

AMS314 : Partie 1

Thibault Mougin

Mars 2024

Vous trouverez le code sur <https://github.com/thibaultmougin/AMS314>

1 Lecture des maillages

On commence par implémenter les fonctions de qualité des triangles données par

$$Q_1(K) = \alpha_1 \frac{\sum_i \ell_i^2}{|K|} \quad \text{et} \quad Q_2(K) = \alpha_2 \frac{h_{\max}}{\rho(K)}$$

Ces histogrammes présentent les distributions (Q_1 à gauche et Q_2 à droite) des qualités des triangles des maillages `carre_05h.mesh`, `carre_h.mesh`, `carre_2h.mesh` et `carre_4h.mesh` :

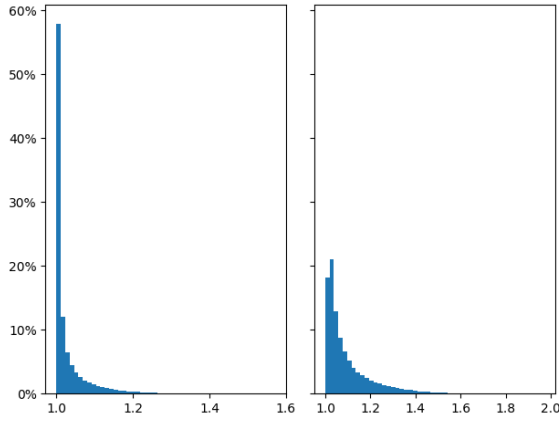


Figure 1: Qualités pour `carre_05h.mesh`

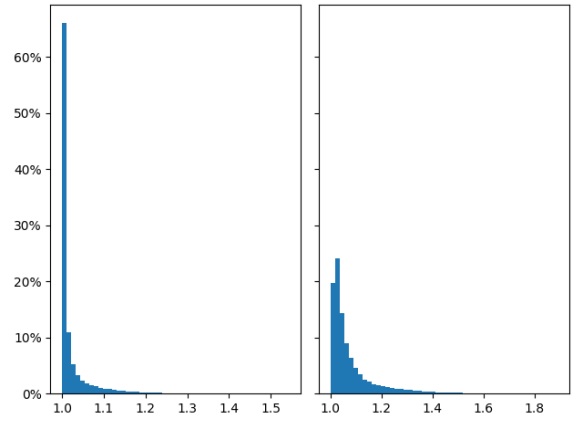


Figure 2: Qualités pour `carre_h.mesh`

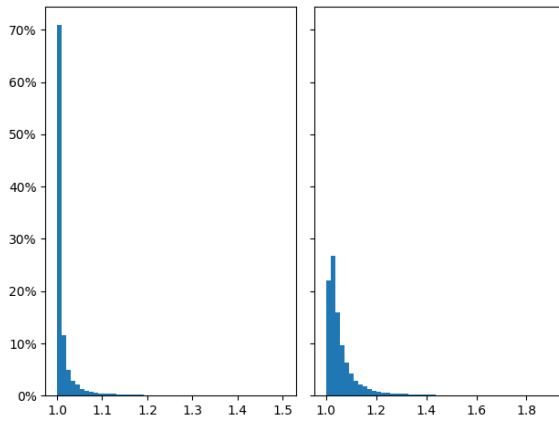


Figure 3: Qualités pour `carre_2h.mesh`

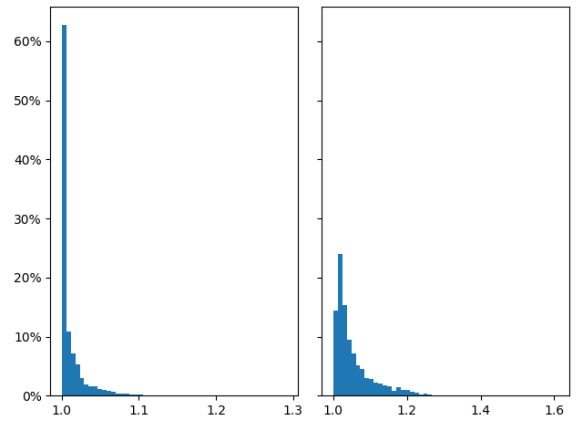


Figure 4: Qualités pour `carre_4h.mesh`

On remarque que Q_2 est plus discriminante et que les maillages les plus fins ont généralement des triangles de qualités moindres.

