

Avancements depuis la dernière réunion :

Calcul de l'objectif
& Amélioration du graphe des fusions
& Problème dans l'algo de fusion identifié
& Méta-heuristiques

1 - Calcul de l'objectif

Calcul de la récompense :

$$\frac{2 \cdot (\sum (\text{nombre de fusions enchainées})^2)}{\text{Nombre de polyèdres} - 1}$$

- ***nombre de fusions enchainées*** :
→ par exemple, = **2** si on fusionne 3 polyèdres ensembles
- On multiplie par 2 pour équilibré par rapport à l'objectif et à la pénalité

1 - Calcul de l'objectif

Calcul de la pénalité :

$$\frac{\left(\frac{\sum(\text{Distances})}{\text{Distance max} \times \text{Nombre de polyèdres}} \right)^2}{3}$$

- Avec **distance minimale** entre deux sommets du graphe = 1
- **Distance max** : distance entre les 2 sommets reliés les plus éloignés + 1
- On divise par 3 pour équilibrer par rapport à l'objectif et à la récompense

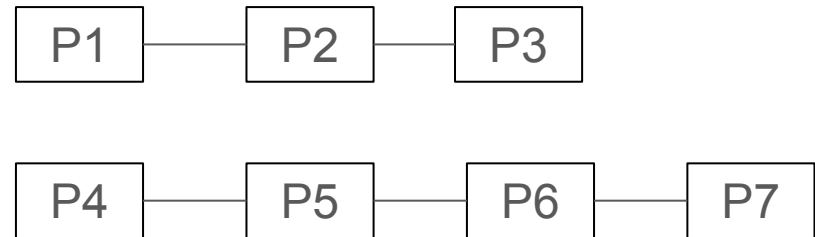
Exemple :

Distance entre **P1** et **P2** = 1

Distance entre **P4** et **P7** = 3

Distance **max** = 3 + 1 = 4

Distance entre **P1** et **P4** = distance **max** = 4



1 - Calcul de l'objectif

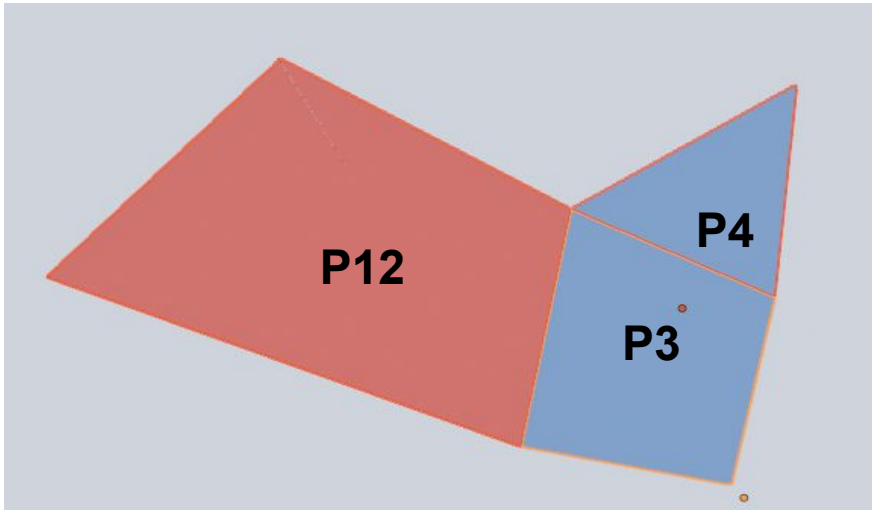
Calcul de la valeur finale de l'objectif à minimiser :

$$1 + \left(\frac{\text{Nombre de polyèdres après fusion}}{\text{Nombre de polyèdres}} \right) + \text{pénalité} - \text{récompense}$$

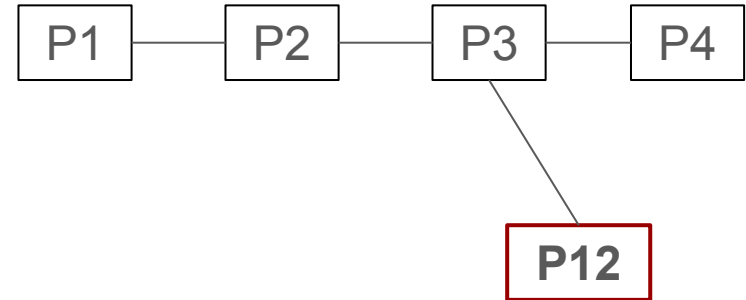
2 - Amélioration du graphe des fusions

Les polyèdres fusionnés (résultants d'une ou plusieurs fusions) sont intégrés dans le graphe dynamiquement au cours de l'exécution et utilisés pour diminuer le nombre de calculs ainsi que pour calculer les distances pour la fonction objectif.

