



Travail d'étude et de recherche

Sandpile on Tiling

Auteurs :

FERSULA Jérémy
DARRIGO Valentin
ZHANG Lin

Encadrant:

PERROT Kévin

May 23, 2019

Abstract

Le but premier de ce projet consiste à développer une application web qui implémente le modèle de piles de sable Abéliennes sur des parties finis du plan pour des pavages de Penrose, afin d'y étudier *a posteriori* le comportement du modèle.

The main goal of this project consists in developing a web app which implements this models on finite parts of the plane for Penrose tilings, in order to study *a posteriori* the model behavior on it.

Contents

1	Introduction	1
2	Contexte théorique	2
3	Implémentation	4
3.1	Motivations	4
3.2	Choix techniques	5
3.3	Spécifications	6
3.4	Piles de sables implémentées	7
4	Penrose	8
4.1	Le pavage kite and dart (P2)	8
4.2	Le pavage Rhombus (P3)	12
5	Tests empiriques	14
6	Conclusion	16

1 Introduction

Le modèle des piles de sable Abéliennes est un modèle d'automate cellulaire relativement récent, puisque formalisé en 1987 par Per Bak, Chao Tang and Kurt Wiesenfeld [1]. La règle de ce modèle est simple : sur un graphe non-orienté quelconque, chaque sommet contient un nombre entier fini de grains de sable - et lorsque un sommet contient exactement ou plus de grains que son degré, il donne un grain à chacun de ses voisins et en perd autant. Ce modèle est en particulier applicable aux pavages du plan. Dans ce cadre, il présente des propriétés géométriques remarquables, par l'apparition sous certaines conditions de formes fractales difficiles à prévoir mais simple à produire expérimentalement. Le projet s'intéresse ainsi au développement d'une application web qui implémente ce modèle sur des pavages potentiellement quelconques, avec pour objectif d'en étudier le comportement sur des pavages de Penrose. Github : <https://github.com/hehaOhyeah/JS-Sandpile>

2 Contexte théorique

Les piles de sables Abéliennes présentent un certain nombre de propriétés non triviales prouvées par un formalisme mathématique dont l'exposition complète dépasserait le cadre de ce projet. Cependant, pour la bonne compréhension de l'application et des comportements observés, il convient d'énoncer quelques-une de ces propriétés, et d'exposer un formalisme simplifié. Pour en savoir plus, voir [3].

Définition 1 *Une configuration sur un graphe (E, V) est une application $z : V \rightarrow \mathbb{N}$ qui associe à chaque sommet son nombre de grains de sable.*

Une configuration est dite stable si et seulement si $\forall x \in V, z(x) < \deg(x)$. Un effondrement (*toppling*) T_v du sommet $v \in V$ avec une dissipation $\alpha \in \mathbb{N}$ est la transformation définit comme suit :

$$z(v) \mapsto z(v) - \deg(v) - \alpha \quad \text{si } z(v) > \deg(v) + \alpha$$

et pour tout voisin u de v :

$$z(u) \mapsto z(u) + 1$$

Les propositions et théorèmes exposés ici sont valables sous réserve qu'au moins un sommet ait une dissipation > 0 .

Remarque : Les effondrements commutent sur les sommets instables.

Théorème 1 *Il existe une énumération $\{v_1, v_2, \dots, v_N\} \in V^N$ unique à la permutation près telle que la transformation $\Pi_{i=1}^N T_i$ aboutisse à une configuration stable. Cette transformation est appelée *stabilisation*.*

On note \star l'opération agissant sur deux piles de sable, consistant à ajouter les grains de chaque tuiles et à stabiliser la configuration obtenue.

Proposition 1 *L'ensemble des configurations stables munit de l'opération \star forme un monoïde commutatif.*

Cette proposition est tout à fait intuitive, lorsque l'on stabilise la somme de deux configurations stables, on obtient une configuration stable, et l'élément identité est la configuration uniformément égale à zero.

