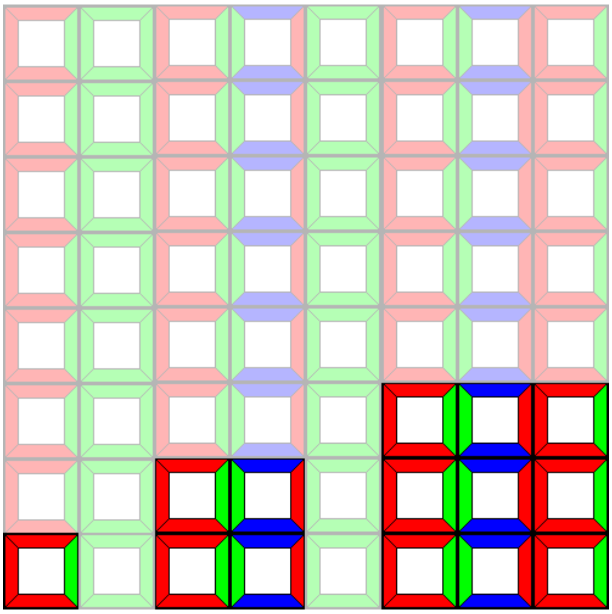
Relation d'ordre

On appelle **relation d'extraction** la relation qui lie deux pavages P_1 et P_2

P_1 est extrait de P_2 si tous les motifs de P_1 sont dans P_2



Pavage 1



Pavage 2

Propriétés:

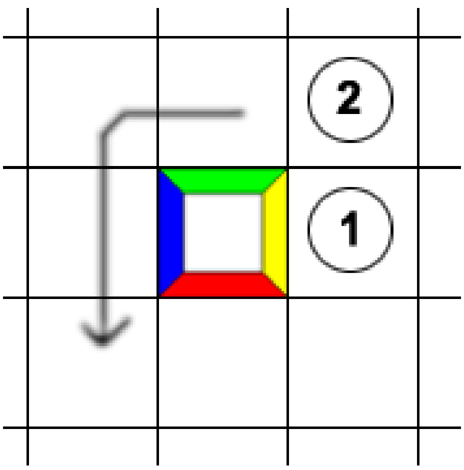
- Non totale
- P_2 est valide $\Rightarrow$ P_1 valide
- M_c critique de P_1, alors M_c critique de P_2

Définition: Un motif est critique dans P si il existe des fenêtres aussi grandes que l'on veut dans P, qui ne contiennent pas ce motif