Pour reprendre l'exemple de la dernière fois :



Graphe des fusions convexes :

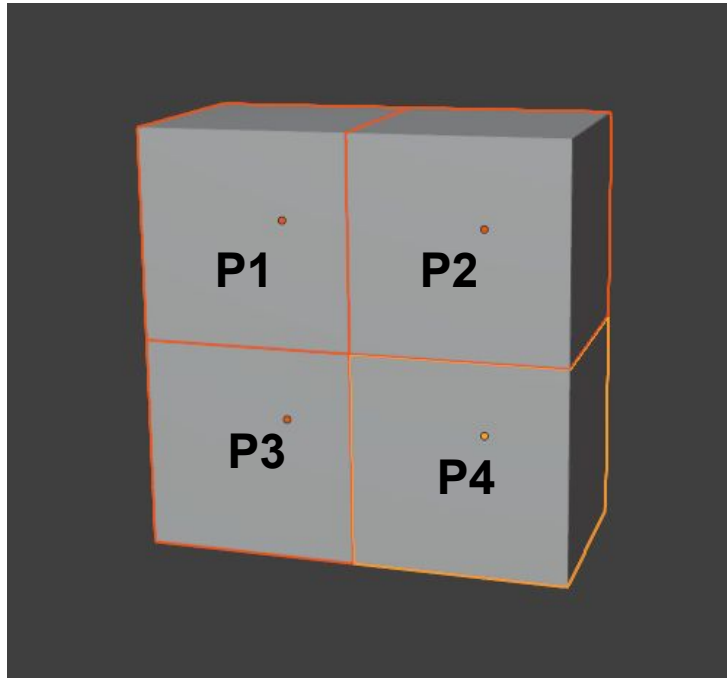


3 - Problème identifié dans l'algorithme de fusion actuel

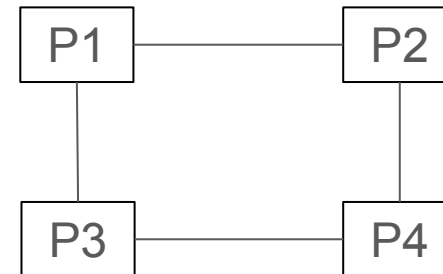
3 - Problème identifié dans l'algorithme de fusion actuel

Exemple avec 4 cubes :

Liste des polyèdres : P1, P2, P4, P3

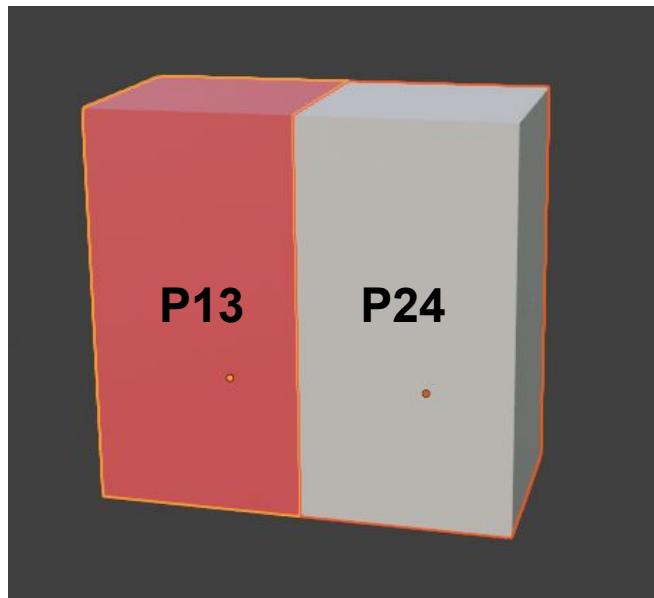
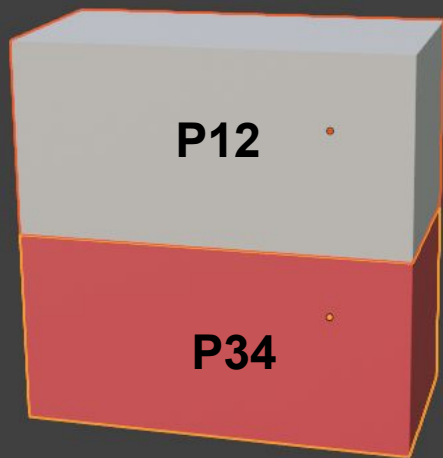