Définition 2 *Une configuration est dite récurrente si il est possible de l'atteindre à partir de n'importe quelle configuration initiale, par ajout de grains et stabilisation successifs.*

Le nombre de configuration récurrente pour un graphe donné est connu [2] :

Théorème 2 *Le nombre de configuration récurrente est exactement égal au déterminant de la matrice de l'opérateur éffondrement, c'est à dire le laplacien du graphe (Δ) auquel on ajoute la matrice diagonale des dissipations.*

Exemple : pour une grille carrée 2x2 :

$$\Delta = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix} \quad D \text{ (dissipation)} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

$$\text{Nombre de configuration stable} = 4^4 = 256$$

$$\text{Nombre de configurations récurrentes} = \text{Det}(\Delta + D) = 192$$

Proposition 2 *L'ensemble des configurations récurrentes munit de l'opération \star forme un groupe commutatif (abélien).*

Cette proposition constitue la clef de voûte du modèle. La configuration identité de ce groupe pour un pavage présente des formes fractales difficiles à déterminer *a priori*. Cependant, il est facile d'étudier expérimentalement ces configurations.

3 Implémentation

3.1 Motivations

Notre logiciel contenons

- Fonctionnalités attendues
 - Les boutons (Play, Pause, choix de la grille, choix des couleurs, ajout de légende, indicateur pour zoom).
 - Augmenter/ baisser La vitesse d'écoulement des grains de sables.
 - L'utilisateur peut modifier les couleurs des grains.
 - Sélection une cellule à la souris.
 - L'utilisateur peut créer des types différentes grilles (carré, triangulaire, hexagonale).
 - Pavage penrose

- Logiciels existants

Nous avons pu tester les applications différentes en Piles de sable Abéliennes.

Exemple sur les applications de piles de sables: [The Abelian Sandpile Model](#), [Wikipédia](#), [Sandpiles](#).

Les comparaisons des 2 models:

Les différents critères de comparaisons:

- Les fonctionnalités:

- * capacité à s'intégrer à votre système informatique

app1 ¹	app2 ²
oui	oui

- * programmer un export/import personnalisé

app1 ¹	app2 ²
non	non

- * normaliser l'arborescence de classement des documents

app1 ¹	app2 ²
oui	non

- * sauvegarde des documents

app1 ¹	app2 ²
oui	non

– Les sécurités des données:

* sauvegarder les configurations

app1 ¹	app2 ²
non	neant

– L’usage:

* ergonomique

app1 ¹	app2 ²
oui	neant

* travailler de manière évolutive

app1 ¹	app2q ²
oui	neant

* facilité d’installation

app1 ¹	app2q ²
oui	neant

* reutilisabilité

app1 ¹	app2 ²
oui	oui

– L’accessibilité:

* opensource

app1 ¹	app2 ²
oui	oui

3.2 Choix techniques

Contraintes : L’application doit pouvoir être lancée sur un navigateur web. Elle être relativement rapide.

JavaScript : Nous avons choisi Langage JavaScript pour interagir avec le HTML permettant ainsi d’apporter du dynamisme à l’intérieur des pages web.

THREE.js : la camera est déjà implémenter, on peut facilement déplacer la vue, faire des zooms. Il est possible d’envoyer un rayon est de récupérer toutes les faces interceptées par ce rayon, on utilise cette méthode pour implémenter le clique (le fait d’ajouter un grain sur une tuile en cliquant par exemple)

3.3 Spécifications

L'application contient deux classes à partir desquelles les pavages sont construits. La première classe `Tile` représente une tuile, la seconde `Tiling` représente un pavage sous la forme d'un tableau de `Tile`. On expose ici les attributs et méthodes principales, sans en faire l'énumération complète. L'intégralité de la documentation est disponible sur la [page github](#).

Tile

- `Number` `id` *Id de la tuile*
- `Array<Number>` `neighbors` *Ids des voisins*
- `Array<Number>` `points` *Points représentant la tuile dans `THREE.js`*
- `Number` `toppleLimit` *Limite d'effondrement de la tuile*

Tiling

- `Array<Tile>` `tiles`
- `THREE.js Mesh` `mesh` *Représentation du pavage*

On construit ainsi différents pavages, de sorte que tous évoluent par l'appel des mêmes méthodes de la classe `Tiling`. Ces méthodes dont pour la plupart le nom est explicite sont par exemple `iterate()` ou encore `addOne(Number index, Number amount)`.

