



FACULTÉ DES SCIENCES

HAI902I AIDE À LA DÉCISION

Conception et implantation d'un système d'aide à la décision

Participants :

Awa SECK
Mitra AELAMI
Amirhossein NASRI
Adrien LINARES
Abdelhalim AZZOUZ

Encadrante :

Pr. Souhila KACI

28 décembre 2023

Table des matières

1	Introduction	2
2	Notre système pas à pas	3
2.1	Génération des préférences aléatoires	3
2.2	Implantation du mariage stable	3
2.3	Interface	6
2.4	Mesure de satisfaction	7
2.5	Tests sur plusieurs jeux de données	9
2.6	Intégration compacte des préférences	10
2.6.1	Préférences des universités par rapport aux étudiants	10
2.6.2	Préférences des étudiants par rapport aux universités	10
2.6.3	Intégration des scores :	10
3	Conclusion et perspectives	12

1 Introduction

L'objectif de ce projet est d'implanter un système d'aide à la décision, en l'occurrence celui de l'algorithme du mariage stable, cela dans le but d'en faire une analyse critique.

Le problème du mariage stable consiste à trouver un appariement dit "stable" entre deux ensembles de tailles égales et selon un ordre de préférence pour chaque élément. Un appariement est une bijection des éléments d'un ensemble vers l'autre. Nos deux ensembles dans le cadre de ce rapport seront les suivants :

$$U = \{\text{l'ensemble des établissements}\}$$

$$E = \{\text{l'ensemble des étudiants}\}$$

Nous avons été 5 étudiants à travailler sur ce projet, qui portait sur la dernière partie de notre cours, nous avons organisé plusieurs réunions de travail afin de mener à bien la réalisation de ce système. Dans la suite de ce rapport nous avons organisé nos parties séquentiellement selon l'ordre des questions qui nous ont été posées, passant d'un système assez primitif à quelque chose de plus complexe et abouti.

Nous présenterons donc notre système, étape par étape, fruit d'une collaboration étroite ayant nécessité avant tout communication et organisation.

2 Notre système pas à pas

2.1 Génération des préférences aléatoires

Tout d'abord, nous avons commencé par fixer un exemple de nombre d'étudiants (E) et d'établissements (U), dans notre cas : $|E| = |U| = 5$. Nous avons ensuite généré des listes de noms pour les étudiants et les établissements en itérant sur ces derniers. Après quoi nous avons généré des listes de préférences aléatoires pour chaque étudiant et chaque établissement avec la fonction built-in `sample()` du package `random`, pour finalement terminer par afficher les préférences générées.

```
Préférences des étudiants :
E1: ['U2', 'U5', 'U3', 'U4', 'U1']
E2: ['U1', 'U2', 'U4', 'U3', 'U5']
E3: ['U5', 'U4', 'U3', 'U1', 'U2']
E4: ['U2', 'U3', 'U4', 'U1', 'U5']
E5: ['U4', 'U5', 'U1', 'U3', 'U2']

Préférences des établissements :
U1: ['E5', 'E3', 'E1', 'E4', 'E2']
U2: ['E4', 'E3', 'E5', 'E1', 'E2']
U3: ['E4', 'E3', 'E1', 'E2', 'E5']
U4: ['E3', 'E1', 'E5', 'E2', 'E4']
U5: ['E1', 'E2', 'E5', 'E4', 'E3']
```

FIGURE 1 – Préférences aléatoires générées

2.2 Implantation du mariage stable

Nous allons décrire brièvement l'algorithme du mariage stable tel que nous l'avons implémenté. Tout d'abord, nous avons utilisé les listes de préférences des étudiants et établissements déjà générées : `preferences_etudiants` et `preferences_etablissement`.

Nous avons par la suite créé deux listes :

- `etudiants_libres` : liste de numéros (index) des préférences des étudiants,
- `mariage` : contenant le résultat de l'algorithme.