Graphe des fusions convexes :

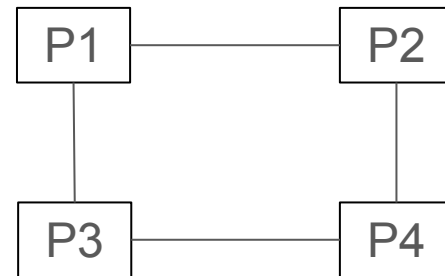


3 - Problème identifié dans l'algorithme de fusion actuel

Les $4! = 24$ combinaisons possibles de la liste de polyèdres donnent ces **2 résultats possibles** :



Graphe des fusions convexes :

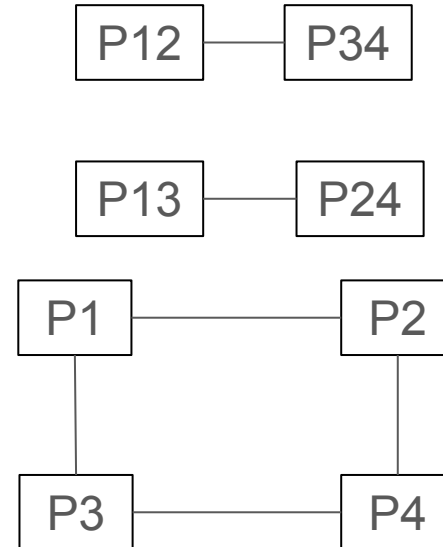


3 - Problème identifié dans l'algorithme de fusion actuel

Or on pourrait encore fusionner **P12 et P34** ou **P13 et P24**
pour avoir au final un cube convexe **P1234**



Graphe des fusions convexes :

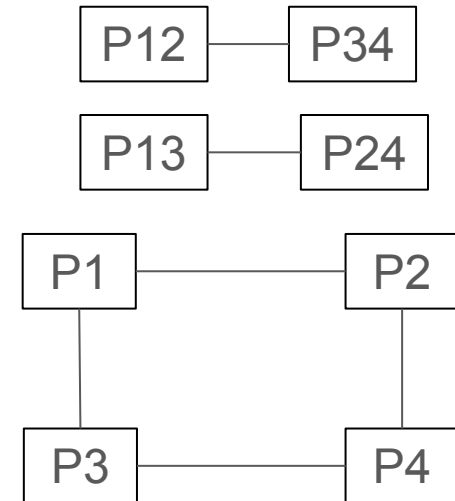


3 - Problème identifié dans l'algorithme de fusion actuel

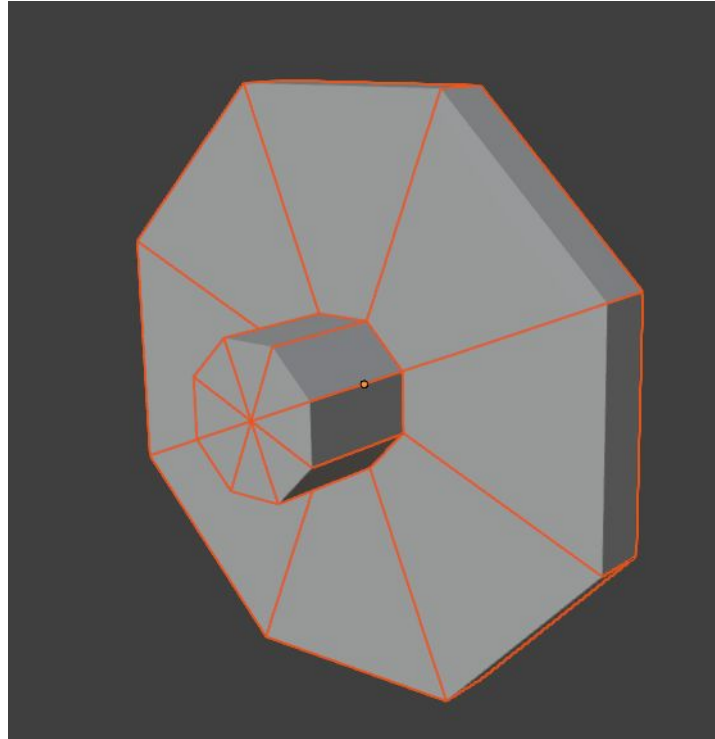
Actuellement, notre algorithme de fusion teste toutes les fusions pour chaque permutations possibles de la liste de polyèdres. **Cependant** pour trouver le meilleur résultat **cela ne suffit pas**, il faudrait repasser les solutions fusionnées dans l'algorithme pour voir si la taille se réduit. Evidemment cela augmenterait encore la complexité de l'algorithme.