2 Algorithme quadratique et table de hachage

On calcule le tableau des voisins par l'algorithme quadratique naïf puis à l'aide d'une table de hachage:

	carre_05h	carre_h	carre_2h	carre_4h
Sommets	178746	43758	10663	2625
Triangles	355946	86742	20938	5084
Algo. quadratique (s)	1746	80.42	4.784	0.2803
Table de hachage (s)	0.05685	0.009661	0.002825	0.0004119

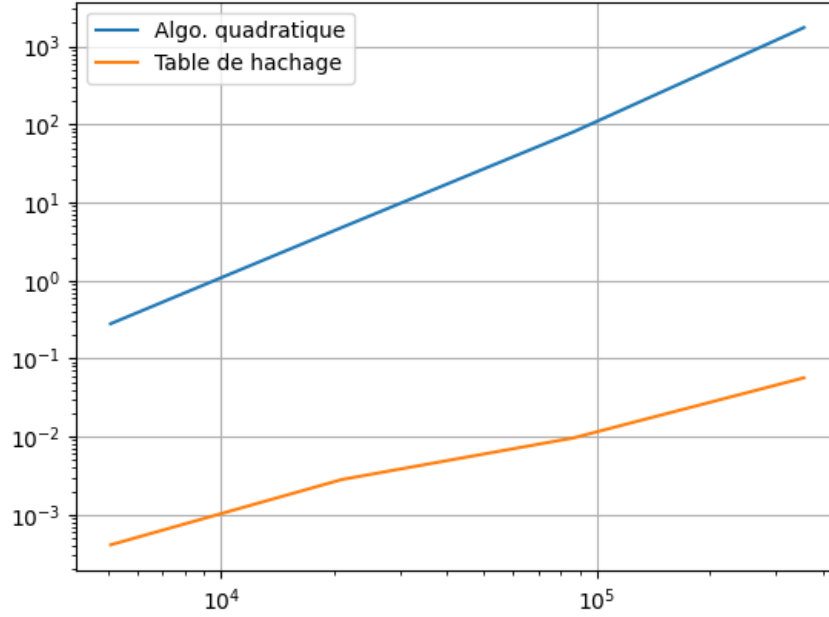


Figure 5: Temps CPU vs nombre N de triangles pour la construction du tableau des voisins

On observe bien les complexités quadratique et en $\mathcal{O}(N \log(N))$ attendues.

La table de hachage fournit aussi le nombre d'arêtes (c'est le nombre **NbrObj** d'éléments listés dans la table) et le nombre d'arêtes frontières (celles qui ont 0 comme deuxième triangle):

	naca0012	ls89	hlcrm
Sommets	7379	2239	2651
Triangles	14266	4098	4616
Nb. arêtes	21646	6338	7270
Nb. arêtes frontières	494	382	692

On s'intéresse au nombre de collisions pour différentes fonctions de hachage (toujours avec `SizeObj=2*msh->NbrVer` et `NbrMaxObj=3*msh->NbrTri`) :

	carre_05h	carre_h	carre_2h	carre_4h
<code>ip1+ip2</code>	0.570	0.571	0.578	0.537
<code>abs(ip1-ip2)</code>	0.731	0.728	0.724	0.741
<code>min(ip1,ip2)</code>	0.707	0.707	0.706	0.705
<code>ip1*ip2</code>	0.364	0.577	0.521	0.593

Table 1: Ratios *collisions/nombre total d'arêtes* pour différents maillages et clés de hachage

On utilisera `ip1+ip2`, qui provoque en général le moins de collisions (`ip1*ip2` aurait aussi été un choix acceptable.. et d'autant plus si on prenait `SizeObj` plus grand).