Tant que la liste `etudiants_libres` n'est pas vide, on prend son premier élément et on regarde sa liste de préférence `preferences_etudiants` :

- Si le premier établissement choisi est vide, alors l'établissement n'a pas encore de candidat, donc un appariement stable est trouvé, et on fait l'affectation.
- Sinon, on vérifie si l'établissement préfère l'étudiant actuel ou le nouvel étudiant en comparant leur position par rapport aux préférences de l'établissement. Si le nouveau est préféré, on met à jour l'appariement.

On fini par retourner les couples stables.

Nous avons fourni ci-dessous (Figure 2) un exemple d'exécution de notre algorithme avec les paramètres précédemment évoqués :

```
Préférences des étudiants :  
E1: ['U2', 'U5', 'U3', 'U4', 'U1']  
E2: ['U1', 'U2', 'U4', 'U3', 'U5']  
E3: ['U5', 'U4', 'U3', 'U1', 'U2']  
E4: ['U2', 'U3', 'U4', 'U1', 'U5']  
E5: ['U4', 'U5', 'U1', 'U3', 'U2']  
  
Préférences des établissements :  
U1: ['E5', 'E3', 'E1', 'E4', 'E2']  
U2: ['E4', 'E3', 'E5', 'E1', 'E2']  
U3: ['E4', 'E3', 'E1', 'E2', 'E5']  
U4: ['E3', 'E1', 'E5', 'E2', 'E4']  
U5: ['E1', 'E2', 'E5', 'E4', 'E3']  
  
Résultat du mariage stable :  
E4 est associé à U2  
E5 est associé à U1  
E1 est associé à U5  
E3 est associé à U4  
E2 est associé à U3
```

FIGURE 2 – Couples stables calculés

Nous avons aussi déroulé à la main notre algorithme avec les mêmes paramètres pour vérifier si nous avons le même résultat, dont nous avons fait la représentation ci-dessous (Figure 3).

Préférences des élèves	Priorités	Capacité des écoles		Préférences des élèves	Priorités	Capacité des écoles	
E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	1	E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	5
E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1		E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1	
E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1		E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1	
E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1		E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1	
E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1		E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1	
Préférences des élèves	Priorités	Capacité des écoles		Préférences des élèves	Priorités	Capacité des écoles	
E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	2	E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	6
E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1		E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1	
E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1		E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1	
E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1		E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1	
E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1		E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1	
Préférences des élèves	Priorités	Capacité des écoles		Préférences des élèves	Priorités	Capacité des écoles	
E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	3	E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	7
E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1		E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1	
E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1		E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1	
E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1		E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1	
E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1		E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1	
Préférences des élèves	Priorités	Capacité des écoles		Préférences des élèves	Priorités	Capacité des écoles	
E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	4	E1:(U2,U5,U3,U4,U1)	U1:(E5,E3,E1,E4,E2)	Q1=1	8
E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1		E2:(U1,U2,U4,U3,U5)	U2:(E4,E3,E5,E1,E2)	Q2=1	
E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1		E3:(U5,U4,U3,U1,U2)	U3:(E4,E3,E1,E2,E5)	Q3=1	
E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1		E4:(U2,U3,U4,U1,U5)	U4:(E3,E1,E5,E2,E4)	Q4=1	
E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1		E5:(U4,U5,U1,U3,U2)	U5:(E1,E2,E5,E4,E3)	Q5=1	

Affectation finale
(E1,U5)
(E2,U3)
(E3,U4)
(E4,U2)
(E5,U1)

FIGURE 3 – Déroulement de l'algorithme à la main

2.3 Interface

Afin d'avoir un système d'aide à la décision plus interactif et intuitif, nous proposons une interface graphique (GUI), guidant l'utilisateur dans l'utilisation de celui-ci. Il est composé plusieurs fenêtres permettant une visualisation plus claire et précise.

Nous avons développé cette application graphique en utilisant la bibliothèque « Tkinter » de Python. Cette interface offre aux utilisateurs la possibilité de générer des préférences aléatoires pour un ensemble d'étudiants et d'établissements, de lancer l'algorithme pour trouver les couples stables, de calculer les scores de satisfaction et aussi visualiser le score de satisfaction.