Graphe des fusions convexes :



4 - Méta-heuristiques & cas de l'exemple de la “roue”



4 - Méta-heuristiques & cas de l'exemple de la “roue”

Le recuit simulé et l'algorithme génétique fonctionnent bien sur les exemples simples, donc nous avons évidemment passé la majorité du temps les tester/modifier sur l'exemple de la roue (32 polyèdres).
(Pour le recuit simulé, nous avons adapté en parti le code que vous nous avez fourni)

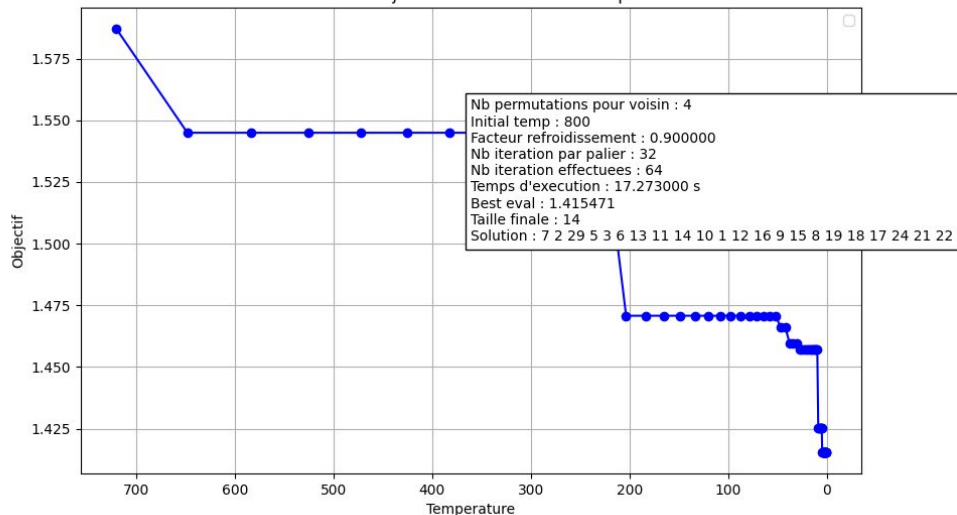
Nous passons pas mal de temps à essayer différents paramètres/méthodes pour améliorer ces algorithmes et changer leur comportement, mais il reste encore à faire

4 - Méta-heuristiques & cas de l'exemple de la "roue"

