



AIX-MARSEILLE UNIVERSITÉ

M3101- SYSTÈMES D'EXPLOITATION

---

# Calcul de surface d'un objet 3D maillé

---

*Auteurs :*  
Lucien Aubert  
Thibaut Jallois

*Enseignant :*  
Romain Raffin

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Environnement d'expérimentation</b>	<b>2</b>
2.1	Machine 1 . . . . .	2
2.2	Machine 2 . . . . .	2
2.3	Machine 3 . . . . .	2
<b>3</b>	<b>Algorithmique et implémentation</b>	<b>3</b>
3.1	Algorithme séquentiel . . . . .	3
3.1.1	Conception . . . . .	3
3.2	Algorithmes parallèles . . . . .	3
3.2.1	Threads . . . . .	3
3.2.2	OpenMP . . . . .	3
<b>4</b>	<b>Résultats</b>	<b>4</b>
<b>5</b>	<b>Conclusion</b>	<b>4</b>

## 1 Introduction

L'objectif consiste en l'optimisation, par parallélisation, du calcul de la surface d'un objet 3D maillé (triangles) au format OFF[1] à l'aide de la formule de Héron[2].

Le programme implémente trois algorithmes

- Classique, séquentiel
- Avec `pthread`[3], parallélisé
- Avec `OpenMP`, parallélisé également

## 2 Environnement d'expérimentation

La phase de test s'est déroulée sur trois machines dont voici les configurations

### 2.1 Machine 1

Intel i7-3612QM 2.10GHz, 8 CPU, 4 cœurs, L1 64K, L2 256K, L3 6144K  
12Go RAM DDR3 800MHz

### 2.2 Machine 2

AMD FX(tm)-8350 4.20GHz, 8 CPU, 4 cœurs, L1 64K, L2 2048K, L3 8192K  
8Go RAM DDR3 2133MHz

### 2.3 Machine 3

Intel i5-4590 3.30GHz, 4 CPU, 4 cœurs, L1 32K, L2 256K, L3 6144K  
8Go RAM DDR3 1600MHz

## 3 Algorithmique et implémentation

### 3.1 Algorithme séquentiel

De manière à pouvoir travailler sur les données contenues dans les fichier OFF on lit ce fichier et on place chaque sommet et chaque face dans deux `std::deque`.

On somme l'aire de chaque triangle du volume, calculée à l'aide de la formule de Héron, ce qui nous donne la surface totale du volume.

#### 3.1.1 Conception

Dans la formule de Héron  $S = \sqrt{p(p-a)(p-b)(p-c)}$  nous avons besoin des longueurs des côtés de chaque triangle. La méthode `Point::distanceFrom(Point*)` nous permet donc d'obtenir les termes  $a$ ,  $b$  et  $c$ .

Ces termes sont également utilisés pour calculer  $p = \frac{a+b+c}{2}$

### 3.2 Algorithmes parallèles

#### 3.2.1 Threads

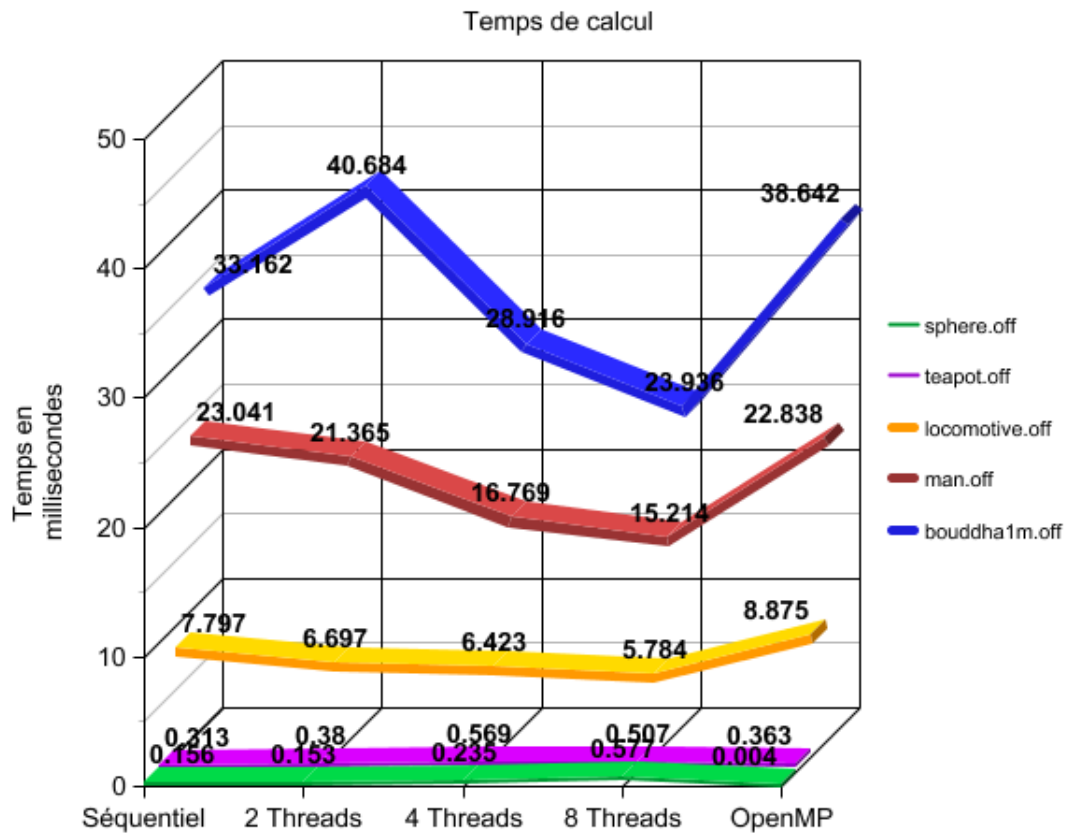
Le `std::deque` de faces est décomposé en  $n$  sous-ensembles correspondants aux faces sur lesquelles chaque thread va travailler.

On lance les thread en leur donnant l'adresse de la fonction `computeSurface(void*)` puis on somme leur sortie en attendant la fin de leur exécution grâce à la fonction `pthread_join(pthread_t, void**)`

#### 3.2.2 OpenMP

Il n'y a pas grand chose à faire pour utiliser OpenMP. L'ajout d'un `#pragma` suffit à paralléliser la boucle `for` qui somme les valeurs de chaque triangle.

## 4 Résultats



## 5 Conclusion

## Références

- [1] Wikipedia. Spécification du format de fichier off.  
[https://en.wikipedia.org/wiki/OFF\\_\(file\\_format\)](https://en.wikipedia.org/wiki/OFF_(file_format)).
- [2] Wikipedia. Formule de héron.  
[https://en.wikipedia.org/wiki/Heron%27s\\_formula](https://en.wikipedia.org/wiki/Heron%27s_formula).
- [3] Franck Hecht. Initiation à la programmation multitâche en c avec pthreads.  
<http://franckh.developpez.com/tutoriels/posix/pthreads/>.