Il y a, en résumé, 5 fenêtres différentes, parmi lesquelles :

- Le prompt utilisateur (Figure 4)
- La génération des préférences (dont l'appariement des éléments) (Figure 5)
- Les scores de satisfaction brut
- L'histogramme de satisfaction
- Le diagramme à bâtons de satisfaction

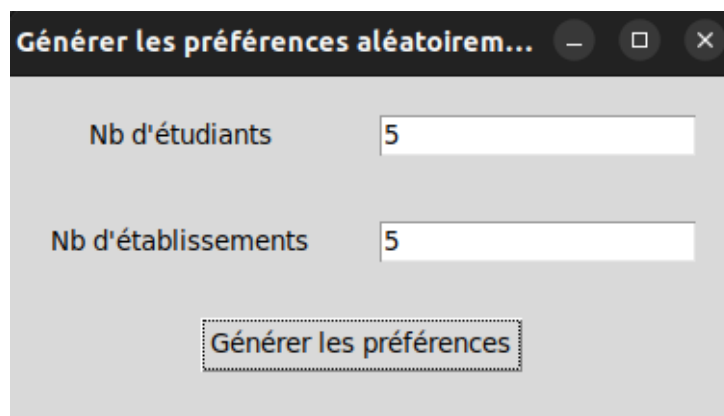


FIGURE 4 – Prompt utilisateur

Dont le résultat est affiché dans une nouvelle fenêtre :

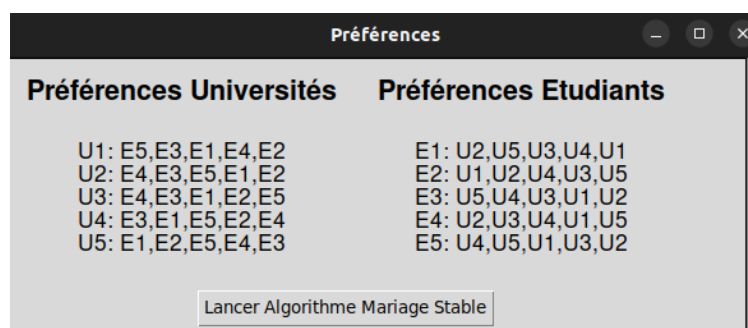


FIGURE 5 – Préférences aléatoires générées

2.4 Mesure de satisfaction

Nous allons à présent définir ce qu'est le score de satisfaction dans le cadre de notre système et donner sa formule par rapport aux étudiants et établissements.

Le score de satisfaction est une valeur comprise dans un intervalle de $]0,1]$; O n'est pas compris car on a toujours une affectation étudiant-établissement.

Il calcule le degré de satisfaction d'un étudiant ou établissement du choix qu'il a eu par rapport à son 1er choix. Il est défini comme suit :

$$\text{score_satisfaction} = \left(\frac{N - i}{N} \right) \times 100$$

- **N** : taille des préférences (étudiants ou établissement).
- **i** : position de l'étudiant (respectivement établissement) dans les choix de l'établissement(respectivement étudiant).

NB : La position dans chaque liste commence par 0 (la position du 1er choix)

On calcule les scores de satisfaction pour chaque étudiant et chaque établissement ci-dessous (Figure 6) :

