



Master MIND/BIOSTAT

Représentation graphique

Multiplication Modulaire

Sobolak Valérian, Goujili Nouhaila, Sene Assane, Berrandou Assia

Université de Montpellier

April 25, 2021

1 Introduction

- Description mathématiques
- Outils graphiques

2 Objectifs

3 Représentation du graphe

- Représentation fixe
- Représentation interactive
- Représentation en continue
- Classe Graph
- Classe Interface_gestion

4 Autres Aspects de l'interface

- Bouton générateur de Gif
- Autres boutons

5 Conclusion

- Problèmes rencontrés
- Généralités
- Perspectives d'amélioration

Introduction

Description mathématiques

Ce projet :

- Fait appel à l'arithmétique modulaire.
 - Soit $p \in \mathbb{N}$ le modulo i.e le reste de la division qui permet de gérer le caractère périodique du calcul.
 - Soit $n \in \mathbb{R}$ la table de multiplication que l'on veut représenter modulo p .
 - On veut représenter graphiquement le resultat du calcul :

$$\forall i \in \{0, \dots, p-1\} \quad n \times i \bmod p.$$

Introduction

Description mathématiques

- Fait appel à la notion de graphe
 - Le graphe fini $G=(V,E)$ est défini par :
 - L'ensemble fini $V = \{0, 1, \dots, p-1\}$ dont les éléments sont appelés sommets.
 - L'ensemble fini $E = \{e_1, e_2, \dots, e_n\}$ dont les éléments sont appelés arêtes tel que

$$\forall i \in \{0, \dots, p-1\} \quad e_i = \{i, j\} \text{ avec } j = n * i \bmod p.$$

- Une matrice d'adjacence symétrique M de dimension $(p \times p)$ dont les lignes et les colonnes représentent les sommets du graphe. Si deux sommets i et j sont adjacents alors $m_{ij} = m_{ji} = 1$.
Remarque : Dans le projet, M n'est pas symétrique.

Le package tkinter :

- Module de référence pour les réalisations d'applications avec une interface graphique.
- Tk : Objet de type Tk crée une fenêtre.
- Canvas : Widget dans lequel on peut dessiner et manipuler des figures géométriques, du texte ou bien encore des images plus ou moins complexes. On appelle ces dernières des items.
- Autres Widgets :
 - Scrollbar : Barre de déroulement
 - Button : Bouton
 - Scale : Slider
 - ...

- Créer un package représentant ces tables de multiplications modulaires de manière interactive.
- Pouvoir les mettre en mouvement.
- Générer un gif à partir des images que l'on a capturées au préalable.
- Large choix de personnalisation graphique.

Représentation du graphe

Représentation fixe(base_vis.py)

- Tous les sommets sont placés sur un cercle de centre C et de rayon R.
- Chaque sommet est espacé de manière proportionnelle d'un angle

$$\theta = \frac{2\pi}{p}$$

- Pour tout sommet $k \in \{0, \dots, p-1\}$, le couple de coordonnées est donné par la formule trigonométrique

$$(R \times \cos(k \times \theta), R \times \sin(k \times \theta))$$

- .
- On associe à chacun des sommets, dans le sens horaire, un nombre compris entre 0 et $p-1$.

Représentation du graphe

Représentation fixe(edge_vis.py)

- Pour un i fixé, on ajoute l'arête caractérisée par un segment entre les sommets i et $j = \text{modulo_result}(i)$.
- On utilise la fonction `Canvas.create_line((xA,yA),(xB,yB))` où (xA,yA) et (xB,yB) sont les couples de coordonnées de i et j respectivement.
- On réitère cela $\forall i \in \{0, \dots, p-1\}$ à l'aide d'une boucle for afin d'afficher tous les items dans le Canvas.

Représentation du graphe

Représentation interactive

- Initialiser la fenêtre et le Canvas à travers laquelle l'utilisateur peut interagir.
- On instancie un objet de type Graph et on définit les variables liées à la représentation circulaire (centre et rayon).
- On représente, pour un n et p fixé, le graphe associé à l'aide des fonctions de la représentation fixe.

Représentation du graphe

Représentation interactive

- On génère les curseurs qui font varier les valeurs n et p entre 2 et 400 et entre 2 et 200 respectivement. Pour chaque n et p sélectionnés par l'utilisateur :
 - On supprime tous les items présents dans le Canvas.
 - On en crée de nouveaux pour le ou les nouveaux paramètres sélectionnés.

Représentation du graphe

Représentation en continue

Le bouton **Play/Pause** permet d'augmenter en continu n par pas de 0.01. Pour ce faire, tant que l'on ne reclique pas sur le bouton :

- On utilise les structures précédemment décrites (Représentation fixe et interactive) i.e on fait appelle au curseur associé à n et on l'augmente par pas de 0.01 automatiquement.
- On définit un délai, en millisecondes, à l'aide de la fonction *Tk.after(m)*. Cette dernière met en veille le programme pendant une durée m avant de passer à la tâche suivante.
- On met à jour la fenêtre.

Représentation du graphe

Classe Graph

La classe Graphe :

- Prend en arguments les paramètres de la multiplication modulaire. De ce fait, on l'appelle dans les autres scripts régulièrement afin de récupérer ces paramètres plus facilement.
- Utilise une matrice creuse/sparse pour représenter la matrice d'adjacence M .
- Affiche, de manière textuelle, le graphe dans la console.

Représentation du graphe

Classe Interface_gestion

La classe Interface_gestion :

- Contient un grand nombre d'arguments liés au design des représentations.
- Fait le lien entre la représentation interactive et la représentation en continue.
- Elle génère aussi d'autres aspects de l'interface graphique.

Autres aspects de l'interface

Bouton générateur de Gif

- A l'aide du bouton **Photo**, on capture et stock une image (.png) du Canvas courant.
- Si plus d'une image est enregistrée, on peut générer, à l'aide du bouton **Vidéo**, un gif assemblant ces dernières dans l'ordre de capture.
- Ce procédé peut être réitéré plusieurs fois et on retrouve les gifs dans le dossier **"/gif "** du projet.

Autres aspects de l'interface

Autres boutons

- Le bouton **Table** permet l'ouverture d'une nouvelle fenêtre contenant l'ensemble des résultats courant de la table n modulo p .
- Le bouton **Quit** supprime l'ensemble des images capturées à l'aide du bouton **Photo**, détruit l'ensemble des widgets et quitte la fenêtre.
- Le bouton **Description** offre une description rapide des différentes fonctionnalités de l'interface.

Conclusion

Problèmes rencontrés

- Problème dans l'intégration continue causé par le package tkinter (workflow) **(Non résolu)**.
- Problème avec la doc du package **(Résolu)**.
- Problème lorsque l'on crée un jupyter notebook online : le package tkinter provoque un problème avec l'environnement des variables **(Non résolu)**.
- Problèmes de connexion lors des réunions

Conclusion

Généralités

- Respect des attentes suggérées dans le sujet.
- Respect des délais.
- Bonne entente au sein du groupe.

Conclusion

Perspectives d'amélioration

- Inclusion de paramètres par défaut si l'utilisateur ne le précise pas.
- Plus de personnalisation dans l'interface (Choix des couleurs - Activer/désactiver les décimaux/entiers - Animation auto du modulo).
- Possibilité de rentrer directement la table et/ou le modulo souhaité dans l'interface (au lieu de passer par les curseurs).
- Pouvoir supprimer les images capturées pour le gif sans avoir besoin de quitter l'interface par le bouton **Quit**.
- Alléger la classe Interface_gestion.