3.4 Piles de sables implémentées

Square Tiling : Pile de sable la plus classique, est une matrice de taille nm dont les tuiles sont des carrés.

Hexagonal Tiling : Matrice de taille nm dont les tuiles sont des Hexagones.

Triangular Tiling : Piles de sables en forme de triangle composée de tuiles triangulaires.

Penrose HK : Pile de sables suivant le pavage de penrose P2 (kite and dart). La construction de ce pavage se fait à partir d'une tuile half-kite.

Penrose HD : Pile de sables suivant le pavage de penrose P2 (kite and dart). La construction de ce pavage se fait à partir d'une tuile half-dart.

Penrose Sun : Pile de sables suivant le pavage de penrose P2 (kite and dart). La construction de ce pavage se fait à partir d'un solei (ensemble de dix tuiles half-kite).

Penrose Star : Pile de sables suivant le pavage de penrose P2 (kite and dart). La construction de ce pavage se fait à partir d'un solei (ensemble de dix tuiles half-dart).

Penrose HK Rhombus : Pile de sables suivant le pavage de penrose P3 (kite and dart). La construction de ce pavage se fait à partir d'une tuile half-kite.

Penrose HD Rhombus: Pile de sables suivant le pavage de penrose P3 (kite and dart). La construction de ce pavage se fait à partir d'une tuile half-dart.

Penrose Sun Rhombus: Pile de sables suivant le pavage de penrose P3 (kite and dart). La construction de ce pavage se fait à partir d'un solei (ensemble de dix tuiles half-kite).

Penrose Star Rhombus: Pile de sables suivant le pavage de penrose P3 (kite and dart). La construction de ce pavage se fait à partir d'un solei (ensemble de dix tuiles half-dart).

Square Moore: Comme la piles de sables carrés mais avec un voisinage de Moore.

4 Penrose

pavages de Penrose ont été inventé en 1970 par le mathématicien et physicien britannique Roger Penrose.

Il existe 3 type de pavage, ici on s'intéressera à seulement deux d'entre eux.

4.1 Le pavage kite and dart (P2)

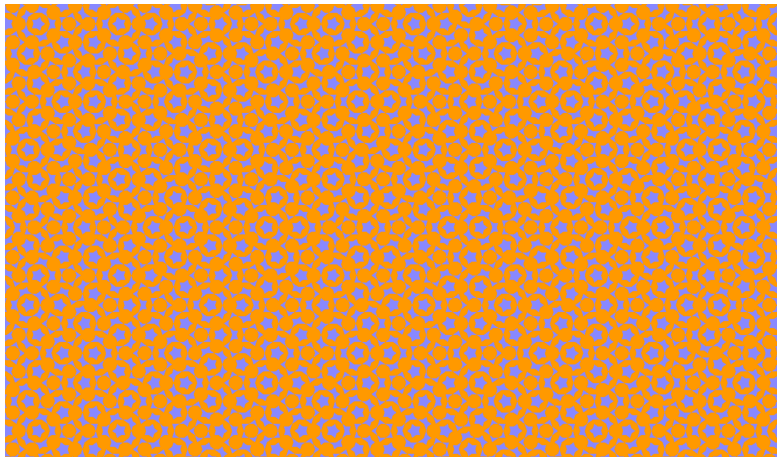


Figure 1: pavage de penrose kite and dart

Il existe plusieurs façons de générer un pavage de penrose, on a choisi d'utiliser la méthode par substitution

Cette méthode consiste à prendre deux sous tuiles : half-kite et half-dart



Figure 2: à gauche la proto tuile half-kite, à droite la proto tuile half-dart

Ces triangles sont aussi connus sous le nom de triangle d'or, le ratio de la longueur du plus long segment du triangle et le plus court est égale au nombre d'or

- la proto tuile half-kite à 1 angle de $\pi/5$ et 2 de $2\pi/5$
- la proto tuile half-dart à 1 angle de $3\pi/5$ et 2 de $\pi/5$

la combinaison de ces sous tuiles forment les deux tuiles du pavage, les tuiles dart et kite :