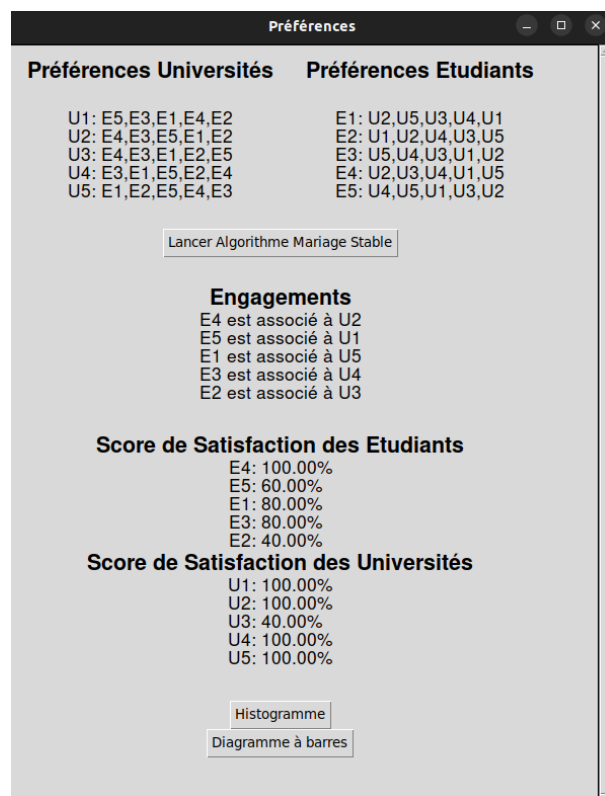


FIGURE 6 – Scores de satisfaction

Visualisation des Scores : L'application propose deux options de visualisation pour examiner les scores de satisfaction :

- **Histogramme :** Cette représentation graphique montre le score de satisfaction en fonction du nombre d'étudiants ou d'établissements. Elle permet de visualiser la distribution des scores et d'identifier rapidement la concentration des niveaux de satisfaction parmi l'ensemble des participants.
- **Diagramme à Barres :** Cette visualisation associe à chaque étudiant ou établissement une barre correspondante qui illustre son score de satisfaction. Ce type de diagramme permet de comparer facilement les scores individuels.

Ces deux méthodes de visualisation offrent des perspectives différentes mais complémentaires sur les résultats. Vous les trouverez ci-après.

