

SUIVI DÉCOUVERTE DE LA RECHERCHE

THEO SAUGE - ALEXANDRE CORNU / ALEXIS DEMAREST - MARCO JORDA

Première séance :

(Copier-coller du mail envoyé le 22/10/2021)

Concernant mon groupe, il est composé de DEMAREST Alexis, SAUGE Théo, CORNU Alexandre ainsi que du chef de projet, Marco JORDA.

Enfin, pour notre premier compte rendu, nous avons discuté pour se départager les tâches et nous avons constitué des "sous-groupes" qui vont s'organiser de cette manière : Alexis et moi-même nous occupons de l'ordonnancement de tâches avec retard, puis Théo et Alexandre s'occupent du remplissage du rectangle avec des cercles.

Nous avons effectué lors de notre première séance des recherches (articles/algorithmes/vidéos explicatives/extraits) pour avoir une meilleure vision globale du problème et tâcher de ne pas oublier certains points. Nous avons ainsi fait un document commun qui se remplissait au fur et à mesure de nos recherches.

Nous avons ainsi pu trouver certaines idées bien concrètes afin d'avancer au mieux sur nos "sous"-projets.

Deuxième séance (04/11/2021) :

- Binôme DEMAREST / JORDA :

Lors de cette séance nous nous sommes occupés de bien comprendre les algorithmes que nous avons trouvé sur vis-à-vis de notre problématique (ici, l'ordonnancement des tâches) et de les appliquer pour pouvoir obtenir certains résultats. Nous n'avons cependant pas abouti à un bon résultat mais nous avons continué de chercher et avons tâché de corriger certains problèmes dans notre implémentation. Nous avons donc ainsi pu avancer sur l'application de ces derniers pour automatiser l'ordonnancement de tâches avec l'algorithme du « Shortest Processing Time » ainsi que celui du « Earliest Due Date ».

- Binôme SAUGE / CORNU :

Lors de cette séance nous nous sommes occupés d'améliorer l'algorithme que nous avons finalement trouvé lors de notre première séance en cherchant un peu plus en profondeur certains points qui se sont avérés plus compliqués que prévu. Cependant, nous avons pu obtenir quelques

pistes afin d'avancer et commencer une implémentation avec un algorithme variant de « wang et al's ». Saugé était cependant plus axé sur l'implémentation d'un algorithme pour l'algo 2 et Cornu faisait des recherches en parallèle pour voir comment résoudre la problématique.

Troisième séance (18/11/2021) :

- Binôme DEMAREST / JORDA :

Lors de cette séance, nous avons ensemble terminé d'implémenter ce qu'il restait des deux premiers algorithmes d'ordonnancement de tâches, nous avons obtenu deux résultats cohérents et étions satisfaits de nos deux algorithmes. Ils ont été implémentés en Python et nous avons ensuite attaqué l'ordonnancement de tâches avec la 3ème problématique "Minimiser le retard moyen sur l'ensemble des tâches". Pour ce faire nous avons consacré le reste de notre séance (un peu moins de la moitié) à faire des recherches à ce sujet là pour trouver un algorithme. Cependant, à part un lien qui pourrait nous être utile pour mieux comprendre la problématique, nous n'avons rien trouvé de nouveau.

- Binôme SAUGE / CORNU :

Lors de cette séance nous avons essayé de continuer la recherche sur un nouveau point d'amélioration de l'algorithme de base trouvé lors de la première séance. Cornu s'est donc principalement occupé de cette recherche, et à réussi à trouver un point qui pourrait potentiellement améliorer l'algorithme de base. Saugé s'est lui préoccupé de créer le programme suivant l'algorithme de base afin de pouvoir le comparer au programme suivant le nouvel algorithme.

Quatrième séance (02/12/2021) :

- Binôme DEMAREST / JORDA :

Cette séance nous a permis de continuer nos recherches sur la problématique de minimiser le retard moyen/maximal. Cependant, après avoir passé un bout de temps à chercher un algorithme ou des ressources permettant de mieux comprendre le problème pour au final ne rien trouver d'utile. Nous avons décidé de faire quelques schémas sur papier pour pouvoir faire l'algorithme nous mêmes et l'implémenter par la suite, nous pensons donc pouvoir commencer une implémentation à la séance prochaine et prévoir une finalisation des deux problématiques à celle d'après.

- Binôme SAUGE / CORNU :

Lors de cette séance, nous avons travaillé ensemble sur la résolution du problème de l'amélioration de l'algorithme : avec les recherches d'Alexandre et l'élaboration d'un nouveau programme faite par Théo. Cela a permis de créer une amélioration de l'algorithme mais pour l'instant de nombreux essais sont encore à faire pour être sûr de l'efficacité de ce nouveau programme. Vers la fin de la séance, nous avons cherché une autre solution et même d'autres idées pour parfaire notre programme. L'inconvénient étant toujours de réussir à optimiser le placement de nos cercles.

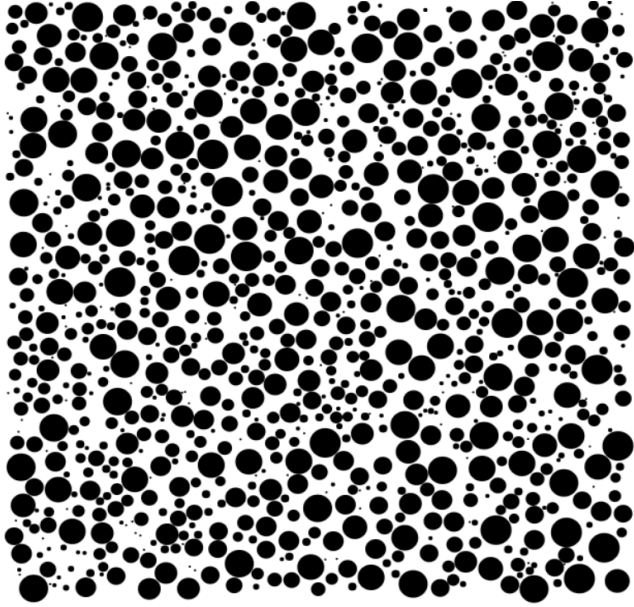
Cinquième séance (06/01/2022) :

- Lors de cette séance, nous avons décidé de nous rejoindre pour faire une mise au point de là où nous en étions en tant que binômes.