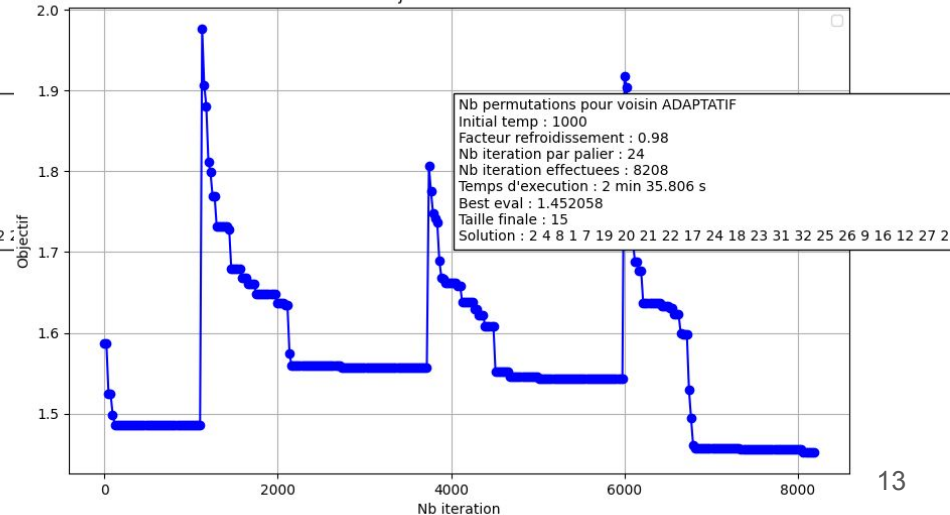
Nous avons aussi écrit du code pour générer des **graphiques** et suivre l'évolution de paramètres au cours de l'exécution des algo (les graphiques sont pour l'instant **plus pratiques qu'esthétique**, parfois même avec des petites erreurs dans les informations affichées). Cela nous permet de comprendre plus facilement l'impact des tests qu'on effectue sur le comportement de l'algo.

Exemples de 2 exécutions du recuit simulé avec des paramètres différents (sachant que tout n'est pas écrit sur les graphiques) :

Evolution de l'objectif en fonction de la température



Evolution de l'objectif en fonction des iterations

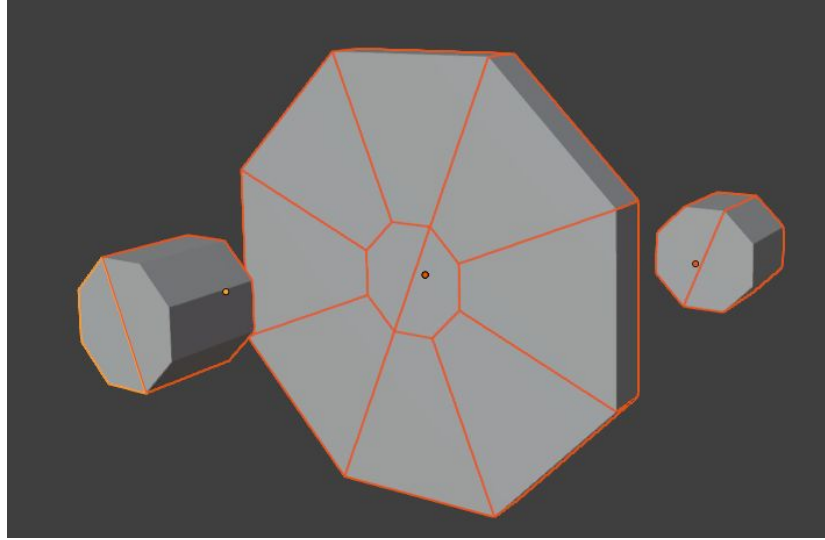


4 - Méta-heuristiques & cas de l'exemple de la "roue"

Actuellement, sur l'exemple de la roue, on arrive à générer une solution qui donne **14 polyèdres** (au lieu de 32) assez rapidement avec le recuit simulé. L'algorithme génétique s'en rapproche.

Cependant, le cas de la roue est similaire à celui des 4 cubes. C'est-à-dire qu'avec notre algorithme actuel il n'est pas possible de descendre en dessous de **14 polyèdres** (Ce qui est encourageant pour nos méta-heuristiques car on arrive à ce résultat en quelques minutes).

Exemple d'un résultat trouvé
avec 14 polyèdres :



Question par rapport au problème de fusion

Par rapport au problème de fusions décrit avec l'exemple des 4 cubes et celui de la roue, qu'en pensez-vous ?

Personnellement je compte modifier les méta-heuristiques pour qu'elles puissent refaire l'algo de fusion sur des solutions déjà fusionnées à **certaines conditions** (car tester tous les cas pour chaque solution revient à faire du brute-force sur un plus petit ensemble certes, mais qui peut quand même prendre un temps infini).

Encore faut-il trouver des conditions pertinentes afin à la fois de continuer à explorer au maximum l'espace des solutions tout en se rapprochant de solutions optimales ou presque.

Enfin une réunion, en présentiel si possible, ne serait pas de trop pour discuter ce problème.