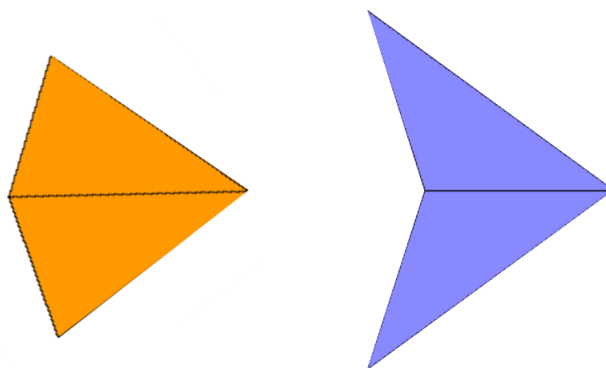


Figure 3: à gauche la tuile kite, à droite la tuile dart

pour obtenir un pavage de penrose il suffit de partir d'une de ces tuiles et d'y appliquer leurs règles de substitution respectives n fois ces règles sont les suivantes :

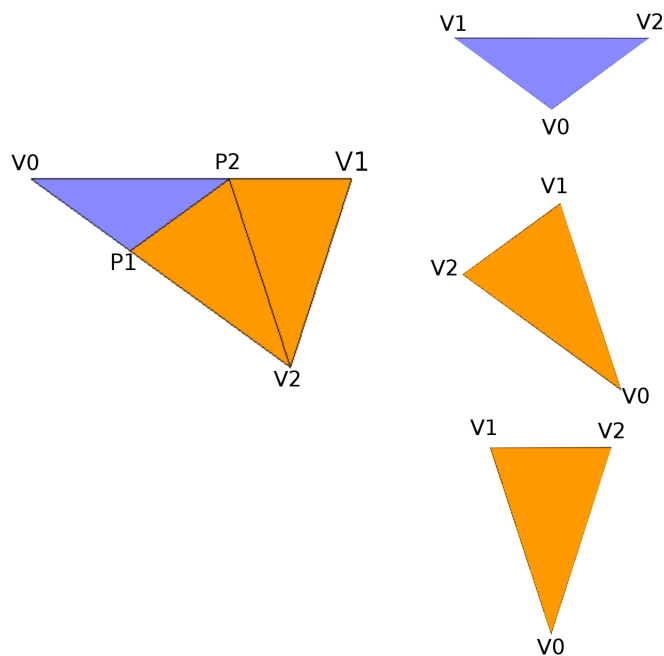


Figure 4: règle de substitution de la sous tuiles half-kite

V_0 , V_1 , V_2 sont les sommets du triangles leurs dispositions sont importantes, ils définissent dans quel sens le triangle sera découpé. P_1 et P_2 sont des points intermédiaires définissant les sommets des triangles pour la substitution.

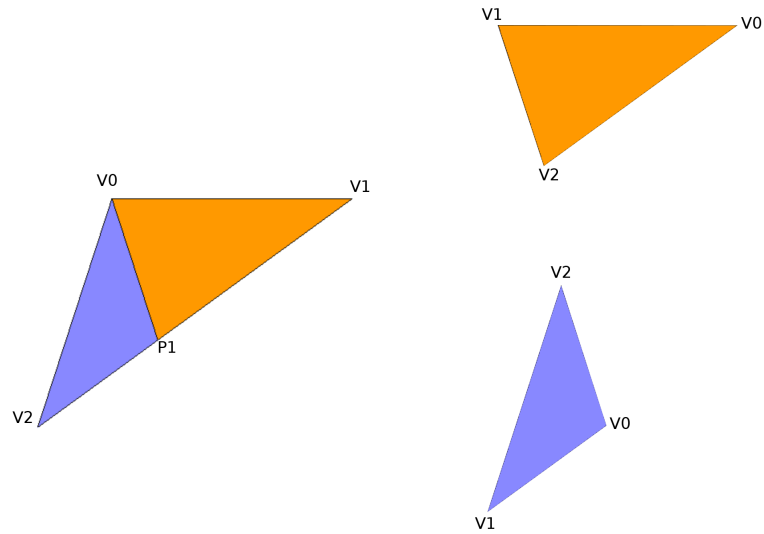


Figure 5: règle de substitution de la sous tuiles half-dart

Dans notre projet il ne suffit pas d'afficher un pavage de penrose il faut aussi quelle tuile est adjacente à quelle tuile. On a trouvé la solution suivante, tous les triangles contiennent les références vers leurs arrêtes et chaque arrêtes contiennent les références de leurs triangles incidents. Ainsi on peut reconstituer les tuiles et déterminer les tuiles adjacentes.