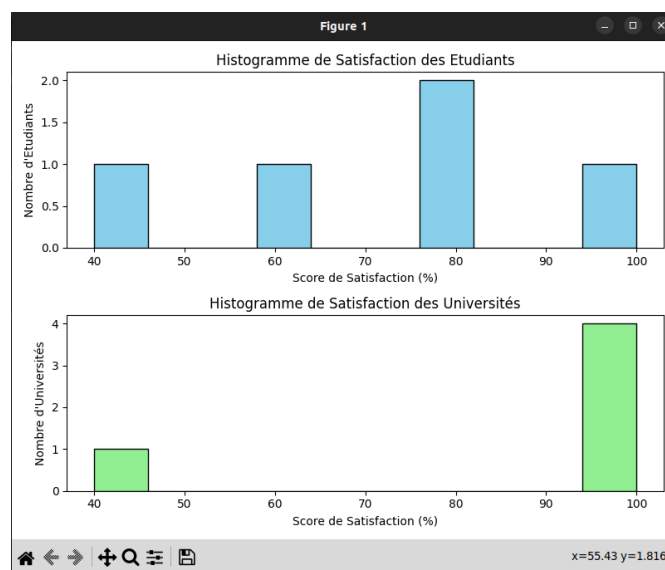


FIGURE 7 – Histogramme de satisfaction

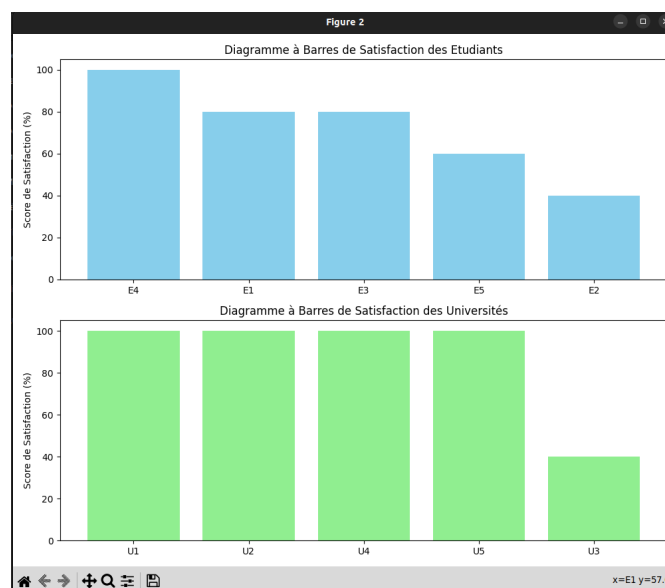


FIGURE 8 – Diagramme à bâtons

2.5 Tests sur plusieurs jeux de données

Nous proposons ici la trace d'exécution sur 2 jeux de données différents (ici, seuls $|I|$ et $|S|$ changent, ainsi que les préférences, donc les engagements). Dans le premier jeu de données nous avons fixé $|I| = |S| = 10$. Puis dans le second $|I| = |S| = 8$. Nous choisissons volontairement de ne montrer que la fenêtre des préférences et celle des scores de satisfaction, nous semblant les plus pertinentes dans le cadre de ce travail.

Préférences	
Préférences Universités	Préférences Etudiants
S1: I6,I1,I4,I5,I7,I2,I10,I8,I3,I9	I1: S4,S3,S5,S9,S10,S6,S2,S7,S8,S1
S2: I7,I1,I9,I2,I3,I8,I10,I6,I4,I5	I2: S2,S4,S10,S3,S8,S1,S6,S5,S9,S7
S3: I2,I7,I5,I9,I10,I1,I6,I4,I8,I3	I3: S1,S9,S6,S4,S7,S2,S3,S10,S5,S8
S4: I1,I3,I8,I4,I10,I5,I7,I6,I2,I9	I4: S5,S10,S8,S9,S6,S4,S1,S3,S2,S7
S5: I6,I5,I2,I3,I10,I8,I1,I9,I4,I7	I5: S5,S8,S1,S4,S10,S2,S7,S6,S3,S9
S6: I4,I9,I6,I3,I10,I5,I7,I8,I1,I2	I6: S1,S10,S3,S4,S5,S7,S8,S9,S2,S6
S7: I2,I5,I9,I1,I3,I7,I10,I6,I8,I4	I7: S9,S2,S6,S1,S7,S4,S3,S8,S5,S10
S8: I10,I2,I5,I1,I8,I6,I4,I3,I7,I9	I8: S5,S4,S9,S1,S6,S3,S7,S10,S2,S8
S9: I5,I10,I6,I1,I8,I3,I2,I4,I7,I9	I9: S4,S6,S2,S8,S3,S7,S9,S5,S1,S10
S10: I4,I10,I2,I8,I5,I6,I9,I7,I3,I1	I10: S5,S2,S3,S6,S8,S4,S9,S1,S7,S10

Lancer Algorithme Mariage Stable

Engagements

I1 est associé à S4
I7 est associé à S2
I6 est associé à S1
I5 est associé à S5
I8 est associé à S9
I9 est associé à S6
I2 est associé à S3
I4 est associé à S10
I3 est associé à S7
I10 est associé à S8

(a) Couples stables

Score de Satisfaction des Etudiants	
I1: 100.00%	
I7: 90.00%	
I6: 100.00%	
I5: 100.00%	
I8: 80.00%	
I9: 90.00%	
I2: 70.00%	
I4: 90.00%	
I3: 60.00%	
I10: 60.00%	
Score de Satisfaction des Universités	
S1: 100.00%	
S2: 100.00%	
S3: 100.00%	
S4: 100.00%	
S5: 90.00%	
S6: 90.00%	
S7: 60.00%	
S8: 100.00%	
S9: 60.00%	
S10: 100.00%	

(b) Scores de satisfaction

FIGURE 9 – Jeux de données 1

Préférences	
Préférences Universités	Préférences Etudiants
S1: I1,I8,I3,I6,I5,I7,I2,I4	I1: S7,S5,S1,S2,S4,S8,S6,S3
S2: I3,I8,I1,I7,I5,I6,I2,I4	I2: S5,S1,S3,S4,S8,S7,S2,S6
S3: I5,I6,I1,I7,I4,I3,I2,I8	I3: S2,S3,S8,S5,S6,S7,S1,S4
S4: I2,I8,I7,I4,I5,I6,I1,I3	I4: S7,S3,S8,S2,S6,S4,S1,S5
S5: I7,I6,I3,I1,I8,I5,I4,I2	I5: S6,S1,S3,S8,S4,S7,S2,S5
S6: I8,I2,I3,I5,I1,I6,I7,I4	I6: S1,S2,S5,S6,S8,S4,S7,S3
S7: I5,I3,I6,I1,I8,I4,I7,I2	I7: S4,S3,S7,S1,S8,S5,S6,S2
S8: I1,I2,I3,I8,I4,I6,I5,I7	I8: S6,S4,S8,S7,S1,S3,S2,S5

