

# Rapport final de TIPE

PEREIRA Romain

4 Juin 2017

## Sommaire

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Corps principal</b>	<b>2</b>
2.1	Modalités d'action . . . . .	2
2.2	Restitution des résultats . . . . .	2
2.3	Analyse - Exploitation - Discussion . . . . .	3
<b>3</b>	<b>Conclusion générale</b>	<b>4</b>
<b>4</b>	<b>Références</b>	<b>5</b>

## Préambule

Un léger inflechissement a eu lieu quant à l'étude spatiale et temporelle de la rasterisation. En effet, ces problématiques me paraissent trop vastes et non appropriées pour être traitées entièrement dans le cadre de mon TIPE. Ainsi, j'ai décidé de me limiter à une présentation rapide du procédé dans mon étude.

Le reste du travail a bien été mené à son terme : j'ai étudié en détail les champs de hauteurs.

## 1 Introduction

Les algorithmes de rendu graphique en temps réel consistent à générer des images numériques, qui sont calculées et affichées suffisamment rapidement pour faire l'illusion de continuité sur l'œil humain. Mon travail s'est orienté sur l'étude de grille uniforme bidimensionnel sur-elevés, basée sur des champs de hauteurs. On peut ainsi synthétiser l'image d'un terrain rapidement grâce à une rasterisation. Ce TIPE décrira le processus de rendu, la façon dont on crée des champs des hauteurs, et les structures de données optimales pour les stocker en mémoire.

## 2 Corps principal

### 2.1 Modalités d'action

J'étudie la théorie de deux méthodes de rendus graphiques : le lancer de rayon et la rasterisation. Dans la synthèse d'image numérique, les objets sont souvent représentés virtuellement sous forme d'une liste de vecteurs, qui forment un maillage surfacique de l'objet à partir de primitives géométriques (triangles, carrés, sphères...)

Je me suis naturellement intéressé aux champs de hauteurs, qui sont des champs scalaires à 2 coordonnées : à une position « (x, z) », elle associe une hauteur « y ». Ils m'ont permis de générer des surfaces de type terrain. Il m'a fallu ensuite être en mesure de triangulariser ces surfaces, afin d'en faire des objets numériques prêts à être injectés dans la rasterisation. Afin de mettre en pratique ces études, j'ai créé un programme capable de travailler avec des terrains à 3 dimensions. J'ai ensuite utilisé une bibliothèque qui implémente le processus de rasterisation pour avoir un rendu concret.

### 2.2 Restitution des résultats

Il s'avère que pour faire de la synthèse d'image en temps réel, la rasterisation est plus adaptée, je ne me suis donc pas attardé sur le lancer de rayon.

La complexité spatiale des champs de hauteurs m'a mené à comparer 3 méthodes de représentation des données. Le résultat de cette étude est donnée ci dessous.

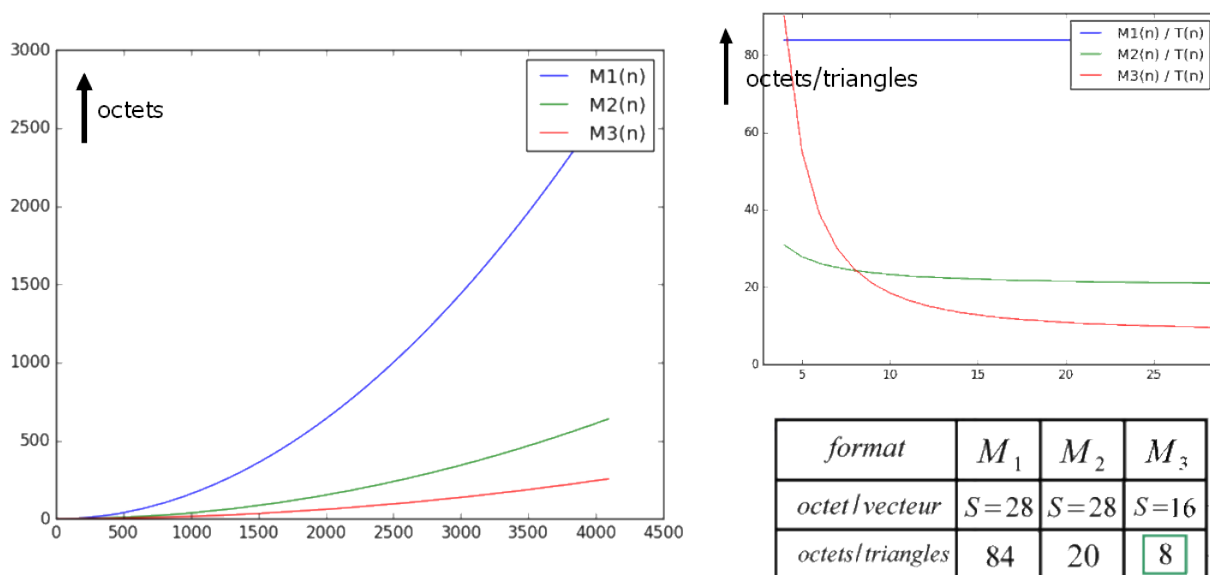


Figure 1: Utilisation mémoire d'un terrain carré avec  $n$  vecteurs par coté

J'ai décomposé l'étude de la complexité temporelle en deux sous-parties.

- Il y a premièrement un temps de préparation, qui consiste à envoyer les données sur

le serveur de calcul (carte graphique), et à initialiser la parallélisation. Ce temps dépend énormément de l'architecture du serveur (nombre de coeurs de calculs à disposition, bande passante...) La bande passante peut varier de 1 à 100 GO/s selon l'architecture... ce qui représente 1Mo/ms à 1Go/ms. Il y a ainsi un lien étroit entre les complexités spatiales et temporelles : la synthèse sera d'autant plus rapide que les données sont légères.

- La deuxième sous-partie est l'exécution de la rasterisation. Elle peut être interprétée comme le nombre de 'triangle par secondes' que le serveur est capable de calculer.

## 2.3 Analyse - Exploitation - Discussion

J'ai cherché à optimiser l'espace mémoire requis car la complexité spatiale est très étroitement liée aux performances temporelles. Ainsi, j'ai pu atteindre une complexité asymptotique que je juge optimale.

$$C(n) = S * n + o(n)$$

Figure 2: *Utilisation mémoire d'un terrain carré n vecteurs (de S octets) par coté*

### 3 Conclusion générale

Mon TIPE a donc été centré sur la synthèse d'image d'un terrain virtuelle, utilisant la rasterisation.

## 4 Références