Extraction diagonale

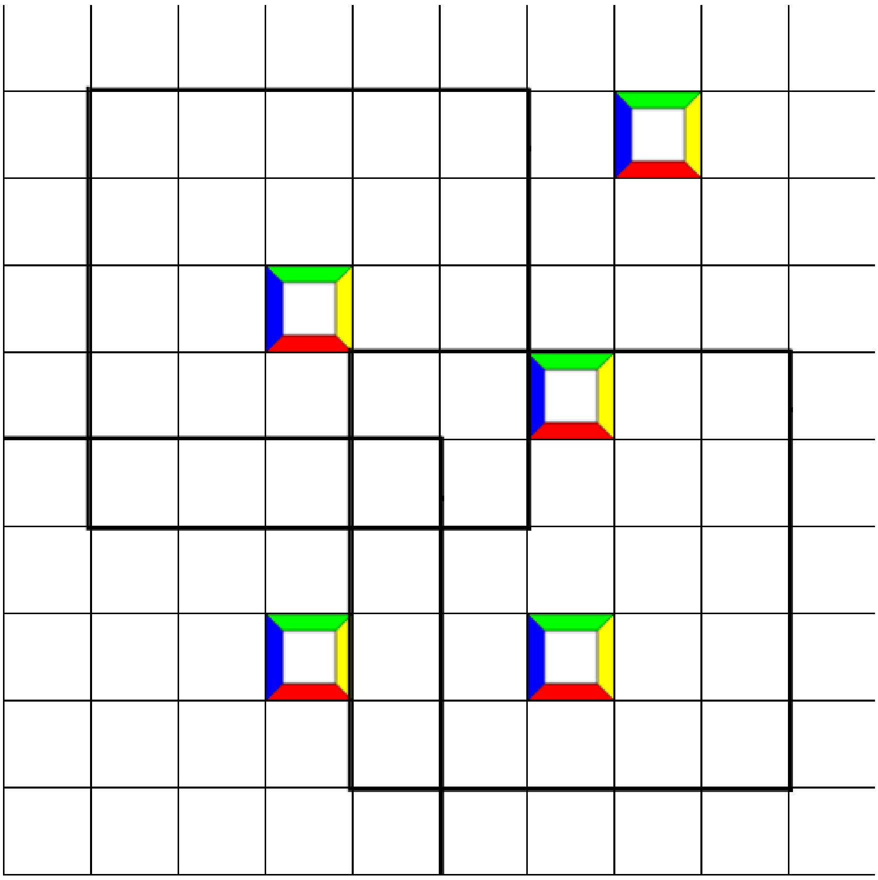
- Nombre fini de tuiles
- On note ***A*** l'ensemble infini des pavages vérifiant la propriété ***P***
- On considère l'emplacement (0,0): il y a une tuile donnée ***t*** qui apparaît un nombre infini de fois
- On note ***A'*** l'ensemble ***A*** restreint aux pavages qui contiennent ***t*** en (0,0)
- Comme ***A'*** est infini, on réapplique l'algorithme sur l'emplacement (0,1), (1,1), ..., de manière à entourer progressivement les emplacements déjà conservés

Au final, le pavage obtenu est extrait de tous les pavages retenus de ***A***



Quasipériodicité

Pavage Quasipériodique: pour tout motif ***m***, extrait du pavage quasipériodique, il existe un entier ***i*** tel que toute fenêtre de taille ***i*** du pavage contienne ***m***



Propriété: Tout pavage périodique est quasipériodique

Preuve: Toute fenêtre de taille n+T (où ***T*** est la période du pavage) contient l'ensemble des motifs de taille ***n***

Fonctions

Fonction de quasipériodicité: On associe à tout ***n***, la taille minimum de la fenêtre qui contient tous les motifs de ***P***, de taille ***n***

Fonction de dénombrement: On associe à tout ***n,*** le nombre de motifs de taille ***n*** que contient le pavage.

On note q_P la fonction de quasipériodicité et g_P la fonction de dénombrement associées au pavage ***P***

Propriété:

- Le pavage ***P*** est quasipériodique est équivalent à: $\exists c \in \mathbb{N} \quad q_{\mathcal{P}}(n) = n + c$
- $\forall n \in \mathbb{N} \quad g_{\mathcal{P}} \leq (q_{\mathcal{P}}(n) + 1 - n)^2$
- Le pavage P est périodique est équivalent à:

$$\exists n \in \mathbb{N} \ tel\ que\ g_{\mathcal{P}}(n) = g_{\mathcal{P}}(n+1)$$

$$\Longleftrightarrow g_{\mathcal{P}} \quad est\ stationnaire$$

Programmation

Sous-Programmes:

- Définition d'un ordre des cases à traiter
- Vérification de validité de la tuile à l'emplacement donné
- Formation de la liste de tuiles qui peuvent aller en position (i,j)

Programme principal

- On envoie au programme principal la liste des tuiles disponibles, qui sont placées en (0,0)
- Le programme principale cherche alors la case suivante à tester
- Il forme alors la liste des tuiles qui peuvent aller dans cette emplacement
- Tant que l'emplacement à tester est différent de (n,n), on rappelle le programme avec comme liste les seuls pavages valides trouvés, et comme emplacement la dernière case remplie

On obtient donc l'ensemble des pavages de taille n x n pouvant être formés grâce au jeu de tuile donné (détails disponibles sur l'annexe donnée)