Lancer Algorithme Mariage Stable

Engagements

I1 est associé à S7
I2 est associé à S5
I3 est associé à S2
I5 est associé à S3
I8 est associé à S6
I6 est associé à S1
I7 est associé à S4
I4 est associé à S8

(a) Couples stables

Score de Satisfaction des Etudiants	
I1: 100.00%	
I2: 100.00%	
I3: 100.00%	
I5: 75.00%	
I8: 100.00%	
I6: 100.00%	
I7: 100.00%	
I4: 75.00%	
Score de Satisfaction des Universités	
S1: 62.50%	
S2: 100.00%	
S3: 100.00%	
S4: 75.00%	
S5: 12.50%	
S6: 100.00%	
S7: 62.50%	
S8: 50.00%	

(b) Scores de satisfaction

FIGURE 10 – Jeux de données 2

2.6 Intégration compacte des préférences

Une manière d'effectuer une intégration compacte des préférences entre les universités et les étudiants est d'utiliser des scores pondérés. Nous pouvons attribuer des scores en fonction de la position de chaque étudiant dans les listes de préférences des universités et vice versa. Plus précisément, nous pouvons suivre les étapes suivantes :

2.6.1 Préférences des universités par rapport aux étudiants

1. Attribution des scores (positions) : Supposons que nous attribuons des scores décroissants en fonction de la position de l'étudiant E_j dans la liste des préférences de l'université U_i .

Maintenant, nous allons attribuer des scores de normalisation (scores initiaux) pour chaque étudiant E_j dans la liste des préférences de l'université U_i en utilisant la formule suivante : $N - \text{position} + 1$, où N est le nombre total des étudiants dans la liste des préférences de l'université et position est l'indice de l'étudiant E_j dans la liste de l'université U_i , sachant que les indices débutent par 1.

2. Matrice des scores normalisée (M_{UE}) : La normalisation des scores pour chaque université peut être faite en divisant chaque score par la somme de tous les scores attribués à cette université.

Pour normaliser les scores de chaque université. Tout d'abord, on doit calculer la somme des scores pour chaque université.

Puis, on crée la matrice M_{UE} telle que M_{UEij} représente le score normalisé de l'étudiant E_j selon les préférences de l'université U_i .

2.6.2 Préférences des étudiants par rapport aux universités

Pour constituer la matrice des scores normalisés M_{EU} pour les étudiants en fonction de leurs listes des préférences pour les universités, on va suivre les mêmes étapes que pour les préférences des universités par rapport aux étudiants.

2.6.3 Intégration des scores :

Intégrons les deux matrices de scores normalisés (M_{UE} et M_{EU}). Pour simplifier, utilisons une moyenne pondérée égale.

Intégration des scores (moyenne pondérée) : Soit SI = Score Intégré.

$$SI_{ij} = \frac{M_{UEij} - M_{EUij}}{2}$$

De cette manière, on aurait défini une matrice des scores pondérés.