En faisant un point, nous nous sommes rendus compte que le binôme Jorda / Demarest était un peu plus avancé au niveau des algorithmes d'ordonnancement de tâches que celui du circle packing. Nous avons donc pris la décision de nous concentrer sur le circle packing en tant que groupe.

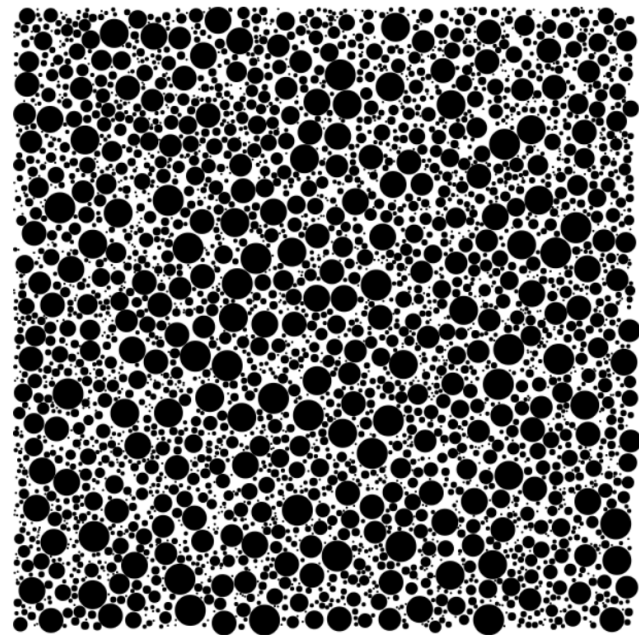
Saugé et Cornu possédaient déjà un algorithme relativement fonctionnel au départ. Avec le temps certains bugs ont pu être corrigés, et le placement des cercles est devenu correct (on a décidé de partir sur un placement au hasard plutôt qu'un placement explicite). Le placement au hasard était implémenté dans le but de laisser le programme faire des tentatives de placements (de cercles aux rayons au hasard aussi) dans un canvas et de voir combien le programme pouvait en placer. Bien entendu nous avons eu quelques problèmes de typage, d'algorithme et de boucles qui nous ont malheureusement fait perdre du temps comme dans chaque programme. Cependant nous avons réussi à aboutir lors de cette séance + une heure passée à l'extérieur de l'UPJV à quelque chose de fonctionnel sans aucun problème de superposition de cercles ni de cercles sortant du cadre (canvas).

Le week end d'après la séance, le binôme Demarest et Jorda ont pu rajouter une fonctionnalité dans l'algorithme afin d'avoir un point de vue plus global sur le placement de cercles : du stockage des données. Nous voulions stocker sur une dizaine d'échantillons combien de cercles étaient placés et en combien de temps. Nous avons donc abouti à similaire à ceci :



```
Elements Console Sources Network
top Filter Default levels
2 Issues: 2
In this attempt : 1037 circles were drawn. fillcircles.js:39
default: 17395.453857421875 ms fillcircles.js:87
In this attempt : 1064 circles were drawn. fillcircles.js:39
default: 17766.06982421875 ms fillcircles.js:87
In this attempt : 920 circles were drawn. fillcircles.js:39
default: 15350.459228515625 ms fillcircles.js:87
In this attempt : 1027 circles were drawn. fillcircles.js:39
default: 17134.096923828125 ms fillcircles.js:87
In this attempt : 1085 circles were drawn. fillcircles.js:39
default: 18100.921875 ms fillcircles.js:87
In this attempt : 1293 circles were drawn. fillcircles.js:39
default: 21567.740234375 ms fillcircles.js:87
In this attempt : 1004 circles were drawn. fillcircles.js:39
default: 16750.47021484375 ms fillcircles.js:87
In this attempt : 940 circles were drawn. fillcircles.js:39
default: 15683.860107421875 ms fillcircles.js:87
In this attempt : 966 circles were drawn. fillcircles.js:39
default: 16116.5791015625 ms fillcircles.js:87
In this attempt : 995 circles were drawn. fillcircles.js:39
(10) [1037, 1064, 920, 1027, 1085, 1293, 1004, 940, 966, 995]
```

Il faut cependant noter que l'algorithme s'arrête lorsque le rayon max possible est à 5. L'algorithme effectue en effet 1000 tentatives de placement d'un cercle au hasard sur le canvas. Et si les 1000 tentatives échouent toutes, le rayon maximum baisse, pour gagner plus de probabilités de trouver un placement à succès. Cependant, sur la capture d'écran du dessus, l'algorithme s'arrête effectivement si le rayon maximum à atteint 5 pour pouvoir avoir un point de vue rapide sur la complétion du canvas. Si l'on met le rayon max à 1, le temps de la procédure devient beaucoup plus long, mais la complétion du canvas devient forcément plus précise et concrète (voir image ci-dessous).



```
Elements Console Sources Network
top Filter Default levels
2 Issues: 2
In this attempt : 3830 circles were drawn. fillcircles.js:39
default: 87771.376953125 ms fillcircles.js:87
```