4.2 Le pavage Rhombus (P3)

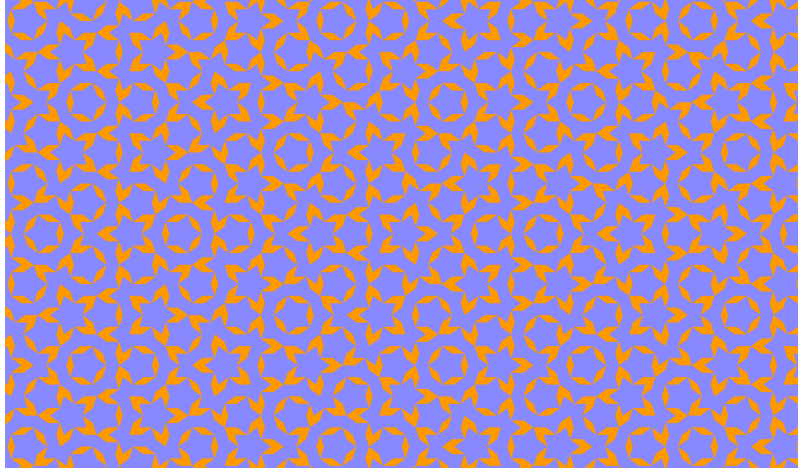


Figure 6: pavage de penrose Rhombus

On peut aussi générer un autre pavage par une méthode de substitution avec les mêmes triangles, cependant les tuiles et les règles de substitutions sont différentes.

Les tuiles composant le pavages sont les tuiles thick et les tuiles thin.

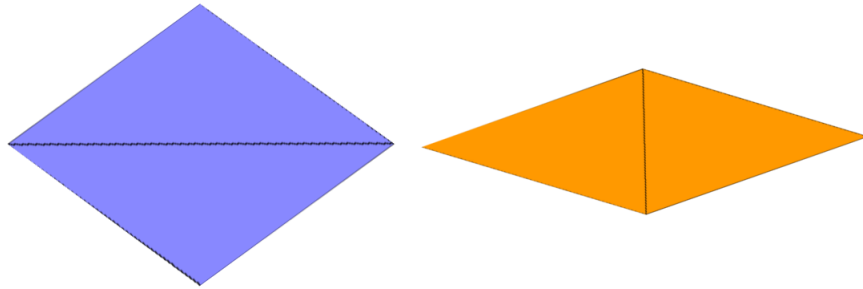


Figure 7: à gauche la tuile thick à droite la tuile thin

Les règles de substitution pour ce pavage sont :