Position de chaque étudiant dans la liste des préférences des universités												
	E1	E2	E3	E4	E5		U1	U2	U3	U4	U5	
U1	3	5	2	4	1		E1	5	1	3	4	2
U2	4	5	2	1	3		E2	1	2	4	3	5
U3	3	4	2	1	5		E3	4	5	3	2	1
U4	2	4	1	5	3		E4	4	1	2	3	5
U5	1	2	5	4	3		E5	3	5	4	1	2
Les scores initiaux												
	E1	E2	E3	E4	E5		U1	U2	U3	U4	U5	
U1	3	1	4	2	5		E1	1	5	3	2	4
U2	2	1	4	5	3		E2	5	4	2	3	1
U3	3	2	4	5	1		E3	2	1	3	4	5
U4	4	2	5	1	3		E4	2	5	4	3	1
U5	5	4	1	2	3		E5	3	1	2	5	4
Matrice des scores normalisés												
	E1	E2	E3	E4	E5		U1	U2	U3	U4	U5	
U1	3/15	1/15	4/15	2/15	5/15		E1	1/15	5/15	3/15	2/15	4/15
U2	2/15	1/15	4/15	5/15	3/15		E2	5/15	4/15	2/15	3/15	1/15
U3	3/15	2/15	4/15	5/15	1/15		E3	2/15	1/15	3/15	4/15	5/15
U4	4/15	2/15	5/15	1/15	3/15		E4	2/15	5/15	4/15	3/15	1/15
U5	5/15	4/15	1/15	2/15	3/15		E5	3/15	1/15	2/15	5/15	4/15
Score intégrés												
	E1	E2	E3	E4	E5							
U1	2/15	3/15	3.5/15	2/15	4.5/15							
U2	3.5/15	2.5/15	3/15	4/15	2/15							
U3	2.5/15	1.5/15	3.5/15	4.5/15	3/15							
U4	3/15	3.5/15	4.5/15	2/15	2/15							
U5	4/15	2.5/15	1.5/15	3.5/15	3.5/15							

FIGURE 11 – Étapes d'exécution de l'algorithme du score intégré

Cette matrice représente les scores intégrés attribués par chaque université à chaque étudiant. Les valeurs dans la matrice indiquent le niveau d'adéquation entre chaque université et chaque étudiant.

L'interprétation des résultats de la matrice du score intégré peut être comme suit : Les valeurs élevées dans la matrice indiquent une forte préférence d'une université pour un étudiant et vice versa. Cela signifie que l'université est très satisfaite de l'étudiant ou que l'étudiant préfère fortement cette université. On peut dérouler cet algorithme sur l'exemple initial de la manière suivante :

On voit bien que $U1$ va recevoir $E5$, $U2$ va recevoir $E4$, $U3$ va recevoir $E4$, $U4$ va recevoir $E3$, $U5$ va recevoir $E1$. Ici, on remarque qu'il ya un conflit autour de $E4$, donc on regarde les score et on remarquera que le score de $U3(E4) > U2(E4)$ donc $E4$ serait affecté à $U3$. Alors on regardera le deuxième score le plus élevé pour $U2$, on voit que c'est celui de $U2(E1)$, mais $E1$ est affecté à $U5$ avec un score plus élevé. De la même manière, on essaiera avec son troisième score plus élevé qui est $U2(E3)$, mais encore une fois $E3$ est affecté avec un score plus élevé à $U4$. On essaie avec son quatrième score le plus élevé qui est $U2(E2)$, et comme $E2$ n'a pas encore été affecté, il sera alors affecté à $U2$.

3 Conclusion et perspectives

Travailler sur ce projet nous a permis de nous extraire de l'aspect très théorique dont ce domaine peut nous faire penser (à tort). Cela nous a montré que l'on peut appliquer les notions vues en cours sur de réels cas concrets, que ce soit sur des exemples se rapprochant de ceux connus de ParcoursSup et APB, pour ce projet, voire dans de nombreux autres cas industriels, évoqués succinctement en classe.

Comme dans d'autres projets nous avons pu perfectionner notre travail d'équipe, tant dans le niveau organisationnel que dans la communication. Nous avons appris plus spécifiquement, de par ce projet, davantage sur la théorie du choix social et en particulier sur l'algorithme d'affectation du mariage stable.

En envisageant l'avenir, des pistes d'amélioration telles que l'optimisation de l'algorithme, l'ajout de fonctionnalités à l'interface utilisateur, et la prise en compte de scénarios avec des ensembles d'étudiants et d'établissements de tailles différentes pourraient être explorées. Ces développements pourraient rendre le système plus robuste et adaptable à une variété de situations.