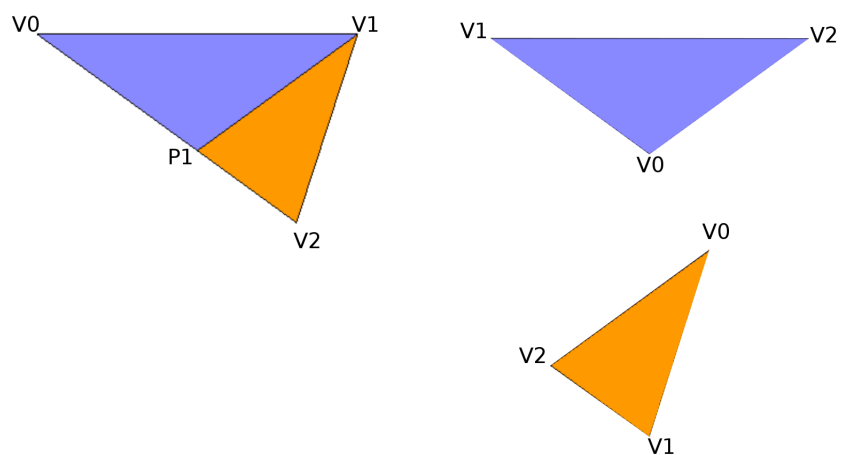


Figure 8: règle de substitution de la sous tuiles half-kite

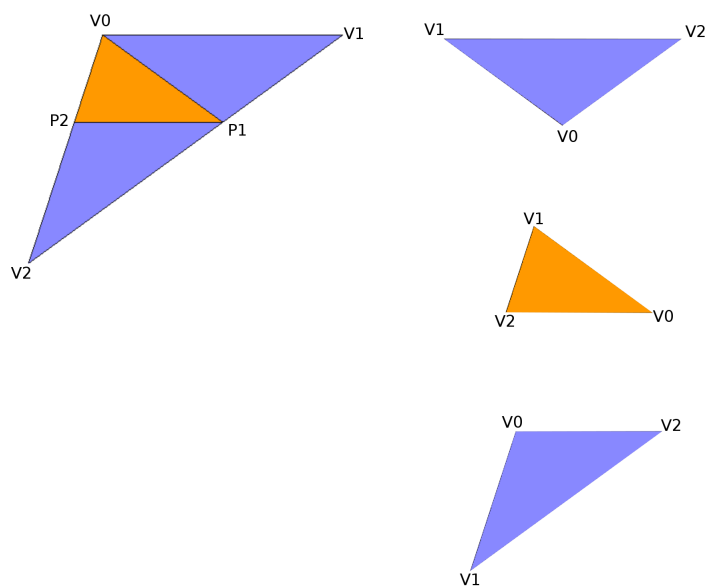
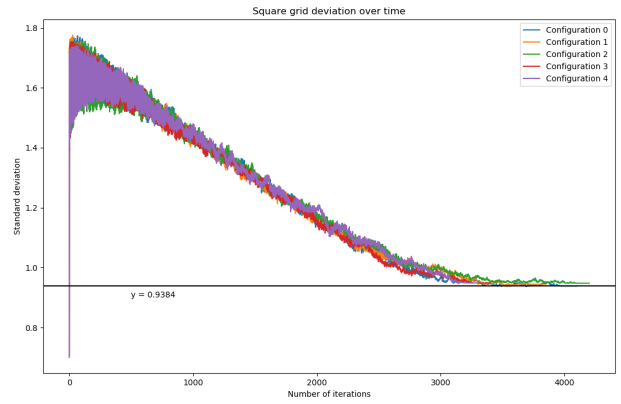
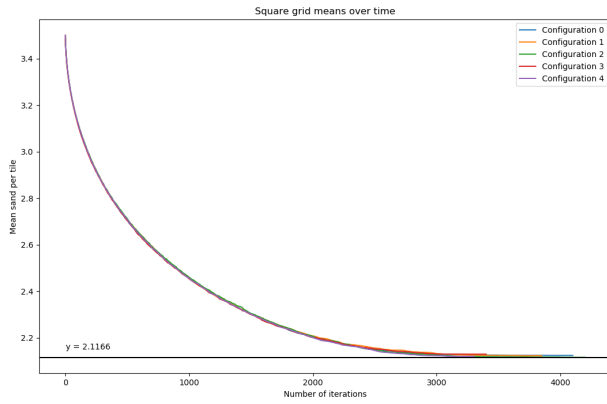


Figure 9: règle de substitution de la sous tuiles half-dart

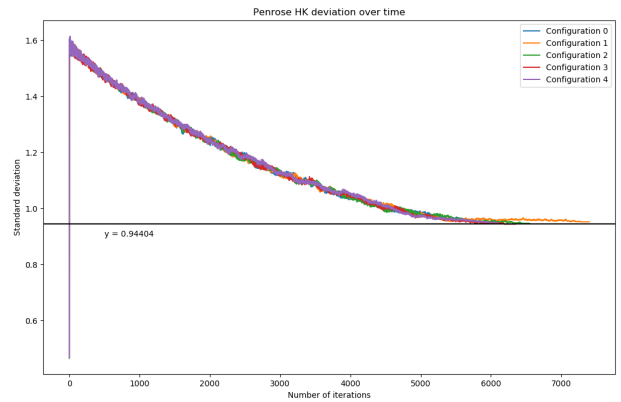
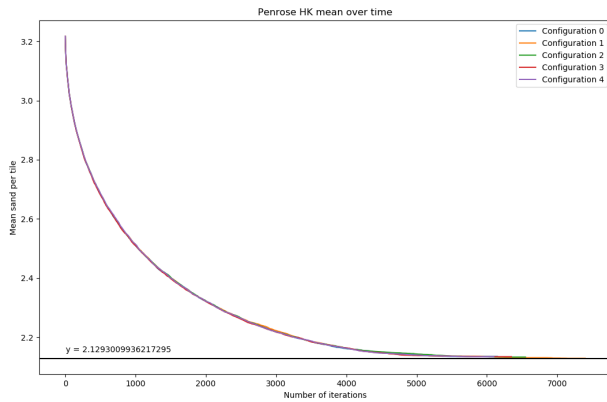
5 Tests empiriques

L'objectif principal du projet rempli, nous avons pu nous pencher sur la génération de données statistiques pour certains effondrement de piles de sable. Les expérimentations conduites ici sont l'ajout du maximum stable de grains partout, puis un ajout de 5000 grains de façon aléatoire.

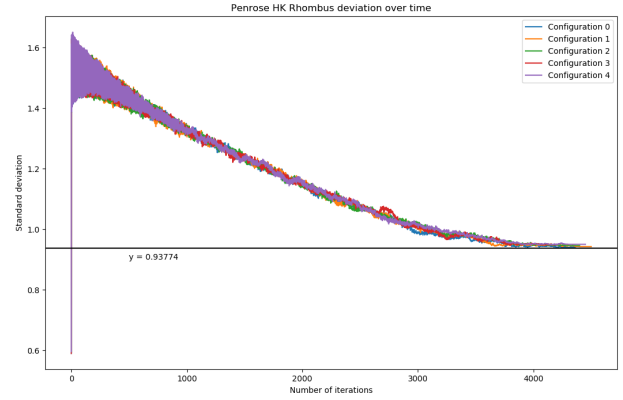
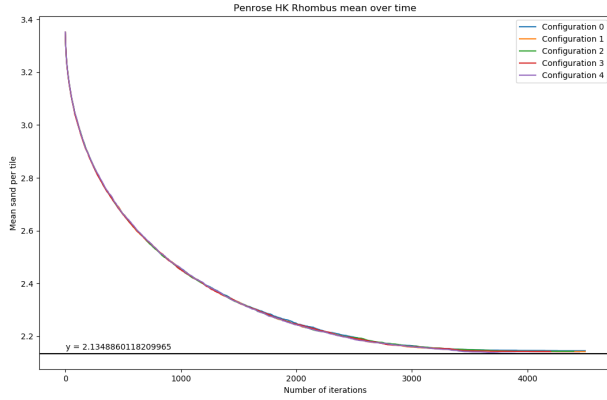
Grille carrée, 10 000 tuiles



Penrose HK, 23 047 tuiles



Penrose HK Rhombus, 14 212 tuiles



On observe sur ces graphiques que la moyenne et l'écart-type semble converger vers une valeur précise lors de la stabilisation. Il apparaît également que l'écart-type semble suivre une loi exponentielle inverse, mais que la moyenne décroît encore plus rapidement.

Une mesure pertinente sur les piles de sable est le nombre d'avalanches au cours du temps, mesure qui n'a pas pu être faite mais qui constitue un point intéressant pour la continuité du projet.

6 Conclusion

Dans la mesure où l'objectif de notre travail était réalisé une application web en JavaScript avec les fonctionnalités indiquées dans le chapitre 3, nous pouvons dire que l'objectif est atteint. L'application, disponible publiquement sur la page github (<https://github.com/heha0hyeah/JS-Sandpile>) est fonctionnelle et peut être utilisée à une fin éducative ou pour effectuer des mesures.

References

- [1] Per Bak, Chao Tang, and Kurt Wiesenfeld. Self-organized criticality: An explanation of the $1/f$ noise. *Physical review letters*, 59(4):381, 1987.
- [2] Deepak Dhar and Ramakrishna Ramaswamy. Exactly solved model of self-organized critical phenomena. *Physical Review Letters*, 63(16):1659, 1989.
- [3] Ronald Meester, Frank Redig, and Dmitri Znamenski. The abelian sandpile; a mathematical introduction. *arXiv preprint cond-mat/0301481*, 2003.